

KLASIFIKASI JENIS KELAMIN BERDASARKAN CITRA WAJAH MENGGUNAKAN ALGORITMA ADABOOST-SVM

Septia Rani¹⁾, Deni Saepudin²⁾

¹⁾ Ilmu Komputer Universitas Gadjah Mada

FMIPA UGM Sekip Utara Bulaksumur Yogyakarta 55281

²⁾ Ilmu Komputasi Fakultas Sains Institut Teknologi Telkom

Jl. Telekomunikasi no.1 Terusan Buah Batu Bandung 40257

email : ranie.septia@gmail.com¹⁾, dns@ittelkom.ac.id²⁾

Abstrak

Dalam penelitian ini digunakan kombinasi algoritma AdaBoost dan Support Vector Machine (AdaBoostSVM) untuk menirukan kemampuan manusia dalam mengklasifikasikan jenis kelamin berdasarkan citra wajah. Dengan menggunakan algoritma AdaBoost sebagai kerangka kerja dan beberapa RBFSVM (SVM dengan RBF kernel) sebagai komponen classifier-nya, dapat dihasilkan sistem klasifikasi yang mempunyai akurasi tinggi. Berdasarkan hasil pengujian, diperoleh akurasi sistem terbaik adalah 86%. Jika dibandingkan dengan akurasi yang dihasilkan oleh single SVM classifier, tingkat akurasi yang dihasilkan oleh AdaBoostSVM ternyata tidak lebih baik. Hal ini terjadi karena pada kasus klasifikasi jenis kelamin ini terdapat dilemma antara akurasi dan diversity. Dimana jika dikombinasikan classifier-classifier yang akurat namun tidak memiliki diversity yang tinggi, sering menyebabkan performansi algoritma AdaBoost menjadi tidak optimal.

Kata kunci :

klasifikasi, AdaBoost, Support Vector Machine

1. Pendahuluan

Wajah merupakan salah satu fitur unik yang dimiliki oleh manusia. Banyak informasi yang bisa diperoleh hanya dengan melihat wajah manusia, seperti tentang ekspresi seseorang, asal ras, umur, dan juga jenis kelamin. Di antara beberapa *task* kategorisasi wajah, klasifikasi jenis kelamin merupakan informasi yang paling penting secara biologis [1]. Beberapa aplikasi yang memerlukan *task* klasifikasi jenis kelamin ini antara lain: menghitung banyaknya wanita yang memasuki sebuah toko baju untuk kepentingan evaluasi penjualan, membatasi akses terlarang sebuah tempat tertentu berdasarkan jenis kelamin, dan mengumpulkan informasi demografi di tempat umum.

Digunakan algoritma AdaBoost karena kemampuannya yang dapat meningkatkan akurasi dari suatu algoritma *learning*. Selain itu, metode *ensemble* (dalam hal ini AdaBoost) biasanya memiliki kemampuan generalisasi

yang lebih baik daripada satu buah *single learner* [18]. AdaBoost sebagai sebuah *framework* yang menerapkan mekanisme Boosting akan memaksa beberapa komponen *classifier* RBFSVM (SVM dengan RBF kernel) untuk fokus pada sampel-sampel yang salah diklasifikasikan. Sampel-sampel yang terklasifikasi dengan benar diberi bobot yang lebih kecil, sedangkan sampel-sampel yang salah terklasifikasi diberi bobot yang lebih besar, sehingga AdaBoost akan fokus pada sampel-sampel dengan bobot yang lebih besar. Dengan mekanisme tersebut, maka akurasi klasifikasi dapat diperbaiki. Di samping itu, berdasarkan penelitian sebe-lumnya [4], disebutkan bahwa algoritma AdaBoostSVM dapat memperbaiki klasifikasi untuk citra satelit di area kota yang susunannya rumit, dibandingkan jika menggunakan *single SVM classifier*.

Perumusan masalah yang akan dibahas pada penelitian ini adalah: Bagaimana tingkat akurasi sistem yang dibangun menggunakan algoritma AdaBoostSVM? Kemudian juga akan dilakukan perbandingan tingkat akurasi sistem AdaBoostSVM dengan SVM biasa untuk kasus klasifikasi jenis kelamin. Dengan batasan masalah sebagai berikut: Data latih dan data uji yang digunakan berupa citra wajah yang diambil secara *frontal* dan menggunakan Principal Component Analysis (PCA) untuk ekstraksi ciri. Dengan adanya implementasi AdaBoostSVM ini, diharapkan performansi untuk melakukan *task* klasifikasi jenis kelamin berdasarkan citra wajah dapat meningkat, terutama dari segi akurasinya.

2. Tinjauan Pustaka

Terdapat beberapa pendekatan untuk permasalahan klasifikasi jenis kelamin, di antaranya menggunakan Support Vector Machine [2], [3], [7], [12], dan metode pembelajaran AdaBoost [17], dimana SVM menunjukkan hasil yang lebih baik dari segi tingkat kebenaran namun membutuhkan komputasi yang lebih intensif, sedangkan AdaBoost jauh lebih cepat dengan performansi yang sedikit menurun. Berikut ini akan dibahas landasan teori yang mendukung penelitian.

2.1 Klasifikasi

Klasifikasi merupakan salah satu *task* analisis data dimana dibentuk sebuah model (*classifier*) untuk menyatakan suatu objek ke salah satu kategori atau kelas yang sudah didefinisikan sebelumnya. Klasifikasi data merupakan proses yang terdiri atas dua langkah [9]. Langkah pertama adalah tahap pelatihan (*learning*), dimana sebuah algoritma klasifikasi akan membentuk *classifier* dengan menganalisis atau “belajar dari” sekumpulan data latih. Pada langkah kedua, model yang terbentuk digunakan untuk mengklasifikasikan data uji. Akurasi dari sebuah *classifier* untuk sekumpulan data uji yang diberikan merupakan persentase dari data-data uji yang diklasifikasikan dengan benar oleh *classifier*.

2.2 Principal Component Analysis

Principal Component Analysis (PCA) menyediakan langkah-langkah tentang bagaimana mengurangi sekumpulan data yang kompleks menjadi dimensi yang lebih rendah untuk mengungkapkan struktur tersembunyi yang lebih sederhana yang sering terdapat di dalam data [16]. Tujuan dari PCA adalah menghitung basis yang paling berarti untuk menyatakan kembali *dataset* yang mengandung *noise*. Hasil dari PCA adalah principal component (PC) yang merupakan matriks pembawa ciri data.

2.3 AdaBoost

Misalkan terdapat permasalahan klasifikasi dengan dua kelas, dengan variabel *output* dikodekan sebagai $Y \in \{-1, 1\}$. Diberikan vektor dari variabel yang akan diprediksi, yaitu X . Sebuah *classifier* $G(X)$ akan menghasilkan prediksi dengan memberikan satu dari dua buah nilai $\{-1, 1\}$. Pada AdaBoost, akan dihasilkan sebuah himpunan berurutan dari beberapa *classifier* lemah $G_m(x), m = 1, 2, \dots, M$. Prediksi dari semua *classifier* kemudian dikombinasikan melalui voting berbobot untuk menghasilkan prediksi akhir.

Modifikasi data pada setiap tahap Boosting dilakukan dengan menerapkan bobot w_1, w_2, \dots, w_N pada setiap pengamatan data latih $(x_i, y_i), i = 1, 2, \dots, N$. Pertama kali, semua bobot diset $w_i = 1/N$, sehingga langkah pertama secara sederhana melatih *classifier* pada data dengan cara yang umum. Untuk setiap iterasi berikutnya $m = 2, 3, \dots, M$ bobot dimodifikasi dengan sendirinya dan algoritma klasifikasi diterapkan kembali pada data latih. Pada langkah ke- m , data-data yang salah diklasifikasikan oleh *classifier* $G_{m-1}(x)$ akan dinaikkan bobotnya, sedangkan bobot akan diturunkan untuk data-data yang diklasifikasikan dengan benar. Dengan iterasi demikian, data-data yang sulit untuk diklasifikasikan dengan benar akan mendapatkan pengaruh yang bertambah. Sehingga *classifier* pada iterasi berikutnya dipaksa untuk berkonsentrasi pada data-data latih yang salah diklasifikasikan oleh *classifier* sebelumnya [10].

2.4 Support Vector Machine

Konsep SVM dapat dijelaskan secara sederhana sebagai usaha mencari *hyperplane* terbaik yang berfungsi sebagai pemisah dua buah kelas pada *input space* [5]. Pendekatan pertama menggunakan teknik *hard margin*, yaitu dimana kedua kelas dapat terpisah secara sempurna oleh *hyperplane*. Pada SVM, *margin* terbesar dapat ditemukan dengan memaksimalkan nilai jarak antara *hyperplane* dan titik terdekatnya, yaitu $1/\|w\|$.

Pada umumnya dua buah kelas pada *input space* tidak dapat terpisah secara sempurna. Untuk mengatasi masalah ini, SVM dirumuskan ulang dengan memperkenalkan teknik *soft margin*. Dalam *soft margin*, dimasukkan variabel slack $\xi_i (\xi_i > 0)$. Penyelesaian SVM dapat dirumuskan sebagai *Quadratic Programming (QP) problem*, yaitu mencari titik minimal persamaan (1), dengan memperhatikan batasan persamaan (2) [13].

$$\min_{w, \xi} \tau(w, \xi) = \frac{1}{2} \|w\|^2 + C \sum_{i=1}^n \xi_i \quad (1)$$

$$y_i(\vec{x}_i \cdot w + b) \geq 1 - \xi_i, \quad \forall i \quad (2)$$

Masalah dalam domain dunia nyata pada umumnya jarang yang bersifat *linearly separable*. Untuk menyelesaikan kasus *non linear* tersebut, perhitungan SVM memanfaatkan konsep yang disebut *Kernel trick*. Kernel Trick dirumuskan sebagai berikut:

$$K(\vec{x}_i, \vec{x}_j) = \Phi(\vec{x}_i) \cdot \Phi(\vec{x}_j) \quad (3)$$

Pada AdaBoostSVM, digunakan kernel Gaussian RBF yang mempunyai bentuk persamaan sebagai berikut:

$$K(x_i, x_j) = \exp\left(-\frac{\|x_i - x_j\|^2}{2\sigma^2}\right) \quad (4)$$

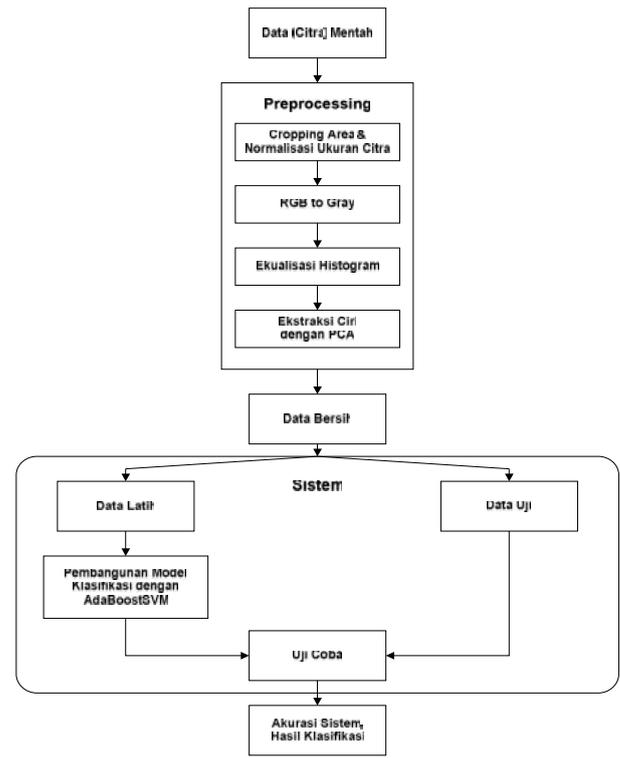
Pada range tertentu, nilai σ yang besar dapat mengurangi kompleksitas dari *classifier*, namun juga mengurangi performansi dari klasifikasi. Sebaliknya, nilai σ yang kecil dapat meningkatkan kompleksitas *learning* dan menyebabkan performansi klasifikasi secara umum dapat meningkat [11].

2.5 AdaBoostSVM

AdaBoostSVM menggunakan SVM sebagai *base learner* pada AdaBoost. SVM yang digunakan adalah SVM dengan RBF kernel (RBF SVM), yang memiliki sebuah parameter yang disebut Gaussian width, σ . Pendekatan AdaBoostSVM secara adaptif mengatur nilai σ pada komponen *classifier* RBF SVM untuk mendapatkan sekumpulan RBF SVM yang cukup akurat untuk AdaBoost [11]. AdaBoostSVM dapat dideskripsikan sebagai berikut:

Tabel 1 Pseudo-code Algoritma AdaBoostSVM [11]

Algorithm AdaBoostSVM	
1. Input:	a set of training samples with labels $\{(\mathbf{x}_1, y_1), \dots, (\mathbf{x}_N, y_N)\}$; the initial σ_{\min} ; the minimal σ_{\max} ; the step of σ , σ_{step} .
2. Initialize:	the weights of training samples: $w_i^1 = 1/N$, for all $i = 1, \dots, N$.
3. Do While ($\sigma > \sigma_{\min}$ and training error $\neq 0$)	(1) Train a RBF SVM component classifier, h_t , on the weighted training set. (2) Calculate the training error of h_t : $\varepsilon_t = \sum_{i=1}^N w_i^t \mathbb{1}_{y_i \neq h_t(\mathbf{x}_i)}$. (3) If $\varepsilon_t \geq 0.5$, decrease σ value by σ_{step} and goto (1). (4) Set the weight of component classifier h_t : $\alpha_t = \frac{1}{2} \ln\left(\frac{1-\varepsilon_t}{\varepsilon_t}\right)$. (5) Update the weights of training samples: $w_i^{t+1} = \frac{w_i^t \exp\{-\alpha_t y_i h_t(\mathbf{x}_i)\}}{C_t}$ where C_t is a normalization constant, and $\sum_{i=1}^N w_i^{t+1} = 1$.
4. Output:	$f(\mathbf{x}) = \text{sign}\left(\sum_{t=1}^T \alpha_t h_t(\mathbf{x})\right)$.



Gambar 1 Skema Sistem

2.6 Pengujian Statistika McNemar

Pengujian statistika McNemar digunakan untuk mengetahui apakah algoritma a memiliki performansi yang lebih baik dibandingkan algoritma b pada suatu *dataset*. Pada sebuah *dataset*, pengujian statistika McNemar dirumuskan sebagai berikut [6]:

$$\frac{(|N_{01} - N_{10}| - 1)^2}{N_{01} + N_{10}} \quad (5)$$

N_{01} : banyaknya data uji yang salah diklasifikasikan oleh algoritma a , namun diklasifikasikan dengan benar oleh algoritma b .

N_{10} : banyaknya data uji yang salah diklasifikasikan oleh algoritma b , namun diklasifikasikan dengan benar oleh algoritma a .

Jika algoritma a dan algoritma b memiliki performansi yang berbeda secara signifikan, maka nilai statistika McNemar akan lebih besar dari $\chi_{1,0.95}^2 = 3.841459$.

3. Metode Penelitian

Skema sistem yang akan digunakan dapat dilihat pada gambar 1. Alur proses yang terjadi dalam merancang sistem secara umum adalah sebagai berikut:

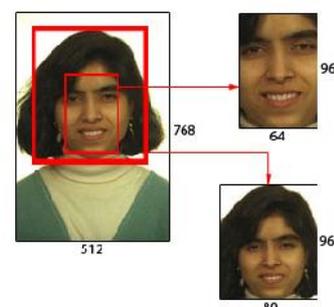
- Preprocessing data.
- Pengembangan AdaBoostSVM.
- Proses pengujian.

3.1 Preprocessing Data

Data yang digunakan dalam penelitian ini adalah data citra wajah (diambil secara *frontal*) yang diambil dari *database* FERET [14,15]. Digunakan 500 citra wajah dimana 400 citra wajah (200 pria dan 200 wanita) digunakan untuk proses pelatihan, 100 citra wajah (50 pria dan 50 wanita) digunakan untuk proses pengujian. Berikut adalah tahapan-tahapan yang dilakukan pada *preprocessing* data:

- Cropping Area dan Normalisasi Ukuran Citra

Cropping area dilakukan untuk mempersempit area citra, sehingga hanya area wajah yang akan digunakan sebagai masukan sistem. Normalisasi ukuran citra dilakukan agar setiap citra memiliki ukuran yang sama. Pada penelitian ini, digunakan dua buah representasi dari citra wajah, seperti diilustrasikan pada gambar 2.



Gambar 2 Citra Wajah Hasil Cropping dan Normalisasi Ukuran

- Konversi RGB to Gray

Konversi citra RGB ke *grayscale* bertujuan untuk menyederhanakan model citra, dari citra RGB yang

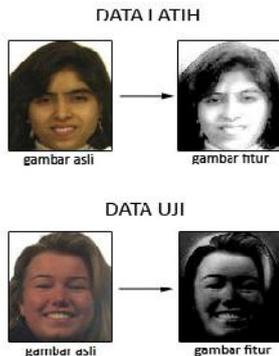
terdiri atas tiga layer menjadi citra *grayscale* yang memiliki satu layer dengan derajat keabuan tertentu.

c. Ekualisasi Histogram

Ekualisasi histogram dilakukan untuk mengatasi kondisi pencahayaan yang berbeda-beda pada setiap citra.

d. Ekstraksi Ciri dengan Principal Component Analysis

Contoh gambar fitur yang dihasilkan oleh PCA dapat dilihat pada gambar 3. Pada gambar fitur dapat dilihat bahwa pada bagian tertentu gambarnya menjadi lebih buram, akan tetapi secara umum masih dapat dikenali apakah termasuk berjenis kelamin pria atau wanita.



Gambar 3 Gambar Fitur yang Dihasilkan oleh PCA

3.2 Pengembangan AdaBoostSVM

Secara umum terdapat dua buah proses sebagai berikut:

a. Proses Pembentukan *Classifier* RBFSVM

Pada proses ini akan dicari *hyperplane* terbaik yang memisahkan kedua kelas pada data latih. Konfigurasi *hyperplane* ini kemudian akan disimpan untuk digunakan pada proses selanjutnya. Implementasi SVM pada program menggunakan rujukan dari Gunn [8].

b. Proses Penggabungan *Classifier*

Proses ini merupakan prosedur iteratif yang menerapkan konsep *ensemble method* menggunakan algoritma AdaBoost. Akan digabungkan beberapa *classifier* RBFSVM agar dihasilkan sebuah *classifier* dengan kemampuan generalisasi yang lebih baik.

Pada tahap inisialisasi, dilakukan perhitungan untuk menentukan nilai σ_{ini} (Gaussian width inisial) dan nilai σ_{min} (Gaussian width minimal) berdasarkan *dataset* yang digunakan. Nilai σ_{ini} di-set dengan nilai jari-jari sebaran data latih pada *input space*. Sedangkan nilai σ_{min} di-set dengan nilai rata-rata jarak minimal di antara dua buah sampel data latih [11].

4. Hasil dan Pembahasan

Pengujian sistem menggunakan dua buah *dataset* citra wajah, enam macam nilai C (0.1, 0.5, 1, 10, 25, 50), dan empat macam nilai C_{step} (25, 50, 100, 200). Sehingga terdapat sebanyak 48 skenario pengujian. Digunakan jumlah PC sebesar 70% untuk setiap skenario pengujian.

4.1 Analisis Tingkat Akurasi Sistem yang Dibangun

Dari pengujian yang telah dilakukan, diperoleh data seperti yang ditunjukkan oleh tabel 2.

Tabel 2 Data Hasil Pengujian

Dataset	C	Skenario Pengujian			
		C_{step}	Banyaknya classifier RBFSVM	Akurasi Pelatihan	Akurasi Pengujian
Citra wajah tanpa rambut	0.1	25	50	78%	76%
		50	46	78%	76%
		100	55	78%	76%
		200	19	78%	76%
	0.5	25	82	82.75%	75%
		50	58	82.75%	75%
		100	52	82.75%	75%
		200	42	82.75%	75%
	1	25	74	87.75%	72%
		50	69	87.5%	71%
		100	39	86.5%	71%
		200	35	85.5%	73%
	10	25	75	100%	76%
		50	72	100%	76%
		100	63	100%	76%
		200	43	100%	76%
	25	25	91	100%	76%
		50	57	100%	76%
		100	39	100%	76%
		200	24	100%	76%
	50	25	67	100%	75%
		50	35	100%	75%
		100	33	100%	75%
		200	32	100%	76%
Citra wajah dengan rambut	0.1	25	15	83.25%	80%
		50	15	83.25%	80%
		100	15	83.25%	80%
		200	16	83.25%	80%
	0.5	25	72	85.5%	82%
		50	48	85.5%	82%
		100	51	85.5%	82%
		200	29	85.5%	82%
	1	25	48	89%	83%
		50	48	89%	83%
		100	47	89%	83%
		200	48	89%	83%
	10	25	58	100%	86%
		50	40	100%	86%
		100	26	100%	86%
		200	20	100%	86%
	25	25	38	100%	86%
		50	31	100%	86%
		100	19	100%	86%
		200	22	100%	86%
	50	25	17	100%	86%
		50	13	100%	86%
		100	10	100%	86%
		200	8	100%	85%

Akurasi sistem terbaik (dilihat dari akurasi pengujian) sebesar 86%. Dapat dilihat bahwa akurasi pelatihan mencapai 100% untuk beberapa kombinasi parameter, akan tetapi akurasi pengujian di bawah 90%. Hal ini terjadi karena *classifier* yang terbentuk memang “belajar” dari data latih. *Dataset* yang digunakan juga berpengaruh terhadap akurasi sistem. *Dataset* citra wajah dengan rambut menghasilkan sistem klasifikasi jenis kelamin dengan akurasi yang lebih tinggi.

Untuk *dataset* citra wajah dengan rambut, sampel-sampel data uji yang salah diklasifikasikan kebanyakan dikarenakan informasi mengenai rambutnya. Pria yang berambut panjang diklasifikasikan sebagai wanita, sedangkan wanita yang berambut pendek diklasifikasi-

kan sebagai pria. Gambar fitur dari sampel yang salah diklasifikasikan dapat dilihat pada gambar 4.



Gambar 4 Gambar Fitur dari Sampel yang Salah Diklasifikasikan

Berdasarkan hasil pengujian pada tabel 2, terlihat bahwa semakin besar nilai C, maka *error* pelatihan dari masing-masing komponen *classifier* semakin kecil. RBFSVM tidak dapat “belajar” dengan baik ketika digunakan nilai C yang kecil. Akan tetapi saat digunakan nilai C yang memenuhi batasan minimal tertentu, pada nilai tertentu akan diperoleh komponen *classifier* RBFSVM dengan *error* pelatihan sebesar 0%. Sesuai dengan mekanisme voting berbobot, maka hasil klasifikasi dari *classifier* tersebut akan mendominasi hasil klasifikasi dari *classifier-classifier* yang terbentuk pada iterasi sebelumnya. Dapat diamati juga bahwa *step* berpengaruh terhadap banyaknya komponen *classifier* RBFSVM yang terbentuk. Semakin besar nilai *step* maka komponen *classifier* RBFSVM yang terbentuk semakin sedikit. Meskipun demikian, nilai *error* pengujian akhir untuk nilai *step* yang berbeda menunjukkan hasil yang relatif stabil.

4.2 Analisis Perbandingan Performansi AdaBoost-SVM dengan SVM

Untuk perbandingan, *classisfier* SVM menggunakan RBFSVM dengan nilai Gaussian width sebesar \min_{step} . Sedangkan untuk AdaBoostSVM, menggunakan nilai \min_{step} 25. Dari pengujian yang telah dilakukan, diperoleh data seperti ditunjukkan pada tabel 3.

Untuk mengetahui apakah kedua algoritma tersebut mempunyai performansi yang berbeda secara signifikan pada sebuah *dataset* maka dilakukan pengujian statistika McNemar menggunakan persamaan (5). Tabel 4 menunjukkan hasil pengujian statistika McNemar. Berdasarkan hasil pengujian statistika McNemar, performansi algoritma AdaBoostSVM tidak berbeda secara signifikan dari SVM biasa untuk kedua *dataset* yang digunakan. Hal ini dapat dilihat dari nilai statistika McNemar yang tidak lebih dari 3.841459.

Tabel 3 Data Perbandingan Akurasi AdaBoostSVM dengan SVM

Dataset	C	Skenario Pengujian			
		AdaBoostSVM		SVM- σ_{normal}	
		Akurasi Pelatihan	Akurasi Pengujian	Akurasi Pelatihan	Akurasi Pengujian
Citra wajah tanpa rambut	0.1	78%	76%	82.25%	78%
	0.5	82.75%	75%	91.25%	70%
	1	87.75%	72%	95.5%	69%
	10	100%	76%	100%	76%
	25	100%	76%	100%	75%
	50	100%	75%	100%	75%
Citra wajah dengan rambut	0.1	83.25%	80%	84.75%	81%
	0.5	85.5%	82%	92.25%	84%
	1	89%	83%	94.25%	85%
	10	100%	86%	100%	87%
	25	100%	86%	100%	87%
	50	100%	86%	100%	87%

Tabel 4 Hasil Pengujian Statistika McNemar

Dataset	C	Statistika McNemar (SVM - AdaBoostSVM)	
		Berbeda secara Signifikan	Berbeda secara Signifikan
Citra wajah tanpa rambut	0.1	0.25	tidak
	0.5	2.285714	tidak
	1	0.8	tidak
	10	0	tidak
	25	0	tidak
	50	0.25	tidak
Citra wajah dengan rambut	0.1	0	tidak
	0.5	0.25	tidak
	1	0.166667	tidak
	10	0	tidak
	25	0	tidak
	50	0	tidak
Rata-rata		0.333532	(0 ya / 12 tidak)

Setelah dilakukan observasi terhadap hasil klasifikasi dari masing-masing *classifier* RBFSVM yang dihasilkan pada setiap iterasi Boosting, diketahui bahwa masing-masing *classifier* tersebut mempunyai akurasi yang sudah cukup tinggi. Selain itu, kesalahan dari masing-masing *classifier* tersebut saling berkorelasi. Dengan kondisi yang demikian maka mekanisme voting berbobot yang terdapat di algoritma AdaBoost tidak dapat meningkatkan akurasi sistem.

Pada algoritma AdaBoost dikenal istilah *diversity*. *Diversity* merupakan ukuran keragaman kesalahan antara satu *classifier* dengan *classifier* lain. Kesalahan dilihat dari sampel-sampel yang salah diklasifikasikan. Jika kesalahan oleh dua buah *classifier* yang berbeda semakin tidak berkorelasi, maka *diversity*-nya dinyatakan tinggi dan begitu sebaliknya. Pada Ada-Boost, jika setiap komponen *classifier* cukup akurat dan mempunyai *diversity* yang cukup tinggi, maka kesalahan-kesalahan klasifikasi yang tidak saling berkorelasi antar *classifier* dapat diatasi dengan mekanisme voting berbobot sehingga diperoleh akurasi yang lebih tinggi. Terdapat dilemma antara akurasi dan *diversity* pada AdaBoost. Maksudnya, semakin tinggi akurasi dua buah *classifier*, maka *diversity*-nya semakin kecil, begitu sebaliknya. Fenomena inilah yang terjadi saat AdaBoostSVM diterapkan untuk kasus klasifikasi jenis kelamin. *Diversity* antar komponen *classifier* yang rendah menyebabkan performansi algoritma AdaBoost-SVM tidak optimal. Selain itu juga terdapat kasus dimana *classifier* pada tahap awal iterasi Boosting (pada umumnya *classifier* paling lemah) diberi bobot *classifier* () yang lebih besar sehingga menjadikannya lebih berpengaruh. Kondisi yang demikian kurang menguntungkan bagi algoritma AdaBoost.

5. Kesimpulan dan Saran

Sistem klasifikasi jenis kelamin berdasarkan citra wajah yang dibangun menggunakan algoritma AdaBoostSVM menghasilkan nilai akurasi terbaik sebesar 86%. Berdasarkan hasil pengujian statistika McNemar, SVM biasa dan AdaBoostSVM mempunyai performansi yang sebanding (tidak berbeda secara signifikan) untuk *dataset* yang digunakan. Kegagalan algoritma AdaBoost dalam meningkatkan akurasi pada SVM disebabkan adanya dilemma antara akurasi dan *diversity*. Pada kasus ini *diversity* antar komponen *classifier* rendah, sehingga menyebabkan performansi algoritma AdaBoostSVM tidak optimal.

Untuk penelitian lebih lanjut, diperlukan mekanisme untuk menghindari dilemma antara akurasi dan *diversity* yang mungkin terjadi pada AdaBoostSVM. Dapat digunakan algoritma *diverse* AdaBoostSVM (yang merupakan pengembangan dari algoritma AdaBoost-SVM) dimana algoritma tersebut dapat mengontrol keseimbangan antara akurasi dan *diversity*, sehingga dapat diperoleh tingkat akurasi yang lebih baik.

Daftar Pustaka

- [1] Abdi, Hervé., Valentin, Dominique., Edelman, Betty., and O'Toole, Alice J., 1995, "More about the difference between men and women: evidence from linear neural networks and the principal component approach", *Perception* 24: 539-562.
- [2] Ari, Andi., 2009, *Klasifikasi Jenis Kelamin Manusia Berdasarkan Citra Wajah Menggunakan Metode Independent Component Analysis dan Support Vector Machine*, Tugas Akhir S1 Teknik Informatika, IT Telkom, Bandung.
- [3] Buchala, Samarasena., Davey, Neil., Frank, Ray J., Gale, Tim M., Loomes, Martin J., and Kanargard, Wanida., 2004, "Gender Classification of Face Images: The Role of Global and Feature-Based Information", *Lecture Notes in Computer Science*, Volume 3316/2004, 763-768.
- [4] Chen, Wei-Chi., Chen, Mei-Hsin., and Chou, Tien-Yin., 2009, "The Study of AdaBoost Algorithm to Enhance Support Vector Machine Application on Urban Rice Land Classification", *Proceeding Asian Conference on Remote Sensing*, Section TS14.
- [5] Cristianini, N., and Shawe-Taylor, J., 2000, *An Introduction to Support Vector Machines and Other Kernel-Based Learning Methods*, Cambridge University Press, Cambridge.
- [6] Dietterich, T. G., 1998, *Approximate Statistical Tests for Comparing Supervised Classification Learning Algorithms*, *Neural Computation* 10, 1895-1923.
- [7] Graf, Arnulf B. A., and Wichmann, Felix A., 2002, "Gender Classification of Human Faces", *Proceedings of the Second International Workshop on Biologically Motivated Computer Vision*, BMCV '02.
- [8] Gunn, Steve., 1998, "Support Vector Machines for Classification and Regression", *ISIS Technical Report*, Image Speech & Intelligent Systems Group, University of Southampton.
- [9] Han, Jiawei., and Kamber, Micheline., 2006, *Data Mining Concepts and Techniques*, Elsevier Inc, San Francisco.
- [10] Hastie, Trevor., Tibshirani, Robert., and Friedman, Jerome., 2001, *The Elements of Statistical Learning*, Springer, Canada.
- [11] Li, Xuchun., Wang, Lei., and Sung, Eric., 2008, "AdaBoost with SVM-based component classifiers", *Engineering Application of Artificial Intelligence* 21: 785-795.
- [12] Moghaddam B. and M.-H. Yang., 2000, "Gender Classification with Support Vector Machines", *Technical Report: TR-2000-01*, Mitsubishi Electric Research Laboratory.
- [13] Nugroho, Anto Satriyo., 2008, "Support Vector Machine: Paradigma Baru dalam Soft Computing", *Konferensi Nasional Sistem dan Informatika*.
- [14] Phillips, P.J., H. Moon, S.A. Rizvi, and P.J. Rauss, "The FERET Evaluation Methodology for Face Recognition Algorithms," *IEEE Trans, Pattern Analysis and Machine Intelligence*, Vol. 22, pp. 1090-1104, 2000.
- [15] Phillips, P.J., H. Wechsler, J. Huang, and P. Rauss, "The FERET database and evaluation procedure for face recognition algorithms," *Image and Vision Computing J*, Vol. 16, No. 5, pp. 295-306, 1998.
- [16] Shlens, Jonathon., 2005, diakses pada tanggal 13 Januari 2011, *A Tutorial on Principal Component Analysis*, URL:<http://www.sn1.salk.edu/ushlens/pub/notes/pca.pdf>.
- [17] Wu, Bo., Ai, Haizhou., and Huang, Chang., 2003, "LUT-Based AdaBoost for Gender Classification", *Proceedings of the 4th International Conference on Audio and Video Based Biometric Person Authentication*, AVBPA'03.
- [18] Wu, Xindong., and Kumar, Vipin., 2009, *The Top Ten Algorithms in Data Mining*, Taylor & Francis Group, New York.

Biodata Penulis

Septia Rani, memperoleh gelar Sarjana Teknik (S.T.), Program Studi Teknik Informatika Institut Teknologi Telkom, lulus tahun 2011. Saat ini melanjutkan studi di Program Magister Ilmu Komputer Universitas Gadjah Mada.

Deni Saepudin, memperoleh gelar Sarjana Sains (S.Si), Program Studi Matematika FMIPA ITB, lulus tahun 1998. Tahun 2003 memperoleh gelar Magister Sains (M.Si) dari Program Studi Matematika ITB. Program Doktor Matematika, lulus tahun 2009. Saat ini sebagai staf pengajar Program Studi Ilmu Komputasi dan Magister Teknik Informatika IT Telkom Bandung.