

Pengaruh Variasi Jumlah Data Pelatihan SVM Terhadap Unjukkerja pada Sistem Pengenalan Pola Tulisan Tangan Aksara Jawa Nglegeno

Effect of SVM Training Set Variation on Handwritten Basic Jawanese Character Recognition System Performance

Azis Wisnu Widhi Nugraha

azis.wwn@unsoed.ac.id

Prodi Teknik Elektro Jurusan Teknik FST Unsoed
Jl. Kampus no.1, Grendeng, Purwokerto

Abstrak— Support Vector Machines (SVM) is one of many classification methods implemented on pattern recognition, including handwritten character recognition system. Numbers of training set will affect the performance of the recognition system. A basic Jawanese handwritten character recognition system based on SVM used to show the effect of training set numbers variation on system performance. The experiment result indicating that SVM has good generalization on limiting training set (on a data set, 30% data used for training set can give more than 90% success rate on all data set).

Keyword— SVM, training set number, generalization rate, handwritten character recognition, basic Jawanese character recognition.

PENDAHULUAN

Support Vector Machine (SVM) merupakan salah satu teknik klasifikasi. Pada dasarnya SVM merupakan sebuah *linear classifier*. Sementara itu permasalahan nyata pada umumnya tidak *linear*. Dengan menggunakan teknik kernel, SVM dapat digunakan untuk menyelesaikan permasalahan yang tidak *linear* (Burges, 1998; Byun and Lee, 2003; Cortes and Vapnik, 1995; Fletcher 2008; Fradkin and Muchnik, 2006; Nugroho dkk, 2003; Oliveira and Sabourin, 2004; Vijayakumar dan Wu, 1999). Hal ini didasari pada teori Cover mengenai pemisahan pola (Nugroho dkk, 2003; Oliveira and Sabourin, 2004; Vijayakumar dan Wu, 1999).

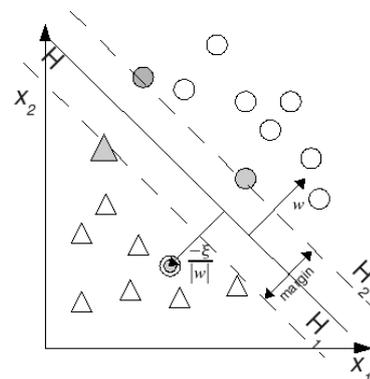
Selain merupakan *linear classifier*, SVM juga merupakan *binary classifier*. Hal ini berarti bahwa SVM hanya mampu melakukan pengelompokan terhadap dua pola. Untuk menyelesaikan permasalahan kelas jamak, SVM dapat dengan mudah dikombinasikan (Burges, 1998; Byun and Lee, 2003; Cortes and Vapnik, 1995; Fradkin and Muchnik, 2006; Oliveira and Sabourin, 2004). Guna menyelesaikan permasalahan kelas jamak, strategi yang dapat diterapkan pada SVM antara lain satu-lawan-sisa (*one-vs-rest*) dan pohon struktur (*pairwise SVM* dan DAG SVM) (Byun and Lee, 2003; Oliveira and Sabourin, 2004).

Nugroho dkk. (2003) menyatakan kelebihan SVM antara lain adalah: kemampuan generalisasi yang baik, kemampuan memecahkan masalah berdimensi tinggi dengan keterbatasan percontoh (*sample*), memiliki

landasar teori yang dapat dianalisa secara jelas, serta relatif mudah untuk diimplementasikan dengan banyak tersedianya pustaka quadratic programming (QP) solver.

Tulisan ini akan menunjukkan bagaimana pengaruh perubahan jumlah data pelatihan terhadap unjuk kerja sistem pengenalan pola tulisan tangan aksara Jawa *noglegeno* dengan menggunakan SVM. Penggunaan SVM dalam pengenalan pola tulisan tangan sendiri telah menunjukkan unjuk kerja yang lebih baik dibandingkan dengan metode yang lain (Byun and Lee, 2003; Oliveira and Sabourin, 2004). Bahkan Cortes and Vapnik (1995) menggunakan obyek pengenalan tulisan tangan untuk menunjukkan keunggulan SVM dibandingkan dengan metode yang lain.

SUPPORT VECTOR MACHINE



Gambar 9 Ilustrasi *hyperplane* pada SVM.

Untuk mendapatkan penjelasan lebih lengkap mengenai *Support Vector Machine* dapat ditemui pada (Burges, 1998; Cortes and Vapnik, 1995; Fletcher 2008; Vijayakumar dan Wu, 1999). Berikut akan disampaikan gambaran singkat mengenai SVM dalam pengelompokan data.

Notasi pengelompokan data dua kelas dapat ditulis $\{(x_i, y_i)\}$, dengan x_i adalah vektor data dengan N dimensi dan $y_i \in \{-1, 1\}$ adalah target. Diantara kedua kelas data dapat dibuat bidang pemisah *hyperplane* H sebagaimana ditunjukkan pada Gambar 9. Jika H_1 dan H_2 adalah bidang sejajar H yang pada keduanya terdapat vektor-vektor pola terluar dari masing-masing kelas yang disebut dengan *Support Vector* (SV), maka algoritma SVM pada dasarnya berusaha mencari H optimal yang memberikan jarak terbesar antara H_1 dan H_2 .

A. Linear Support Vector Machine

Pada kasus pemisahan data yang *linear*, persamaan *hyperplane* H dapat dinyatakan oleh persamaan (1), dengan w adalah vektor bobot, x adalah vektor pola, dan b adalah bias.

$$w \cdot x + b = 0 \quad (1)$$

Jika dari l buah vektor data pelatihan dapat dibuat sebuah matriks H dengan komponen penyusun H_{ij} seperti pada persamaan (2) berikut,

$$H_{ij} = y_i y_j (x_i \cdot x_j), \text{ dengan } i, j = 1, 2, 3, \dots, l \quad (2)$$

maka algoritma SVM akan mencari nilai optimasi persoalan *quadratic programming* pada persamaan (3) berikut. Persamaan (3) tersebut dapat diselesaikan dengan menggunakan *quadratic programming solver* yang telah banyak tersedia.

$$\max \left[\sum_{i=1}^l \alpha_i - \frac{1}{2} \alpha^T H \alpha \right], \text{ subject to } \alpha_i \geq 0, \sum_{i=1}^l \alpha_i y_i = 0$$

dengan :

- α_i = pengali *Lagrange* untuk vektor data pelatihan ke- i
- C = variabel *slack* yang besarnya ditentukan sendiri
- α = $[\alpha_1, \alpha_2, \alpha_3, \dots, \alpha_l]$

$$(3)$$

Penyelesaian persamaan (3) di atas menghasilkan pengali *lagrange* untuk setiap vektor data pelatihan α_i . Semua pengali *lagrange* dengan nilai $0 \leq \alpha_i \leq C$ merupakan *support vector* yang terletak pada bidang H_1 dan H_2 . Vektor bobot w dan nilai bias b dapat dinyatakan dengan persamaan (4) dan persamaan (5) berikut, dengan n_s adalah jumlah *support vector* dan S adalah himpunan semua *support vector*.

$$w = \sum_{m \in S} \alpha_m y_m x_m \quad (4)$$

$$b = \frac{1}{n_s} \sum_{n \in S} \left(y_n - \sum_{m \in S} \alpha_m y_m x_m \cdot x_n \right) \quad (5)$$

Klasifikasi terhadap sebuah vektor data x dapat dilakukan dengan persamaan (6) berikut.

$$y = \text{sign} \left(\sum_{m \in S} \alpha_m y_m x_m \cdot x + b \right) \quad (6)$$

B. Non-Linear Support Vector Machine

Teori *Cover* mengenai pemisahan pola (9) menyatakan bahwa secara umum pengelompokan pola kompleks yang ditempatkan pada ruang berdimensi tinggi secara tidak *linear* terlihat lebih terpisah secara *linear* dibandingkan pada ruang berdimensi lebih rendah. Hal ini berarti jika kita memiliki vektor x pada ruang berdimensi N , maka biasanya akan terdapat sebuah pemetaan $\Phi(x)$ ke ruang berdimensi yang lebih tinggi M sehingga vektor masukan x akan terpisah secara *linear*.

Dari persamaan (2), (4), (5), dan (6) dapat diketahui bahwa semua perkalian antara dua vektor selalu berada dalam bentuk perkalian skalar (*dot product*). Dengan demikian jika semua vektor masukan terlebih dahulu dipetakan dengan menggunakan pemetaan $\Phi(x)$, maka $x_i \cdot x_j$ akan berubah menjadi $\Phi(x_i) \cdot \Phi(x_j)$. Pemetaan $\Phi(x)$ biasanya sangat rumit, namun $\Phi(x_i) \cdot \Phi(x_j)$ adalah sebuah skalar. Perkalian skalar $\Phi(x_i) \cdot \Phi(x_j)$ dapat digantikan dengan sebuah fungsi kernel $K(x_i, x_j)$ tanpa harus mengetahui fungsi pemetaan $\Phi(x)$ secara langsung. Fungsi kernel yang umum digunakan dalam SVM dapat dilihat pada Tabel 6.

TABEL 6 FUNGSI KERNEL YANG SERING DIGUNAKAN DALAM SVM

Kernel	$K(x_i, x_j)$
Linear	$K(x_i, x_j) = (x_i^T x_j)$
Polinomial	$K(x_i, x_j) = (x_i^T x_j + 1)^d$
Gaussian RBF	$K(x_i, x_j) = \exp \left(- \frac{\ x_i - x_j\ ^2}{2\sigma^2} \right)$
Sigmoid	$K(x_i, x_j) = \tanh(\beta_0 x_i^T x_j - \beta_1)$

C. Support Vector Machine kelas jamak

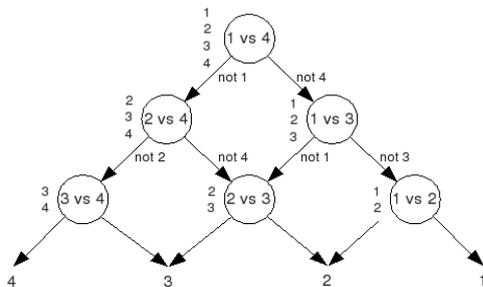
Untuk menyelesaikan persoalan kelas jamak (N kelas) dengan menggunakan SVM pada dasarnya dapat digunakan dua strategi: strategi *one-vs-rest* dan strategi struktur pohon (Byun and Lee, 2003; Oliveira and Sabourin, 2004).

1) *One-vs-rest SVM*: Strategi ini pada dasarnya membangun N buah SVM untuk setiap kelas. Pada tahap pelatihan SVM kelas ke- i , data pelatihan akan diberikan label target $+1$ untuk vektor data pelatihan kelas ke- i dan label target -1 untuk vektor data pelatihan kelas lainnya. Pada tahap klasifikasi akan diperoleh N buah keluaran $d_i(x) = w_i \cdot x + b_i$ dengan $i=1, 2, \dots, N$.

Vektor masukan x masuk ke dalam kelas j jika memenuhi persamaan (7) berikut.

$$d_j(x) = \max_{i=1,2,\dots,N} d_i(x) \quad (7)$$

2) *SVM struktur pohon*: Pada strategi ini SVM hanya dilatih untuk membandingkan dua kelas saja. Dengan demikian untuk permasalahan N kelas akan dibutuhkan $N(N-1)/2$ SVM. Pengelompokan SVM disusun dalam bentuk pohon dengan masing-masing titik pada pohon menyatakan sebuah SVM. Pohon dari bawah ke atas yang mirip dengan pohon eliminasi turnamen tenis diusulkan oleh Pontil dan Veri serta diterapkan oleh Guo *et al.* (Byun and Lee, 2003). Struktur pohon dari atas ke bawah disebut dengan *Decission Directed Acyclic Graph* (*Decission DAG*) yang kemudian disebut dengan DAG SVM diusulkan dalam Platt, *et al.* (2000). Strategi DAG SVM untuk permasalahan empat kelas dapat dilihat pada Gambar 10. Hsu dan Lin dalam Byun and Lee (2003) mengindikasikan bahwa strategi struktur pohon lebih cocok untuk keperluan praktis.



Gambar 10 Struktur DAG SVM untuk permasalahan 4-kelas.

PENGENALAN POLA TULISAN TANGAN AKSARA JAWA

Komputerisasi aksara Jawa dapat digunakan sebagai media pembelajaran dan melestarikan keberadaan aksara Jawa yang sudah mulai jarang digunakan Sayogo (2006a, 2006b). Sementara itu penelitian mengenai pengenalan aksara Jawa hingga saat ini masih jarang dilakukan (Widiarti, 2006; Asriani dkk., 2006; Nugraha, 2009). Tulisan ini merupakan bagian dari penelitian pengenalan pola aksara Jawa yang bertujuan untuk melestarikan keberadaan aksara Jawa.

D. Aksara Jawa

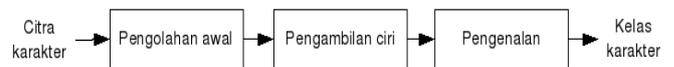
Tabel 7 berikut menunjukkan dua puluh karakter dasar dari aksara Jawa (aksara *nglegeno*) yang digunakan dalam tulisan ini.

TABEL 7 DUA PULUH AKSARA JAWA *NGLEGENO*

Huruf	Lafal								
a	Ha	n	Na	c	Ca	r	Ra	k	Ka
ɛ	Da	ɛ	Ta	s	Sa	w	Wa	l	La
p	Pa	d	Dha	j	Ja	y	Ya	v	Nya
m	Ma	g	Ga	b	Bha	q	Tha	z	Nga

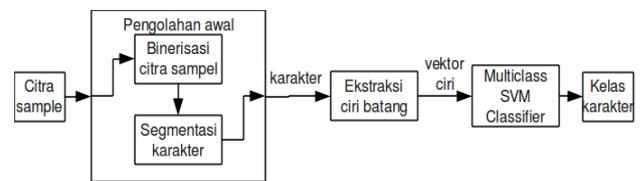
E. Model sistem pengenalan pola tulisan tangan aksara Jawa

Secara umum, model sistem pengenalan pola tulisan tangan terdiri atas: pengolahan awal, pengambilan ciri dan pengenalan karakter (Widiarti, 2006; Asriani dkk., 2006; Nugraha, 2009) yang ditunjukkan pada Gambar 11.



Gambar 11 Model sistem pengenalan pola tulisan tangan.

Adapun diagram blok sistem pengenalan pola tulisan tangan aksara Jawa *nglegeno* yang digunakan dalam tulisan ini sebagaimana yang dijelaskan dalam [16] seperti pada Gambar 12 berikut.



Gambar 12 Diagram blok sistem pengenalan pola tulisan tangan aksara Jawa *nglegeno*.

F. Ciri batang

Ciri batang (Gader *et al.*, 1995) menggambarkan berapa ketebalan obyek dan ketebalan latar belakang dalam empat arah mata angin. Ciri batang pada dasarnya tidak memerlukan normalisasi ukuran citra karakter terlebih dahulu. Namun demikian, jika ukuran citra karakter terlalu besar, ukuran citra karakter perlu dikecilkan terlebih dahulu untuk mengurangi beban komputasi.

Untuk mendapatkan ciri batang (Gader *et al.*, 1995) pertama-tama dibuat delapan citra ciri dari citra biner karakter. Tiap-tiap citra ciri berhubungan dengan arah (timur, timur laut, utara, dan barat laut) untuk obyek maupun latar belakang. Citra ciri diperoleh dengan menerapkan algoritma pelolosan ganda pada citra obyek dan citra latar belakang. Citra obyek adalah citra dengan piksel pada obyek bernilai '1' dan '0' pada latar belakang. Sedangkan citra latar belakang adalah citra dengan piksel latar belakang bernilai '1' dan '0' pada obyek.

Dengan $nbaris$, $nkolom$, $t(i,j)$, $tl(i,j)$, $u(i,j)$, dan $bl(i,j)$ berturut-turut adalah jumlah baris, jumlah kolom, citra ciri arah timur, citra ciri arah timur laut, citra ciri arah utara dan citra ciri arah barat laut, algoritma pelolosan ganda yang digunakan untuk mendapatkan citra ciri adalah sebagai berikut.

/* PELOLOSAN MAJU */

FOR i = 1,2,...,nbaris DO

FOR j = 1,2,...,mkolom DO

$$\begin{aligned} t(i,j) &= t(i,j-1) + 1 \\ tl(i,j) &= tl(i-1,j+1) + 1 \\ u(i,j) &= u(i-1,j) + 1 \\ bl(i,j) &= bl(i-1,j-1) + 1 \end{aligned}$$

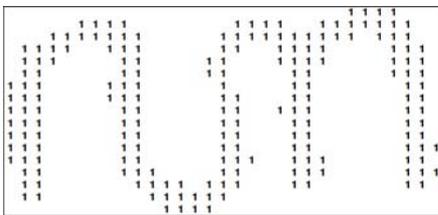
/* PELOLOSAN MUNDUR */

FOR i = nbaris,nbaris-1,...,1 DO

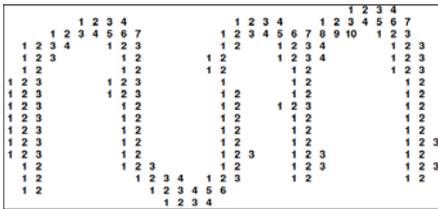
FOR j = mkolom,mkolom-1,...,1 DO

$$\begin{aligned} t(i,j) &= \max(t(i,j), t(i,j+1)) \\ tl(i,j) &= \max(tl(i,j), tl(i+1,j-1)) \\ u(i,j) &= \max(u(i,j), u(i+1,j)) \\ bl(i,j) &= \max(bl(i,j), bl(i+1,j+1)) \end{aligned}$$

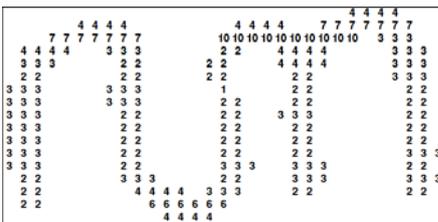
Ilustrasi citra ciri hasil penerapan algoritma pelolosan ganda di atas terhadap citra obyek aksara 'a' (ha) (Gambar 13) pada arah timur dapat dilihat berturut-turut pada Gambar 14 dan Gambar 15.



Gambar 13 Citra obyek aksara 'ha'.



Gambar 14 Citra obyek aksaran 'ha' hasil pelolosan maju pada arah timur.



Gambar 15 Citra obyek aksara 'ha' hasil pelolosan mundur pada arah timur.

Vektor ciri dihitung dari citra ciri dengan daerah tumpang tindih. Jika diketahui f_c , $c = 1, 2, \dots, 8$ adalah citra ciri berkaitan dengan citra karakter masukan, t dan l adalah tinggi dan lebar citra masukan, m dan n adalah jumlah daerah pada vertikal dan horisontal, maka dimensi tiap daerah untuk menghitung vektor ciri ditentukan dengan persamaan (8).

$$\text{dimensi} = \frac{t}{(m+1)/2} \times \frac{l}{(n+1)/2} \quad (8)$$

$$f_c(b_i, k_j) = \left((i-1) \times \frac{t}{(m+1)} + (j-1) \times \frac{l}{(n+1)} \right) \quad (9)$$

$$z_{i,j,c} = \frac{1}{N} \sum_{b_i=1}^{b_{i+2}} \sum_{k_j=1}^{k_{j+2}} f_c(b_i, k_j) \quad (10)$$

Sedangkan nilai pendekatan koordinat kiri atas (b_i, k_j) setiap daerah $z(i,j)$, $i=1, 2, \dots, m$, $j=1, 2, \dots, n$ dapat dihitung dengan persamaan (9). Nilai ciri untuk setiap daerah $z(i,j)$ dapat dihitung dengan persamaan (10) dengan N adalah nilai maksimal jumlahan yang mungkin pada tiap-tiap daerah. Panjang vektor ciri batang adalah $m \times n \times 8$ komponen.

METODE PENELITIAN

Metode penelitian yang digunakan dalam tulisan ini adalah sebagai berikut.

A. Percontoh (sample)

Percontoh yang digunakan adalah tulisan tangan aksara Jawa *nglegeno* yang dikumpulkan dari empat belas responden. Setiap responden diminta untuk menuliskan sepuluh set aksara Jawa *nglegeno* pada lembar pengambilan percontoh. Dengan demikian diperoleh 2800 percontoh tulisan tangan aksara Jawa *nglegeno* dan jumlah percontoh tiap karakter sebanyak 140 percontoh. Setiap lembar percontoh dipayar dengan pemayar digital untuk mendapatkan citra digital dari setiap lembar percontoh.

B. Sistem pengenalan pola tulisan tangan aksara Jawa

Sistem pengenalan pola tulisan tangan aksara Jawa dibuat berdasarkan diagram blok pada Gambar 12. Sistem pengenalan aksara Jawa *nglegeno* dibuat dengan menggunakan perangkat lunak *Octave* (Eaton, 2002).

Pengelompok yang digunakan adalah DAG SVM dengan kernel *Gaussian RBF* parameter 1. Sedangkan ciri yang digunakan untuk pengenalan adalah ciri batang (Gader *et al.*, 1995) sebagaimana dijelaskan di atas dengan $m = 3$ dan $n = 5$.

C. Uji coba

Dilakukan dua uji coba, yaitu.

1) *Uji coba 1*: Pada ujicoba ini diambil sepuluh lembar percontoh pertama sehingga terdapat 2000 percontoh tulisan tangan karakter aksara Jawa *nglegeno*. DAG SVM akan dilatih dengan variasi jumlah data pelatihan 200, 400, 600, 800, 1000, ..., 2000. Pengujian akan dilakukan menggunakan sisa percontoh dari percontoh yang telah diambil sebagai data pelatihan.

2) *Uji coba 2*: Uji coba ini pada dasarnya sama dengan uji coba 1. Perbedaannya adalah pada jumlah percontoh yang digunakan. Pada uji coba ini digunakan seluruh

lembar percontohan yang telah diambil, sehingga terdapat 2800 percontohan tulisan tangan karakter aksara Jawa nglegeno.

Untuk setiap variasi akan dihitung tingkat keberhasilan pengenalan memorisasi ($%_{mem}$), generalisasi ($%_{gen}$), dan tingkat keberhasilan pengenalan total ($%_{tot}$). Jika jumlah data pelatihan adalah n_{lat} , jumlah data pengujian adalah n_{uji} , dan jumlah data total adalah $n_{tot} = n_{lat} + n_{uji}$, maka tingkat keberhasilan dapat dihitung dengan persamaan (11), (12), dan (13).

$$\%_{mem} = \frac{\text{jmlh data pelatihan dikenali benar}}{n_{tot}} \times 100\% \quad (11)$$

$$\%_{gen} = \frac{\text{jmlh data pengujian dikenali benar}}{n_{uji}} \times 100\% \quad (12)$$

$$\%_{tot} = \frac{\text{jmlh seluruh data dikenali benar}}{n_{tot}} \times 100\% \quad (13)$$

HASIL UJI COBA

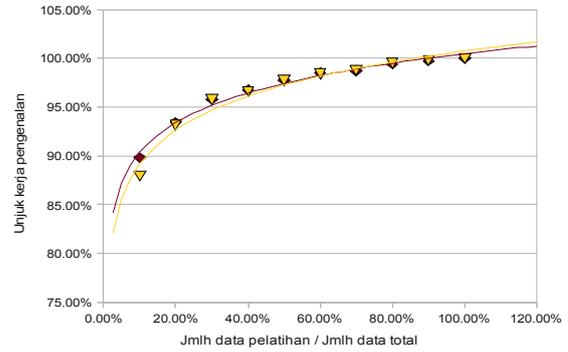
Hasil ujicoba dapat dilihat pada Tabel 8 (ujicoba 1), Tabel 9 (ujicoba 2) dan Gambar 16.

TABEL 8 HASIL UJICOBA 1

Nlat	200	400	600	800	1000
Nuji	1800	1600	1400	1200	1000
nuji/ntot	10.00%	20.00%	30.00%	40.00%	50.00%
%mem	100.00%	100.00%	100.00%	100.00%	100.00%
%gen	86.61%	91.44%	94.00%	94.33%	95.50%
%tot	87.95%	93.15%	95.80%	96.60%	97.75%
Nlat	1200	1400	1600	1800	2000
Nuji	800	600	400	200	0
nuji/ntot	60.00%	70.00%	80.00%	90.00%	100.00%
%mem	100.00%	100.00%	100.00%	100.00%	100.00%
%gen	96.00%	96.00%	97.50%	98.00%	
%tot	98.40%	98.80%	99.50%	99.80%	100.00%

TABEL 9 HASIL UJICOBA 2

nlat	280	560	840	1120	1400
nuji	2520	2240	1960	1680	1400
nuji/ntot	10.00%	20.00%	30.00%	40.00%	50.00%
%mem	100.00%	100.00%	100.00%	100.00%	100.00%
%gen	88.69%	91.79%	94.03%	94.64%	95.57%
%tot	89.82%	93.43%	95.82%	96.79%	97.79%
nlat	1680	1960	2240	2520	2800
nuji	1120	840	560	280	0
nuji/ntot	60.00%	70.00%	80.00%	90.00%	100.00%
%mem	100.00%	100.00%	100.00%	100.00%	100.00%
%gen	96.25%	95.83%	97.14%	97.50%	
%tot	98.50%	98.75%	99.43%	99.75%	100.00%



Gambar 16 Unjuk kerja pengenalan ujicoba dengan variasi jumlah data pelatihan.

Terlihat dari Tabel 8, Tabel 9 dan Gambar 16 unjuk kerja pengenalan sistem telah mencapai lebih dari 90% persen pada saat jumlah data pelatihan sekitar 20%. Terlihat pula bahwa penambahan jumlah data pelatihan antara 20% hingga 40% masih memberikan perubahan unjuk kerja yang cukup besar. Namun jika jumlah data pelatihan ditambah, peningkatan unjuk kerja kurang terlalu signifikan.

KESIMPULAN

Pada sistem pengenalan pola tulisan tangan aksara Jawa nglegeno dengan menggunakan SVM, variasi jumlah data pelatihan hingga 30% masih memberikan perubahan unjuk kerja yang cukup besar. Penambahan jumlah data pelatihan di atas 40% terlihat kurang memberikan perubahan yang signifikan. Untuk mendapatkan unjukkerja di atas 90% penggunaan data pelatihan sebesar 30% telah mencukupi. Hal ini menunjukkan bahwa SVM mampu melakukan generalisasi dengan cukup baik bahkan pada jumlah data pelatihan yang terbatas.

REFERENSI

Burges, Christopher J.C., 1998, *A Tutorial on Support Vector Machines for Pattern Recognition*, Data Mining and Knowledge Discovery, 2, pp. 121 – 167.

Byun, H. and Lee, S.W., 2003, *A Survey on Pattern Recognition Applications of Support Vector Machines*, International Journal of Pattern Recognition and Artificial Intelligence, Vol 17, No 3, pp. 459 – 486.

Cortes, C. and Vapnik, V., 1995, *Support-Vector Networks*, Machine Learning, Vol 20, No 3, pp 273 – 297.

Fletcher, T, 2008, *Support Vector Machines Explained*, available: <http://www.tristanfletcher.co.uk/SVM%20Explained.pdf>, [2009, 18 Februari].

Fradkin, D. and Muchnik, I., 2006, *Support Vector Machines for Classification*, in Abello, J. and Carmode, G. (Eds), *Discrete Methods in Epidemiology*, DIMACS Series in Discrete Mathematics and Theoretical Computer Science, 70, pp. 13-20, available: <http://paul.rutgers.edu/~dfradkin/papers/svm.pdf>, [2009, 18 Februari].

Nugroho, A.S., Witarto, A.B., dan Handoko, D., 2003, *Support Vector Machine – Teori dan Aplikasinya dalam Bioinformatika*, Kuliah Umum IlmuKomputer.com, available: www.ilmukomputer.com [2009, 23 Februari].

Oliveira, L.S. and Sabourin R., 2004, *Support Vector Machines for Handwritten Numerical String Recognition*, Proceedings of the 9th

- International Workshop on Frontiers in Handwriting Recognition (IWFHR-9), available: <http://www.livia.etsmtl.ca/people/soares/download/iwfhf-svm.pdf>, [2009, 25 Februari].
- Vijayakumar, S. and Wu, S., 1999, *Sequential Support Vector Classifiers and Regression*, Proc. International Conference on Soft Computing (SOCO '99), Genoa, Italy, pp. 610 – 619.
- Cover, T.M., 1965, Geometrical and Statistical, Properties of Systems of Linear Inequalities with Applications in Pattern Recognition, IEEE Transactions on Electronic Computers EC-14, 326-334.
- Haykin, Simon, 1994, *Neural Network: A Comprehensive Foundation*, Mcmillan College Publishing Company, Inc., New York, 237-242.
- Platt, J., Cristianini, N., and Shawe-Taylor, J., 2000, *Large Margin DAGs for Multiclass Classification*, Advances in Neural Information Processing Systems, 12, pp. 547 – 553.
- Sayogo, T. B., 2006a, *Komputerisasi Aksara Jawa Sebagai Media Pembelajaran Membaca dan Menulis Aksara Jawa*, disusun sebagai sumbangan pemikiran untuk peserta Konggres Bahasa Jawa IV Semarang 5-10 Juli 2006, Purwokerto, available: <http://hanacaraka.fateback.com>, [2006, 26 September].
- Sayogo, T. B., 2006b, Registrasi Unicode Aksara Jawa Kepentingan dan Keuntungannya : Sebuah wacana untuk persiapan pengajuan proposal resmi registrasi Unicode Aksara Jawa, disusun sebagai sumbangan pemikiran untuk peserta Konggres Bahasa Jawa IV Semarang 5-10 Juli 2006, Purwokerto, available: <http://hanacaraka.fateback.com>, [2006, 26 September].
- Widiarti, A.R., 2006, *Pengenalan Citra Dokumen Sastra Jawa Konsep dan Implementasinya*, Tesis S-2 Program Studi Ilmu Komputer Kelompok Bidang Studi Ilmu Matematika dan Pengetahuan Alam, Program Pascasarjana UGM, Yogyakarta *unpublished*.
- Asriani, F., Widhi Nugraha, A.W., dan Nuraeni, E., 2006, *Pengenalan Pola Aksara Jawa Dengan Jaringan Syaraf Tiruan Perambatan-Balik Sebagai Dasar Sistem Pengenalan Dokumen dengan Huruf Jawa*, Laporan Penelitian DIPA II tahun 2006, Program Sarjana Teknik, Universitas Jenderal Soedirman, Purwokerto *unpublished*.
- Nugraha, A.W.W., 2009, *Pengenalan Pola Tulisan Tangan Aksara Jawa Nglegeno dengan Multiclass SVM*, Tesis S-2 Teknik Elektro Program Pascasarjana UGM, Yogyakarta, *unpublished*.
- Gader, P.D., Mohammed, M., and Chiang, J.H., 1995, *Comparison of Crisp and Fuzzy Character Neural Networks in Handwritten Word Recognition*, IEEE Transactions on Fuzzy Systems, Vol 3, No. 3, pp. 357 – 363.
- Eaton, John W., 2002, *GNU Octave Manual*, Network Theory Limited.