KLASIFIKASI CITRA DENGAN SUPPORT VECTOR MACHINE PADA SISTEM TEMU KEMBALI CITRA

Yeni Herdiyeni, Agus Buono, Vita Yulia Noorniawati
Departemen Ilmu Komputer, Fakultas matematika dan Ilmu Pengetahuan Alam
Institut Pertanian Bogor, Kampus Dramaga. Wing 20 Level V, Bogor, Jawa Barat, Indonesia
yeni_herdi@ipb.ac.id

ABSTRACT
This paper presents a Support Vector Machine (SVM) algorithm for image classification in Content Based Image Retrieval (CBIR). Expectation Maximization is applied to image segmentation. The classification is based on color image. The algorithm is optimized using Sequential Minimal Optimization (SMO). This research is also compared to the Euclidean distance for measurement image similarity. The experiment on Caltech Database and images from www.flower.vg shows that the algorithm can improve the performance. The average precision of the experiment using SVM and SMO is 76,76% while Euclidean distance is 50.91%. This algorithm is a promising algorithm to be used for Content Based Image Retrieval (CBIR).
***Keywords**:* SVM, S*equential Minimal Optimization. EM Algorithm, CBIR*

1. Pendahuluan

Seiring dengan banyaknya aplikasi multimedia dan perkembangan internet, jumlah citra yang dikelola meningkat secara tajam. Para pengguna internet sangat mudah untuk mengakses ratusan bahkan ribuan citra, akan tetapi seringkali tidak mudah mendapatkan citra-citra yang sesuai dengan yang dibutuhkan pengguna. Oleh karena itu, perlu dikembangkan metode pencarian citra untuk mempermudah pencarian data. Pencarian citra dapat dilakukan berdasarkan isi citra atau sering disebut *content-based image retrieval* (CBIR). CBIR ini mencari citra dengan mencocokkan *content*-nya yang berupa tekstur, bentuk, atau warna.

Pada temu kembali citra yang hanya berdasarkan jarak *Euclidean* antar citra, citra yang ditemukembalikan adalah citra di dalam basis data yang mempunyai tingkat kemiripan yang tinggi secara visual dengan citra kueri. Banyak kemungkinan citra yang ditemukembalikan adalah citra yang bukan dari jenisnya sendiri. Hal ini dapat menyebabkan tingkat keefektifan temu kembali menjadi kurang baik. Klasifikasi citra merupakan salah satu cara untuk memperbaiki tingkat keefektifan hasil temu kembali citra.

Platt[6] telah menggunakan algoritma *Sequential Minimal Optimization* (SMO) untuk proses pelatihan dalam metode *Support Vector Machine* (SVM) dengan waktu komputasi yang lebih cepat. Zhang *et al.* [8] telah menggunakan metode SVM untuk klasifikasi pada temu kembali citra berdasarkan ciri warna. Berdasarkan hasil penelitian Gosselin dan Cord[4], metode SVM memberikan hasil klasifikasi yang lebih baik dibandingkan dengan metode klasifikasi *Bayes* dan *k-Nearest Neighbors* (kNN). Penelitian ini mengimplementasikan algoritme SVM dan SMO untuk klasifikasi warna pada sistem temu kembali citra.

Susunan makalah ini terdiri dari Bab 2 menjelaskan proses klasifikasi citra dengan SVM, Bab 3 menjelaskan percobaan, Bab 4 menjelaskan hasil percobaan dan Bab 5 menjelaskan kesimpulan penelitian.

2. Klasifikasi Citra

2.1 Segmentasi dengan *Expectation-Maximization*

Tahap pertama di dalam penelitian ini adalah melakukan segmentasi terhadap seluruh citra di dalam basis data. Algoritma segmentasi yang digunakan adalah *Expectation-Maximization* (EM)[2]. Setiap citra akan disegmentasi untuk mengelompokkan warna yang dikandung oleh setiap piksel dari citra ke beberapa segmen yang sudah ditentukan jumlahnya. Segmen ini merupakan representasi dari warna-warna dominan citra. Peluang sebuah piksel masuk ke dalam segmen adalah sebagai berikut:

$$p(x \mid \Theta) = \sum_{l=1}^{g} p(x \mid \theta_l)\pi_l$$

Masing-masing segmen diasumsikan mempunyai distribusi normal *Gauss*, sehingga peluang piksel dari segmen ℓ dapat dihitung dengan formulasi sebagai berikut:

$$p(x \mid \theta_l) = \frac{1}{(2\pi)^{\frac{d}{2}} \det(\Sigma_l)^{\frac{1}{2}}} \exp\left\{-\frac{1}{2}(x-\mu_l)^T \Sigma_l^{-1}(x-\mu_l)\right\}$$

2.2 Ekstraksi dengan Fuzzy Color Histogram

2.2.1 Fuzzy C-Means Clustering

Fuzzy C-Means Clustering digunakan untuk mengkuantisasi warna. Kuantisasi warna dilakukan dengan memetakan seluruh warna piksel hasil segmentasi ke *n bin* histogram. Kuantisasi ini didasarkan pada distribusi warna citra di dalam basis data. Terdapat 10 kelas citra, untuk tiap kelas diambil 10 warna piksel yang muncul terbanyak sehingga dihasilkan 100 warna tanpa ada warna yang sama. Jumlah *bin* yang terlalu banyak membuat perbedaan warna antar dua *bin* yang berdekatan terlalu kecil. Jumlah cluster optimal yang diperoleh pada penelitian yaitu 25 cluster.

2.2.2 Fungsi *Cauchy*

Fungsi *Cauchy* $C : R^n \rightarrow [0,1]$ didefinisikan sebagai berikut [5]:

$$C(\vec{x}) = \frac{1}{1 + \left(\|\vec{x} - \vec{v}\| / d\right)^{\alpha}}, \quad \vec{v} \in R; d, \alpha \in R; d > 0, \alpha \geq 0$$

dengan $\vec{v}$ adalah titik tengah dari lokasi *fuzzy set*, *d* merepresentasikan lebar dari fungsi dan *α* mendiskripsikan tingkat *fuzziness* (kekaburan). Derajat keanggotaan dinyatakan dengan fungsi *Cauchy* yaitu:

$$\mu_{c'}(c) = \frac{1}{1 + (d(c', c) / \sigma)^{\alpha}},$$

dengan d(c’,c) adalah jarak *Euclid*ean antara warna c dengan c’, c’ adalah warna pada *bin* FCH, c adalah semesta warna (*universe color*), α digunakan untuk menentukan bentuk atau kehalusan dari fungsi, dan σ untuk menentukan lebar dari fungsi keanggotaan. Nilai parameter $\alpha = 2$ dan $\sigma = 15$ [1]. Matriks derajat keanggotaan ini hanya dihitung sekali untuk setiap warna kuantisasi FCH. Di dalam proses ini, diperoleh 25×100 matriks derajat keanggotaan.

2.2.3 *Fuzzy Color Histogram*

Fuzzy Color Histogram (FCH) adalah salah satu metode yang merepresentasikan informasi warna dalam citra digital ke dalam bentuk histogram. Hubungan antar warna dimodelkan dengan fungsi keanggotaan (*membership function*) terhadap *fuzzy set*. *Fuzzy set F* pada ruang ciri *Rn* didefinisikan oleh $\eta F : Rn \rightarrow [0,1]$ yang biasa disebut *membership function.* Untuk tiap vektor ciri $f \in \Re^n$, nilai dari $\eta_F(f)$ disebut derajat keanggotaan dari *f* terhadap *fuzzy set F* (derajat keanggotaan *F*). Nilai dari $\eta_F(f)$ yang mendekati 1 lebih representatif terhadap vektor ciri *f* dan terhadap *fuzzy set f* [8].

Pada FCH, setiap warna *C* yang ditemukan pada citra akan mempengaruhi seluruh kuantisasi warna berdasarkan kemiripan warna dengan warna *C* tersebut. Hal ini dapat dinotasikan dengan:

$$h_2(c') = \sum_{c \in \mu} \mu_{c'}(c) * h(c),$$

dimana $h(c)$ adalah histogram warna, dan $\mu_{c'}(c)$ adalah nilai keanggotaan dari warna *c* ke warna *c’*.

2.3 *Support Vector Machine*

Support Vector Machine (SVM) adalah salah satu teknik klasifikasi data dengan proses pelatihan (*supervised learning*). Salah satu ciri dari metode klasifikasi SVM adalah menemukan *hyperplane* terbaik sehingga diperoleh ukuran *margin* yang maksimal. *Margin* adalah jarak antara *hyperplane* tersebut dengan titik terdekat dari masing-masing kelas. Titik yang paling dekat ini disebut dengan *support vector*. Ilustrasi SVM untuk *linear separable data* dapat dilihat pada Gambar 1.

Diberikan data pelatihan $(x_1, y_1), (x_2, y_2), ..., (x_n, y_n)$, dimana $x \in \Re^n$, $y \in \{+1, -1\}$. Jika data terpisah secara *linear* seperti pada Gambar 1, maka akan berlaku fungsi diskriminan *linear*:

$$u = w.x - b,$$

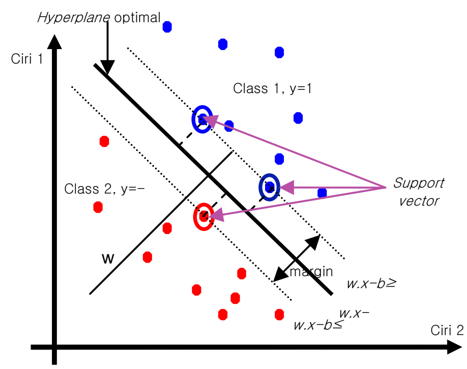
dimana *w* adalah vektor bobot normal terhadap garis pemisah (*hyperplane*), *x* adalah data yang diklasifikasi, dan *b* adalah bias. *Hyperplane* adalah garis *u*=0. *Margin* antara dua kelas adalah $m = \frac{2}{\|w\|_2}$. *Margin* dapat dimaksimalkan dengan menggunakan fungsi optimisasi *Lagrangian* seperti berikut:

$$L(\vec{w}, b, \alpha) = \frac{1}{2}\|\vec{w}\|^2 - \sum_{i=1}^{l} \alpha_i \left(y_i \left((\vec{x}_i, \vec{w} + b) - 1\right)\right)$$

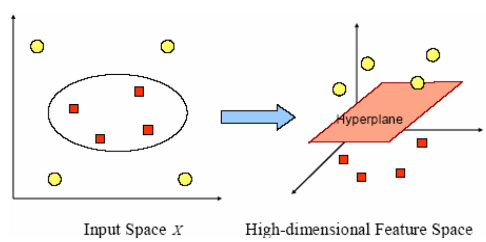
Dengan memperhatikan sifat gradien: $\frac{\partial L}{\partial w} = w - \sum_{i=1}^{l} \alpha_i y_i x_i = 0$ dan $\frac{\partial L}{\partial b} = \sum_{i=1}^{l} \alpha_i y_i = 0$, persamaan *Lagrangian* dapat dimodifikasi sebagai maksimalisasi L yang hanya mengandung α_i, sebagai berikut:

$$L = \sum_{i=1}^{l} \alpha_i - \frac{1}{2} \sum_{i=1}^{l} \sum_{j=1}^{l} \alpha_i \alpha_j y_i y_j (\vec{x}_i . \vec{x}_j)$$

dengan $0 \le \alpha_i \le C, (i = 1,2,...,l), \sum_{i=1}^{l} \alpha_i y_i = 0$, dan α_i adalah *lagrange multiplier*. Data yang berkorelasi dengan α_i yang positif disebut sebagai *support vector*.



Gambar 1. Data terpisah secara *linear*



Gambar 2. Fungsi *Kernel* memetakan data ke ruang vektor berdimensi lebih tinggi.

Jika data terpisah secara *non-linear*, maka data terlebih dahulu diproyeksikan oleh fungsi *Kernel* ke ruang vektor baru yang berdimensi tinggi sedemikian sehingga data itu dapat terpisah secara linear, seperti pada Gambar 2. Selanjutnya di ruang vektor yang baru itu, SVM mencari *hyperplane* yang memisahkan kedua kelas. Pencarian ini hanya bergantung pada *dot product* dari data yang sudah ditransformasikan pada ruang baru yang berdimenasi lebih tinggi, yaitu $\Phi(\vec{x}_i).\Phi(\vec{x}_j)$. Fungsi *Kernel* dirumuskan sebagai berikut: $K(\vec{x}_i, \vec{x}_j) = \Phi(\vec{x}_i).\Phi(\vec{x}_j)$, sehingga persamaan *Lagrangian* menjadi seperti berikut:

$$L = \sum_{i=1}^{l} \alpha_i - \frac{1}{2} \sum_{i=1}^{l} \sum_{j=1}^{l} \alpha_i \alpha_j y_i y_j K(\vec{x}_i . \vec{x}_j)$$

dan persamaan fungsi diskriminan menjadi seperti berikut:

$$u = \sum_{i=1}^{nsv} \alpha_i y_i K(\vec{x}_i, \vec{x}) - b$$

dengan NSV adalah data pelatihan yang termasuk *support vector*. Fungsi *Kernel* yang digunakan untuk SVM adalah fungsi *Kernel Gaussian RBF* [8]:

$$K(\vec{x}_i, \vec{x}_j) = \exp\left(- \frac{\left\| \vec{x}_i - \vec{x}_j \right\|^2}{2\sigma^2} \right)$$

Untuk mengoptimalkan proses pelatihan SVM, penelitian ini menggunakan *Sequential Minimal Optimization* (SMO)[5]. SMO akan memilih dua *lagrange multipliers* α_i untuk dioptimisasi bersama-sama, mencari nilai yang paling optimal untuk *lagrange multiplier* tersebut, dan memperbaharui SVM dengan nilai optimal yang baru.

3. Percobaan

K-Fold Cross Validation dilakukan untuk membagi data pelatihan dan data pengujian. *K-Fold Cross Validation* membagi data *sample* secara acak ke dalam *K* subset yang saling bebas. Satu subset sebagai data pengujian dan *K-1* subset sebagai data pelatihan. Proses *cross-validation* akan diulang sampai *K* kali[3].

Data yang digunakan diambil dari *database Caltech* dan www.flowerz.vg sebanyak 300 citra untuk berbagai macam objek. Format citra adalah JPEG berukuran 50×50 piksel. Sesuai dengan metode *10-fold cross validation*, seluruh data hasil ekstraksi ciri warna yang terdapat di dalam basis data dibagi menjadi 10 subset, yaitu $S_1,S_3,...,S_{10}$. Masing-masing subset memiliki ukuran yang sama. Pembagian data dilakukan secara acak dengan mempertahankan perbandingan jumlah citra setiap kelas. Pada proses pertama $S_2,...,S_{10}$ menjadi data pelatihan dan S_1 menjadi data pengujian, pada proses kedua $S_1,S_3,...,S_{10}$ menjadi data pelatihan dan S_2 menjadi data pengujian, dan seterusnya.

Proses pengujian dilakukan berdasarkan metode *10-fold cross validation*. Model klasifikasi dikatakan terbaik jika mencapai nilai akurasi yang tertinggi ketika diaplikasikan ke data uji dengan nilai *C* dan σ terbaik. Model klasifikasi inilah yang akan digunakan untuk menentukan hasil klasifikasi akhir.

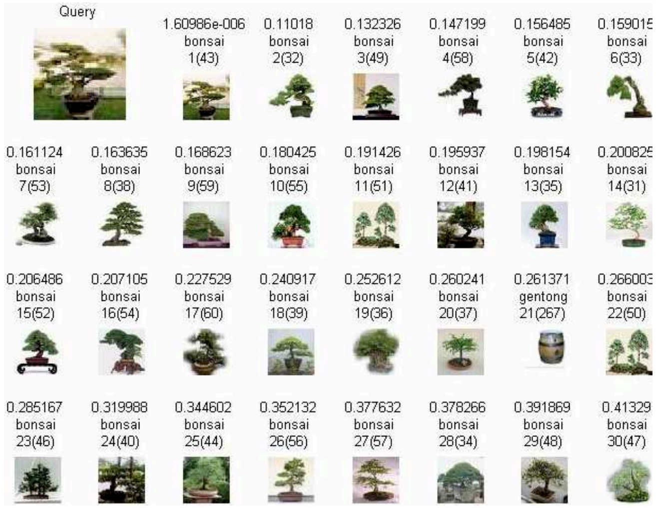
4. Hasil Percobaan

Berdasarkan hasil percobaan, klasifikasi citra dengan menggunakan SVM mencapai akurasi sebesar 86.67%. Hasil klasifikasi tersebut selanjutnya digunakan untuk penemuan kembali citra (*image retrieval*). Pada penelitian ini, temu kembali citra diujicobakan kedua metode yang berbeda untuk melihat perbedaan tingkat keefektifan hasil temu kembali citra. Dua metode ini adalah temu kembali citra tanpa menggunakan SVM dan temu kembali citra menggunakan SVM.

Hasil temu kembali citra tanpa menggunakan SVM hanya didasarkan pada kemiripan ciri warna menggunakan perhitungan jarak *Euclidean*. Contoh hasil temu kembali tanpa menggunakan SVM dapat dilihat pada Gambar 3.

Gambar 3. Contoh hasil temu kembali tanpa menggunakan SVM.

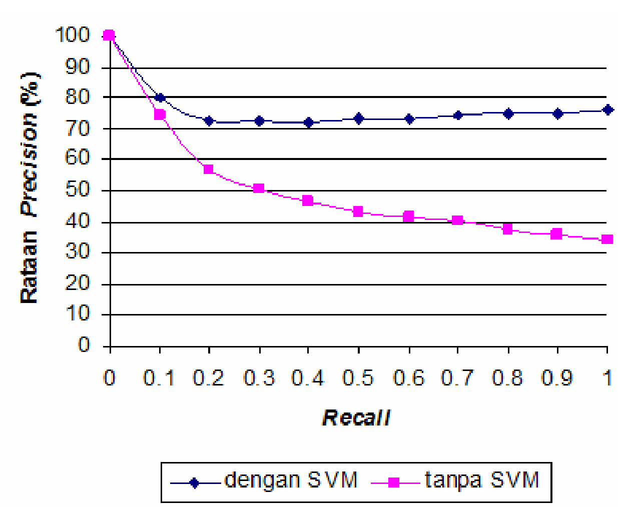
Gambar 3 memperlihatkaan bahwa citra hasil temu kembali tidak sepenuhnya berasal dari jenis citra yang sama dengan citra kueri. Citra-citra tersebut cenderung berwarna biru, berbeda sekali dengan warna citra kueri yang cenderung berwarna hijau kekuningan. Hal ini dikarenakan sistem hanya menemukembalikan citra di dalam basis data yang mempunyai tingkat kemiripan yang tinggi dengan citra kueri. Kekurangan sistem ini diperbaiki oleh temu kembali citra menggunakan SVM yang dapat dilihat pada Gambar 4.



Gambar 4. Contoh hasil temu kembali menggunakan SVM.

Berdasarkan Gambar 4, dapat dilihat bahwa citra hasil temu kembali dengan menggunakan SVM lebih baik Hal ini dikarenakan sistem mempunyai model klasifikasi untuk memprediksi baik citra kueri maupun seluruh citra di dalam basis data sehingga diperoleh kelas yang baru untuk setiap citra. Citra yang ditemukembalikan adalah citra hasil klasifikasi di dalam basis data yang terdapat dalam kelas citra yang sama dengan citra kueri dan citra dari kelas lain yang memiliki tingkat kemiripan yang tinggi dengan citra kueri.

Tingkat keefektifan temu kembali citra diukur dengan menghitung nilai *recall* dan *precision*. Berdasarkan Gambar 5, dapat dilihat bahwa nilai rataan *precision* temu kembali citra menggunakan SVM cenderung selalu lebih baik daripada nilai rataan *precision* temu kembali citra tanpa menggunakan SVM pada setiap nilai *recall*. Berdasarkan hasil percobaan, klasifikasi dengan SVM menghasilkan average precision sebesar 76,76% sedangkan tanpa menggunakan SVM mencapai 50,91%. Untuk temu kembali citra menggunakan SVM, dapat dilihat bahwa nilai rataan *precision* mengalami penurunan pada nilai *recall* 0.1, sedangkan pada nilai *recall* lainnya cenderung stabil. Hal ini karena pada peringkat *recall* 0.1 banyak kueri citra yang tidak menemukan citra yang relevan di dalam basis data. Untuk temu kembali citra tanpa menggunakan SVM, dapat dilihat bahwa nilai rataan *precision* cenderung selalu mengalami penurunan pada setiap nilai *recall*.



Gambar 5. Grafik rataan *precision* hasil temu kembali citra menggunakan SVM dan tanpa menggunakan SVM.

5. KESIMPULAN

Makalah ini mengimplementasikan algoritme *Support Vector Machine* (SVM) dan *Sequential Minimal Optimization* (SMO) untuk mengklasifikasikan citra pada sistem temu kembali citra. Penelitian menggunakan dua pendekatan berbeda untuk mengukur kemiripan citra (*image similarity measurement*). Dua pendekatan tersebut adalah temu kembali citra tanpa menggunakan SVM dan menggunakan SVM dengan metode optimasi SMO. Hasil percobaan menunjukkan bahwa rataan *precision* hasil temu kembali citra menggunakan SVM cenderung lebih baik dibandingkan tanpa menggunakan SVM untuk setiap nilai *recall*. Klasifikasi dengan SVM menghasilkan average precision sebesar 76,76% sedangkan tanpa menggunakan SVM mencapai 50,91%. Dengan demikian metode klasifikasi SVM dapat digunakan untuk sistem temu kembali citra.

DAFTAR PUSTAKA

[1] Balqis, DP. 2006. Metode Fuzzy Color Histogram untuk Temu Kembali Citra Bunga. [Skripsi]. Bogor:Fakultas Matematika dan Ilmu Pengetahuan Alam, Institut Pertanian Bogor.

[2] Belongie S, *et al.* 1998. *Color- and Texture-Based Image Segmentation Using EM and Its Application to Content-Based Image Retrieval*. Computer Science Division, University of California at Berkeley, Berkeley, CA 94720.

[3] Fu L. 1994. *Neural Network In Computer Intelligence*. Singapura : McGraw Hill.

[4] Gosselin PH, Cord M. 2004. *A Comparison of Active Clasification Methods for Content-Based Image Retrieval.* http://perso-etis.ensea.fr/~gosselin/gosselin04cvdb.pdf [28 Oktober 2006].

[5] Mak G. 2000. *The Implementation of Support Vector Machines Using The Sequential Minimal Optimization Algorithm.* http://www.cs.mcgill.ca/~hu/publications/99.04.McGill.thesis.gmak.pdf [28 Oktober 2006].

[6] Platt JC. 1998. *Sequential Minimal Optimization: A Fast Algorithm for Training Support Vector Machines.* http://reseach.microsoft.com/users/jplatt/smoTR.pdf [28 Oktober2006].

[7] Zang Lei, *et al.* 2001. *Support Vector Machine Learning for Image Retrieval*. http://research.microsoft.com/users/leizhang/Paper/ICIP01.pdf [28 Oktober 2006].

[8] Zhang R, Zhang Z. *A Robust Color Object Analysis Approach to Efficient Image Retrieval*. EURASIP Journal on Applied Signal Processing, 6, 2004, 871-885.