

Klasifikasi Bentuk Lengkung Gigi Menggunakan Algoritma Propagasi Balik Berdasarkan Fitur Tekstur Pada Citra Digital

Shinta Puspasari

Program Studi Teknik Informatika
STMIK Global Informatika MDP
Jl. Rajawali 14, Palembang , 30113
email: shinta@mdp.ac.id

Abstrak— Tulisan ini menyajikan model jaringan saraf tiruan yang digunakan untuk klasifikasi bentuk lengkung gigi setelah diberikan perawatan ortodontik. Arsitektur jaringan saraf tiruan dengan algoritma pembelajaran propagasi balik yang dirancang untuk menerima parameter input berupa empat parameter lengkung gigi dan fitur tekstur Haralick yang diambil dari citra cetakan gigi sebelum menjalani perawatan ortodontik. Usulan model dievaluasi kinerjanya menggunakan data set yang terdiri dari data numerik dan data citra cetakan gigi pasien ortodonti. Dalam hal efektivitas, hasil tes menunjukkan bahwa kinerja model jaringan saraf tiruan yang dirancang efektif mengklasifikasi secara akurat hingga 66%. Penelitian lebih lanjut diperlukan untuk menentukan fitur yang efektif untuk klasifikasi bentuk lengkung gigi sehingga dapat diimplementasikan dalam aplikasi pendukung perawatan ortodonti.

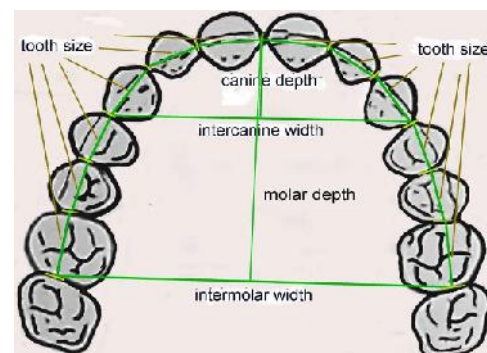
Keywords: algoritma propagasi balik, bentuk lengkung gigi, fitur tekstur

I. PENDAHULUAN

Secara alamiah, lengkung gigi manusia memiliki bentuk yang berbeda. Bentuk lengkung gigi dapat dikategorikan menjadi tiga, yaitu bulat telur, meruncing dan siku [1]. Gigi tersusun pada tulang rahang untuk membentuk struktur lengkung yang berbeda secara alami, dalam hal ukuran dan bentuk [2] dipengaruhi oleh bentuk lengkungan tulang gigi, erupsi, dan kerusakan gigi [3]. Banyak masalah yang dapat terjadi pada struktur gigi secara fungsional dan estetika. Semua masalah tersebut dapat diatasi dengan cara menjalani perawatan ortodontik. Perawatan terhadap ketidakrapian susunan geligi dilakukan oleh dokter gigi dengan spesialisasi orthodontik. Banyak contoh kasus yang ditangani oleh dokter gigi, masalah kurang teraturnya gigi yang terjadi karena ketidakseimbangan antara ukuran gigi dan ukuran rahang

sehingga gigi pasien tidak tertata dengan rapi. Perawatan ortodonti dilakukan untuk merapikan gigi berdasarkan diagnosa komposisi bentuk lengkung gigi [4]. Dalam menentukan bentuk lengkung gigi pasien dilakukan secara manual berdasarkan pengetahuan dan pengalaman dengan cara ortodontist melihat model bentuk cetakan gigi pasien. Suatu model cetakan gigi dibuat dengan biaya yang tidak murah serta membutuhkan waktu yang tidak singkat, dan hasil perkiraan yang diperoleh bisa saja salah karena faktor kelalaian manusia. Selain itu, komposisi bentuk lengkung gigi, yang bulat telur, meruncing, dan persegi, memiliki kesamaan yang sangat tinggi sehingga sulit untuk membedakan. Selain itu, dengan tidak ada pengaturan rapi gigi membuat bentuk melengkung menjadi semakin tidak jelas. Kesalahan dalam menentukan bentuk lengkung akan mempengaruhi keberhasilan dan stabilitas hasil perawatan ortodonti [5].

Studi dalam penelitian ini mencoba untuk membangun model jaringan syaraf tiruan untuk klasifikasi bentuk lengkung gigi pasien berdasarkan empat parameter lengkung gigi sebagai fitur input yang lebar inter, kedalaman Canine, lebar intermolar, dan kedalaman molar [6] dan Haralic fitur tekstur diekstraksi dari citra cetakan gigi.



Gambar 1. Parameter lengkung gigi [6]

Algoritma propagasi balik jaringan saraf tiruan yang telah terbukti efektivitasnya untuk klasifikasi [7] [8] akan digunakan untuk mengklasifikasikan bentuk lengkung gigi dari pasien

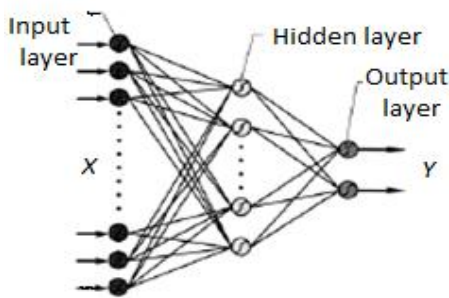
setelah pengobatan. Arsitektur ini dikenal untuk mengklasifikasikan bentuk yang didasarkan pada nilai masukan. Input akan diberikan pada data pelatihan, lebar inter, kedalaman Canine, lebar intermolar, dan kedalaman molar. Neural network dilatih dengan menggunakan 30 set data dan diuji dengan 50 set data. Output yang akan diperoleh adalah klasifikasi bentuk lengkung gigi setelah pasien menjalani perawatan ortodontik, apakah bentuknya bulat telur, meruncing, atau persegi.

II. JARINGAN SARAF TIRUAN

A. Arsitektur Jaringan Saraf Tiruan

Penelitian ini menggunakan arsitektur algoritma propagasi balik dengan tiga lapisan. *Multilayers feed forward* NN dengan tiga lapisan yang cukup untuk digunakan dalam klasifikasi, di mana satu lapisan tersembunyi cukup baik untuk perkiraan fungsi kontinu [9]. Dalam [10] dinyatakan bahwa pengenalan pola, jika jumlah neuron pada lapisan tersembunyi kurang atau sama dengan jumlah neuron pada lapisan masukan, NN tidak cukup baik untuk menentukan klasifikasi kendala output. *Node* dalam lapisan tersembunyi dapat ditambahkan satu per satu dalam percobaan [11] uji coba dilakukan pengaturan jumlah *node* dalam setiap lapisan sehingga dapat menghasilkan NN dengan kinerja yang optimal khususnya untuk klasifikasi.

Input *layer* menerima nilai parameter masukan berdasarkan hasil dari proses pemilihan dan ekstraksi fitur. *Node* dari lapisan berikutnya akan menghasilkan output yang dikirim sebagai input ke *node* dalam lapisan tersembunyi. Pada lapisan tersembunyi, nilai masukan akan berubah dan hasilnya dikirimkan kembali sebagai input ke lapisan output dari NN. *Multilayers feed forward* NN yang dirancang untuk aplikasi klasifikasi dapat diilustrasikan seperti pada Gambar 3 di bawah ini [7],



Gambar 2. Arsitektur *Multilayers feed forward*

B. Algoritma Pembelajaran

Proses pelatihan NN dengan algoritma propagasi balik membutuhkan tiga tahap [12], input data untuk pelatihan, update kesalahan dan penyesuaian nilai bobot dari setiap node setiap lapisan NN. Dimulai dengan umpan maju nilai-nilai masukan, setiap unit masukan-*i* (x_i) menerima sinyal untuk

input kemudian akan diteruskan ke lapisan tersembunyi Z_1, \dots, Z_p .

$$Z_j = \theta_{1j} + \sum_i x_i v_{ij} \tag{1}$$

dan

$$z_j = f(Z_j) \tag{2}$$

dimana θ_{1j} = error hidden unit. Nilai bias dan bobot awal dapat diambil secara acak. Setiap unit keluaran ke-*k* atau (Y_k) dan kemudian menghitung nilai fungsi aktivasi,

$$Y_k = \theta_{2k} + \sum_j z_k w_{jk} \tag{3}$$

dan

$$y_k = f(Y_k) \tag{4}$$

dimana θ_{2k} = error hidden unit ke-*k*.

Selanjutnya node tersembunyi ke-*j* akan menghitung besar nilai sinyal (Z_j), yang akan dikirim ke lapisan output, menggunakan fungsi aktivasi *f* berikut,

C. Fungsi Aktifasi

Karakteristik yang harus dimiliki oleh fungsi aktivasi, yang terus-menerus, dan monoton menurun terdiferensialkan. Terdapat sejumlah fungsi aktivasi yang dapat digunakan dalam arsitektur jaringan saraf tiruan khususnya algoritma propagasi balik untuk klasifikasi. Fungsi aktivasi yang sering digunakan salah satunya adalah fungsi sigmoid dengan rentang (0,1),

$$f(x) = \frac{1}{1 + \exp(-x)} \tag{5}$$

dan banyak lagi fungsi lain.

D. Fitur Tekstur Haralick's

Pada tahun 1973 Haralick memperkenalkan co-terjadinya matriks dan tekstur fitur untuk klasifikasi otomatis batuan menjadi enam kategori [13]. Hari ini, fitur ini banyak digunakan untuk berbagai jenis gambar, misalnya, untuk gambar mikroskop sel-sel biologis. Tekstur Fitur Haralick terdiri 14 fitur. Implementasi penelitian mengoptimalkan Tekstur Fitur persamaan pertama 13 Haralick ini (6) sampai (18) dan tidak menghitung Fitur nomor 14 (Koefisien Korelasi Maksimum) [13],

$$f_1 = \sum_{i=1}^N \sum_{j=1}^N P(i, j)^2 \tag{6}$$

$$f_2 = \sum_{k=0}^{N-1} k^2 \left(\sum_{i=1}^N \sum_{j=1}^N P(i, j) \right)^{|i-j|=k} \tag{7}$$

$$f_3 = \frac{1}{2} \sum_{i=1}^N \sum_{j=1}^N (ij) P(i, j) - \sim^2 \tag{8}$$

$$f_4 = \sum_{i=1}^N \sum_{j=1}^N (i - \sim^2) P(i, j) \tag{9}$$

$$f_5 = \sum_{i=1}^N \sum_{j=1}^N \frac{P(i, j)}{1 + (1 - j)^2} \tag{10}$$

$$f_6 = \sum_{k=0}^{2N-2} k P_{x+y}(k) \tag{11}$$

$$f_7 = \sum_{k=0}^{2N-2} (k - f_6)^2 P_{x+y}(k) \tag{12}$$

$$f_8 = \sum_{k=0}^{2N-2} P_{x+y}(k) \log[P_{x+y}(k)] \tag{13}$$

$$f_9 = - \sum_{i=1}^N \sum_{j=1}^N P_{(i,j)} \log[P_{(i,j)}] \tag{14}$$

$$f_{10} = \sum_{k=0}^{N-1} [P_{x-y}(k) (k - \sum_{i=0}^{N-1} i P_{x-y}(k))^2] \tag{15}$$

$$f_{11} = - \sum_{k=0}^{N-1} P_{|x-y|}(k) \log[P_{|x-y|}(k)] \tag{16}$$

$$f_{12} = \frac{f_9 - HXY1}{H} \tag{17}$$

$$f_{13} = \sqrt{1 - \exp(-2 | HXY2 - f_9 |)} \tag{18}$$

dimana masing-masing variabel dari fitur tekstur Haralick (f_i) di atas merujuk pada properti Matrik *co-occurrence* dari citra model cetakan gigi.

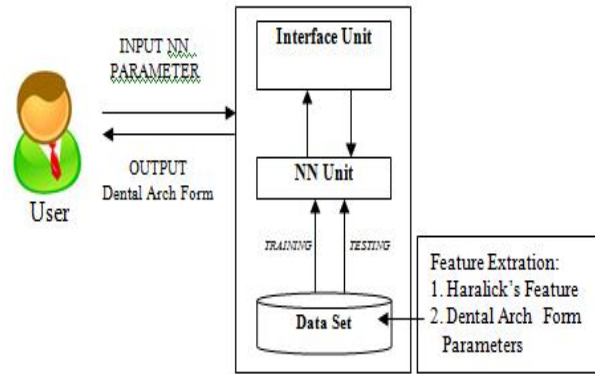
III. HASIL UJI COBA

Percobaan yang dilakukan menggunakan data set berisi 50 (lima puluh) sampel gambar dan fitur numerik yang dilakukan pada model gambar rahang atas cetakan gigi sebelum dan sesudah perlakuan perawatan ortodonti diperoleh dengan melakukan pemindaian terhadap model 3D asli gigi cetakan (Gbr.3). Ekstraksi gambar tekstur dilakukan langsung pada model gambar yang dicetak. Parameter lengkung gigi diukur, lebar yaitu intercanine, kedalaman canine, lebar inter, dan kedalaman molar



Gambar 3. Sampel citra uji coba

Untuk meminimalisasi kesalahan dalam pengukuran, proses pengukuran dilakukan tiga kali, dan kemudian mengambil nilai rata-rata untuk selanjutnya digunakan sebagai data sampel. Dalam perkembangannya, sistem pengujian dirancang menggunakan MATLAB 9.0. Berikut adalah rencana untuk sistem yang digunakan untuk pengujian,



Gambar 4. Rancangan Arsitektur Sistem

Dari Gambar 4 di atas, dapat dilihat bahwa ada dua unit dalam sistem, yang masing-masing memiliki peran sebagai berikut:

a. Unit Antar Muka

Untuk input dan output dari pengguna berupa sejumlah nilai yang diterima dalam bentuk bilangan real (Gbr.5). Unit kemudian akan mengirimkan nilai yang telah diterima ke dalam unit NN. Unit antar muka, juga, yang akan menampilkan output dari NN. Melalui unit ini pengguna juga dapat menentukan tingkat pembelajaran dan jumlah *epoch* dan pelatihan data untuk NN.



Gambar 5. GUI Pelatihan NN

b. Unit Jaringan Saraf (NN Unit)

Berperan dalam pengolahan data. Algoritma propagasi balik diterapkan di unit ini. Database yang digunakan oleh unit lengkung gigi untuk pelatihan dan pengujian NN. Selanjutnya masukan yang diterima akan diproses dalam jaringan saraf untuk mendapatkan output dari bentuk lengkung gigi. Hasil akhir yang kemudian dikirim ke unit antarmuka yang akan ditampilkan kepada pengguna..

Arsitektur propagasi balik dirancang dalam percobaan ini terdiri dari tiga lapisan. Lapisan input terdiri dari enam unit, sedangkan lapisan output terdiri dari tiga unit di mana fungsi sigmoid digunakan sebagai fungsi aktivasi di semua lapisan dalam NN. Selanjutnya, percobaan dilanjutkan dengan pelatihan NN dengan 30 data JST dilatih menggunakan sejumlah *node* tersembunyi laju belajar yang berbeda. Hal ini dilakukan dalam rangka untuk menemukan arsitektur dengan hasil terbaik dari sejumlah arsitektur yang mencoba untuk membedakan jumlah *node* pada lapisan tersembunyi.

Indikator yang digunakan untuk mengukur keberhasilan arsitektur dalam penelitian ini adalah persentase pengenalan oleh NN. Nilai kesalahan dan bobot awal untuk algoritma propagasi balik yang diambil secara acak, sedangkan bobot akhir pada lapisan output digunakan untuk menentukan bentuk lengkung gigi apakah bulat telur, meruncing atau persegi. Setelah tahapan uji coba, kinerja yang didapat dengan menerapkan arsitektur NN hasil perancangan diperoleh hasil optimal yang terdiri dari sejumlah parameter dengan nilai-nilai seperti yang digambarkan dalam Tabel I,

TABEL I. TOPOLOGI OPTIMAL JARINGAN SARAF TIRUAN

Parameter	Nilai
Jumlah Input Node	17
Jumlah Output Node	3
Jumlah Hidden Node	8
Jumlah Hidden layer	1
Learning Rate	0.1
Epoch	1000
Maximum Error	0.1

Pengujian kinerja dari NN telah dilakukan sebanyak 10 kali untuk melihat stabilitas hasil yang diperoleh. Selama pengujian ada perbedaan dalam nilai persentase klasifikasi sukses lengkung oleh NN. Kondisi tersebut terjadi karena penentuan nilai acak dari sejumlah parameter dilakukan oleh MATLAB setiap kali proses pelatihan dan pengujian.

Hasil pengujian menunjukkan jumlah yang berbeda untuk tiap jumlah simpul pada lapisan tersembunyi, hasil lengkap disajikan pada Tabel II. Dari tabel tersebut diketahui bahwa efektivitas NN untuk memprediksi bentuk lengkung gigi setelah perawatan ortodontik menggunakan 17 fitur gigi cetakan hingga 66% dan 60,4% untuk rata-rata pengenalan. Kesalahan klasifikasi occurs ketika bentuk-bentuk persegi dan meruncing diklasifikasikan sebagai bulat telur. Selain itu, sejumlah kecil data sampel untuk membentuk meruncing dan persegi dapat menjadi faktor yang mempengaruhi hasil klasifikasi.

TABLE II. PERSETASE HASIL KLASIFIKASI

10 Nodes	8 Nodes	5 Nodes
66	64	62

66	64	60
64	62	60
62	62	58
58	60	58
54	60	58
52	60	54
52	58	54
52	58	52
52	56	52

IV. KESIMPULAN

Model jaringan saraf dalam penelitian yang disajikan hasilnya dalam tulisan ini dirancang untuk dapat mengenali ketiga bentuk lengkung gigi, yang bulat telur, meruncing, dan bentuk persegi yang memiliki kesamaan yang sangat tinggi. Sejumlah parameter NN ditetapkan, sehingga sistem aplikasi dapat memiliki hit kinerja yang optimal. Parameter masukan NN untuk dipilih berdasarkan hasil uji statistik pada penelitian sebelumnya untuk menentukan parameter apa yang mempengaruhi bentuk lengkung gigi. Dari hasil pengujian, empat parameter yang mempengaruhi bentuk lengkung gigi, yaitu lebar intercanine, lebar intermolar, tinggi canine, dan tinggi intermolar. Algoritma propagasi balik yang telah dirancang dan dilatih serta diuji menggunakan sampel data hasil numerik dari model pengukuran citra gigi rahang atas dan gambar cetakan gigi yang diekstrak fitur tekstur Haralick-nya dan selanjutnya digunakan pada tahapan pembelajaran jaringan saraf tiruan yang telah dirancang.

Hasil percobaan menunjukkan bahwa efektivitas model NN diusulkan hingga 66% dan 60,4% rata-rata memprediksi bentuk lengkung gigi setelah perawatan ortodontik akurat. Namun, masih ada sebesar 34% salah klasifikasi oleh model NN yang dirancang. Hal ini karena kemiripan tinggi antara tiga bentuk, yang diklasifikasikan sebagai persegi, meruncing, dan bulat telur. Kinerja dari model jaringan saraf masih perlu ditingkatkan untuk dapat diterapkan di dunia nyata untuk mendukung perawatan ortodonti efisien. Penelitian lebih lanjut diperlukan untuk menentukan fitur yang efektif menentukan bentuk lengkung gigi.

REFERENSI

- [1] H.Naroozi, T.H.Nik and R.Saeeda, "The Dental Arch Form Revisited", *Angle Orthodontist*, Vol. 71, No. 5, 2001, hal 386-389
- [2] R.McLaughlin and J. Bennett, "Arch Form Considerations for Stability and Esthetics", *Rev. Esp. Orthod*, Vol.29, No.2, hal.46-63, 1999.
- [3] S.,Stanley; W.Hnat, D.Fender; H. Legan, "The Form of The Human Dental Arch", *The Angel Orthodontist*, Vol.68, No.1, hal. 29-36, 1998.
- [4] Y.A.Ko, K.Nojima, H.B.Moon, R. McLaughlin, P.Sinclair, "Comparison of Arch Forms Between Korean and North American White Populations", *American Journal of Orthodontics and Dentofacial Orthopedics*, Vol.126, No.6, hal. 680-686, 2004.
- [5] T.Taner, S.Ciger, H. El, D.Germec, A. Es, "Evaluation of Dental Arch Width and Form Changes After Orthodontic Treatment and Retention with a New Computerized Method", *American Journal of Orthodontics and Dentofacial Orthopedics*, Vol.126, NO.4, hal. 464-476, 2004.

- [6] J.A. Budiman, "Neural Network in developing Software for Identifying Arch Form", *International Journal of Artificial Intelligence & Applications*, Vo.4. No. 3, May 2013.
- [7] K. Mumtaz, S.S. Sherriff, and K. Duraiswamy, " Evaluation of Neural Network based Classification System for Clinical Cancer Data Classification", *Journal of Computer Applications*, Vo.1. No. 4, 2008.
- [8] G.P. Zhang, " Neural Network for Classification: A Survey", *IEEE Transaction on System, Man, and Cybernetics*, Vo.1.30 No. 4, 2000.
- [9] Hertz,J; Krogh.A; Palmer,R.G; *Introduction to The Theory of Neural Computation*, Addison Wesley, 1991.
- [10] M. Gori, and F. Scarselli, "Are Multilapisan Perceptrons Adequate for Pattern Recognition and Verification ?", *IEEE Transaction on Pattern Analysis and Machine Intelligence*, Vol. 20, No. 11, hal 1121-1131,1998.
- [11] O. Fujita.; "Statistical Estimation of the Number of Hidden Units for Feedward Neural Networks", *IEEE Transactions on Neural Networks*, Vol. 11, No.5, hal 851-860, 1998.
- [12] L. Fausett.; *Fundamental Of Neural Networks: Architectures, Algorithms, Application*, Prentice Hall, 1994.
- [13] R. M. Haralick and K. Shanmugam, "Computer Classification of Reservoir Sandstones," *IEEE Transactions on Geoscience Electronics*, vol. 11, pp. 171-177, 1973.
- [14] M. Gipp, G.Marcus, N. Harder, A. Suratane, K. Rohr, R. Konig, R. Manner, " Haralick's Textures Computed by GPUs for Biological Applications" *IAENG International Journal Of Computer Science*, 36:1, 2009.

Halaman Ini Sengaja Dikosongkan