



SISTEM PENDETEKSI POLA IDZHAR PADA CITRA AL-QUR'AN MENGUNAKAN *LEARNING VEKTOR QUANTIZATION (LVQ)*

Fadliyah

Teknik Informatika Universitas Malikussaleh Lhokseumawe
Jl. Kampus Unimal BI Blang Pulo, Aceh Utara, Indonesia

Email: ffadliyah@gmail.com

ABSTRAK

Hukum bacaan pada citra Qur'an secara visual membentuk berbagai pola. Salah satu hukum bacaan yang dipilih untuk dijadikan objek dalam penelitian ini adalah pola idzhar. Pola idzhar terbentuk apabila pola nun mati (نْ) atau tanwin (ـن , ـان , ـان) bertemu dengan huruf hamzah (ء), kha (خ), ha (ح), ain (ع), ghain (غ), dan ha' (هـ). Pada penelitian ini, pola idzhar akan dideteksi melalui teknik-teknik pengolahan citra dan selanjutnya pengukuran kelas pola dilakukan dengan menggunakan metode jaringan *Learning Vektor Quantization (LVQ)*. Hasil penelitian menunjukkan bahwa sistem pendeteksi pola menggunakan pendekatan jaringan LVQ memiliki detection rate berkisar dari 65% hingga 90%. Persentase detection rate tersebut menunjukkan bahwa jaringan LVQ dapat digunakan sebagai salah satu pendekatan untuk pendeteksian pola-pola yang sejenis. Keunggulan jaringan LVQ adalah kemampuan pengenalan pola dengan jumlah data pelatihan yang tidak terlalu besar, sangat tepat untuk diterapkan pada sistem.

Kata kunci: *Learning Vektor Quantization (LVQ)*.

Pendahuluan

Ilmu tajwid merupakan bagian dari ilmu ulumul Quran yang perlu dipelajari, mengingat ilmu ini berkaitan dengan bagaimana seseorang dapat membaca Al – Quran dengan baik. Tajwid (تَجْوِيدٌ) merupakan bentuk masdar, berakar dari fiil madhi (جَوَّدَ) yang berarti “membaguskan“. Ilmu tajwid adalah ilmu yang berguna untuk mengetahui bagaimana cara melafalkan huruf yang benar dan di benarkan, baik berkaitan dengan sifat, mad, dan sebagainya, misalnya Tarqiq, Tafhim dan selain keduanya. Ada berbagai macam hukum bacaan tajwid dalam Al-Qur'an. Di antaranya hukum nun sukun dan tanwin, hukum mim sukun, hukum mim dan nun bertasydid, hukum mad, hukum idgham shaghir, dan qalqalah. Hukum nun sukun dan tanwin terdiri dari idzhar halqi, idgham bighunnah dan idgham bilaghunnah, iqlab, dan ikhfa hakiki. Kemudian, hukum mim sukun terdiri dari ikhfa syafawi, idgham mitslain, dan izhar syafawi. Selanjutnya, pada hukum mad terdiri dari mad ashli dan mad far'i, sedangkan hukum idgham shaghir terdiri dari idgham mutamatsilain, idgham mutajansain, dan idgham mutaqaribain. Sementara itu, hukum qalqalah terdiri dari qalqalah sugra dan qalqalah kubra [1].

Berbagai hukum-hukum bacaan yang dijelaskan di atas secara visual membentuk berbagai macam pola yang dapat dijadikan objek dalam penelitian di bidang

pengolahan citra [2]. Salah satu hukum bacaan yang dijadikan objek dalam penelitian ini adalah pola idzhar. idzhar adalah bayan atau jelas, sedangkan menurut istilah adalah membaca nun mati (نْ) atau tanwin (ـً, ـٍ, ـٌ) dengan jelas tanpa suara dengung atau disamarkan bila bertemu dengan huruf idzhar, yaitu hamzah (ء), kha (ح), ha (خ), ain (ع), ghain (غ), dan ha' (هـ) [1].

Pada penelitian ini, berbagai pola idzhar akan dideteksi melalui teknik-teknik pengolahan citra dan pengukuran kelas pola menggunakan metode jaringan syarafa tiruan. Untuk itu dipilih metode yang efisien baik dalam pembelajaran pola atau dalam pengujian pola. Metode yang digunakan dalam penelitian pendeteksian pola ini adalah jaringan *Learning Vektor Quantization* (LVQ).

Citra

Citra merupakan fungsi kontinu dari intensitas cahaya pada bidang dua dimensi. Secara matematis fungsi intensitas cahaya pada bidang dua dimensi dinotasikan sebagai $f(x, y)$, di mana (x, y) merupakan koordinat pada bidang dua dimensi, dan $f(x, y)$ merupakan intensitas cahaya pada bidang dua dimensi. Karena cahaya merupakan bentuk energi, maka intensitas cahaya bernilai $0 \leq f(x, y) \leq \infty$. Nilai $f(x, y)$ merupakan hasil kali dari $i(x, y)$ dengan $r(x, y)$, di mana $i(x, y)$ merupakan jumlah cahaya yang berasal dari sumber cahaya, nilainya antara 0 sampai tidak terhingga, dan $r(x, y)$ merupakan derajat kemampuan objek memantulkan cahaya, nilainya berkisar 0 hingga 1.

Agar dapat diolah dengan komputer digital, maka suatu citra harus direpresentasikan secara numerik dengan nilai-nilai diskrit. Representasi citra dari fungsi kontinu menjadi nilai-nilai diskrit disebut digitalisasi, dan citra yang dihasilkan merupakan citra digital [3].

Grey-Scale

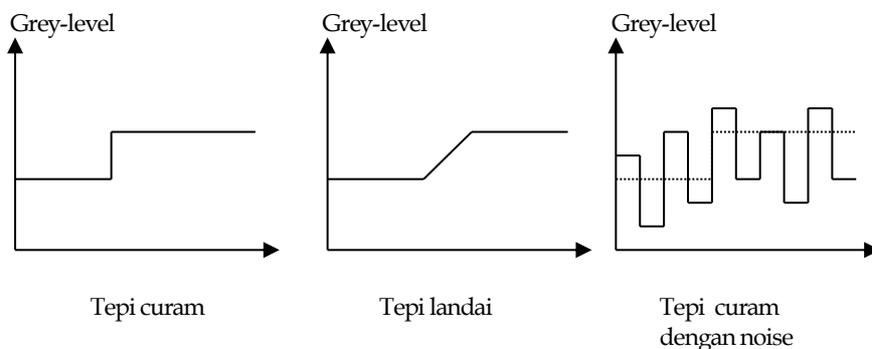
Sebelum citra video 24 bit (RGB) dimanipulasi dengan berbagai operasi tertentu, biasanya, citra dikonversikan dahulu menjadi citra grey-scale. Penkonversian ini bertujuan agar citra tersebut dapat dimanipulasi secara efisien tanpa perlu terjadinya perhitungan yang sama berulang-ulang pada setiap kanal citra [4].

Untuk mengkonversi citra RGB menjadi citra grey-scale, kita gunakan formulasi berikut :

$$bit_{RGB}(x, y) = 0,11 \times bit_R(x, y) + 0,59 \times bit_G(x, y) + 0,3 \times bit_B(x, y)$$

Edge-Detection

Tepi (edge) didefinisikan sebagai perubahan intensitas grey-level secara mendadak, dalam jarak yang singkat. Ada tiga macam tepi (edge) yang sering muncul di dalam citra digital : tepi curam, tepi landai, dan tepi yang mengandung noise [5].



Gambar 1 Jenis-jenis tepi

Untuk mendeteksi keberadaan tepi-tepi pada citra digunakan berbagai teknik Operator Sobel .

Tinjau berbagai pixel di sekitar pixel (x, y)

$$a_0 \quad a_1 \quad a_2$$

$$a_7 \quad (x, y) \quad a_3$$

$$a_6 \quad a_5 \quad a_4$$

Operator sobel merupakan magnitudo dari gradien

$$M = \sqrt{s_x^2 + s_y^2}$$

Turunan parsial dihitung dengan

$$s_x = (a_2 + ca_3 + a_4) - (a_0 + ca_7 + a_6)$$

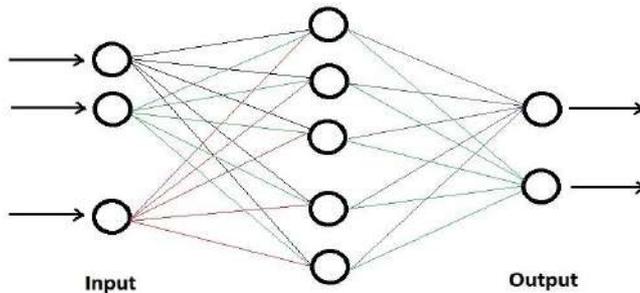
$$s_y = (a_0 + ca_1 + a_2) - (a_6 + ca_5 + a_4)$$

dengan konstanta $c = 2$. dalam bentuk *mask*, s_x dan s_y dapat dinyatakan sebagai :

$$s_x = \begin{bmatrix} -1 & 0 & 1 \\ -2 & 0 & 2 \\ -1 & 0 & 1 \end{bmatrix} \quad \text{dan} \quad s_y = \begin{bmatrix} 1 & 2 & 1 \\ 0 & 0 & 0 \\ -1 & -2 & -1 \end{bmatrix}$$

Learning Vector Quantization

Jaringan LVQ merupakan salah satu jaringan syaraf tiruan yang melakukan pembelajaran secara supervisi. LVQ mengklasifikasikan input secara berkelompok ke dalam kelas yang sudah didefinisikan melalui jaringan yang telah dilatih. Dengan kata lain LVQ mendapatkan n input dan mengelompokkannya ke dalam m output. Arsitektur jaringan LVQ terdiri dari lapisan input, lapisan kohonen, dan lapisan output. Gambar 2 mengilustrasikan contoh arsitektur LVQ [6][7][8].



Gambar 2 Contoh arsitektur LVQ

Pada proses pelatihan jaringan LVQ menggunakan layer kohonen, di mana pada proses pelatihan jaringan ini akan dibandingkan nilai vektor yang dilatih dengan semua elemen pemroses. Jarak terpendek antara vektor yang dilatih dengan elemen pemroses akan menentukan kelas dari data yang dilatih.

Pada beberapa referensi akan ditemukan berbagai algoritma LVQ yang berbeda, tetapi secara umum algoritma LVQ adalah sebagai berikut :

1. Tentukan kelas output, bobot, dan menetapkan *learning rate* α .
2. Ukur jarak input x_i terhadap bobot w_j melalui persamaan $\|x_i - w_j\|$
3. Ambil jarak terpendek, dan perbaharui bobot. Untuk input dan bobot dalam kelas yang sama, berlaku $w'_j = w_j + \alpha(x_i - w_j)$ dan Untuk input dan bobot dalam kelas yang berbeda, berlaku $w'_j = w_j - \alpha(x_i - w_j)$.

Pada dasarnya langkah-langkah algoritma di atas akan dilakukan berulang-ulang sampai nilai bobot tidak berubah jika ada input baru. Hal ini tentu akan mengkosumsi memori yang besar dengan waktu yang lama, maka dalam proses perhitungan LVQ diperlukan batas maksimal perulangan atau *Epoch*.

Ambil kasus berikut :

Ditentukan bobot awal kelas A = (0,24, 0,36, 0,2, 0,32),
Dan kelas B = (0,32, 0,24, 0,2, 0,2)

learning rate $\alpha = 0,05$

Epoch = 5.

Jika diambil secara acak vektor pola input = (0,32, 0,24, 0,2, 0,16)

Hitung jarak vektor input ke bobot kelas A = (0,24, 0,36, 0,2, 0,32)

$$= \sqrt{(0,32 - 0,24)^2 + (0,24 - 0,36)^2 + (0,2 - 0,2)^2 + (0,16 - 0,32)^2} = 0,215$$

Hitung jarak vektor input ke bobot kelas B = (0,32, 0,24, 0,2, 0,2)

$$= \sqrt{(0,32 - 0,32)^2 + (0,24 - 0,24)^2 + (0,2 - 0,2)^2 + (0,16 - 0,2)^2} = 0,04.$$

Diperoleh jarak terpendek adalah jarak vektor pola input ke bobot kelas B.

Untuk vektor pola input dengan kelas yang sama terhadap nilai bobot, maka bobot terbaru kelas B adalah :

$$w'_j = w_j + \alpha(x_i - w_j) = 0,32 + (0,05) \times (0,32 - 0,32) = 0,32$$

$$w'_j = w_j + \alpha(x_i - w_j) = 0,24 + (0,05) \times (0,24 - 0,24) = 0,24$$

$$w'_j = w_j + \alpha(x_i - w_j) = 0,2 + (0,05) \times (0,2 - 0,2) = 0,2$$

$$w'_j = w_j + \alpha(x_i - w_j) = 0,16 + (0,05) \times (0,16 - 0,2) = 0,198$$

Dan bobot terbaru kelas B adalah : (0,32, 0,24, 0,2, 0,198)

Iterasi ke-2 :

Misalkan vektor pola input ke-2 = (0,24, 0,36, 0,24, 0,32)

jarak vektor input ke bobot kelas A = (0,24, 0,36, 0,2, 0,32)

$$= \sqrt{(0,24 - 0,24)^2 + (0,36 - 0,36)^2 + (0,24 - 0,2)^2 + (0,32 - 0,32)^2} = 0,04$$

Hitung jarak vektor input ke bobot kelas B = (0,32, 0,24, 0,2, 0,2)

$$= \sqrt{(0,24 - 0,32)^2 + (0,36 - 0,24)^2 + (0,24 - 0,2)^2 + (0,32 - 0,2)^2} = 0,19.$$

Diperoleh jarak terpendek adalah jarak vektor pola input ke bobot kelas A.

Untuk vektor pola input dengan kelas yang sama terhadap nilai bobot, maka bobot terbaru kelas B adalah :

$$w'_j = w_j + \alpha(x_i - w_j) = 0,24 + (0,05) \times (0,24 - 0,24) = 0,24$$

$$w'_j = w_j + \alpha(x_i - w_j) = 0,36 + (0,05) \times (0,36 - 0,36) = 0,36$$

$$w'_j = w_j + \alpha(x_i - w_j) = 0,2 + (0,05) \times (0,24 - 0,2) = 0,202$$

$$w'_j = w_j + \alpha(x_i - w_j) = 0,32 + (0,05) \times (0,32 - 0,32) = 0,32$$

Dan bobot terbaru kelas B adalah : (0,24, 0,36, 0,202, 0,32)

Lakukan perulangan sampai epoch ke-5, dan diperoleh

$$w_1 = (0,32, 0,24, 0,2, 0,183)$$

$$w_2 = (0,24, 0,36, 0,206, 0,32)$$

Untuk pengujian, misalkan diketahui sebuah vektor pola = (0,28, 0,24, 0,2, 0,2), tentukan kelas vektor pola tersebut ?

jarak vektor input ke bobot kelas A = (0,32, 0,24, 0,2, 0,183)

$$= \sqrt{(0,28 - 0,32)^2 + (0,24 - 0,24)^2 + (0,2 - 0,2)^2 + (0,2 - 0,183)^2} = 0,043$$

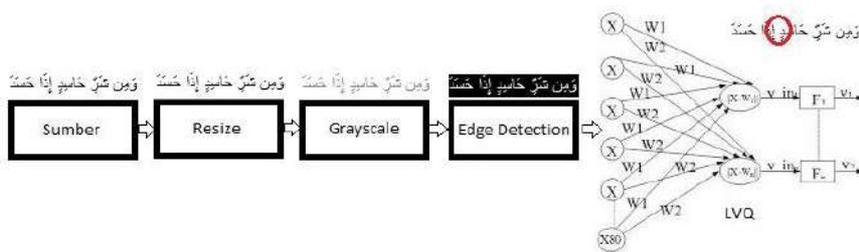
Hitung jarak vektor input ke bobot kelas B = (0,24, 0,36, 0,206, 0,32)

$$= \sqrt{(0,28 - 0,24)^2 + (0,24 - 0,36)^2 + (0,2 - 0,206)^2 + (0,2 - 0,32)^2} = 0,184.$$

Diperoleh jarak terdekat adalah jarak vektor pola input ke bobot kelas A, maka citra uji termasuk kelas A.

Skema Sistem

Skema sistem pengenalan pola yang dirancang dalam penelitian ini diilustrasikan pada Gambar 3.



Gambar 3 Skema sistem pendeteksi pola

Sistem menerima input sebuah citra format .bmp, yang selanjutnya diproses ke bagian *Resize*. *Resize* berfungsi untuk menskalakan ukuran citra sumber yang memiliki ukuran yang variatif menjadi ukuran *default*. Setelah melalui tahapan *resize*, citra akan melewati tahapan *grayscale* untuk penentuan tepi-tepi pola pada tahapan akhir pengolahan citra. Pada tahapan klasifikasi, citra tepi akan melalui beberapa tahapan proses pelatihan. Pelatihan dilakukan dengan mengunjukkerjakan pendekatan jaringan *Learning Vector Quantization (LVQ)*. Setelah mencapai epoch maksimum, pelatihan LVQ akan menghasilkan nilai-nilai bobot. Nilai-nilai bobot yang dihasilkan akan menjadi acuan perhitungan jarak dalam proses pengujian pola.

Hasil dan Pembahasan

Pengukuran unjuk kerja sistem dilakukan dengan pelatihan bertahap. Pada citra yang memuat pola nun mati (ن) atau tanwin (ـ, ـ, ـ) bertemu dengan huruf idzhar hamzah (ء) (k_1), sistem memiliki tingkat keakuratan sebesar 90%. Tabel 1 memaparkan beberapa hasil pengukuran unjuk kerja sistem pendeteksi pola (k_1). Hasil evaluasi sistem menunjukkan bahwa peningkatan *Detection Rate* sangat terkait dengan peningkatan jumlah pelatihan. Pada citra yang memuat pola nun mati (ن) atau tanwin (ـ, ـ, ـ) bertemu dengan huruf idzhar kha (خ) (k_2), sistem pendeteksi pola memiliki tingkat keakuratan sebesar 82%. Tabel 2 memaparkan beberapa hasil pengukuran unjuk kerja sistem pendeteksi pola (k_2). Pada citra yang memuat pola

nun mati (نْ) atau tanwin (ـً, ـٍ, ـٌ) bertemu dengan huruf idzhar ha (ح) (k_3), sistem pendeteksi pola memiliki tingkat keakuratan sebesar 70%. Tabel 3 memaparkan beberapa hasil pengukuran unjuk kerja sistem pendeteksian pola (k_3). Pada citra yang memuat pola nun mati (نْ) atau tanwin (ـً, ـٍ, ـٌ) bertemu dengan huruf idzhar ain (ع) (k_4), sistem pendeteksi pola memiliki tingkat keakuratan sebesar 65%. Tabel 4 memaparkan beberapa hasil pengukuran unjuk kerja sistem pendeteksian pola idzhar (k_4). Pada citra yang memuat pola nun mati (نْ) atau tanwin (ـً, ـٍ, ـٌ) bertemu dengan huruf idzhar ghain (غ) (k_5), sistem pendeteksi pola memiliki tingkat keakuratan sebesar 65%. Tabel 5 memaparkan beberapa hasil pengukuran unjuk kerja sistem pendeteksian pola (k_5). Pada citra yang memuat pola nun mati (نْ) atau tanwin (ـً, ـٍ, ـٌ) bertemu dengan huruf idzhar ha' (هـ) (k_6), sistem pendeteksi pola memiliki tingkat keakuratan sebesar 65%. Tabel 6 memaparkan beberapa hasil pengukuran unjuk kerja sistem pendeteksian pola (k_6).

Tabel 1. Hasil unjuk kerja sistem pendeteksi pola (k_1).

Jumlah Citra Pelatihan	Jumlah Citra Pengujian	Jumlah Pendeteksian yang benar	False Positive Rate	Detection Rate
25	100	60	0,4	0,6
50	100	67	0,33	0,67
75	100	85	0,15	0,85
100	100	90	0,1	0,9

Tabel 2. Hasil unjuk kerja sistem pendeteksi pola (k_2).

Jumlah Citra Pelatihan	Jumlah Citra Pengujian	Jumlah Pendeteksian yang benar	False Positive Rate	Detection Rate
25	100	50	0,5	0,5
50	100	59	0,41	0,59
75	100	70	0,3	0,7
100	100	82	0,18	0,82

Tabel 3. Hasil unjuk kerja sistem pendeteksi pola (k_3).

Jumlah Citra Pelatihan	Jumlah Citra Pengujian	Jumlah Pendeteksian yang benar	False Positive Rate	Detection Rate
25	100	45	0,55	0,45
50	100	52	0,48	0,52
75	100	61	0,39	0,61
100	100	70	0,3	0,7

Tabel 4. Hasil unjuk kerja sistem pendeteksi pola (k_4).

Jumlah Citra Pelatihan	Jumlah Citra Pengujian	Jumlah Pendeteksian yang benar	False Positive Rate	Detection Rate
25	100	40	0,6	0,4
50	100	49	0,51	0,49
75	100	55	0,35	0,55
100	100	65	0,25	0,65

Tabel 5. Hasil unjuk kerja sistem pendeteksi pola (k_5).

Jumlah Citra Pelatihan	Jumlah Citra Pengujian	Jumlah Pendeteksian yang benar	False Positive Rate	Detection Rate
25	100	40	0,6	0,4
50	100	49	0,51	0,49
75	100	55	0,35	0,55
100	100	65	0,25	0,65

Tabel 6. Hasil unjuk kerja sistem pendeteksi pola (k_6).

Jumlah Citra Pelatihan	Jumlah Citra Pengujian	Jumlah Pendeteksian yang benar	False Positive Rate	Detection Rate
25	100	40	0,6	0,4
50	100	49	0,51	0,49
75	100	55	0,35	0,55
100	100	65	0,25	0,65

Kesimpulan dan Saran

Hasil penelitian menunjukkan bahwa sistem pendeteksi pola menggunakan pendekatan jaringan LVQ memiliki *detection rate* berkisar dari 65% hingga 90%. Persentase *detection rate* tersebut menunjukkan bahwa jaringan LVQ dapat digunakan sebagai salah satu pendekatan untuk pendeteksian pola-pola yang sejenis. Keunggulan jaringan LVQ adalah kemampuan pengenalan pola dengan jumlah data pelatihan yang tidak terlalu besar, sangat tepat untuk diterapkan pada sistem.

Untuk peningkatan kualitas unjuk kerja sistem, penelitian pendeteksian pola menggunakan LVQ dapat mengintegrasikan sistem dengan salah satu dari : uji kekompleksitasan citra, segmentasi, uji statistika, dan lain-lain. Penambahan pendekatan tersebut tentunya akan mempengaruhi kecepatan komputasi pendeteksian pola tersebut.

Daftar Pustaka

- [1] Zarkasyi, 2016. *Panduan Tajwid Ringkas dan Praktis*. HUDA Lhokseumawe.
- [2] Fadlisyah, 2013. *Sistem Pendeteksian Wajah Pada Video Menggunakan Jaringan Adaptive Linear Neuron (ADALINE)*. Tesis. Universitas Sumatra Utara.
- [3] Munir, Rinaldi. 2004. *Pengolahan Citra Digital Dengan Pendekatan Algoritmik*. Informatika : Bandung.
- [4] Fadlisyah. 2020. *Pengolahan Citra dan Suara*, Penerbit Graha Ilmu. Yogyakarta
- [5] Fadlisyah dan Rizal. 2011. *Pemrograman Computer Vision pada Video*, Penerbit Graha Ilmu. Yogyakarta
- [6] Putra, Darma. 2010. *Pengolahan Citra Digital*. Yogyakarta: PenerbitAndi
- [7] Bustami, Dahlan Abdullah, dan Fadlisyah. 2014. *Statistika : terapannya di informatika*, Penerbit Graha Ilmu. Yogyakarta. (978-979-756-929-7)
- [8] Fadlisyah. 2007. *Computer Vision dan Pengolahan Citra*, Penerbit Andi. Yogyakarta