

Perbandingan identifikasi citra penderita tumor otak menggunakan jaringan syaraf tiruan LVQ dan MLVQ

Veronika Pitri, Yudha Arman*, Hasanuddin

Program Studi Fisika, Fakultas MIPA, Universitas Tanjungpura

Jl. Prof. Dr. H.Jl. Profesor Dokter H. Hadari Nawawi, Bansir Laut, Kec. Pontianak Tenggara,
Kota Pontianak, Kalimantan Barat

*e-mail: yudha_arman@physics.untan.ac.id

Abstrak – Telah dilakukan identifikasi citra magnetic resonance imaging (MRI) pada penderita tumor otak menggunakan metode learning vector quantization termodifikasi (MLVQ). Modifikasi dilakukan pada bagian pembaharuan learning rate metode LVQ, menerapkan konsep peluang pada sebaran data berdistribusi gaussian. Sebanyak citra yang terdiri atas citra otak normal dan citra otak terdiagnosa tumor digunakan pada penelitian ini. Proses pengolahan citra yang dilakukan adalah pemotongan, pengubahan ukuran, pengubahan skala abu-abu, pengubahan skala hitam putih dan deteksi tepi. Hasil yang didapatkan kemudian dibandingkan dengan hasil yang diperoleh menggunakan metode LVQ. Berdasarkan hasil perhitungan yang dilakukan, metode MLVQ memiliki akurasi yang lebih tinggi jika dibandingkan dengan metode LVQ untuk kasus ini.

Kata Kunci: learning rate, LVQ, MLVQ, tumor otak

Abstract – Identification of magnetic resonance imaging (MRI) has been performed on brain tumor patients using the modified learning vector quantization (MLVQ) method. Modifications were made to update the learning rate of the LVQ method, applying the concept of opportunity to the distribution of data with a Gaussian distribution. A total of 3.000 images consisting of 1.500 typical brain images and 1.500 tumor-diagnosed brain images were used in this study. Image processing includes cropping, resizing, grayscale conversion, black-and-white scale conversion, and edge detection. The results were then compared with those obtained using the LVQ method. Based on the calculation results, the MLVQ method has higher accuracy when compared to the LVQ method for this case.

Key words: brain tumor, learning rate, LVQ, MLVQ

PENDAHULUAN

Tumor otak merupakan pertumbuhan sel abnormal di sekitar otak. Pertumbuhan tersebut akan mengganggu fungsi koordinasi ke bagian lain dari tubuh manusia karena otak merupakan pusat kendali seluruh pergerakan tubuh manusia. Menurut *International Agency for Research on Cancer*, penderita tumor otak di setiap tahun mencapai lebih dari 126.000 jiwa dan lebih dari 97.000 jiwa meninggal dunia setiap tahunnya [1].

Keberadaan tumor otak dapat dideteksi menggunakan *magnetic resonance imaging* (MRI) [2]. Alat ini menggunakan medan magnet sebesar 0,064 sampai 1,5 tesla (1 tesla = 1000 gauss), yang segaris dengan arah perputaran pada inti atom hidrogen [3]. Citra hasil MRI dilaporkan memiliki kualitas yang lebih baik daripada alat diagnosis radiologi lainnya, sehingga MRI sering digunakan pada bidang medis [1].

Analisis hasil MRI umumnya dilakukan secara manual sehingga akurasi yang dihasilkan perlu ditingkatkan. Keterbatasan mata manusia dalam mengidentifikasi objek yang cukup kompleks menjadi salah faktor penyebab hal ini. Diperlukan perangkat bantu yang dapat meminimalisir kesalahan dalam proses diagnosa tersebut [2].

Teknik komputasi yang umum digunakan untuk pemecahan masalah pengenalan pola karakteristik serta memiliki akurasi perhitungan yang tinggi adalah jaringan syaraf tiruan (JST). Metode *Learning Vector Quantization* (LVQ) merupakan salah satu varian dari metode JST. LVQ menggunakan sistem pembelajaran kompetitif terawasi [4]. Metode ini dilaporkan telah digunakan untuk mengklasifikasi penyakit ginjal kronis [5]. Learning rate sebesar 0,002 dilaporkan menghasilkan model prediksi dengan akurasi terbaik untuk kasus tersebut. Penentuan parameter *learning rate* ini sangat menentukan keberhasilan dari penggunaan metode LVQ.

Modifikasi dapat dilakukan pada metode LVQ untuk mendapatkan hasil dengan akurasi yang lebih baik. Metode ini biasa disebut sebagai *Modified Learning Vector Quantization* (MLVQ). Umumnya, modifikasi dilakukan pada proses pemutakhiran *learning rate* untuk meningkatkan akurasi perhitungan. Metode LVQ termodifikasi digunakan untuk menentukan centroid pada identifikasi pola [6]. Akurasi yang dihasilkan melalui modifikasi ini dilaporkan lebih tinggi jika dibandingkan dengan metode LVQ. Peningkatan akurasi juga dijumpai dalam mengenali citra wajah [4]. Penggunaan MLVQ untuk kasus tersebut diketahui memiliki akurasi 50,37%, lebih tinggi jika dibandingkan dengan metode propagasi balik (48,14%) dan metode LVQ (49,24%).

Berdasarkan hal tersebut, penelitian ini menggunakan JST metode MLVQ dalam mengidentifikasi tumor otak citra MRI. Citra MRI akan melalui beberapa proses untuk mendapatkan karakteristik yang bisa digunakan pada proses analisis. Hasil dari metode MLVQ ini kemudian dibandingkan dengan hasil yang diperoleh menggunakan metode LVQ.

Citra yang digunakan pada studi ini adalah citra hasil MRI pada bagian otak manusia yang terdiri dari citra otak normal dan abnormal (tumor) dengan ekstensi .jpg. Sebanyak 3.000 citra digunakan sebagai masukan untuk sistem yang dibuat. Citra yang digunakan sebanyak 3.000, terdiri atas 1.500 citra otak normal dan 1.500 citra otak abnormal (tumor) diambil dari laman internet www.kaggle.com.

METODE PENELITIAN

Penelitian ini secara garis besar terdiri dari 2 tahapan, yaitu, pengolahan citra dan perancangan jaringan syaraf tiruan. Pengolahan citra terdiri dari pemotongan citra, pengubahan ukuran, pengubahan skala abu-abu, pengubahan skala hitam putih dan deteksi tepi. Pengolahan citra dilakukan untuk mendapatkan ciri khas khusus citra otak yang didiagnosis tumor otak dan normal. Pengolahan ini dimulai dengan melakukan pemotongan bagian yang tidak digunakan untuk proses identifikasi. Proses tersebut menghasilkan ukuran citra yang tidak seragam, sehingga diperlukan pengubahan ukuran citra. Ukuran citra yang digunakan pada penelitian ini sebesar 100×100 piksel berformat merah, hijau dan biru (RGB). Citra RGB ini kemudian diubah ke skala abu-abu (hitam, putih, dan abu-abu) untuk selanjutnya

diubah ke citra berformat biner. Citra dengan skala abu-abu ($f_0(x, y)$) diperoleh dengan merata-ratakan komponen warna merah ($f^R(x, y)$), hijau ($f^G(x, y)$), dan biru ($f^B(x, y)$) pada x dan y seperti yang ditunjukkan oleh persamaan (1) berikut,

$$f_0(x, y) = \left(\frac{f^R(x, y) + f^G(x, y) + f^B(x, y)}{n} \right), \quad (1)$$

dengan n merupakan banyaknya data.

Pengubahan citra ke skala biner dilakukan agar perbedaan antara bagian yang terdiagnosa tumor otak dan bagian normal dapat terlihat. Pengubahan ke citra biner dilakukan dengan menggunakan nilai threshold (T) sebesar 127,5 yang diperoleh dari rata-rata nilai piksel ($f(x, y)$) maksimum dan minimum sesuai persamaan berikut,

$$T = \frac{f(x, y)_{\text{maksimum}} + f(x, y)_{\text{minimum}}}{n}, \quad (2)$$

dengan ketentuan apabila nilai $f(x, y)$ lebih besar atau sama dengan 127,5 maka nilai piksel yang baru $g(x, y)$ berubah menjadi 1 (putih), sedangkan jika $f(x, y)$ lebih kecil dari atau sama dengan 127,5 maka akan berubah menjadi 0 (hitam). Kondisi ini dituliskan sebagai:

$$(x, y) = \begin{cases} 1, & f(x, y) \geq T \\ 0, & f(x, y) < T \end{cases} \quad (3)$$

Pengolahan citra diakhiri dengan proses deteksi tepi. Proses ini bertujuan untuk mempertegas detail-detail yang ada pada citra dengan mempertajam garis-garis tepi. Proses deteksi tepi pada penelitian ini menggunakan penjumlahan dari operasi operator sobel (M) pada absis x (S_x) dan ordinat y (S_y). Operator sobel yang digunakan disajikan pada Gambar 1.

$$\begin{matrix} & & S_x & & S_y \\ \begin{bmatrix} -1 & 0 & 1 \\ -2 & 0 & 2 \\ -1 & 0 & 1 \end{bmatrix} & & \begin{bmatrix} 1 & 2 & 1 \\ 0 & 0 & 0 \\ -1 & -2 & -1 \end{bmatrix} \end{matrix}$$

Gambar 1. Operator 2 buah kernel [4] (Sumber: Ketaren, 2016)

Jaringan syaraf tiruan yang dibangun, yaitu LVQ dan MLVQ terbagi menjadi 2 proses, yaitu pelatihan dan uji coba. Perancangan dilakukan dengan menggunakan bahasa pemrograman python.

Proses pelatihan pada metode LVQ diawali dengan menetapkan jumlah k -folds, n -codebooks

(W_{ji}), *epoch* dan *learning rate* (α). *Learning rate* yang digunakan pada penelitian ini sebesar 0,02. Selanjutnya dilakukan perhitungan jarak (S_j) antara vektor masukan (X_i) dengan *codebooks* (W_{ji}) menggunakan persamaan (4).

$$S_j = \sqrt{\sum_{i=1}^n (X_i - W_{ji})^2} \quad (4)$$

Kelas bagi vektor masukan ditentukan berdasarkan jarak terkecil ke *codebooks*. Setelah proses ini, pembaharuan nilai *learning rate* (a_{baru}) dilakukan sesuai persamaan berikut:

$$a_{baru} = 0,4(a_{lama}) \quad (5)$$

dengan a_{lama} adalah *learning rate* sebelumnya.

Proses terakhir dari siklus perhitungan adalah memutakhirkan *codebooks* dengan beberapa syarat tertentu. Apabila hasil yang didapatkan menunjukkan target kelas (T_j) dan kelas *codebooks* (C_j) sama maka *codebooks* diperbaharui menggunakan:

$$W_{ji_{baru}} = W_{ji_{lama}} + \alpha_{baru} (X_i - W_{ji_{lama}}) \quad (6)$$

Sebaliknya, apabila T_j dan C_j tidak sama maka W_{ji} diperbaharui menggunakan persamaan (7) berikut,

$$W_{ji_{baru}} = W_{ji_{lama}} - \alpha_{baru} (X_i - W_{ji_{lama}}) \quad (7)$$

dengan $W_{ji_{baru}}$, $W_{ji_{lama}}$, α_{baru} , dan X_i masing-masing merupakan *codebooks* baru, *codebooks* lama, *learning rate* baru, dan vektor masukan.

Pelatihan metode LVQ diakhiri dengan proses uji coba. Proses ini dimulai dengan mencari jarak terpendek yang menjadi kelasnya seperti langkah pertama pada proses pelatihan menggunakan persamaan (4). Persentase akurasi kemudian dihitung menggunakan:

$$Akurasi = \frac{J_b}{J_s} \times 100\% \quad (8)$$

dengan J_b adalah jumlah benar dan J_s adalah jumlah salah.

Proses yang sama juga dilakukan pada metode MLVQ. Namun terdapat perbedaan pada pembaharuan *learning rate* (a) yang tidak lagi menggunakan 0,02 tetapi menggunakan persebaran data pada fungsi gaussian (f) sesuai persamaan berikut:

$$f = \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{1}{2}\left(\frac{y-\mu}{\sigma}\right)^2} \quad (9)$$

dengan σ merupakan simpangan baku, μ adalah nilai rata-rata, e adalah bilangan euler dan y adalah nilai dari variabel acak. Dengan mengganti beberapa variabel pada fungsi gaussian persamaan (9) dengan nilai jarak (S_j) antara vektor masukan (X_i) dan *codebooks* (W_{ji}), maka pemutakhiran *learning rate* (a) dilakukan sesuai persamaan berikut:

$$a = \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\left(\frac{S_j}{2\sigma^2}\right)} \quad (10)$$

Kendala yang dihadapi saat mengimplementasikan persamaan (10) ke sistem adalah pada hasil perhitungan $\frac{1}{\sqrt{2\pi\sigma^2}}$. Nilai yang dihasilkan terlalu besar mengakibatkan sistem tidak stabil. Untuk mengatasi hal ini, nilai kemudian diperkecil sesuai persamaan berikut:

$$a = \left(0,01 \sqrt{2\pi\sigma_b^2}\right) \left(e^{-\left(\frac{S_j}{2\sigma_b^2}\right)}\right) \quad (11)$$

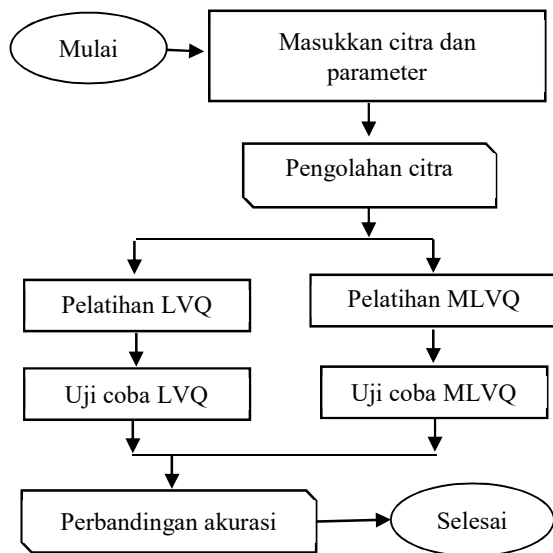
Variable σ_b pada persamaan (11) adalah simpangan baku dari data sebanyak n yang dihitung menggunakan persamaan berikut,

$$\sigma = \sqrt{\frac{1}{n} \sum_{i=1}^N (m_i - \mu_n)^2} \quad (12)$$

dengan m_i adalah data ke- i dan μ_n adalah nilai rata-rata data ke- n . Karena $(m_i - \mu_n)^2$ memiliki kesamaan karakteristik dengan $(X_i - W_{ji})^2$ maka,

$$\sigma = \sqrt{\frac{1}{n} \sum_{i=1}^N (X_i - W_{ji})^2} \quad (13)$$

Sama seperti proses terakhir metode LVQ, pada metode MLVQ juga dilakukan proses perhitungan akurasi dengan menggunakan persamaan (8). Hasil akurasi yang didapat kemudian dibandingkan dengan akurasi metode LVQ melihat metode dalam mengidentifikasi citra hasil MRI penderita tumor otak. Adapun serangkaian proses pada penelitian ini disajikan pada Gambar 2.



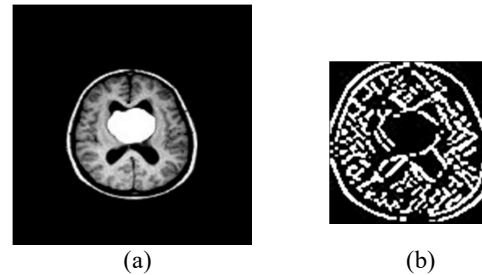
Gambar 2. Diagram alir penelitian

HASIL DAN PEMBAHASAN

Pengolahan citra merupakan proses awal pada penelitian ini yang bertujuan untuk mendapatkan karakteristik unik pada citra sehingga mudah untuk dianalisis. Proses diawali dengan pemotongan bagian yang tidak

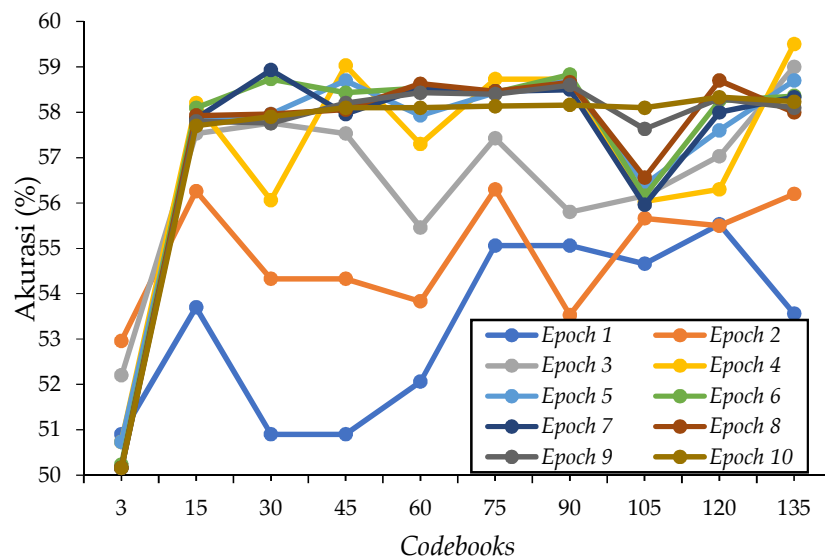
digunakan pada citra dan diakhiri dengan deteksi tepi menggunakan operator sobel.

Gambar 3 memperlihatkan perbedaan antara citra asli dengan yang telah melewati proses pengolahan citra.



Gambar 3. Citra (a) asli dan (b) hasil pengolahan citra

Parameter yang digunakan pada penelitian ini terdiri atas *codebooks* dengan variasi jumlah 3,15,30,45,60, 75, 90, 105, 120 dan 135, 2 kelompok *epoch* (1-10 dan 10-100), *k-folds* sebanyak 2, serta *learning rate* yang digunakan pada metode LVQ adalah 0,002.



Gambar 4. Hasil pengujian metode LVQ (epoch 1-10)

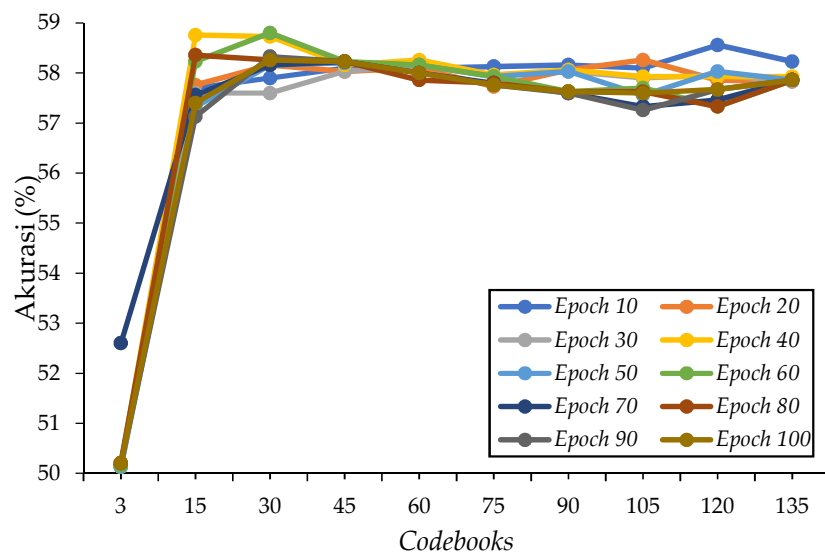
Gambar 4 memperlihatkan grafik hasil akurasi metode LVQ untuk *epoch* 1-100. Dari grafik tersebut dapat dilihat bahwa akurasi untuk *epoch* 1-10 mengalami fluktuasi. *Epoch* 1 menghasilkan akurasi terendah pada pengujian ini dengan rata-rata akurasi sebesar 53,23%, sedangkan akurasi tertinggi berada pada *epoch* 6

dengan rata-rata akurasi sebesar 57,40%. Akurasi cenderung mengalami kenaikan pada saat *epoch* 1, *epoch* 2, dan *epoch* 3, namun untuk *epoch* setelahnya akurasi yang dihasilkan hampir berada di titik yang sama. Hal ini disebabkan karena pembaharuan *learning rate* bergantung pada jumlah *epoch* yang digunakan.

Epoch merupakan jumlah maksimum dari iterasi. Semakin besar jumlah *epoch* yang digunakan maka besar nilai *learning rate* yang dihasilkan akan semakin kecil. Pada kasus ini, besarnya pembaharuan *learning rate*, sudah tidak memiliki pengaruh besar pada pembaharuan nilai *codebooks* untuk *epoch* 4 sampai *epoch* 10.

Namun, apabila dilihat berdasarkan variasi *codebooks*, akurasi terendah berada pada *codebooks* 3 dengan rata-rata akurasi sebesar 50,84%, sedangkan akurasi tertinggi berada

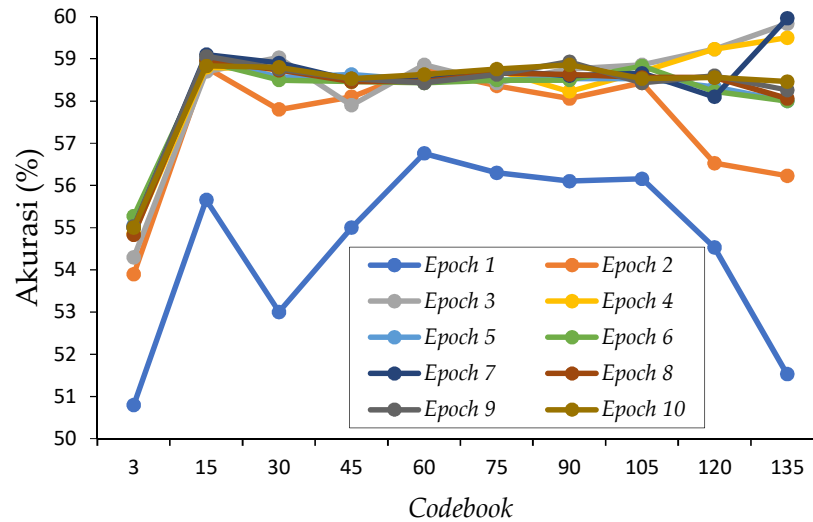
pada *codebooks* 75 dengan rata-rata akurasi sebesar 57,78%. Nilai akurasi berdasarkan variasi *codebooks* tersebut terlihat sangat berfluktuasi. Banyaknya *codebooks* juga mempengaruhi besarnya akurasi yang didapatkan. Biasanya, semakin banyak menggunakan *codebooks* maka semakin baik akurasi yang didapatkan. Tetapi, karena pembaharuan *learning rate* yang bergantung pada penggunaan jumlah *epoch* maka mengakibatkan hasil dari variasi *codebooks* ini tidak begitu terlihat jelas kenaikannya.



Gambar 5. Hasil pengujian metode LVQ (*epoch* 10-100)

Akurasi pengujian metode LVQ untuk *epoch* 10-100 ditunjukkan oleh grafik pada Gambar 5. Nilai akurasi juga menunjukkan fluktuasi untuk variasi *epoch* tersebut namun dengan amplitudo yang lebih kecil. Akurasi terendah berada masih berada pada *epoch* 90 dengan rata-rata akurasi sebesar 57,00%, sedangkan untuk akurasi tertinggi berada pada *epoch* 40 dengan rata-rata akurasi sebesar 57,38%. Fluktuasi yang dihasilkan untuk variasi *epoch* ini, hampir berada di titik yang sama. Hal tersebut, disebabkan oleh variasi yang digunakan cukup besar, sehingga mengakibatkan pembaharuan nilai *learning rate* yang didapatkan cukup kecil. *Learning rate* yang bisa dikatakan sangat kecil ini, sudah tidak lagi memiliki pengaruh besar pada pembaharuan *codebooks* yang menyebabkan menurunnya kualitas program

dalam mengidentifikasi. Penurunan tersebut, yang menyebabkan akurasi yang dihasilkan cenderung menurun disetiap *epoch*-nya. Analisis berdasarkan variasi *codebooks* memperlihatkan nilai akurasi terendah ditunjukkan oleh *codebook* 3 dengan rata-rata akurasi sebesar 50,40%, sedangkan akurasi tertinggi berada pada *codebook* 30 dengan rata-rata akurasi sebesar 58,24%. Fluktuasi nilai Akurasi yang ditunjukkan oleh bagian ini tidak sebesar dari variasi sebelumnya. Sama seperti pada pengujian *epoch* sebelumnya, akurasi berdasarkan *codebooks* ini bergantung pada jumlah *epoch* dan penurunan pembaharuan *learning rate* yang menyebabkan *codebooks* diperbaharui terlalu kecil sehingga kurang efektif dalam proses identifikasi.

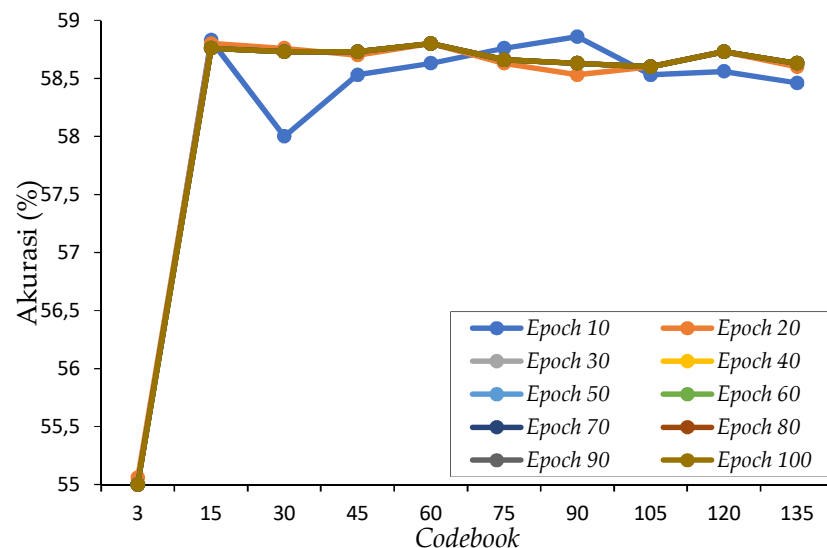


Gambar 6. Hasil pengujian metode MLVQ (epoch 1-10)

Hasil pengujian metode MLVQ pada epoch 1-10 yang ditunjukkan oleh Gambar 6 memperlihatkan bahwa epoch 1 dan epoch 7 masing-masing menunjukkan nilai akurasi terendah dan tertinggi dengan akurasi rata-rata masing-masing epoch sebesar 54,58% dan 58,40%. Akurasi pada pengujian ini, masih mengalami fluktuasi tetapi amplitudo yang dimiliki tidak sebesar pengujian LVQ dengan variasi epoch yang sama. Hasil memperlihatkan bahwa akurasi sudah tidak mengalami kenaikan yang signifikan pada saat epoch 20 sampai dengan epoch 100 sehingga grafik yang dihasilkan hampir berada dititik yang sama. Hal tersebut, menunjukan bahwa pembaharuan

learning rate yang dihasilkan sudah hampir berada pada nilai maksimum pada saat epoch 20.

Apabila fluktuasi dilihat berdasarkan codebooks terlihat bahwa akurasi terendah berada pada codebook 3 dan tertinggi berada pada codebooks 15 dengan rata-rata hasil akurasi pada masing-masing codebooks sebesar 54,40% dan 58,58%. Fluktuasi akurasi berdasarkan codebooks cenderung stabil daripada pengujian LVQ sebelumnya dengan kasus yang sama. Hal ini disebabkan karena kebergantungan pembaharuan codebooks dengan jumlah epoch.

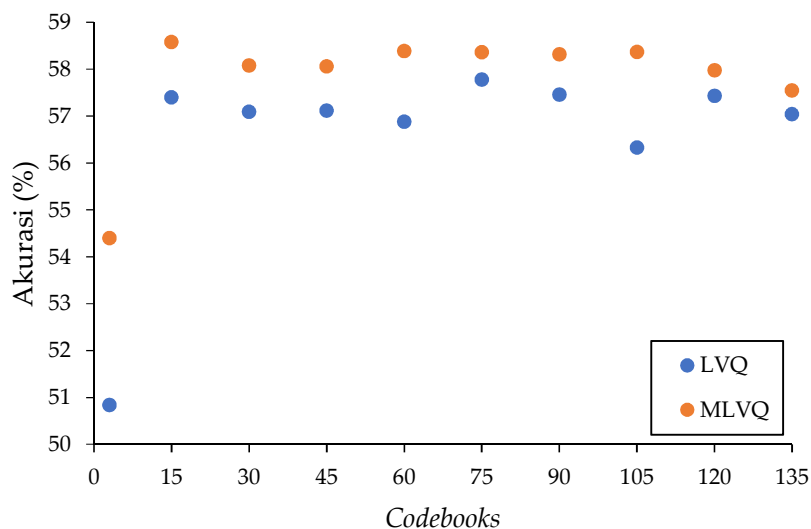


Gambar 7. Hasil pengujian metode MLVQ (epoch 10-100)

Hasil pengujian metode MLVQ pada *epoch* 10-100 diperlihatkan oleh gambar 7. Gambar menunjukkan bahwa fluktuasi hasil akurasi pada setiap *epoch* cukup konstan. Akurasi terendah berada pada *epoch* 10 dengan rata-rata akurasi sebesar 58,21% dan akurasi terendah berada pada *epoch* 30, *epoch* 40, *epoch* 50, *epoch* 60, *epoch* 70, *epoch* 80, *epoch* 90 dan *epoch* 100 dengan besar rata-rata akurasi yaitu, 58,32%. Hal ini terjadi karena pembaharuan *learning rate* yang dihasilkan sudah sangat kecil bahkan sudah mendekati nol. Nilai pembaharuan *learning rate* yang sudah sangat kecil ini, tidak lagi memiliki pengaruh terhadap pembaharuan *codebooks* sehingga akurasi yang dihasilkan selalu sama (konstan).

Jika dilihat berdasarkan pada *codebooks* akurasi mengalami penurunan untuk *epoch* 10,

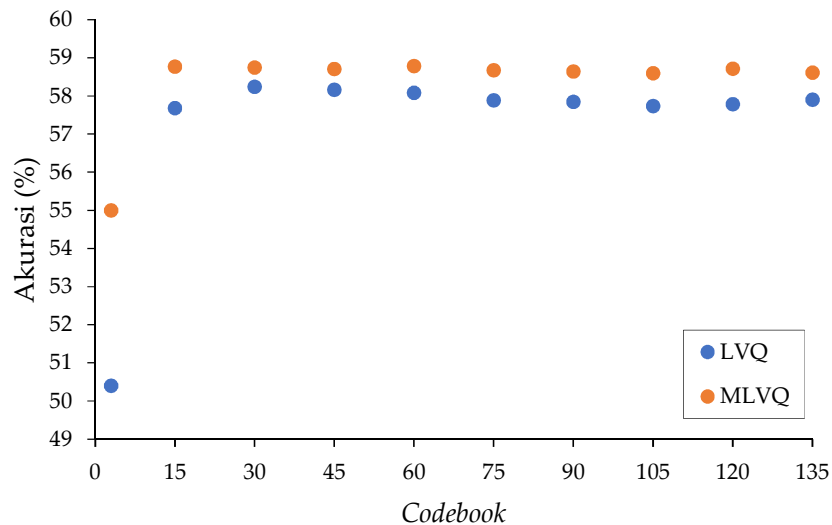
epoch 20, dan *epoch* 30 pada beberapa *codebooks*. Untuk *epoch* di atas 30 akurasi mengalami penurunan maksimum yang menyebabkan akurasi pada *epoch* selanjutnya (*epoch* 40, *epoch* 50, *epoch* 60, *epoch* 70, *epoch* 80, *epoch* 90, dan *epoch* 100) selalu konstan. Penurunan maksimum terjadi karena *codebooks* sudah tidak lagi mengalami perubahan akibat besar *learning rate* yang mendekati nol. Amplitudo yang dihasilkan sangat kecil sehingga sulit untuk terlihat. Namun berdasarkan data dapat diketahui bahwa akurasi tertinggi berada pada *codebook* 3 dan akurasi terendah berada pada *codebook* 60 dengan rata-rata akurasi masing-masing *codebooks* adalah sebesar 55% dan 58,78%.



Gambar 8. Perbandingan metode LVQ dan metode MLVQ (*epoch* 1-10)

Gambar 8 merupakan perbandingan antara metode LVQ dan metode MLVQ untuk *epoch* 1-10. Perbandingan ini didapatkan dengan merata-ratakan hasil akurasi *codebooks* di setiap *epoch* yang sudah didapatkan sebelumnya. Pada gambar terlihat bahwa metode MLVQ memiliki tingkat akurasi yang lebih tinggi daripada metode LVQ. Selisih akurasi tertinggi terjadi pada *codebooks* 3 sedangkan selisih akurasi terendah pada *codebooks* 135 dengan besar selisih akurasi masing-masing *codebooks* yaitu 3,35% dan 0,51%.

Perbandingan metode LVQ dan metode MLVQ (*epoch* 10-100) didapatkan sama seperti pada perbandingan metode LVQ dan metode MLVQ (*epoch* 10-100). Pada gambar 9 terlihat bahwa metode MLVQ juga memiliki tingkat akurasi rata-rata tertinggi daripada metode LVQ. Selisih akurasi tertinggi juga masih terjadi pada *codebooks* 3, sedangkan terendah terjadi saat *codebooks* 30 dengan besar selisih akurasi masing-masing *codebooks* yaitu 4,60% dan 0,50%.



Gambar 9. Perbandingan metode LVQ dan metode MLVQ (epoch 10 – 100)

KESIMPULAN

Adapun kesimpulan dari penelitian ini, didapatkan bahwa metode MLVQ memiliki tingkat akurasi yang lebih tinggi dari pada metode LVQ. Pada metode MLVQ *learning rate* didapatkan dari hasil perhitungan fungsi *gaussian*, sedangkan metode LVQ masih menggunakan *learning rate* yang ditentukan dari penelitian sebelumnya. Fluktuasi yang didapatkan pada metode MLVQ cenderung lebih stabil jika dibandingkan dengan fluktuasi pada metode LVQ yang masih acak.

DAFTAR PUSTAKA

- [1] J. Sofian, & R. H. Laluma, "Klasifikasi Hasil Citra Mri Otak Untuk Memprediksi Jenis Tumor Otak Dengan Metode Image Threshold Dan GLCM Menggunakan Algoritma K-Nn (Nearest Neighbor) Classifier Berbasis Web", *Infotronik: Jurnal Teknologi Informasi Dan Elektronika*, vol. 4, no. 2, p 51, 2019. <https://doi.org/10.32897/infotronik.2019.4.2.258>
- [2] K. Amalia, "Klasifikasi Penyakit Tumor Otak Pada Citra MRI Menggunakan Metode CNN Dengan Arsitektur Alexnet", *e-Proceeding of Engineering*, vol. 9, no. 5), pp. 3289–3297, 2022.
- [3] R. Wood, et al, "1.5 Tesla Magnetic Resonance Imaging Scanners Compared with 3.0 Tesla Magnetic Resonance Imaging Scanners: Systematic Review of Clinical Effectiveness", *CADTH Technology Overviews*, vol. 2, no. 2, pp. 1–5, 2012.
- [4] E. Ketaren, "Modifikasi Jaringan Syaraf Tiruan Learning Vector Quantization Pada Pengenalan Wajah", Skripsi, FMIPA Universitas Sumatera Utara, 2016.
- [5] I. G. Bgs, B. Adi, I. M. Widiartha, & L. Gede, "Implementation Learning Vector Quantization (LVQ) for Chronic Kidney Disease Classification", *Jurnal Elektronik Ilmu Komputer Udayana*, vol. 9, no. 2, pp. 241–248, 2020.
- [6] R. Y. Liu, C. Y. Wu, & I. C. Jou, "A CMOS current-mode design of modified learning-vector-quantization neural networks", *Analog Integrated Circuits and Signal Processing*, vol. 8, no. 2, pp. 157–181, 1995. <https://doi.org/10.1007/BF01239108>