

# Implementasi Metode *Extreme Learning Machine* untuk Klasifikasi Tumor Otak pada Citra *Magnetic Resonance Imaging*

Radical Rakhman Wahid<sup>1</sup>, Fetty Tri Anggraeni<sup>1\*</sup>, Budi Nugroho<sup>1</sup>

<sup>1</sup> Informatika, Universitas Pembangunan Nasional “Veteran” Jawa Timur

<sup>1</sup>[roploverz@gmail.com](mailto:roploverz@gmail.com)

<sup>1</sup>[budinugroho.if@upnjatim.ac.id](mailto:budinugroho.if@upnjatim.ac.id)

\*Corresponding author email: [fettyanggraeny.if@upnjatim.ac.id](mailto:fettyanggraeny.if@upnjatim.ac.id)

**Abstrak**— Tumor otak merupakan penyakit yang menyerang otak makhluk hidup di mana sel otak tumbuh secara tidak normal pada daerah sekitar otak. Berbagai macam cara telah dilakukan untuk mendeteksi penyakit ini, bahkan dengan bantuan teknologi digital, salah satunya dengan melalui pendekatan anatomi citra kesehatan. Pada penelitian ini kami mengusulkan metode *Extreme Learning Machine (ELM)* sebagai pengklasifikasi citra tumor otak yang didapat melalui pemindaian *Magnetic Resonance Imaging*. *ELM* dipilih karena keunggulannya pada proses pelatihan, yaitu lebih cepat dibanding algoritma pembelajaran mesin yang sifatnya iteratif. *Data set* yang kami gunakan adalah *Brain MRI Images for Tumor Detection*. *Data set* tersebut berisi dua kategori citra yaitu Tumor dan Otak tanpa tumor (sehat). Citra pada *data set* tersebut dipraproses sebelum masuk tahap klasifikasi, *ELM* dilatih pada data latih (70% data) dan melakukan pengujian performa pada data uji (30% data). Hasilnya *ELM* yang memiliki jumlah node pada lapisan tersembunyi sebanyak 2500 memiliki performa yang paling baik dibandingkan *ELM* yang memiliki node pada lapisan tersembunyi sebanyak 3000, 3500, dan 4000. Ini dibuktikan melalui nilai rata-rata presisi, *recall*, dan *F1-score* nya adalah 0.86 sedangkan akurasi ujinya sebesar 86%.

**Kata Kunci**—Klasifikasi, *Extreme Learning Machine*, Pengolahan Citra Digital, Tumor Otak, Sistem Diagnosis Komputer.

## I. PENDAHULUAN

Tumor otak merupakan penyakit yang menyerang otak makhluk hidup yang di mana sel otak tumbuh secara tidak normal pada daerah sekitar otak [1]. Adapun pertumbuhan ini disebabkan oleh mutasi DNA pada sel. Bila terjadi mutasi DNA sel yang seharusnya tumbuh dan mati pada suatu waktu justru masih hidup serta berkembang biak secara tidak terkendali hingga menjadi tumor [1]. Beberapa faktor yang diyakini sebagai peningkat resiko kanker otak antara lain bertambahnya usia, terkena paparan radiasi ionisasi/bom atom, serta riwayat keluarga pendahulu. Meski demikian, memiliki satu atau lebih faktor resiko bukan berarti seseorang tersebut pasti akan terkena tumor [1].

Berbagai macam cara dilakukan untuk mendeteksi dini keberadaan tumor otak bahkan sampai menggunakan dukungan peralatan digital, salah satunya dengan melalui pendekatan anatomi citra kesehatan [2]. Beberapa contoh citra kesehatan sendiri antara lain, sinar-X, *CT-Scan* dan *Magnetic Resonance Imaging (MRI)*. Kualitas citra pada sinar-X masih kurang bagus, sehingga banyak informasi yang tidak

didapatkan, pemanfaatan *CT-Scan* sendiri lebih cocok untuk melihat perubahan dalam struktur tulang. *MRI* sangat sensitif dan sukses memberikan informasi citra yang baik [3], sehingga *MRI* mampu memberikan gambaran yang jelas antara *soft tissue* dan *hard tissue* pada otak. Hasil pindaian dari *MRI* dapat memberikan informasi yang penting bagi dokter untuk mendiagnosis, melakukan perencanaan perawatan, dan evaluasi tindak lanjut. Namun diagnosis secara manual oleh manusia membutuhkan waktu yang tidak sebentar, sehingga diagnosis otomatis dengan bantuan komputer sangat dapat dimanfaatkan untuk membantu dokter dalam mengambil sebuah keputusan.

Pada penelitian ini penulis mengusulkan sebuah sistem diagnosis yang memanfaatkan metode klasifikasi pembelajaran mesin, yaitu *Extreme Learning Machine (ELM)*. Model pembelajaran *ELM* yang terbentuk bisa digunakan untuk mendeteksi keberadaan tumor otak dengan melakukan klasifikasi pada data citra *MRI* yang pada tahap pelatihannya memiliki dua kategori, yaitu tumor dan otak tanpa tumor. *ELM* dipilih karena memiliki tahap pelatihan yang singkat dan arsitektur lebih sederhana jika dibandingkan dengan metode berbasis jaringan saraf tiruan lainnya [4]. Beberapa skenario percobaan juga dilakukan, seperti memilih jumlah node pada lapisan tersembunyi di struktur *ELM* yang digunakan. Untuk melakukan pelatihan dan pengujian, penulis menggunakan data *Brain MRI Images for Tumor Detection* [5]. *Data set* tersebut berisikan citra digital *MRI* yang terbagi menjadi dua kelas yaitu tumor (*yes*) dan otak tanpa tumor (*no*).

Jika dibandingkan pada penelitian yang dilakukan oleh Febrianti dkk. [6] kontribusi yang kami tambahkan adalah menggunakan metode pembelajaran mesin yang berbeda, mempersingkat tahapan praproses, serta hasil pengujian yang kami dapatkan juga lebih baik.

## II. PENELITIAN TERDAHULU

Para peneliti telah melakukan penelitian terkait klasifikasi citra tumor otak dengan metode *ELM* ataupun metode lainnya pada *data set* yang sama. Berikut adalah penelitian terdahulu yang penulis jadikan acuan dalam penyusunan penelitian ini.

Penelitian yang dilakukan oleh Sharif dkk. [7] membahas mengenai pemanfaatan *Extreme Learning Machine* untuk mendeteksi keberadaan tumor otak. Sebelum masuk ke tahap

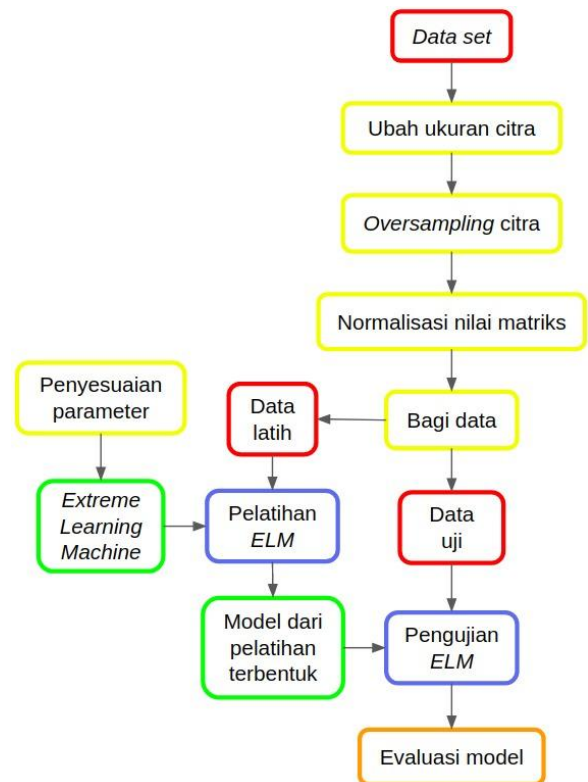
pelatihan dan klasifikasi, dilakukan tahapan praproses citra terlebih dahulu adapun metode praproses yang digunakan adalah *image enhancement*, segmentasi citra, *texture feature vector*, dan *1D Gabor filter*. Sharif dkk. juga melakukan tambahan pada *ELM* yang digunakan, yaitu dengan menambahkan metode *leave-one-out* untuk optimasi pelatihan. Penelitian ini dilakukan pada *data set* BraTS tahun 2012-2015 dengan hasil yang lebih baik dari pada penelitian-penelitian sebelumnya yang disinggung di penelitian tersebut. Untuk *data set* BraTS 2012 *ELM* usulan penulis mendapatkan akurasi 99%, BraTS 2013 akurasi yang dicapai 99%, BraTS 2014 87,6% akurasinya, serta pada *data set* BraTS 2015 mendapatkan akurasi 96,5%.

Khan M dkk. [8] pada penelitiannya melakukan percobaan klasifikasi multimodal pada *data set* BraTS tahun 2015, 2017, dan 2018. Adapun tahapan yang para peneliti tersebut lakukan ada lima. Yang pertama praproses citra dengan memanfaatkan metode *edge-based histogram equalization* dan *discrete cosine transform*. Kedua melakukan ekstraksi fitur yang diambil dari arsitektur *Convolutional Neural Network* (CNN) VGG16 dan VGG19. Ketiga, mengkombinasi metode *Correntropy Mutual Learning* dan *ELM* yang digunakan untuk seleksi fitur. Keempat, menginisialisasi kan sebuah matriks dengan metode *Partial Least Square*. Matriks yang terbentuk akan digabung dengan *ELM* untuk melakukan proses klasifikasi pada tahap akhir. Metode yang diusulkan menghasilkan akurasi uji sebesar 97,8%, 96,9%, 92,5% pada *data set* BraTS 2015, BraTS 2017, dan BraTS 2018.

*Data set* [5] yang penulis gunakan juga pernah digunakan oleh Febrianti dkk. dalam penelitiannya [6] di mana metode pembelajaran mesin yang mereka gunakan adalah *Support Vector Machine* (SVM). Sebelum masuk ke tahap klasifikasi oleh SVM *data set* yang berupa citra terlebih dahulu dipraproses adapun metode praproses yang digunakan antara lain *Gaussian blur*, *binary thresholding*, dilatasi dan erosi, mencari kontur terbesar, *extreme point* pada kontur, dan melakukan pemangkasan citra otomatis. Setelah dipraproses kemudian diekstrak fiturnya lalu diklasifikasikan dengan SVM *Classifier* (SVC) dan *nu-SVC*. Hasil akurasi tertinggi didapatkan oleh SVC yang memanfaatkan kernel linier dan *radial basis function*, yaitu 76%. Sementara *nu-SVC* hanya mendapatkan akurasi sebesar 72% saja.

### III. METODOLOGI PENELITIAN

Bagian ini merupakan penjelasan dari metode usulan kami yang digunakan untuk melakukan klasifikasi pada citra MRI tumor otak. Bagian ini meliputi penjelasan dari *data set* yang digunakan, praproses data, *Extreme Learning Machine*, evaluasi performa, serta skenario pengujian. Gbr. 1 adalah alur percobaan yang terdapat pada penelitian ini.



Gbr. 1 Ilustrasi alur percobaan

Pada tahap awal, citra dipraproses terlebih dulu dengan mengubah ukurannya, *oversampling* data yang persebarannya tidak merata, kemudian melakukan normalisasi citra yang direpresentasikan sebagai matriks. Setelah tahapan praproses selesai, *data set* yang sudah dalam bentuk matriks dibagi menjadi dua bagian, 70% data menjadi data latih sedangkan 30% sisanya menjadi data uji. *ELM* awalnya diatur dulu beberapa parameternya seperti jumlah node pada lapisan tersembunyi dan fungsi aktivasi apa yang digunakan. Kemudian *ELM* dapat langsung dilatih menggunakan data latih yang tersedia, jika proses pelatihan selesai model hasil pembelajaran dapat langsung diujikan performanya pada data uji. Setelah itu penulis dapat melakukan evaluasi dari performa *ELM*.

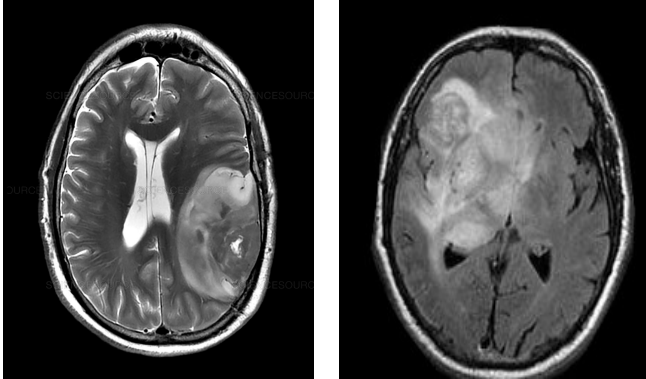
Spesifikasi perangkat keras yang digunakan oleh penulis antara lain Intel® Core™ i5-7200U 2.5-3.1 GHz sebagai prosesor dan RAM berukuran 12 giga byte. *ELM* dilatih pada CPU dengan sistem operasi Ubuntu 20.04. Adapun perangkat lunak pendukung penelitian adalah NumPy untuk membangun arsitektur *ELM* dan operasi matriks, Scikit-Image untuk mengolah data citra digital, serta Visual Studio Code yang berfungsi sebagai lingkungan pengembangan.

#### A. Data set

Kami menggunakan *data set Brain MRI Images for Tumor Detection* [5] di penelitian ini, di mana *data set* tersebut bersifat publik sehingga dapat dimanfaatkan oleh siapapun. *Data set* tersebut terdiri dari 253 citra MRI yang terlihat seperti grayscale namun pada dasarnya ada beberapa citra yang bukan grayscale (RGB). Melalui situs Kaggle, Navoneel

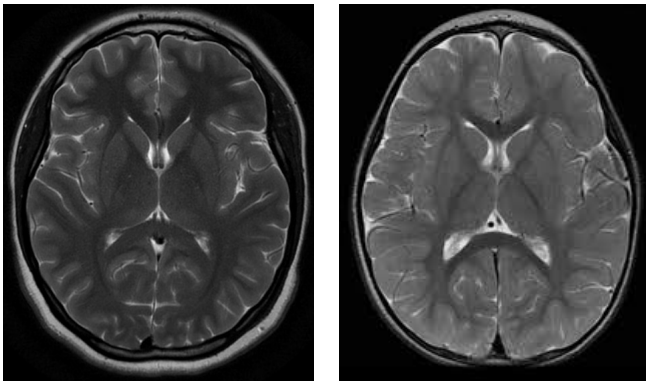
Chakrabarty mengunggah *Brain MRI Images for Tumor Detection data set* juga ia merupakan seorang yang mengumpulkan dan menyusun *data set* tersebut.

*Brain MRI Images for Tumor Detection data set* terdiri dari dua direktori utama yang masing-masing direktori merupakan representasi kategori dari citra tersebut. Kategori yang tersedia adalah Tumor (*yes*) dan Otak sehat (*no*). Direktori Otak sehat memiliki data sejumlah 98 sedangkan direktori Tumor memiliki 155 data citra *MRI*. Berikut adalah beberapa contoh data citra dari *data set* yang kami gunakan :



Gbr. 2 Citra *MRI* dengan tumor otak

Gbr. 2 merupakan contoh data citra dari kategori Tumor. Setidaknya citra pada kategori tersebut memiliki jumlah 61,3% dari *data set* keseluruhan.



Gbr. 3 Citra *MRI* otak (sehat) tanpa tumor

Adapun Gbr. 3 merupakan contoh data citra dari kategori Otak sehat. Citra pada kategori tersebut memiliki jumlah 38,7% dari *data set* keseluruhan.

Ukuran dari seluruh citra sangatlah beragam oleh karena itu diperlukan tahapan praproses sebelum masuk ke tahapan klasifikasi citra, agar hasil yang didapatkan nantinya menjadi baik [9].

#### B. Praproses Citra

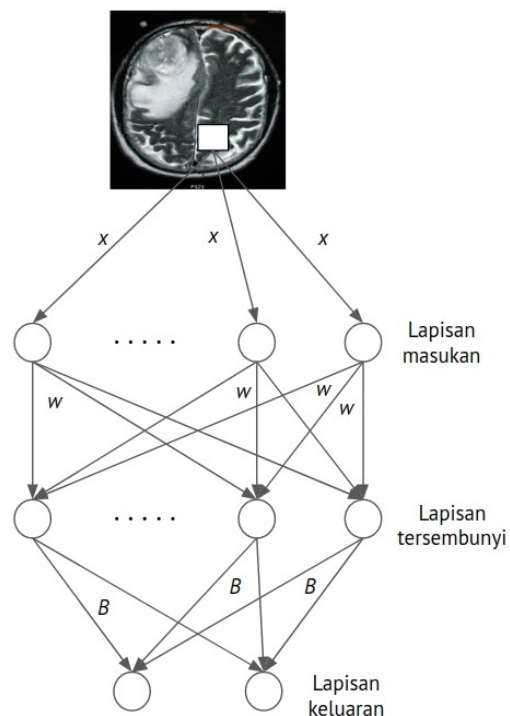
Untuk melakukan klasifikasi pada data citra oleh algoritma pembelajaran mesin, seluruh citra harus melewati tahapan praproses terlebih dahulu [9]. Tahap awal yang kami lakukan adalah mengkonversi tiap citra ke dalam bentuk matriks kemudian berubah ukurannya menjadi 50x50 piksel. Adapun citra yang memiliki 3 *channel* (*RGB*) maka kami ubah ke dalam bentuk *grayscale* (1 *channel*). Mengingat persebaran jumlah data dari tiap kategori citra tidak sama kami

menerapkan teknik *oversampling* data guna menyamakan jumlah data dari kategori yang memiliki data sedikit [10]. Dalam hal ini kami melakukan *oversampling* ( $180^\circ$  *flip*) pada citra dengan kategori Tidak memiliki tumor sehingga dari jumlah citra yang awalnya 98 menjadi 155, jumlahnya sama seperti citra dengan kategori ada Tumor otak. Akhirnya total keseluruhan data setelah dilakukan *oversampling* adalah 310 data. Setelah itu nilai tiap elemen pada matriks kami normalisasikan dengan mengubah rentangnya. Di mana rentang awal adalah 0-255 kami ubah menjadi 0-1.

Lalu, setelah dilakukan tahapan praproses kami bagi data menjadi dua bagian, yaitu data latih dan data uji. Di mana data latih yang terdiri dari 217 citra (70% data) digunakan untuk melatih model *ELM* sedangkan data uji yang terdiri dari 93 citra (30% data) digunakan untuk menguji performa model *ELM* yang terlatih.

#### C. Extreme Learning Machine

*ELM* merupakan metode jaringan saraf tiruan yang proses pelatihannya tidak memerlukan propagasi balik sehingga proses pelatihan dapat lebih cepat karena tidak perlu memperbarui nilai bobot dan bias pada tiap epoch nya [4]. Proses pelatihan dengan menggunakan metode propagasi balik dapat memakan waktu karena sifatnya yang iteratif. Oleh karenanya diperlukan metode pelatihan yang sifatnya yang tidak iteratif (*single feed forward pass*), cepat, dan juga menghasilkan akurasi yang tidak kalah baik dengan propagasi balik, maka *ELM* keluar untuk memberi solusi [4]. Parameter yang dapat diatur selain pemilihan fungsi aktivasi pada *ELM* adalah jumlah node pada lapisan tersembunyi, yang jumlahnya dapat disesuaikan oleh pemrogram.



Gbr. 4 Ilustrasi arsitektur *ELM* usulan penulis

Pada *ELM* nilai bobot antara lapisan masukan dan lapisan tersembunyi diinisialisasikan secara acak. Rumus perhitungan antara lapisan masukan dan lapisan tersembunyi pada *ELM*



sama sebagaimana perhitungan *feed forward pass* pada jaringan saraf tiruan pada umumnya. Bobot yang dilatih adalah bobot antara lapisan tersembunyi dan lapisan keluaran. Adapun rumus perhitungan untuk mencari nilai bobot antara lapisan keluaran dan lapisan tersembunyi adalah :

$$B = (X^T X)^{-1} X^T y \quad (1)$$

di mana :

$X$  adalah nilai hasil perhitungan dari lapisan tersembunyi

$X^T$  adalah nilai  $X$  yang telah ditranspose

$y$  adalah nilai target/label pada data latih ( $y_{train}$ ).

#### D. Evaluasi Performa

Evaluasi performa dibutuhkan untuk menilai seberapa baik model pada tahap pengujian. Dalam penelitian ini kami menggunakan matriks konfusi untuk mengevaluasi performa dari *ELM*. Dari sebuah matriks konfusi kita bisa mendapatkan nilai-nilai seperti *true positive* (TP), *false positive* (FP), *true negative* (TN), dan *false negative* (FN). Setiap nilai yang disebutkan dapat digunakan untuk mendapatkan presisi, F1, *recall* dan akurasi [11]. Persamaan (2) - (5) di bawah ini menunjukkan fungsi dari keempat istilah tersebut :

1. Presisi adalah prediksi rasio TP dibanding keseluruhan prediksi positif

$$\frac{TP}{TP + F} \quad (2)$$

2. Recall adalah prediksi rasio TP dibanding keseluruhan data yang benar

$$\frac{TP}{TP + F} \quad (3)$$

3. F1-score adalah perbandingan sebuah rata-rata presisi dan *recall*

$$\frac{2 \cdot (Recall \cdot Presisi)}{Recall + Presisi} \quad (4)$$

4. Akurasi adalah hasil prediksi yang benar dari seluruh data uji

$$\frac{TP + TN}{TP + FP + FN + TN} \quad (5)$$

#### E. Skenario Pengujian

Pengujian dilakukan pada 93 citra yang terdapat pada data uji. Urutan dari citra pada tiap pengujian selalu acak. Pengujian dilakukan dengan empat skenario yang berbeda disesuaikan dengan pilihan jumlah node pada lapisan tersembunyi *ELM*. Adapun jumlah node di lapisan tersembunyi pada pengujian pertama adalah 2500 node (sama dengan jumlah node pada lapisan masukan), pengujian kedua 3000 node, pengujian ketiga 3500 node, dan pengujian keempat adalah 4000 node. Perlu diingat bahwa semakin

banyak node pada lapisan tersembunyi semakin banyak pula memori (*RAM*) yang harus dialokasikan oleh komputer [12].

#### IV. HASIL DAN PEMBAHASAN

Hasil yang dibahas pada bab ini merupakan hasil pengujian dari model *ELM* yang telah diujikan melalui data uji. Hasil pengujian tersebut sebelum disajikan sebagaimana TABEL 1 adalah berbentuk matriks konfusi, yang telah penulis jelaskan pada bab sebelumnya. Mengingat cara kerja *ELM* hanyalah *single feed forward pass* maka evaluasi metrik di sini hanya ditampilkan dengan melalui sebuah tabel yang sederhana. Berikut adalah hasil dari pengujian tersebut :

TABEL I  
EVALUASI METRIK PENGUJIAN

Jumlah Node	Kategori	Presisi	Recall	F1	Akurasi (%)
2500	Sehat	0.85	0.87	0.86	86
	Tumor	0.87	0.85	0.86	
3000	Sehat	0.74	0.76	0.75	75,3
	Tumor	0.76	0.74	0.75	
3500	Sehat	0.82	0.84	0.83	81,8
	Tumor	0.81	0.79	0.8	
4000	Sehat	0.84	0.73	0.78	78,5
	Tumor	0.74	0.84	0.79	

Pada pengujian dengan jumlah node 2500 yang terdapat di lapisan tersembunyi menghasilkan presisi, *recall*, dan *F1-score* rata-rata per kategori sebesar 0.86 sedangkan akurasi 86%. Untuk *ELM* dengan 3000 node pada lapisan tersembunyi memiliki rata-rata nilai presisi, *recall*, dan *F1-score* sebesar 0.75 dengan akurasi 75,3%. Adapun *ELM* dengan 3500 node pada lapisan tersembunyi memiliki akurasi 81,8% dengan nilai presisi, *recall*, dan *F1-Score* beragam pada dua kategori. Struktur *ELM* yang memiliki 4000 node pada lapisan tersembunyi mendapatkan akurasi sebesar 78,5% dengan nilai presisi, *recall*, dan *F1-Score* yang beragam pula pada kategori Tumor dan (Otak) Sehat.

Berdasarkan TABEL I beserta penjelasannya, dapat ditarik kesimpulan bahwa *ELM* yang memiliki performa terbaik untuk melakukan klasifikasi dalam penelitian ini adalah yang memiliki node lapisan tersembunyi dengan jumlah 2500. Kami menyadari bahwa hal tersebut itu masih belum bisa dijadikan standar dalam menyusun struktur *ELM*. Karena masih terdapat inisialisasi nilai secara acak pada bobot dan bias serta urutan data yang acak pula pada tiap simulasi/program pertama kali dijalankan.

#### V. KESIMPULAN DAN SARAN

Adapun kesimpulan yang dapat diperoleh adalah ada beberapa teknik praproses citra yang berperan signifikan pada hasil akhir percobaan seperti mengubah ukuran citra dan mengubahnya ke *grayscale*, *oversampling* data, serta normalisasi nilai. Parameter yang mempengaruhi dari

performa *ELM* yang dapat diatur selain pemilihan fungsi aktivasi adalah jumlah node pada lapisan tersembunyi. Hasil akhirnya *ELM* dengan node sejumlah 2500 pada lapisan tersembunyi mampu mendapatkan nilai rata-rata presisi, *F1-score*, dan *recall* 0.86 serta akurasi sebesar 86%. Jika dibandingkan jumlah node yang lainnya maka *ELM* dengan 2500 node lapisan tersembunyi memiliki performa paling baik.

Berdasarkan hasil implementasi yang telah dilakukan, terdapat beberapa saran yang perlu diperhatikan dalam mengembangkan penelitian ini kedepannya, yaitu data yang digunakan sekiranya dapat lebih bervariasi lagi. Adapun untuk praprosesnya sendiri, dapat ditambahkan tahapan segmentasi citra. Pada struktur *ELM* nya dapat ditambahkan proses ekstraksi fitur sebelum masuk ke tahap klasifikasi dengan memanfaatkan algoritma lain, misalnya *CNN*, guna mendapatkan hasil evaluasi yang lebih baik. Juga model pembelajaran yang terlatih bisa dibenamkan pada sebuah aplikasi sistem diagnosis daring agar tidak berhenti pada tahap penelitian saja.

#### UCAPAN TERIMA KASIH

Kami ucapkan terima kasih kepada seluruh pihak yang telah banyak membantu kami dalam penelitian ini baik secara materi maupun non materi. *Jazakumullahu khairan*.

#### REFERENSI

- [1] Fadila, I. (2020). Tumor Otak : Gejala, Penyebab, dan Pengobatan. Hello Sehat.
- [2] Suta, I., Hartati, R., Divayana, Y. (2019). Diagnosa Tumor Otak Berdasarkan Citra MRI (Magnetic Resonance Imaging). Bali : Majalah Ilmiah Teknologi Elektro Universitas Udayana.
- [3] R. Riley, J. Murphy, and T. Higgins, "MRI imaging in pediatric appendicitis," *J. Pediatr. Surg. Case Reports*, vol. 31, no. January, pp. 88–89, 2018.
- [4] Guang-Bin Huang, Qin-Yu Zhu, and Chee-Kheong Siew 2006. Extreme learning machine: Theory and applications. *Neurocomputing*, 70(1), p.489 - 501.
- [5] N. Chakrabarty, "Brain MRI Images for Brain Tumor Detection." <https://www.kaggle.com/navoneel/brain-mri-images-for-brain-tumor-detection>.
- [6] Febrianti, Ainani Shabrina, et al. "Klasifikasi Tumor Otak Pada Citra Magnetic Resonance Image Dengan Menggunakan Metode Support Vector Machine." *Jurnal Teknik ITS*, vol. 9, no. 1, July 2020, pp. A118–23. DOI.org (Crossref), doi:10.12962/j23373539.v9i1.51587.
- [7] Sharif, M., Amin, J., Raza, M., Anjum, M. A., Afzal, H., & Shad, S. A. (2020). *Brain tumor detection based on extreme learning*. *Neural Computing and Applications*. doi:10.1007/s00521-019-04679-8
- [8] Khan, Muhammad & Ashraf, Imran & Alhaisoni, Majed & Damasevicius, Robertas & Scherer, Rafal & Rehman, Amjad & Ahmad, Syed & Bukhari, Syed Ahmad Chan. (2020). Multimodal Brain Tumor Classification Using Deep Learning and Robust Feature Selection: A Machine Learning Application for Radiologists. *Diagnostics*. 10. 565. 10.3390/diagnostics10080565.
- [9] Pal, K.K., & Sudeep, K.S. (2016). Preprocessing for image classification by convolutional neural networks. *2016 IEEE International Conference on Recent Trends in Electronics, Information & Communication Technology (RTEICT)*, 1778-1781.
- [10] Mikołajczyk, A., and Grochowski, M. 2018. Data augmentation for improving deep learning in image classification problem. In *undefined* (pp. 117-122).
- [11] Powers, David & Ailab. (2011). Evaluation: From precision, recall, and F-measure to ROC, informedness, markedness & correlation. *Journal of Machine Learning Technologies*, BioInfo Publications
- [12] Hanlon, Jamie. 2016. "Why is so Much Memory Needed for Deep Neural Networks?" Graphcore. <https://www.graphcore.ai/posts/why-is-so-much-memory-needed-for-deep-neural-networks>