

# ANALISIS GAMBAR MRI OTAK UNTUK MENDETEKSI TUMOR OTAK MENGGUNAKAN ALGORITMA CNN

Valliant Benvenuto Gianzurriell<sup>1</sup>, Ferdi Husnal<sup>2</sup>, Fiky Ari Wijaya<sup>3</sup>, Fahmi Fauzi<sup>4</sup>, Iman Paryudi<sup>5</sup>, Ionia Veritawati<sup>6</sup>, Sri Rezeki Candra Nursari<sup>7</sup>

Program Studi Teknik Informatika  
Fakultas Teknik Universitas Pancasila  
Jl. Srengseng Sawah, Jagakarsa, Jakarta Selatan, 12640

[valliantbenvenutog@gmail.com](mailto:valliantbenvenutog@gmail.com), [ferdihusnal25@gmail.com](mailto:ferdihusnal25@gmail.com), [fikiariwijaya@gmail.com](mailto:fikiariwijaya@gmail.com),  
[4518215001@univpancasila.ac.id](mailto:4518215001@univpancasila.ac.id), [iman.paryudi@univpancasila.ac.id](mailto:iman.paryudi@univpancasila.ac.id), [ionia.veritawati@univpancasila.ac.id](mailto:ionia.veritawati@univpancasila.ac.id),  
[sri.rezeki.candra.n@univpancasila.ac.id](mailto:sri.rezeki.candra.n@univpancasila.ac.id)

**Abstrak**— Penyakit tumor otak merupakan satu dari sekian penyakit yang paling mematikan. Penyakit ini bermula dari berkembangnya sel abnormal pada jaringan otak manusia adalah tanda penyakit ini. Teknologi citra digital yang disebut *pencitraan resonansi magnetic* (MRI) dapat diperuntukan guna mendeteksi tumor otak ini. Teknologi ini dimaksudkan untuk membantu para dokter mengidentifikasi dan mengklasifikasikan berbagai jenis tumor otak. Dibutuhkan metode yang efektif dan akurat untuk melakukan klasifikasi citra MRI. Salah satu solusi yang efektif untuk masalah ini adalah *Convolutional Neural Network* (CNN). CNN merupakan suatu algoritma yang dapat belajar sendiri dari kasus sebelumnya. Metode deep learning, seperti CNN dengan model VGG16, dapat digunakan secara baik untuk menyelesaikan kasus ini. Proses pada penelitian ini melalui beberapa tahapan pembuatan *Image Detection* yaitu akuisisi gambar, *preprocessing*, ekstraksi, klasifikasi, dan identifikasi data gambar. Penelitian ini menggunakan 3 dataset dimana setiap dataset memiliki gambar hasil MRI pasien sebanyak 1311 gambar. Dalam dataset tersebut dipisahkan menjadi 3 data yang berbeda yaitu data train, data validasi, dan data uji. Hasil dari pengujian ketiga dataset ini mampu mengidentifikasi gambar yang di uji ke dalam sistem dengan persentase keakuratan 99%.

**Kata Kunci**—Tumor Otak; MRI; CNN; Deep Learning; Klasifikasi; Citra Digital

## I. PENDAHULUAN

Tumor otak merupakan tumbuhnya sel dalam atau sekitar otak yang abnormal, yang mana hal tersebut tidaklah wajar karena dapat menyebabkan kematian. Tumor otak mempunyai dua jenis diantaranya tumor primer dan tumor sekundär[1]. Penyakit ini dikenal karena keganasannya, dan berbagai teknik telah digunakan untuk mendeteksinya sedini mungkin, salah satunya adalah gambar MRI digital.

Salah satu cara untuk mengidentifikasi keberadaan tumor pada otak adalah dengan menggunakan teknik anatomi citra kesehatan. Contohnya adalah citra tomografi komputasi (CT) dan pencitraan resonansi magnetik (MRI). CT-scan efektif untuk memberikan gambaran struktur

tulang. MRI, di sisi lain, dapat memberitahu suatu data yang jelas tentang jaringan keras dan lunak otak, yang memungkinkan dokter untuk melakukan evaluasi dan dokter sering menggunakan biopsi dan observasi langsung saat diagnosis manual untuk membuat keputusan tentang tumor otak[2].

Salah satu jenis *deep learning* adalah algoritma *convolutional neural network* (CNN) ini. Jenis pembelajaran komputer yang disebut *deep learning* memiliki kemampuan untuk mengarahkan perangkat keras untuk melaksanakan tugas seperti manusia[3]. CNN ideal untuk pemrosesan gambar karena terinspirasi oleh sistem saraf biologis[4] dan menampilkan setiap neuron dalam 2D[5]. CNN beroperasi secara hierarki, artinya input dari *layer* konvolusi pertama digunakan untuk masukkan dari *layer* konvolusi berikutnya. Proses klasifikasi melibatkan fungsi aktivasi softmax dan penuh terhubung, yang menghasilkan hasil klasifikasi[6].

## II. LANDASAN TEORI

### A. Tumor Otak

Tumor otak bermula dari berkembangnya sel yang tidak normal pada otak. Terdapat dua jenis tumor otak yaitu primer dan sekunder. Tumor otak primer merupakan pertumbuhan sel yang abnormal yang berawal dari sel otak itu tersebut. Di sisi lain, tumor otak sekunder ialah tumor yang bermula dari kanker tubuh lain ke otak. Glioma, meningioma, dan medulloblastoma adalah beberapa jenis tumor otak primer[1]. Pasien yang mengidap penyakit ini akan merasakan sakit kepala yang tidak berhenti, gangguan perilaku, kejang, mual, muntah, dan gangguan neurokognitif[7].

## B. Deep Learning

*Deep learning* merupakan bagian dari *AI* yang merupakan pengembangan dari jaringan neural berlapis berlapis untuk mendeteksi objek, seperti pengenalan suara dan terjemahan bahasa. Dalam pembelajaran mesin, *deep learning* adalah sekumpulan algoritma yang akan belajar dalam banyak tingkat, sesuai dengan berbagai tingkat abstraksi[8].

Dalam model statistik yang dipelajari, tingkat konsep dimulai dengan tingkat di bawah dan yang ada di atasnya dapat membantu mendefinisikan konsep yang berbeda pada tingkat yang berbeda. Selain itu, *deep learning* adalah subbidang *machine learning* dengan algoritma yang diambil dari *jaringan neural buatan*, sebuah struktur dalam otak manusia[9].

Terdapat dua jenis *deep learning* sebagai berikut:

1. *Deep learning* untuk *Unsupervised learning*, dimana label tanpa variabel tujuan dan nilai korelasi yang lebih besar harus dicari dari unit yang terlihat untuk memeriksa polanya, metode pengajaran mendalam ini digunakan.
2. *Hybrid deep learning*. Metode ini berfungsi untuk mencapai hasil yang lebih bagus dengan *supervised learning* untuk menganalisa pola atau pembelajaran tanpa pengawasan.

*Deep Learning* memiliki banyak macamnya, yang paling populer adalah *Convolutional Neural Network (CNN)*, *Multilayer Perceptrons (MLP)*, dan *Recurrent Neural Network (RNN)*. Memiliki kemampuan terbaik untuk menyelesaikan masalah yang rumit adalah keuntungan dari *deep learning*, menurunkan keperluan akan rekayasa karakteristik dan memiliki arsitektur yang dapat dengan mudah menyesuaikan diri dengan masalah baru.

## C. Convolutional Neural Network (CNN)

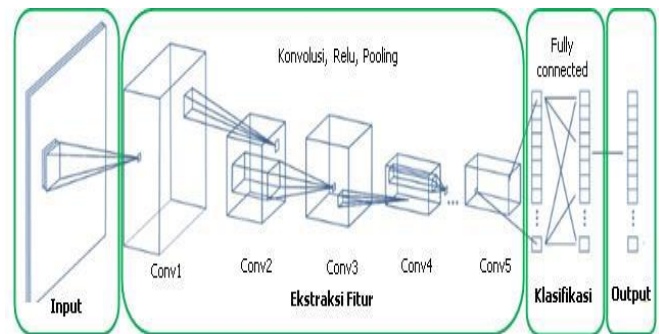
Algoritma ini tersusun lebih dari satu lapisan serta biasanya diperuntukan guna memproses data berbentuk *image* atau objek citra[8]. *CNN* merupakan algoritma yang dibuat spesial untuk mengatasi masalah dengan pengolahan gambar. Karena arsitekturnya sama dengan bagaimana otak manusia memproses gambar, sehingga *CNN* dapat paham akan detail bentuk dengan apik. *CNN* menggunakan data 2D, diantaranya gambar, suara, dan menggunakan perhitungan konvolusi untuk matriks dan bobot 4D[9].

Salah satu jenis *neural network* yaitu *CNN*, dibuat untuk menangani masalah pengolahan gambar. Karena arsitekturnya yang seperti yang dilakukan oleh otak manusia untuk memproses data visual, *CNN* dapat menggolong-golongkan, menemukan, dan mengidentifikasi pola dalam gambar. *CNN* hanya diperuntukan data dengan struktur data dua dimensi. karena *CNN* menggunakan operasi konvolusi dalam matriks dan sekumpulan kernel konvolusi empat dimensi yang terdiri dari bobot[9].

Gambar 1 Arsitektur CNN

## D. Relu

Relu atau *Rectification Linear Unit* adalah perhitungan yang mengenalkan yang tidak linearitas serta memperbaiki ilustrasi model. Fungsi aktivasi ReLU adalah  $f(x) = \max(0, x)$ [10]. Hasil akhir dari *neuron* dapat dikatakan 0 jika nilai masukannya nilai yang negatif. Akan tetapi, jika nilai



Gambar 1 Pooling

masukannya positif, maka nilai keluarannya adalah nilai input aktivasi itu sendiri[11].

## E. Pooling

*Pooling*, juga dikenal sebagai *sub-sample*, merupakan Langkah untuk mengurangi dimensi matriks. Jenis *pooling* yang paling umum adalah *average pooling* dan *max pooling*[12]. Nilai maksimum gabungan adalah nilai maksimal, sedangkan nilai rata-rata gabungan adalah nilai rata-rata[13].

## F. Fully Connected Layer

Lapisan terhubung sepenuhnya adalah kelompok proses konvolusi[14] yang berfungsi untuk menggabungkan semua node ke dalam satu dimensi[15], dan menerima masukan dari prosedur sebelumnya untuk mengidentifikasi fitur yang paling terkait dengan kelas tertentu.

## G. Fungsi Aktivasi Softmax

Supaya menghasilkan klasifikasi, maka langkah ini digunakan. Fungsi tersebut mampu memberikan hasil yang dianggap sebagai kemungkinan yang belum distandardisasi pada setiap kelasnya[16].

## III. METODOLOGI PENELITIAN

Dalam penelitian ini penulis melakukan langkah-langkah sistematis antara lain:

### A. Studi Literatur

Penelitian ini salah satu metode yang digunakan adalah analisis literatur dimana penulis mencari sumber bacaan

rujukan yang berkaitan langsung terhadap topik yang penulis angkat yang berdasarkan jurnal.

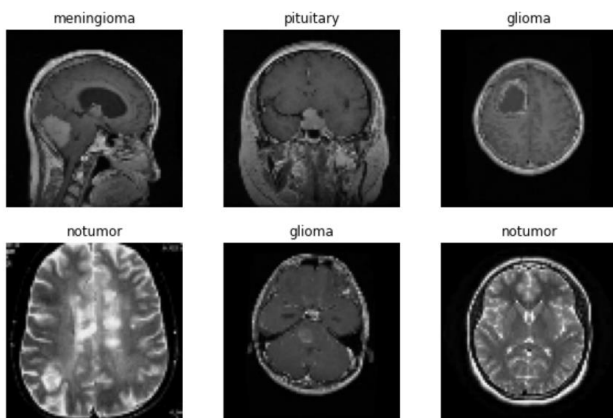
**B. Identifikasi Masalah**

Mengidentifikasi masalah adalah langkah pertama menuju masalah yang akan diangkat, algoritma yang akan digunakan, dan rekomendasi.

**C. Pengumpulan Data**

Dalam penelitian ini data masukan adalah data MRI penyakit tumor otak. Data yang digunakan untuk penelitian ini dikumpulkan menggunakan Kaggle. Data yang dibutuhkan untuk proses deteksi otak terdiri dari tiga bagian: latihan (latihan), tes, dan evaluasi. Latihan digunakan untuk mengajar program dan disimpan dalam jumlah besar; tes adalah data yang digunakan untuk membandingkan latihan dan tes yang dipelajari; dan evaluasi adalah data yang digunakan untuk membandingkan latihan dan tes yang dipelajari. Hasil dari pembandingan ini akan digunakan untuk proses deteksi otak.

Program akan dilatih dengan lebih banyak data yang digunakan untuk meningkatkan akurasi. Data tersebut berasal dari gambar MRI otak pasien dan dikumpulkan dalam satu folder yang disebut dataset. Setiap dataset dibagi menjadi beberapa folder berdasarkan nama penyakit yang termasuk dalam kategori tumor otak. jumlah nama penyakit yang diambil sebanyak 4 dengan jumlah data 1311.



Gambar 2 Dataset MRI

**D. Analysis**

Pada tahap ini dimulai dengan mendefinisikan kebutuhan dari apa yang diperlukan dalam permasalahan yang diangkat yaitu mengenai tumor otak. Pada analisis ini penulis menggunakan *Convolutional Neural Network* sebagai algoritma yang paling tepat untuk data. Pada penelitian analisis ini tahapan yang dilakukan untuk mendapatkan *output* yang diinginkan melalui beberapa tahapan yaitu sebagai berikut:

- Load Dataset

```
In [2]: #input dataset
data_train_path = 'dataset_tumor/train'
data_test_path = 'dataset_tumor/test'
data_val_path = 'dataset_tumor/valid'

In [3]: #Resize Image
img_width = 80
img_height = 80

In [4]: data_train = tf.keras.utils.image_dataset_from_directory(
    data_train_path,
    shuffle=True,
    image_size=(img_width, img_height),
    batch_size=32,
    validation_split=False)

Found 1311 files belonging to 4 classes.

In [5]: data_cat = data_train.class_names

In [6]: data_cat
Out[6]: ['glioma', 'meningioma', 'notumor', 'pituitary']
```

Gambar 3 Load Dataset

- Model

```
In [10]: from tensorflow.keras.models import Sequential
In [11]: data_train
Out[11]: <BatchDataset element_spec=(TensorSpec(shape=(None, 80, 80, 3), dtype=tf.float32, name=None), TensorSpec(shape=(None, 1), dtype=tf.int32, name=None))>

In [12]: model = Sequential([
    layers.Rescaling(1./255),
    layers.Conv2D(16, 3, padding='same', activation='relu'),
    layers.MaxPooling2D(),
    layers.Conv2D(32, 3, padding='same', activation='relu'),
    layers.MaxPooling2D(),
    layers.Conv2D(64, 3, padding='same', activation='relu'),
    layers.MaxPooling2D(),
    layers.Flatten(),
    layers.Dropout(0.2),
    layers.Dense(128),
    layers.Dense(len(data_cat))
])

In [19]: model.compile(optimizer='adam', loss='sparse_categorical_crossentropy', metrics=['accuracy'])

In [14]: epochs_size = 20
history = model.fit(data_train, validation_data=data_val, epochs=epochs_size)
```

Gambar 4 Pemodelan Dataset

- Training

```
In [14]: epochs_size = 20
history = model.fit(data_train, validation_data=data_val, epochs=epochs_size)

Epoch 1/20
41/41 [=====] - 15s 285ms/step - loss: 1.1404 - accuracy: 0.5050 - val_loss: 0.8822 - val_accuracy: 0.6575
Epoch 2/20
41/41 [=====] - 14s 344ms/step - loss: 0.6817 - accuracy: 0.7216 - val_loss: 0.5473 - val_accuracy: 0.7796
Epoch 3/20
41/41 [=====] - 12s 295ms/step - loss: 0.5444 - accuracy: 0.7895 - val_loss: 0.6378 - val_accuracy: 0.7929
Epoch 4/20
41/41 [=====] - 11s 255ms/step - loss: 0.4657 - accuracy: 0.8177 - val_loss: 0.3372 - val_accuracy: 0.8772
Epoch 5/20
41/41 [=====] - 11s 258ms/step - loss: 0.3782 - accuracy: 0.8505 - val_loss: 0.2626 - val_accuracy: 0.9376
Epoch 6/20
41/41 [=====] - 11s 265ms/step - loss: 0.2713 - accuracy: 0.8932 - val_loss: 0.2762 - val_accuracy: 0.8968
Epoch 7/20
41/41 [=====] - 11s 261ms/step - loss: 0.2270 - accuracy: 0.9207 - val_loss: 0.1829 - val_accuracy: 0.9321
Epoch 8/20
41/41 [=====] - 12s 283ms/step - loss: 0.1614 - accuracy: 0.9405 - val_loss: 0.1284 - val_accuracy: 0.9527
Epoch 9/20
```

Gambar 5 Training Dataset

- Identify

```
In [47]: from keras.models import load_model

In [48]: # ..... Load Keras CNN model .....
model = load_model('BrainImage_classify.h5')
print("[INFO] finish load model...")
[INFO] finish load model...

In [50]: #First Image
image = 'C:/Users/USER/DESKTOP/2024/X-ray_detection/MRI.jpg'
img = tf.keras.utils.load_img(image, target_size=(img_height, img_width))
img_arr = tf.keras.utils.array_to_img(img)
img_batch = tf.expand_dims(img_arr, 0)

In [51]: predict = model.predict(img_batch)
1/1 [=====] - 0s 120ms/step

In [52]: score = tf.nn.softmax(predict)

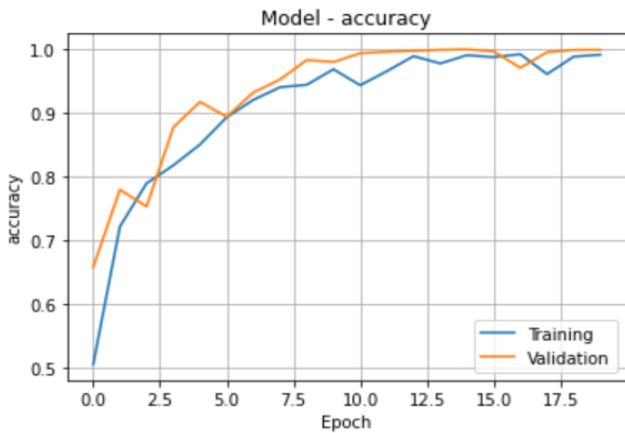
In [53]: print("MRI Brain detection in Image is {} with accuracy of {:.2f}.".format(data_cat[np.argmax(score)], np.max(score)*100))
MRI Brain detection in Image is notumor with accuracy of 99.99

In [54]: model.save('BrainImage_classify.h5')
```

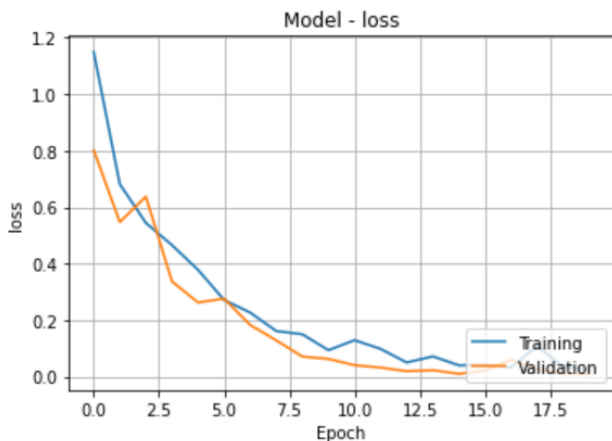
Gambar 6 Identify Dataset

IV. HASIL DAN PEMBAHASAN

Hasil penelitian menunjukkan bahwa tingkat pembelajaran model yang dibangun mencapai nilai akurasi sebesar 99%. Pengujian yang dilakukan pada penelitian ini menggunakan epoch 20.



Gambar 7 Model-Accuracy



Gambar 8 Model-Loss

Gambar 7 dan gambar 8 menunjukkan bahwa nilai akurasi berubah seiring waktu selama pelatihan. Ini karena pada awal pelatihan, model masih sering salah memprediksi hasil. Namun, beberapa saat kemudian, model menjadi lebih baik karena telah menemukan pola yang tepat sehingga kemampuan untuk meramalkan *output* menjadi lebih baik. Nilai kehilangan akurasi dan validasi yang baik hampir nol, dan nilai kehilangan validasi dan akurasi yang lebih kecil berarti model tersebut lebih baik dalam memprediksi hasil yang benar.

Berdasarkan penelitian yang telah dilakukan, berikut penulis uraikan berupa tabel yang didalamnya menampilkan total epoch dengan nilai los dan akurasi serta nilai val\_loss dan val\_akurasi yang didapatkan. Proses pelatihan *Neural Network* telah melewati semua data saat ini sehingga secara bertahap dikembalikan ke awal, yang merupakan parameter

penting dalam pengolahan citra yang menghasilkan nilai akurasi tertentu[17],[18].

Dari tabel tersebut dapat dilihat bahwa hasil akurasi terbesar terdapat pada total epoch yang digunakan yaitu dengan 99%.

Table 1 Perbandingan Hasil Nilai Epoch

Total Epoch	Loss	akurasi	Val_Loss	Val_akurasi
1	1.1494	0.5050	0.8022	0.8022
2	0.6817	0.7216	0.5473	0.7796
3	0.5444	0.7895	0.6378	0.7529
4	0.4657	0.8177	0.3372	0.8772
5	0.3782	0.8505	0.2626	0.9176
6	0.2713	0.8932	0.2762	0.8940
7	0.2270	0.9207	0.1829	0.9321
8	0.1614	0.9405	0.1284	0.9527
9	0.1501	0.9443	0.0710	0.9832
10	0.0936	0.9687	0.0628	0.9802
11	0.1293	0.9436	0.0407	9939
12	0.0982	0.9657	0.0321	0.9962
13	0.0501	0.9893	0.0191	0.9977
14	0.0716	0.9779	0.0229	0.9992
15	0.0402	0.9908	0.0102	1.0000
16	0.0413	0.9878	0.0210	0.9969
17	0.0325	0.9924	0.0603	0.9710
18	0.1090	0.9611	0.0247	0.9954
19	0.0352	0.9886	0.0119	0.9992
20	0.0235	0.9916	0.0100	0.9992

V. PENUTUP

A. Kesimpulan

Penelitian ini menghasilkan sebuah simpulan terkait Image Detection dengan dilakukannya sebuah pengujian maka

dihasilkan anatar lain, gambar MRI Otak untuk mendeteksi tumor dapat dilakukan dengan algoritma *Convolutional Neural Network* (CNN). Akurasi jaringan dapat dipengaruhi oleh jumlah data pelatihan. Semakin banyak data pelatihan, jaringan akan belajar lebih banyak, meningkatkan ketelitian. Untuk melakukan ini, bagaimanapun, diperlukan peralatan komputasi yang besar. Hasil pelatihan dengan epoch 20 menunjukkan nilai akurasi sebesar 99% pada model yang dibangun.

Untuk membantu interpreter mengidentifikasi jenis objek tumor otak, penelitian ini hanya menggunakan klasifikasi semantik. Penerjemah memasukkan sampel data untuk menentukan klasifikasinya, dan sebagai hasilnya, jaringan menghasilkan nama objek. Ini bisa meringankan penerjemah yang tidak tahu tentang kondisi kesehatan otak untuk mempertimbangkan tindakan medis selanjutnya apakah ada tumor otak.

#### DAFTAR PUSTAKA

- [1] I. B. L. M. Suta, R. S. Hartati, and Y. Divayana, "Diagnosa Tumor Otak Berdasarkan Citra MRI (Magnetic Resonance Imaging)," *Majalah Ilmiah Teknologi Elektro*, vol. 18, no. 2, Jun. 2019, doi: 10.24843/mite.2019.v18i02.p01.
- [2] R. Andre, B. Wahyu, and R. Purbaningtyas, "KLASIFIKASI TUMOR OTAK MENGGUNAKAN CONVOLUTIONAL NEURAL NETWORK DENGAN ARSITEKTUR EFFICIENTNET-B3," 2021. [Online]. Available: <https://jurnal.umj.ac.id/index.php/just-it/index>
- [3] Deng, L., & Yu, D. (2013). Deep Learning: Methods and Applications. *Foundations and Trends® in Signal Processing*, 7(3–4), 197–387. <https://doi.org/10.1136/bmj.319.7209.0a>
- [4] Hu, F., Xia, G. S., Hu, J., & Zhang, L. (2015). Transferring deep convolutional neural networks for the scene classification of high-resolution remote sensing imagery. *Remote Sensing*, 7(11), 14680–14707. <https://doi.org/10.3390/rs71114680>
- [5] Maggiori, E., Tarabalka, Y., Charpiat, G., & Alliez, P. (2016). Convolutional Neural Networks for Large-Scale Remote-Sensing Image Classification. *IEEE Transactions on Geoscience and Remote Sensing*, 55(2), 645–657. <https://doi.org/10.1109/TGRS.2016.2612821>
- [6] Katole, A. L., Yellapragada, K. P., Bedi, A. K., Kalra, S. S., & Siva Chaitanya, M. (2015). Hierarchical Deep Learning Architecture for 10K Objects Classification. *Computer Science & Information Technology ( CS & IT )*, (September), 77–93. <https://doi.org/10.5121/csit.2015.51408>
- [7] A. Putri, "Wanita 31 Tahun dengan Tumor Otak A 31 Years Old Woman with Brain Tumor."
- [8] A. Raup, W. Ridwan, Y. Khoeriyah, Q. Yuliaty Zaqiah, and U. Islam Negeri Sunan Gunung Djati Bandung, "Deep Learning dan Penerapannya dalam Pembelajaran." [Online]. Available: <http://jiip.stkipyapispomp.ac.id>
- [9] A. ANHAR and R. A. PUTRA, "Perancangan dan Implementasi Self-Checkout System pada Toko Ritel menggunakan Convolutional Neural Network (CNN)," *ELKOMIKA: Jurnal Teknik Energi Elektrik, Teknik Telekomunikasi, & Teknik Elektronika*, vol. 11, no. 2, p. 466, Apr. 2023, doi: 10.26760/elkomika.v11i2.466.
- [9] A. Y. W. R. S. I Wayan Suartika E. P, "Klasifikasi Citra Menggunakan Convolutional Neural Network (Cnn) pada Caltech 101," 2016.
- [10] Heaton, J. (2015). *Artificial Intelligence for Humans: Deep learning and neural networks of Artificial Intelligence for Humans Series*. Createspace Independent Publishing Platform.
- [11] Kim, J., Sangjun, O., Kim, Y., & Lee, M. (2016). Convolutional Neural Network with Biologically Inspired Retinal Structure. *Procedia Computer Science*, 88, 145–154. <https://doi.org/10.1016/j.procs.2016.07.418>
- [12] Bejiga, M. B., Zeggada, A., Nouffidj, A., & Melgani, F. (2017). A convolutional neural network approach for assisting avalanche search and rescue operations with UAV imagery. *Remote Sensing*, 9(2). <https://doi.org/10.3390/rs9020100>
- [13] Zhi, T., Duan, L. Y., Wang, Y., & Huang, T. (2016). Two-stage pooling of deep convolutional features for image retrieval. In 2016 IEEE International Conference on Image Processing (ICIP) (hal. 2465–2469). <https://doi.org/10.1109/ICIP.2016.7532802>
- [14] Hijazi, S., Kumar, R., & Rowen, C. (2015). *Image Recognition Using Convolutional Neural Networks*. Cadence Whitepaper, 1–12.
- [14] Albelwi, S., & Mahmood, A. (2017). A Framework for Designing the Architectures of Deep Convolutional Neural Networks. *Entropy*, 19, 242.
- [16] Vedaldi, A., & Lenc, K. (2015). *MatConvNet: Convolutional Neural Networks for MATLAB*. In *Proceedings of the 23rd ACM International Conference on Multimedia* (hal. 689–692). New York, NY, USA: ACM. <https://doi.org/10.1145/2733373.2807412>
- [17] I. G. T. Isa and B. Junedi, "Hyperparameter Tuning Epoch dalam Meningkatkan Akurasi Data Latih dan Data Validasi pada Citra Pengendara," *Prosiding Sains Nasional dan Teknologi*, vol. 12, no. 1, p. 231, Nov. 2022, doi: 10.36499/psnst.v12i1.6697.
- [18] A. Nur, A. Thohari, and G. Boy Hertantyo, "Implementasi Convolutional Neural Network untuk Klasifikasi Pembalap MotoGP Berbasis GPU," 2018.