

Penggunaan Algoritma *Extreme Learning Machine* Berbasis Android Untuk Mengidentifikasi Daging Sapi dan Babi

Agung Hartono¹, Suendri², Aninda Muliani Harahap³

^{1,2,3} Sistem Informasi, Universitas Islam Negeri Sumatera Utara Medan

Email: ¹hartonoagung06@gmail.com, ²suendri@uinsu.ac.id, ³anindamh@gmail.com

Email Penulis Korespondensi: hartonoagung06@gmail.com

Article History:

Received Dec 12th, 2022

Revised Jan 03th, 2023

Accepted Jan 10th, 2023

Abstrak

UD. Zahra merupakan sebuah unit usaha yang bergerak dibidang perdagangan daging sapi yang berdiri pada tahun 2014 oleh bapak Dwi Sanopa. Dalam hal membedakan antara daging sapi dan daging babi terdapat potensi kecurangan yang akan dilakukan oleh supplier, yaitu dengan mencampurkan daging babi kedalam daging sapi tersebut dikarenakan kurangnya ilmu dalam membedakan hal tersebut. UD. Zahra memerlukan sistem identifikasi untuk membantu dalam membedakan daging sapi dan babi kedepannya. Algoritma *Extreme Learning Machine* pada *machine learning* dapat memberikan kemudahan bagi UD. Zahra untuk mengidentifikasi daging sapi dan babi. Aplikasi Identifikasi ini dibangun menggunakan Bahasa Python sehingga mengurangi kecurangan yang akan terjadi dalam pemasokan daging.

Kata Kunci : Android, ELM, *Extreme Learning Machine*, Identifikasi, *Machine learning*

Abstract

UD Zahra is a business unit engaged in beef trading which was founded in 2014 by Mr. Dwi Sanopa. In terms of distinguishing beef and pork, there is a potential for fraud that can be carried out by suppliers, namely by mixing pork into beef due to lack of knowledge in distinguishing these things. UD Zahra needs an identification system to help differentiate beef and pork in the future. The *Extreme Learning Machine* algorithm in *Machine learning* can make it easier for UD Zahra to identify beef and pork. This Identification application is built using the Python language so that it can reduce fraud that will occur in the provision of meat.

Keyword : Android, ELM, *Extreme Learning Machine*, Identification, *Machine learning*

1. PENDAHULUAN

Zaman semakin berkembang tanpa bisa dibendung lagi, oleh karena itu kita di tuntut untuk terus mengikuti perkembangan zaman, apabila kita tidak mengikuti perkembangan zaman, maka kita akan tertinggal. Orang-orang telah mempertimbangkan bagaimana komputer dapat belajar dari pengalaman sejak komputer pertama dibuat. Ini menjadi jelas pada tahun 1952 ketika Arthur Samuel mengembangkan *game of checkers*, di komputer IBM. Komputer dapat mengingat gerakan yang diperlukan untuk memenangkan permainan catur setelah mempelajarinya. Proses pembelajaran komputer dari data dikenal sebagai *machine learning* (*learn from data*). Komputer tidak dapat mempelajari apa pun tanpa data [1]. Oleh karena itu, kita pasti akan terus terlibat dengan data jika kami ingin menguasai pembelajaran mesin. Data harus selalu dilibatkan dalam keahlian *machine learning*. Sekalipun datanya mungkin sama, teknik dan metode yang berbeda digunakan untuk mencapai hasil terbaik.

UD. Zahra yang beralamat di Jalan Sei Mencirim gang Ikhlas merupakan usaha dagang yang menjual daging sapi di pasar. Berdasarkan observasi penulis pada bulan Agustus lalu, kebutuhan akan daging sapi terus meningkat dikalangan masyarakat. Tentunya ini menjadi sebuah tantangan bagi UD. Zahra. Dalam persediaan barang, UD. Zahra tidak mempunyai ternak sapi sendiri, melainkan mengambil daging sapi yang sudah di potong-potong melalui Agen Sapi. Permasalahannya adalah dengan meningkatnya kebutuhan daging sapi tersebut memberikan celah kepada pihak-pihak tertentu untuk mengoplosnya dengan babi. UD. Zahra belum mempunyai suatu teknologi yang dapat membedakan daging sapi dan daging babi.

Identifikasi atau pengenalan daging selama ini hanya dilakukan secara manual menggunakan indera penglihatan manusia. Pendekatan ini memiliki banyak masalah. Oleh karena itu, sistem yang dapat membedakan daging sapi dan babi harus dikembangkan. Agar dapat mencegah terjadi kecurangan yang akan dialami oleh UD Zahra. Pada era saat ini, seluruh teknologi yang diciptakan dapat membantu kegiatan manusia. Oleh karena itu, pemanfaatan teknologi harus digunakan sebaik-baiknya. Jika suatu usaha yang sudah memanfaatkan teknologi, maka usaha tersebut dapat bersaing di Era sekarang.

Berdasarkan permasalahan tersebut, penulis melakukan penelitian menggunakan algoritma *Extreme Learning Machine* untuk mengidentifikasi daging sapi dan babi berdasarkan ekstraksi fitur warna dan tekstur berbasis android. Dengan teknologi tersebut, penulis berharap proses identifikasi daging sapi dan daging babi akan lebih efektif, sehingga dapat membantu guru dan santri pada pondok pesantren tersebut dalam mengidentifikasi daging sapi dan daging babi secara tepat tanpa membutuhkan pengetahuan tentang karakteristik dari daging itu sendiri.

Pada penelitian sebelumnya oleh Deppi Ariyanto [2], meneliti dan mengidentifikasi mengenai daging sapi dan babi menggunakan citra RGB, sedangkan penelitian yang dilakukan oleh penulis menggunakan Algoritma *Extreme Learning Machine*. Kemudian penelitian yang dilakukan oleh Elvia Budianita [3] mengidentifikasi perbedaan daging babi dengan daging sapi menggunakan algoritma K-Nearest Neighbour serta mengimplementasikannya dalam bentuk aplikasi berbasis web. Penulis dalam mengimplementasikan aplikasi berbasis android dengan algoritma yang berbeda pula. Sehingga diharapkan jika implementasi menggunakan aplikasi berbasis android lebih memudahkan pengguna dalam mengecek daging sapi atau babi.

Sistem pemrosesan yang dikenal sebagai jaringan syaraf tiruan dibuat dan diajarkan untuk melakukan proses pembelajaran dengan mengubah bobotnya untuk menyelesaikan masalah yang menantang [4]. Serupa dengan otak manusia, JST memiliki neuron-neuron yang terhubung dan terhubung. Informasi yang diterima oleh neuron dikirim ke neuron lain. Informasi ini adalah jumlah tertentu yang dikenal sebagai berat [5].

Extreme Learning Machine (ELM) adalah sebuah metode jaringan syaraf tiruan yang bersifat *feed-forward* dan hanya mempunyai satu *hidden layer* atau biasa dikenal dengan istilah *single-hidden layer feed-forward neural networks* (SLFNs) [6]. Semua parameter dalam *JST feedforward* yang menggunakan *conventional gradient based learning algorithm*, seperti *backpropagation* (BP), harus dipilih secara manual. Parameter yang dimaksud adalah *input weight* dan *hidden bias*. Parameter-parameter ini juga terkait antar lapisan yang berbeda, oleh karena itu perlu waktu lama untuk dipelajari dan sering terjebak di local minima. Sementara di ELM, parameter seperti *input weight* dan *hidden bias* dipilih secara acak, sehingga ELM memungkinkan *learning speed* yang cepat dan menghasilkan *good generalization performance* [7].

Pendekatan pelatihan digunakan untuk menghitung secara iteratif semua bobot dalam jaringan FFNN, termasuk bobot *input nodes* ke *hidden nodes*, bobot bias ke *hidden nodes*, dan bobot *hidden nodes* ke *output*. Dalam ELM, bobot awal yang berupa bobot *input nodes* ke *hidden nodes* dan bobot bias ke *hidden nodes* dipilih secara sembarang. Moore-Penrose Generalized Inverse kemudian digunakan untuk melakukan perhitungan analitik untuk menentukan bobot akhir (bobot *hidden nodes* ke *output*) [8].

2. METODOLOGI PENELITIAN

2.1 Tahapan Penelitian

Metode R&D mengacu pada prosedur yang terlibat dalam menciptakan barang baru atau meningkatkan yang ada saat ini yang dapat diperhitungkan [9]. Metode penelitian R&D (*Research and Development*) dipilih karena teknik tersebut menciptakan item-item spesifik dan mengevaluasi kinerjanya sehingga dapat digunakan dalam komunitas yang lebih besar, yang memerlukan penelitian analitis untuk menguji produk-produk ini.

Adapun tahapan pada penelitian ini adalah sebagai berikut

- a. Potensi dan Masalah
Penelitian dapat berasal dari adanya potensi dan masalah. Segala sesuatu yang memiliki potensi akan menjadi berharga ketika digunakan. Masalah merupakan variasi antara apa yang diantisipasi dan apa yang sebenarnya terjadi. Pada tahap ini penulis melakukan sebuah observasi atau pra riset di UD Zahra untuk menemukan potensi dan masalah yang ada pada UD Zahra.
- b. Pengumpulan Informasi
Sangat penting untuk mengumpulkan informasi yang dapat digunakan sebagai bahan untuk merancang produk tertentu yang diharapkan dapat mengatasi masalah ini ketika potensi dan tantangan dapat dibuktikan dan dimutakhirkan secara faktual.
- c. Pengembangan Design
Produk dibuat untuk mengatasi masalah yang sudah ada.
- d. Uji Coba Produk
Sekelompok kecil orang digunakan untuk menguji produk.
- e. Revisi Produk
Cacat produk yang ditemukan dalam sampel kecil kemudian diperbaiki untuk menciptakan produk yang lebih baik.
- f. Uji Coba Pemakaian

Kelompok yang lebih besar digunakan untuk menguji produk.

g. Revisi Produk

Jika masalah ditemukan dalam pengujian skala besar, produk dapat ditingkatkan.

h. Pembuat Produk Masal

Produk jadi diproduksi secara massal untuk memastikan pemanfaatan terbaiknya.

2.2 Algoritma *Extreme Learning Machine*

Cara kerja proses identifikasi menggunakan algoritma *Extreme Learning Machine* [10]:

a. Menentukan jumlah *hidden nodes* pada *hidden layer*

Jumlah *hidden nodes* sangat berpengaruh pada besarnya akurasi klasifikasi. Jumlah *nodes* yang tidak sesuai dapat menyebabkan kesalahan dalam pemrosesan informasi seperti *underfitting* dan *overfitting*.

b. Menentukan fungsi aktivasi yang digunakan

Setelah menentukan jumlah *hidden nodes* yang akan digunakan pada *hidden layer*, tahap selanjutnya yaitu menentukan fungsi aktivasi yang akan digunakan untuk perhitungan nilai keluaran pada *hidden layer*. Pada penelitian ini digunakan fungsi aktivasi sigmoid.

c. Proses *Training*

Proses *training* terdiri dari beberapa tahapan, yaitu

1. Pengacakan bobot masukan (*input weight*) dan bias
2. Menghitung *matrix output hidden layer*
3. Menghitung bobot keluaran (*output weight*)

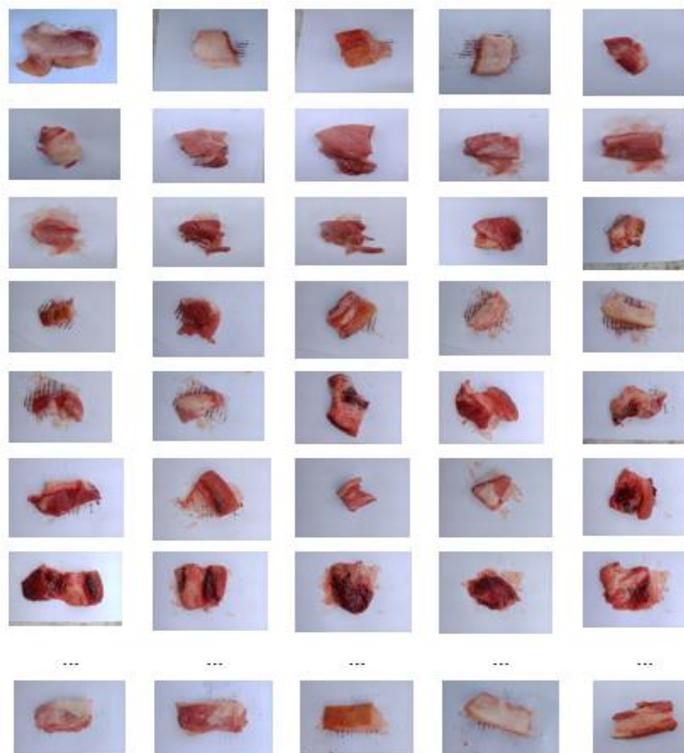
d. Proses *Testing*

Proses *testing* bertujuan untuk mengetahui keefektifan kinerja algoritma *Extreme Learning Machine* dalam identifikasi. Pada proses ini perhitungan dilakukan menggunakan parameter yang telah diperoleh pada proses *training* dan akan menghasilkan output berupa informasi identifikasi daging sapi dan babi.

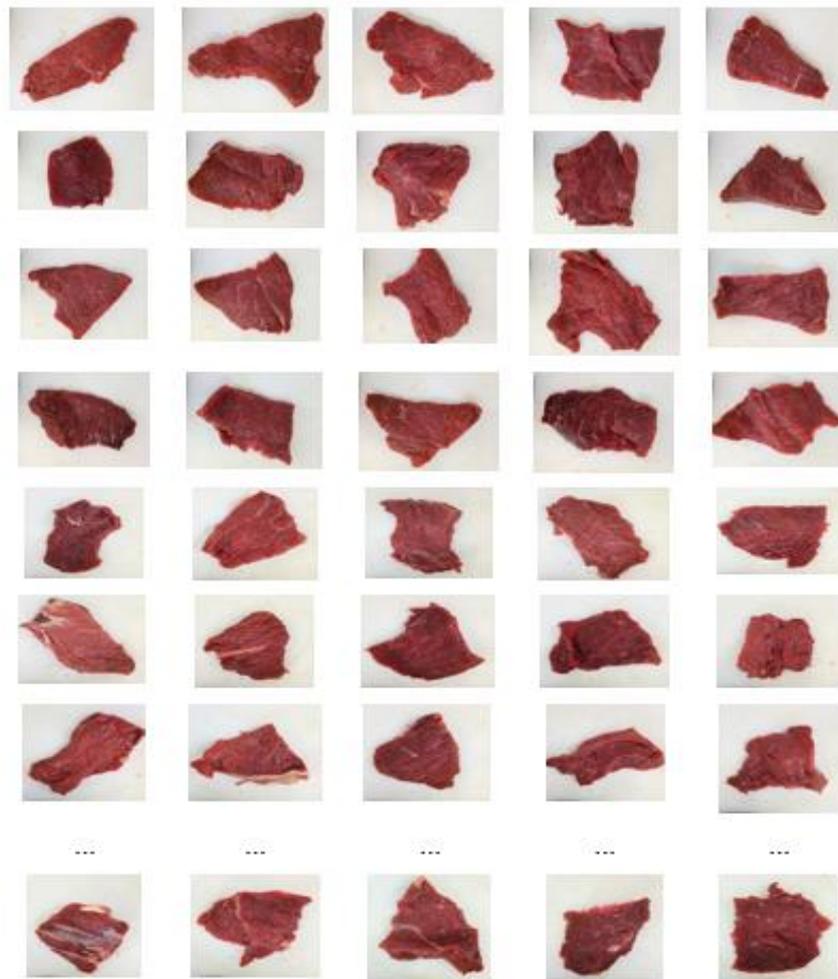
3. HASIL DAN PEMBAHASAN

3.1 Data Hasil Observasi

Data yang digunakan dalam penelitian ini berjumlah 217 citra, dimana Citra daging sapi dan babi diambil menggunakan kamera smartphone beresolusi 13 MP dengan format .jpg/.jpeg Contoh citra daging babi pada gambar 1 dan citra daging sapi yang digunakan pada penelitian ini dapat dilihat pada gambar dibawah ini.



Gambar 1. Data Daging Babi



Gambar 2. Data Daging Sapi

3.2 Proses Training

Perhitungan manual peramalan pemakaian konsumsi listrik dengan menggunakan percobaan 10 *input* neuron pada *hidden layer*, berikut hasil perhitungan matriks Data 1, untuk matriks *W* dan biasnya diperoleh secara acak dengan interval $[-1,1]$, dengan perintah `coding Self.w = rnd.uniform(-1,1,(columns, hid_num))` pada algoritma ELM dan memanggilnya menggunakan `coding print (weight)` pada file `trainingfinal.ipynb`. Dimana *columns* itu merupakan jumlah fitur yang terdapat pada data *preprocessing* dan *hid num* merupakan jumlah neuron yang dimasukkan. Perintah di atas merupakan perintah dari algoritmanya langsung yang dibuat secara acak oleh sistem, maka didapatkan hasil *input weight* yang terdapat pada tabel 1 berikut.

Tabel 1. *Input Weight*

8.85597011	-4.56931200	...	3.58784094	2.66364462
-7.53360665	-6.54572453	...	-9.07068185	5.87041164
-1.09302567	-6.29242954	...	7.93798360	6.58029261
...
-6.15466115	1.17891575	...	-3.76130929	1.41071655
-6.35294221	-8.34087692	...	2.48694880	-2.53181270
-1.42071646	4.36531030	...	9.5736464	-8.78197431

Data bias yang terdapat pada tabel 4 didapatkan dari perintah coding yang dibuat secara random dengan perintah : `Self.b = np.zeros(-1,1,(row, hid_num), dtype=float)` pada algoritma ELM dan memanggilnya menggunakan `coding print(bias)` pada file `trainingfinal.ipynb`.

Dengan dilakukannya perintah di atas, maka terbentuklah matrix bias secara acak yang terdapat pada tabel 2.

Tabel 2. Matrix Bias

0.34797398	0.34797398	...	0.34797398	0.34797398
0.10734301	0.10734301	...	0.10734301	0.10734301
-0.15939198	-0.15939198	...	-0.15939198	-0.15939198
...
0.7058356	0.7058356	...	0.7058356	0.7058356
-0.33148046	-0.33148046	...	-0.33148046	-0.33148046
0.14703211	0.14703211	...	0.14703211	0.14703211

Langkah selanjutnya adalah menggunakan persamaan berikut untuk mendapatkan hasil matriks H. Berikut adalah perhitungan manualnya.

$$\begin{aligned}
 H &= g(x_1 \cdot w_1, 1 + b_1) \\
 H_{(1,1)} &= \text{sigmoid}(g(x_1 \cdot w_1, 1 + b_1)) \\
 &= \text{sigmoid}(0.01369 \cdot 8.85) + (143.738 \cdot -7.53) + \dots + (0.34)) \\
 &= 0.40187581 \\
 H_{(1,2)} &= \text{sigmoid}(g(x_2 \cdot w_2, 1 + b_2)) \\
 &= \text{sigmoid}(0.01369 \cdot -4.56) + (143.738 \cdot -6.54) + \dots + (0.10)) \\
 &= 0.59833966 \\
 H_{(1,3)} &= \text{sigmoid}(g(x_3 \cdot w_3, 1 + b_3)) \\
 &= \text{sigmoid}(0.01369 \cdot -5.19) + (143.738 \cdot 8.02) + \dots + (-0.15)) \\
 &= 0.4366177
 \end{aligned}$$

Dari hasil perhitungan di atas maka dapatlah *matrix hidden layer* seperti pada tabel 3 sebagai berikut:

Tabel 3. Matrix Hidden Layer

0.401875	0.519130	0.496150	...	0.492454	0.547067	0.428386
0.598339	0.466641	0.636643	...	0.395745	0.227635	0.235614
0.436617	0.493175	0.486840	...	0.447996	0.338469	0.409115
...
0.173852	0.160170	0.125628	...	0.150060	0.312558	0.218480
0.361747	0.371670	0.402506	...	0.313268	0.133750	0.223635
0.228855	0.321715	0.322607	...	0.298262	0.150728	0.214048

Tahap selanjutnya adalah membentuk matrix Moore-Penrose Generalized Invers dengan memasukkan code : `Self.H_ = np.linalg.pinv(self.h)` pada program algoritma ELMnya dan memanggilnya menggunakan coding `print(hinit)` pada file `trainingfinal.ipynb`. Rumus dari perhitungan Moore-Penrose Generalized Invers adalah sebagai berikut [11]:

$$H^+ = (H^T \cdot H)^{-1} \cdot H^T$$

Keterangan:

- H⁺ = Moore-Penrose Generalized Invers
- (H^T * H)⁻¹ = Hasil inversi dari perkalian matriks H transposisi dengan matriks H
- H^T = Matriks H transposisi

Hasil dari perhitungan matrix H+ dapat dilihat di tabel 4 berikut.

Tabel 4. Matrix H⁺

0.04532795	0.04999822	...	0.17329525	0.10303569
0.01045334	-0.10041215	...	0.25299586	-0.4861286
...
0.5303318	-0.00534039	...	0.28606027	0.03676203
-0.29535385	0.00067232	...	-0.57418868	0.36798634

Nilai yang terdapat Tahap selanjutnya adalah membentuk *Output Weight* dengan membentuk code : `Self.beta_ = np.dot(self.H_, y_train)` pada program algoritma ELMnya dan memanggilnya menggunakan `coding print(outputweight)` pada file `trainingfinal.ipynb`. Rumus perhitungan *Output Weight* adalah sebagai berikut [12].

$$\beta = H^+ \cdot T$$

Keterangan:

β = *Output Weight Matrix*
 H^+ = *Moore Penrose Matrix*
T = Target

Hasil perhitungan matrix *output weight* dapat dilihat di tabel 5 berikut ini.

Babi	-1.44	2.40	-0.29	0.56	-0.23	0.37	-3.07	0.27	-0.62	0.58
Sapi	1.85	-2.28	0.52	-0.27	0.01	0.69	2.60	0.44	1.15	-1.31

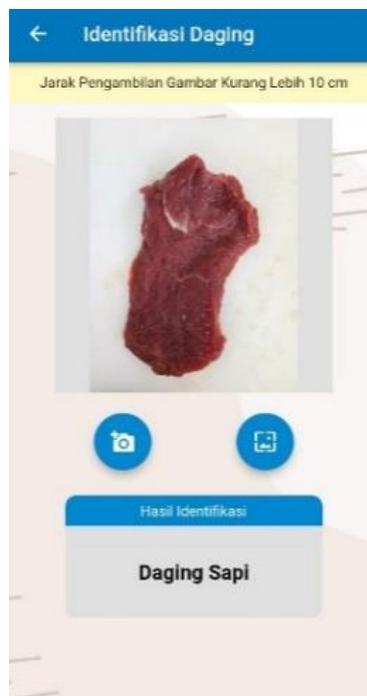
Bobot di atas inilah yang akan disimpan dan dipikel untuk dijadikan model dengan nama `par11.pkl` yang berisi bobot *beta/output weight, bias dan input weight* yang akan di *deploying*.

3.3 Implementasi

Dibawah ini merupakan hasil implementasi pada aplikasi untuk mengidentifikasi daging babi dan daging sapi.

a. Halaman hasil identifikasi daging sapi

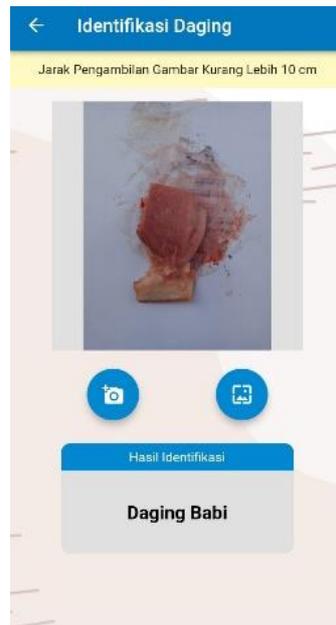
Halaman ini merupakan halaman hasil dari identifikasi daging sapi oleh pengguna. Pada gambar 3 terlihat upload dari daging sapi yang akan diidentifikasi. Kemudian di proses dan pada kotak hasil identifikasi akan mengeluarkan hasil dari daging itu sendiri.



Gambar 3. Halaman Hasil Identifikasi Daging Sapi

b. Halaman hasil identifikasi daging babi

Halaman ini merupakan halaman hasil dari identifikasi daging babi oleh pengguna. Pada gambar 4 terlihat upload dari daging babi yang akan diidentifikasi. Kemudian di proses dan pada kotak hasil identifikasi akan mengeluarkan hasil dari daging itu sendiri.



Gambar 4. Halaman Hasil Identifikasi Daging Babi

4. KESIMPULAN

Berdasarkan penelitian yang dilakukan oleh penulis, dapat disimpulkan bahwa penerapan algoritma *Extreme Learning Machine* pada identifikasi daging sapi dan babi dapat dikatakan sangat efektif dengan keakuratan 0.93 -> 93% untuk *training* dan 90,62% untuk *testing*. Aplikasi ini dibangun melalui proses *modelling* dan *deploying* dengan menggunakan bahasa pemrograman Python dengan *tools* Jupyter Notebook untuk pembuatan *modelling* dan Flask untuk *deploying*. Model yang telah dibangun menggunakan Bahasa pemrograman Python sebelumnya, di *coding* kembali kedalam *framework flutter* atau tahap *deploying*.

DAFTAR PUSTAKA

- [1] R. M. Awangga dan N. A. Batubara, *Tutorial Object Detection Plate Number With Convolution Neural Network (CNN)*. Bandung: Kreatif Industri Nusantara, 2020.
- [2] D. Ariyanto, "Identifikasi Daging Sapi dan Daging Babi Berdasarkan Citra Red Green Blue Menggunakan Algoritma K-Nearest Neighbor," Universitas Pembangunan Nasional "Veteran" Yogyakarta, 2020.
- [3] E. Budianita, J. Jasril, dan L. Handayani, "Implementasi Pengolahan Citra dan Klasifikasi K-Nearest Neighbour Untuk Membangun Aplikasi Pembeda Daging Sapi dan Babi Berbasis Web," *J. Sains dan Teknol. Ind.*, vol. 12, no. Vol 12, No 2 (2015): Juni 2015, hal. 242–247, 2015.
- [4] E. Setyowati dan S. Mariani, "Penerapan Jaringan Syaraf Tiruan dengan Metode Learning Vector Quantization (LVQ) untuk Klasifikasi Penyakit Infeksi Saluran Pernapasan Akut (ISPA)," in *PRISMA, Prosiding Seminar Nasional Matematika*, 2021, vol. 4, hal. 514–523.
- [5] V. V. Nurdiansyah, I. Cholissodin, dan P. P. Adikara, "Klasifikasi Penyakit Tuberkulosis (TB) menggunakan Metode Extreme Learning Machine (ELM)," *J. Pengemb. Teknol. Inf. dan Ilmu Komput.*, vol. 4, no. 5, hal. 1387–1393, 2020.
- [6] L. H. Harum, N. Hidayat, dan R. K. Dewi, "Implementasi Metode Extreme Learning Machine (ELM) Untuk Memprediksikan Penjualan Roti (Studi Kasus: Harum Bakery)," vol. 2, no. 11, hal. 1–119, 2018.
- [7] Z. A. Fikriya, M. I. Irawan, dan Soetrisno, "Implementasi Extreme Learning Machine untuk Pengenalan Objek Citra Digital," *J. Sains dan Seni ITS*, vol. 6, no. 1, 2017, doi: 10.12962/j23373520.v6i1.21754.
- [8] N. A. Izati, B. Warsito, dan T. Widiharih, "Prediksi Harga Emas Menggunakan Feed Forward Neural Network dengan Metode Extreme Learning Machine," *J. Gaussian*, vol. 8, no. 2, hal. 171–183, 2019.
- [9] N. S. Sukmadinata, *Metode Penelitian Pendidikan*. Bandung: PT Remaja Rosdakarya, 2013.
- [10] R. A. Chandra, E. Santoso, dan S. Adinugroho, "Optimasi Metode Extreme Learning Machine Dalam Penentuan Kualitas Air Sungai Menggunakan Algoritme Genetika," *J. Pengemb. Teknol. Inf. dan Ilmu Komput.*, vol. 2, no. 10, hal. 3265–3273, 2018.

- [11] M. D. Puspaningrum, E. Santoso, dan N. Yudistira, “Prediksi Persentase Penyelesaian Permohonan Hak Milik menggunakan Metode Fuzzy Time Series (Studi Kasus: Badan Pertanahan Nasional (BPN) Kabupaten Malang),” *J. Pengemb. Teknol. Inf. dan Ilmu Komput.*, vol. 4, no. 7, hal. 2851–2857, 2020.
- [12] K. N. D. Putra, M. T. Furqon, dan N. Yudistira, “Prediksi Penjualan Hijab menggunakan Metode Extreme Learning Machine (ELM) (Studi Kasus : Vie Hijab Store),” *J. Pengemb. Teknol. Inf. dan Ilmu Komput.*, vol. 4, no. 6, hal. 1750–1757, 2020.