

Identifikasi Moncong Sapi menggunakan Metode Jaringan Saraf Tiruan Konvolusional (CNN)

Recognizing Cow Muzzle Patterns using the Convolution Neural Network (CNN) Algorithm

¹Sulthon Zamroni*, ²Giri Wahyu Wiriasto, ³Bulkis Kanata

^{1,2,3}Teknik Elektro, Teknik, Universitas Mataram

^{1,2,3}Jl. H.L. Anggrat, Mesanggok, Kec. Gerung, Kab. Lombok Barat, NTB, Indonesia

*e-mail: sulthonzamroni@gmail.com

(received: 10 September 2024, revised: 25 September 2024, accepted: 16 October 2024)

Abstrak

Pada era serba digital saat ini, apapun pekerjaan ataupun masalah dapat diselesaikan hanya menggunakan sedikit usaha, terutama untuk mengenali hewan ternak seperti sapi, banyak sistem dan algoritma yang digunakan untuk mengenali sapi, mulai dari bentuk tubuh, corak warna bulu, sampai dengan bagian tertentu pada sapi. Penelitian ini bertujuan untuk membuat sistem identifikasi moncong sapi menggunakan metode jaringan saraf konvolusi dengan arsitektur AlexNet, dan menemukan masalah apa saja yang dapat menurunkan tingkat akurasi hasil prediksi. Hasil dari penelitian ini dapat membantu peternak sapi dalam mengelola data ternak dengan lebih efektif, dikarenakan metode dengan sistem identifikasi tradisional dapat membuat sapi merasa tidak nyaman dan stres. Penelitian ini juga berfungsi sebagai acuan atau referensi tambahan untuk para peneliti selanjutnya dalam mengembangkan penelitian tentang pengenalan sapi. Selain itu penelitian ini juga dapat digunakan sebagai bantuan dalam program asuransi seperti Asuransi Usaha Ternak Sapi (AUTS) untuk melindungi peternak dari kerugian usaha budidaya ternak yang disebabkan kecurian dan kematian ternak. Pengenalan sapi melalui moncongnya menggunakan metode CNN dapat memberikan hasil yang cukup tinggi, dengan menggunakan arsitektur AlexNet yang sedikit dimodifikasi, sistem ini dapat mengenali sapi 85%.

Kata kunci: alexnet, gambar, hidung, jaringan syaraf konvolusional, moncong, pengenalan, sapi, ternak

Abstract

In today's digital era, any task or problem can be solved with minimal effort, especially in livestock identification such as cattle. Numerous systems and algorithms have been developed to recognize cattle, ranging from body shape, fur patterns, to specific parts of the cattle. This research aims to develop a cattle muzzle identification system using convolutional neural networks method with Alexnet architecture and to identify the factors that can decrease the accuracy of prediction results. The results of this research can help cattle farmers manage their livestock data more effectively, as traditional identification methods can cause discomfort and stress to the cattle. This research also serves as a reference for future researchers in developing cattle recognition research. Additionally, this research can be used to support insurance programs such as Cattle Farming Insurance (AUTS) to protect farmers from losses due to cattle theft and death. Cattle recognition through their muzzles using the CNN method can produce relatively high results. By slightly modifying the AlexNet architecture, this system can recognize cattle with an accuracy of 85%.

Keywords: alexnet, convolutional neural network, cow, image, livestock, nose, recognition, snout

1 Pendahuluan

Identifikasi adalah adalah suatu cara untuk membedakan atau menyamakan sebuah identitas dari suatu benda maupun individu, dalam hal ini yang dibutuhkan adalah sebuah sistem yang dapat mengenal atau mengidentifikasi informasi yang didapat. Dalam sejumlah bidang bisnis seperti asuransi,

perdagangan hewan, dan manajemen pertanian, identifikasi dan verifikasi hewan sangatlah penting [1],[2],[3]. Pada hewan ternak, terutama sapi sangat sulit untuk membedakannya, meskipun ciri fisik antara sapi identik sama, tetapi bisa jadi sapi tersebut berbeda jenis dan varian. Sapi biasanya ditandai oleh peternaknya menggunakan tag nomor, tetapi hal ini masih kurang efektif karena tag nomor bisa hilang, mengalami kerusakan ataupun diganti oleh pihak yang tidak bertanggung jawab.

Tentunya dengan adanya sistem yang dapat mengklasifikasikan sapi akan mempermudah identifikasi pada sapi, sistem ini akan memindai moncong atau muzzle sapi yang merupakan ciri khas dari setiap sapi, karena yang membedakan setiap sapi adalah pada moncongnya, maka pembelajaran mendalam bisa dilakukan untuk mencegah terjadinya pendekatan secara invasif [3]. Moncong sapi mirip seperti sidik jari manusia, disinilah terletak biometriknya, sehingga membuat identifikasi sapi bisa dilakukan dengan mengamati moncong atau muzzle sapi, namun hal ini sulit untuk dilakukan karena perlu ketelitian yang sangat tinggi untuk dapat membedakan moncong sapi secara manual dengan mata telanjang.

Moncong sapi dapat digunakan sebagai sampel pengenalan pola. Beberapa tahun terakhir, teknik non-invasif yang didukung oleh kemajuan dalam pembelajaran mesin dan pemrosesan gambar telah membuka kemungkinan baru untuk identifikasi hewan yang akurat dan efisien [4],[5]. Metode pengenalan pola dapat digunakan untuk pengembangan sistem identifikasi sapi ternak memanfaatkan biometrik yang terdapat pada moncong menggunakan teknik pengolahan citra digital.

Penelitian ini bertujuan untuk mengetahui seberapa akurat metode identifikasi moncong sapi menggunakan jaringan saraf tiruan konvolusi, dan seberapa berpengaruh spesifikasi komputasi untuk menjalankan jaringan saraf tiruan konvolusi ini, dengan mengetahui bahwa kelebihan dari metode CNN lebih efisien dibandingkan metode neural network lainnya terutama untuk memori dan kompleksitas, selain itu metode CNN juga dapat secara otomatis mengekstraksi ciri penting dari setiap citra tanpa bantuan manusia. CNN merupakan salah satu metode deep learning yang mampu melakukan proses pembelajaran mandiri untuk pengenalan objek, ekstraksi objek dan klasifikasi serta dapat diterapkan pada citra resolusi tinggi yang memiliki model distribusi nonparametrik [7]. Metode ini banyak digunakan untuk mengidentifikasi sidik jari, telapak tangan dan wajah dengan keakuratan yang cukup tinggi, sehingga diharapkan mampu mengidentifikasi dan mengklasifikasi hewan ternak sapi.

2 Tinjauan Literatur

Identifikasi sapi dilakukan berdasarkan hidung sapi, dengan menggunakan metode Statistic Region Merging (SRM), memisahkan pola suatu objek (latar depan) dari informasi yang tidak diperlukan (latar belakang) menciptakan suatu sistem identifikasi yang dapat mengenali sapi. Dengan metode ini, individu sapi dapat dibedakan berdasarkan pola hidungnya dan dapat memberikan hasil kinerja yang baik, terbukti dengan nilai presisi dan recall sebesar 87% serta nilai ROC sebesar 0,976 [6].

Selanjutnya Jaringan saraf konvolusional (CNN) digunakan untuk mengklasifikasikan tanaman dengan resolusi tinggi dan membedakan lima kelas spesies tanaman, termasuk padi, bawang merah, cabai, pisang, dan kelapa. Data yang digunakan adalah gambar udara yang diambil menggunakan teknologi UAV (unmanned aerial vehicle) dengan menggunakan kamera jenis Canon PowerShot S100. Data diperoleh dari pemandangan gambar RGB dengan koreksi geometrik 4,64 mm pada area seluas 311 hektar dan resolusi spasial 6,5 cm. Lokasi penelitian adalah Kecamatan Kretek Daerah Istimewa Yogyakarta. Akurasi metode CNN adalah 82%. [7].

Jaringan syaraf konvolusional (CNN) juga dimanfaatkan dalam keamanan lingkungan dari kecelakaan yaitu menentukan terjadinya kebakaran di suatu lokasi. Data yang digunakan adalah data gambar yang diambil oleh CCTV pada lokasi-lokasi untuk memantau terjadinya kebakaran, dan diklasifikasikan menjadi dua kelas yaitu api dan non-api. Untuk citra api digunakan 950 data citra yang terdiri dari 700 data latih, 170 data validasi, dan 80 data uji. Arsitektur model CNN yang digunakan terdiri dari dua lapisan konvolusional dan dua lapisan pooling. Hasil CNN mencapai akurasi 98,8% [8].

Selanjutnya penelitian yang dilakukan pada hewan ternak, dalam studi literaturnya pengenalan sapi yang lebih baik menggunakan deep learning Convolution Neural Network (CNN), Stacked Denoising Auto Encoder (SDAE) dan Deep Belief Network (DBN), hasilnya penelitiannya Deep Belief Network

(DBN) menunjukkan akurasi 98,99% dengan data 4000 foto, 400 sapi dan moncong sapi masing-masing sapi 10 foto untuk training dan 10 foto untuk testing [9].

Kemudian penelitian tentang memanfaatkan kumpulan data komprehensif yang terdiri dari 4923 gambar moncong yang telah dibersihkan dan dipotong dari 268 ras sapi yang berbeda, dengan menggunakan VGG Face model, hasil akurasi dari data latihnya adalah 98,88% dan akurasi pengujian sebesar 100%. Penelitiannya menghindari prosedur invasif dan menunjukkan kemampuan beradaptasi, serta menangani pengenalan hewan baru ke dalam sistem secara efektif [10].

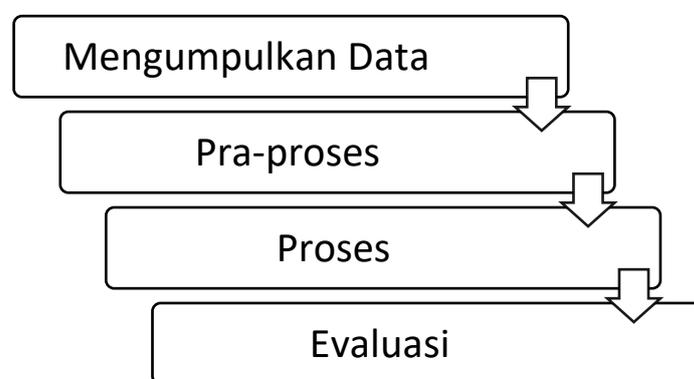
Metode CNN juga sering digunakan untuk identifikasi berbagai macam makhluk hidup maupun benda mati, seperti pada hewan ternak sapi di lombok timur, Nusa Tenggara Barat, menggunakan CNN dengan arsitektur MobileNetV2, dengan tingkat akurasi 83% [13]. Kombinasi CNN juga sering dilakukan, sering sekali CNN dikombinasikan atau disesuaikan, kemudian menghasilkan algoritma yang lebih kompleks, untuk tujuan yang lebih khusus, seperti YOLO, YOLOv3 dan ResNet50 dua algoritma yang diusulkan memiliki potensi besar dalam membentuk sistem otomatis yang sangat akurat, dengan akurasi deteksi moncong sapi 99,13% resolusi gambar 1024 x 1024 [14].

Berikutnya pada bagian tubuh hewan ternak, banyak yang membuat penelitian dengan menggunakan jaringan syaraf konvolusional. Untuk badan sapi, dapat dikenali dengan baik, menghasilkan akurasi 92,59% untuk data pelatihan, dan 89,95% untuk data pengujian [15], bukan hanya pada bagian badan, tetapi banyak juga yang melakukan penelitian di bagian lain, seperti bagian kepala [16], bagian wajah [17], bagian wajah dan badan [18], bagian rambut punggung [19] dan ada juga pada aktivitasnya [20].

Setelah mempelajari beberapa penelitian yang terkait, terutama penelitian tentang klasifikasi dan identifikasi hewan ternak, dapat disimpulkan bahwa penelitian untuk mengenali sapi bisa dilakukan dengan metode yang cukup beragam, dengan menggunakan metode deep learning, bisa mengenali sapi dengan tingkat akurasi yang tinggi dengan jumlah data yang banyak, hanya saja di dalam beberapa penelitian tidak mencantumkan dengan jelas apa saja pengaruh faktor menurunnya tingkat akurasi, dan perangkat dan apa saja yang dibutuhkan untuk membuat sistem tersebut, maka dari itu penelitian ini bertujuan untuk memperjelas faktor apa saja yang mempengaruhi akurasi sistem identifikasi pada hewan ternak khususnya sapi dengan mengecualikan faktor augmentasi data [22], apa saja perangkat yang dibutuhkan untuk bisa membuat sebuah sistem identifikasi ini, hasil perbandingan dataset, dan perbandingan model yang dibutuhkan untuk mendapatkan hasil yang lebih baik.

3 Metode Penelitian

Adapun alur penelitian yang akan dilaksanakan pada penelitian ini, selain untuk memudahkan peneliti dalam mengkategorikan kegiatan penelitiannya, juga dapat mengetahui adanya kesalahan atau kekurangan dalam penelitian ini melalui tahap per tahap, diantaranya disajikan sebagai berikut



Gambar 1. Alur penelitian

Gambar 1 merupakan alur penelitian yang dibagi menjadi 4 tahapan, mengumpulkan data, pra-proses, proses, dan terakhir adalah evaluasi, berikut penjelasan dari tiap – tiap tahapan.

3.1 Mengumpulkan Data

Pengumpulan data adalah tahapan awal untuk membangun sistem identifikasi sapi pada penelitian ini. Data yang akan dikumpulkan adalah data primer, artinya data yang dikumpulkan secara langsung oleh peneliti yang diambil dari sumber aslinya, kemudian data ini belum pernah diolah atau dipublikasikan sebelumnya, sehingga dianggap sebagai data yang paling original dan relevan dengan penelitian yang sedang dilakukan.

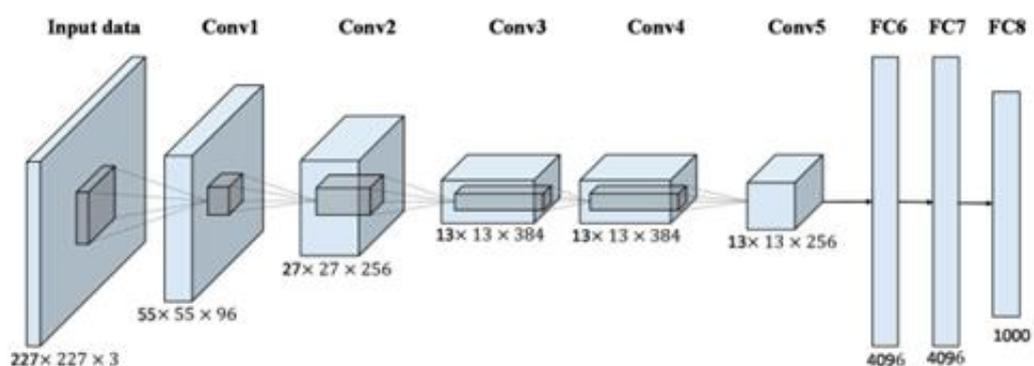
Data yang dibutuhkan pada sistem ini adalah data foto moncong sapi, sehingga harus mencari tempat ternak sapi terlebih dahulu, peneliti mengumpulkan data di peternakan sapi yang bertempat di Korporasi Ridho Ilahi, Lombok Timur. Data diambil dengan menggunakan kamera dari smartphone, baik di foto secara langsung moncong sapi, maupun divideokan terlebih dahulu untuk nanti bisa diambil per-frame.

3.2 Pra-proses

Pada tahapan ini, data yang sudah dikumpulkan akan ditinjau apakah layak atau tidak untuk dijadikan sebuah dataset penelitian, kemudian diubah agar sesuai dengan model yang akan dibangun. Setelah semua data sudah sesuai, selanjutnya data akan dikategorikan menjadi 3 kategori, yang masing – masing berada dalam folder latih, validasi, dan folder uji.

Pada tahap ini juga membangun model arsitektur untuk sistem identifikasi sapi. Model arsitektur yang digunakan untuk identifikasi dataset sapi dari penelitian ini yaitu mengacu pada model arsitektur AlexNet yang merupakan bagian dari sekian banyaknya arsitektur convolution neural network (CNN). AlexNet, yang pertama kali diusulkan oleh Alex Krizhevsky dkk. dalam ImageNet Large Scale Visual Recognition Challenge (ILSVRC-2012) tahun 2012 [10], merupakan arsitektur CNN yang mendasar, sederhana, dan efektif, yang utamanya terdiri dari tahapan bertingkat, yaitu lapisan konvolusi, lapisan penyatuan, lapisan unit linier teraktifasi (ReLU), dan lapisan yang terhubung penuh.

Secara khusus, AlexNet terdiri dari lima lapisan konvolusional, lapisan pertama, lapisan kedua, lapisan ketiga, dan lapisan keempat diikuti oleh lapisan penyatuan, dan lapisan kelima diikuti oleh tiga lapisan yang terhubung penuh. Untuk arsitektur AlexNet, kernel konvolusional diekstraksi selama prosedur optimasi backpropagation dengan mengoptimalkan seluruh fungsi biaya dengan algoritma penurunan gradien stokastik atau stochastic gradient descent (SGD). Secara umum, lapisan konvolusional bertindak atas peta fitur input dengan kernel konvolusional geser untuk menghasilkan peta fitur yang dikonvolusionalkan, dan lapisan pengumpulan beroperasi pada peta fitur yang dikonvolusionalkan untuk menggabungkan informasi dalam jendela lingkungan yang diberikan dengan operasi pengumpulan maksimum atau operasi pengumpulan rata-rata [11].



Gambar 2. Arsitektur alexnet [12]

Gambar 2 merupakan model arsitektur AlexNet yang diperkenalkan pada tahun 2012. Model ini memenangkan kompetisi ImageNet Large Scale Visual Recognition Challenge (ILSVRC) tahun itu dengan margin yang sangat signifikan, mengalahkan model-model lainnya [12].

3.3 Proses

Tahap ini dilakukan pelatihan dataset moncong sapi menggunakan jaringan saraf tiruan konvolusi arsitektur AlexNet, pelatihan dilakukan sebanyak empat kali, yang pertama adalah pelatihan menggunakan dataset pertama dengan model pertama, yang kedua pelatihan dataset kedua dengan model pertama, yang ketiga pelatihan dataset pertama dengan model kedua, yang terakhir adalah pelatihan dataset kedua dengan model kedua.

3.4 Evaluasi

Tahap ini akan memaparkan hasil dari pelatihan dataset moncong sapi, kemudian disusun dengan rapi mulai dari proses akurasi saat pelatihan, kinerja model, akurasi model dalam identifikasi sistem, yang dimana semua itu merupakan bagian terpenting dalam identifikasi moncong sapi menggunakan jaringan saraf tiruan konvolusi.

Akurasi, presisi, recall dan juga f1-score dapat diketahui melalui Tabel 1 confusion matrix berikut :

Tabel 1. Tabel confusion matrix

n = jumlah prediksi	Aktual positif	Aktual negatif
Prediksi positif	True Positif	False Positif
Prediksi negatif	False Negatif	True Negatif

Adapun hasil akurasi, presisi, recall dan f1-score dapat diketahui dengan menggunakan rumus atau persamaan berikut :

$$akurasi = \frac{(TP + TN)}{(TP + FP + FN + TN)} \quad (1)$$

$$Presisi = \frac{TP}{(TP + FP)} \quad (2)$$

$$Recall = \frac{TP}{(TP + FN)} \quad (3)$$

$$F1 - Score = \frac{(2 * Recall * Presisi)}{(Recall + Presisi)} \quad (4)$$

Setelah mengetahui akurasi, presisi, recall dan f1-score dapat membantu menentukan baik atau tidaknya sistem dan model tersebut.

4 Hasil dan Pembahasan

Pada tahap ini hasil dan pembahasan dilakukan dengan mengacu pada metode penelitian yang sudah dibuat, mulai dari mengumpulkan data, pra-proses, proses dan terakhir adalah evaluasi, berikut adalah hasil dari masing – masing tahapan :

4.1 Mengumpulkan Data

Dataset pertama berupa data foto moncong sapi yang diambil langsung menggunakan kamera smartphone. proses pengambilan dilakukan baik dengan mengambil foto langsung moncong sapi tersebut, maupun dengan cara merekam video.



Gambar 3. Contoh hasil foto

Gambar 3 adalah contoh hasil pengambilan foto yang dilakukan dengan jarak yang cukup aman sehingga sapi tidak merasa terancam atau panik.

4.2 Pra-proses

Data yang berupa foto akan dipotong dengan skala 1:1, kemudian data yang berupa video diambil per-*frame* kemudian dipotong juga dengan skala 1:1. Data yang sudah terkumpul berupa foto moncong sapi berjumlah 350, data ini dijadikan dataset pertama. Dataset ini dibagi terlebih dahulu dalam 3 kategori, setiap kategori dibuatkan folder dengan nama Latih, Validasi, dan Uji, untuk data Latih berjumlah 270 foto, data Validasi berjumlah 40 foto, dan data Uji berjumlah 40 foto, dengan jumlah individu sapi yaitu 32, dan masing - masing sapi memiliki 10 - 16 foto. Sapi yang digunakan sebagai dataset adalah sapi lokal yang bertempat di Korporasi Ridho Ilahi, Lombok Timur.



Gambar 4. Contoh foto dataset 1

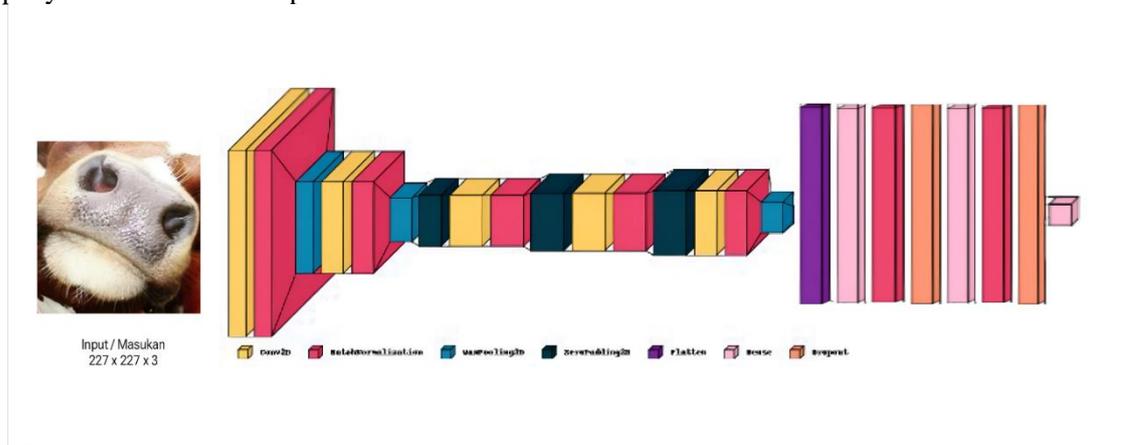
Gambar 4 merupakan contoh dataset primer, diambil dengan kamera smartphone resolusi 1:1, berjumlah 350 data, dengan format Joint Photographic Experts Group (jpg).



Gambar 5. Contoh foto dataset 2

Gambar 5 adalah Dataset ke-2, yang merupakan hasil dari dataset pertama yang sudah dipotong bagian muzzle atau moncongnya saja sehingga penulis percaya ini akan menjadi perbandingan yang bagus untuk hasil proses identifikasi pada penelitian ini.

Kemudian dalam merancang model arsitektur, peneliti ingin membandingkan 2 model dengan arsitektur yang sama, yaitu AlexNet, tetapi untuk model kedua sedikit dilakukan penyesuaian dari model pertama. Berikut adalah ilustrasi model AlexNet:



Gambar 6. Model alexnet default

Gambar 6 merupakan model AlexNet, dimana model ini terdiri dari 5 lapisan konvolusi (convolutional layers) dan 3 lapisan fully connected (FC layers), yang membentuk total 8 lapisan utama. Dalam struktur ini, konvolusi dan pengecilan resolusi (subsampling) dilakukan secara bergantian untuk mengambil fitur-fitur berbeda dalam citra.

Model pertama yang definisikan dalam penelitian ini adalah implementasi dari AlexNet, sebuah arsitektur *Convolutional Neural Network* (CNN) yang terkenal. Berikut adalah penjelasan dari setiap layer pada model tersebut:

1) Layer 1:

Conv2D(96, (11, 11), strides=(4, 4), padding='valid', input_shape=(227, 227, 3), activation='relu'): Layer konvolusi pertama mengekstrak 96 fitur menggunakan filter 11x11 dengan langkah 4 piksel. Padding 'valid' berarti tidak ada padding yang ditambahkan, sehingga ukuran output berkurang. Fungsi aktivasi ReLU memperkenalkan non-linearitas. Layer ini menerima gambar input berukuran 227x227 piksel dengan 3 saluran warna.

BatchNormalization(): Menormalkan aktivasi dari layer sebelumnya, meningkatkan stabilitas dan kecepatan pelatihan.

MaxPooling2D(pool_size=(3, 3), strides=(2, 2)): Layer *pooling* pertama mengurangi dimensi spasial dengan mengambil nilai maksimum dalam setiap wilayah 3x3 dengan langkah 2 piksel.

2) Layer 2:

Conv2D(256, (5, 5), strides=(1, 1), padding='same', activation='relu'): Layer konvolusi kedua mengekstrak 256 fitur menggunakan filter 5x5 dengan langkah 1 piksel. Padding 'same' mempertahankan ukuran spasial. ReLU digunakan untuk non-linearitas.

BatchNormalization(): Menormalkan aktivasi.

MaxPooling2D(pool_size=(3, 3), strides=(2, 2)): Layer *pooling* kedua mengurangi dimensi spasial lebih lanjut.

3) Layer 3:

ZeroPadding2D((1, 1)): Menambahkan padding nol di sekitar peta fitur untuk mempertahankan ukuran spasial.

Conv2D(384, (3, 3), strides=(1, 1), padding='same', activation='relu'): Layer konvolusi ketiga mengekstrak 384 fitur menggunakan filter 3x3.

BatchNormalization(): Menormalkan aktivasi.

4) Layer 4:

ZeroPadding2D((1, 1)): Menambahkan padding nol.

Conv2D(384, (3, 3), strides=(1, 1), padding='same', activation='relu'): Layer konvolusi keempat mengekstrak 384 fitur.

BatchNormalization(): Menormalkan aktivasi.

5) Layer 5:

ZeroPadding2D((1, 1)): Menambahkan padding nol.

Conv2D(256, (3, 3), strides=(1, 1), padding='same', activation='relu'): Layer konvolusi kelima mengekstrak 256 fitur.

BatchNormalization(): Menormalkan aktivasi.

MaxPooling2D(pool_size=(3, 3), strides=(2, 2)): Layer *pooling* ketiga mengurangi dimensi spasial.

6) Layer 6:

Flatten(): Mengubah peta fitur menjadi vektor satu dimensi.

Dense(4096, activation='relu'): Layer yang terhubung sepenuhnya pertama dengan 4096 neuron.

BatchNormalization(): Menormalkan aktivasi.

Dropout(0.5): Secara acak menonaktifkan 50% neuron selama pelatihan untuk mencegah *overfitting*.

7) Layer 7:

Dense(4096, activation='relu'): Layer yang terhubung sepenuhnya kedua dengan 4096 neuron.

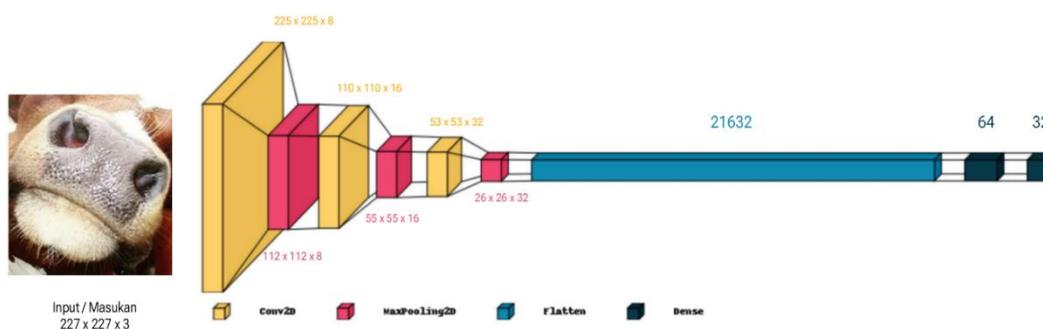
BatchNormalization(): Menormalkan aktivasi.

Dropout(0.5): Secara acak menonaktifkan neuron.

8) Layer 8:

Dense(32, activation='softmax'): Layer output menghasilkan distribusi probabilitas di atas 32 kelas, mewakili prediksi model untuk gambar input.

Untuk model 2, merubah model AlexNet dibagian konvolusi, kami mengurangi konvolusinya menjadi 3 konvolusi, dimana setiap konvolusi hanya membawa 1 layer pooling saja, yaitu maxpooling, kami melihat bahwa penambahan fitur yang tidak diperlukan akan membuat prediksi dan akurasi menjadi overfitting, overfitting adalah permasalahan pembelajaran mesin dimana mesin menjadi kelebihan pintar, ini disebabkan karena banyaknya parameter yang dipelajari [21]. Sehingga apabila dilakukan tes dengan menggunakan data yang berbeda dapat mengurangi akurasi. Kemudian di layer dense kami menghapus fitur dropout, jadi total layer adalah 6, dimana 3 layer konvolusi dan 3 layer terhubung semua, berikut adalah ilustrasi model AlexNet dengan model AlexNet yang kami sesuaikan:



Gambar 7. Model alexnet yang disesuaikan

Gambar 7 merupakan model AlexNet yang sudah disesuaikan, menghilangkan 2 layer konvolusi, dan mengurangi fitur yang dipercaya menyebabkan overfitting atau berkurangnya akurasi sistem.

Berikut penjelasan model ini :

1) Layer 1 Conv2D

Layer konvolusi pertama yang mengekstrak fitur dari gambar input. Memiliki 16 filter dengan ukuran kernel 3x3. Fungsi aktivasi ReLU memperkenalkan non-linearitas untuk mempelajari pola yang kompleks. `input_shape` mendefinisikan ukuran gambar input (227x227 piksel dengan 3 saluran warna), setelah itu pool layer, layer ini merupakan bagian layer pertama yang mengurangi dimensi spasial dari peta fitur, mempertahankan fitur yang paling menonjol. Ukuran pool 2x2 berarti setiap blok 2x2 piksel dipetakan ke satu piksel, mengambil nilai maksimum dalam blok tersebut.

2) Layer 2 Conv2D

Layer konvolusi kedua yang mengekstrak fitur tingkat yang lebih tinggi dari peta fitur yang di-pool. Memiliki 32 filter dengan ukuran kernel 3x3. Fungsi aktivasi ReLU kembali digunakan untuk non-linearitas kemudian maxpooling layer ukuran pool 2x2.

3) Layer 3 Conv2D

Layer konvolusi terakhir yang mengekstrak fitur tingkat yang lebih tinggi dari peta fitur yang di-pool, memiliki 64 filter dengan ukuran kernel 3x3. Fungsi aktivasi ReLU kembali digunakan untuk non-linearitas kemudian maxpooling layer terakhir dengan ukuran pool 2x2.

4) Layer 4 Flatten

Layer yang mengubah peta fitur multidimensi menjadi vektor satu dimensi, mempersiapkannya untuk layer yang terhubung sepenuhnya.

5) Layer 5 Dense

Layer yang terhubung sepenuhnya (atau dense) dengan 128 neuron, fungsi aktivasi ReLU memperkenalkan non-linearitas untuk mempelajari pola yang kompleks dalam representasi vektor.

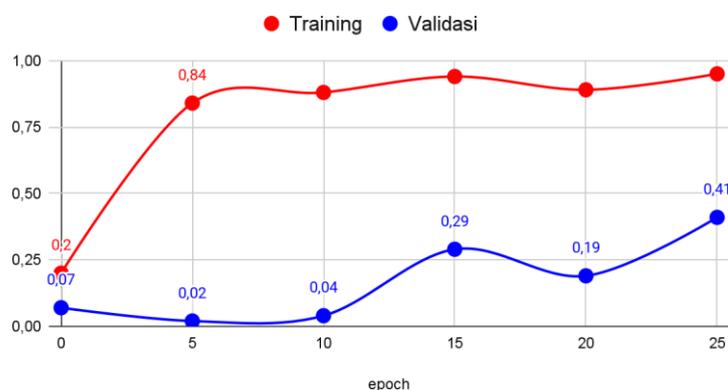
6) Layer 6 Dense

Layer output dengan 32 neuron, mewakili jumlah kelas dalam masalah klasifikasi pada penelitian ini, fungsi aktivasi softmax menghasilkan distribusi probabilitas di atas 32 kelas, menunjukkan probabilitas gambar input milik setiap kelas.

4.3 Proses

Tahap ini dilakukan pelatihan model. Model pembelajaran mesin dilatih dengan menggunakan TensorFlow, menggunakan kerangka kerja dari sumber terbuka. Pelatihan dilakukan dengan bahasa pemrograman python 3 di google colab, hardware yang digunakan adalah hardware CPU dengan kapasitas RAM 12.7 GB dan Disk 107.7 GB kemudian untuk versi trial dengan hardware TPU v2 dengan RAM 334.6 GB, Disk 225.3 GB. Pelatihan ini juga menggunakan Laptop dan komputer pribadi penulis dengan spesifikasi windows 10, intel core i5 generasi ke 6 dan 4 CPU, kapasitas RAM 4 GB, Disk 128 GB.

Akurasi Training dan Validasi



Gambar 8. Grafik akurasi training dan validasi alexnet

Gambar 8 merupakan grafik perbandingan akurasi proses pelatihan data latih dan data validasi pada pelatihan dataset 1 dengan model pertama, dimana sumbu x horizontal merupakan

banyaknya epoch, dan sumbu y vertikal adalah tingkat akurasinya, dapat dilihat tingkat akurasi tertinggi pada epoch 25.

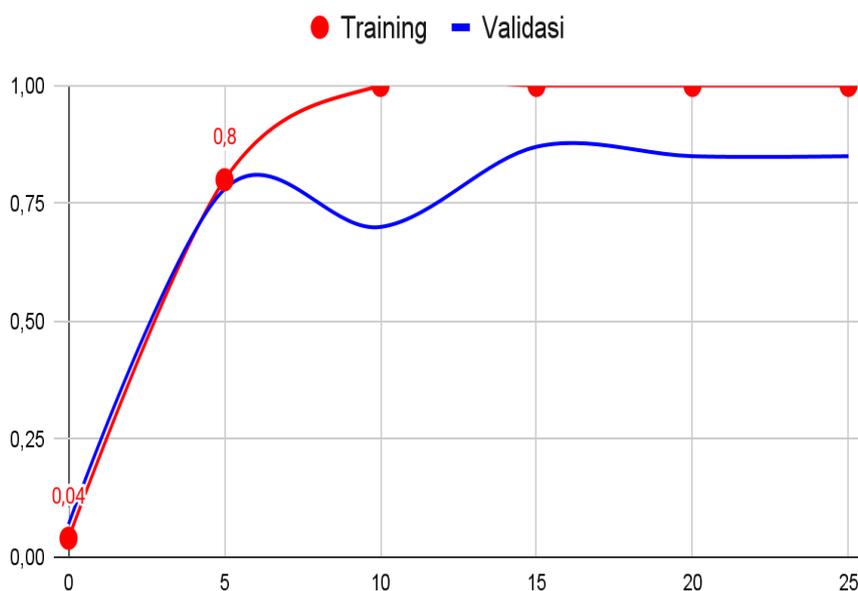
Pelatihan model dengan arsitektur AlexNet juga memerlukan resource yang besar, kapasitas RAM yang cukup besar dan memerlukan waktu yang lebih lama saat proses pelatihan. saat proses pelatihan menggunakan runtime gratis memerlukan waktu setidaknya 1742 detik atau 29 menit untuk dukungan hardware CPU. Hasil dari proses pelatihan dengan model 1 dipaparkan pada Tabel 2 sebagai berikut :

Tabel 2. Tabel Proses Pelatihan Model 1

Epoch	Train	Valid
0	0.2%	0.07%
5	0.84%	0.02%
10	0.88%	0.04%
15	0.94%	0.29%
20	0.89%	0.19%
25	0.95%	0.41%

Pelatihan dengan dataset yang sama akan tetapi modelnya sedikit ada penyesuaian menghasilkan tingkat akurasi yang signifikan, yaitu 85%. model ini juga mampu melatih data yang sama dengan waktu 870 detik atau dalam waktu 14 menit 30 detik untuk hardware CPU, dan dengan waktu 100 detik untuk dukungan hardware TPU v2, dimana per-epoch hanya membutuhkan waktu 2 detik. Untuk tampilan grafik hasil pelatihan dicontohkan melalui Gambar 9.

Training dan Validasi



Gambar 9. Grafik akurasi training dan validasi alexnet yang disesuaikan

Gambar 9 merupakan grafik perbandingan akurasi proses pelatihan data latih dan data validasi pada pelatihan dataset 2 dengan model AlexNet yang sudah disesuaikan, dimana sumbu x horizontal merupakan banyaknya epoch, dan sumbu y vertikal adalah tingkat akurasinya, dapat dilihat tingkat akurasi tertinggi pada epoch 10 dan seterusnya. Hasil dari proses pelatihan dengan model 2 dipaparkan pada Tabel 3 sebagai berikut :

Tabel 3. Tabel proses pelatihan model 2

Epoch	Train	Valid
0	0.04	0.07
5	0.8	0.78
10	1	0.7
15	1	0.87
20	1	0.85
25	1	0.85

4.4 Evaluasi

Pada tahap ini hasil dari semua kegiatan penelitian dicantumkan, dilakukan dengan membandingkan hasil penelitian model pertama dengan kedua, kemudian dipaparkan dengan sederhana. Berikut hasil evaluasi penelitian ini :

4.4.1 Model AlexNet

Pelatihan model dengan arsitektur AlexNet juga memerlukan resource yang besar, kapasitas RAM yang cukup dan memerlukan waktu yang lebih lama saat proses pelatihan. Proses pelatihan dengan arsitektur AlexNet ini memerlukan waktu setidaknya 1742 detik atau 29 menit untuk 50 epoch, bergantung pada kecepatan baca tulis dan kapasitas RAM dan ROM, jika menggunakan dukungan hardware TPU v2 dengan RAM 334.6 GB, Disk 225.3 GB, proses pelatihan memerlukan waktu yang lebih sedikit, untuk epoch 25 dalam waktu kurang lebih 600 detik atau setara dengan 10 menit. Hasil akurasi pada dataset 1 dipaparkan pada Tabel 4 sebagai berikut :

Tabel 4. Tabel hasil akurasi dataset 1

bs/epoch	5	10	25	50
8	20%	47.5%	57.5%	75%
16	10%	45%	52.5%	80%
32	5%	10%	12.5%	57.5%
64	2.5%	2.5%	22.5%	42.5%
128	2.5%	2.5%	10%	15%

Kami mengambil salah satu hasil akurasi tertinggi pada tabel 4 untuk perhitungan pada tabel 5 Confusion Matrix sebagai berikut :

Tabel 5. Tabel confusion matrix model 1

n = 40	Aktual positif	Aktual negatif
Prediksi positif	TP : 32	FP : 8
Prediksi negatif	FN : 0	TN : 0

$$\begin{aligned}
 \text{Akurasi} &= (TP + TN) / (TP + FP + FN + TN) \\
 &= (32 + 0) / (32 + 8 + 0 + 0) \\
 &= 0.8 * 100\% \\
 &= 80\%
 \end{aligned}$$

$$\begin{aligned}
 \text{Presisi} &= TP / (TP + FP) \\
 &= 32 / (32 + 8)
 \end{aligned}$$

$$= 0.8 * 100\%$$

$$= 80\%$$

Recall

$$= TP / (TP + FN)$$

$$= 32 / (32 + 0)$$

$$= 1 * 100\%$$

$$= 100\%$$

F1-Score

$$= (2 * Recall * Presisi) / Recall + Presisi)$$

$$= (2 * 1 * 0.8) / (1 + 0.8)$$

$$= 0.888 * 100\%$$

$$= 88.8\%$$

Hasil pada tabel 3 bahwa akurasi paling tinggi adalah 80% pada batch size (bs) 16 dengan epoch 50. Kemudian setelah datasetnya diubah dan dibersihkan, menghasilkan akurasi yang sedikit lebih baik dengan epoch yang sama, bisa dilihat di Tabel 6.

Tabel 6. Tabel hasil akurasi dataset 2

bs/epoch	5	10	25	50
8	25%	50%	62.5%	77.5%
16	17.5%	50%	47.5%	80%
32	7.5%	30%	57.4%	45%
64	7.5%	7.5%	25%	45%
128	7.5%	7.5%	20%	52.5%

Dari kedua hasil akurasi pada tabel 3 dan tabel 5, bisa dilihat bahwa akurasinya sedikit mengalami kenaikan pada tabel 5. Model ini awalnya dilatih dengan 10 epoch dengan batch size 16, proses ini menghasilkan akurasi 45% pada tabel 3, kemudian menggunakan dataset kedua, akurasinya bertambah menjadi 50% dengan batch size dan epoch yang sama, karena banyaknya noise atau gangguan pada dataset saat sistem melakukan pelatihan, dan juga tingkat bobot pelatihan setiap kelas yang berbeda menghasilkan akurasi di Tabel 5 lebih tinggi dibandingkan akurasi di Tabel 3.

4.4.2 Model AlexNet yang disesuaikan

Pelatihan model dengan arsitektur AlexNet yang disesuaikan juga memerlukan resource yang besar, Proses pelatihan dengan arsitektur AlexNet dengan dukungan hardware CPU ini memerlukan waktu setidaknya 870 detik atau setara dengan 14 menit 30 detik untuk 50 epoch, bergantung pada kecepatan baca tulis dan kapasitas RAM dan ROM, jika menggunakan dukungan hardware TPU v2 dengan RAM 334.6 GB, Disk 225.3 GB, proses pelatihan memerlukan waktu yang lebih sedikit, untuk epoch 50 dalam waktu kurang lebih 330 detik atau setara dengan 5 menit 30 detik.

Hasil akurasi pada dataset 1 menggunakan model ke-2 dipaparkan pada tabel 7 sebagai berikut :

Tabel 7. Tabel hasil akurasi dataset 1

bs/epoch	5	10	25	50
8	32.5%	62.5%	65%	50%
16	22.5%	70%	80%	85.5%
32	17.5%	67.5%	80%	85%
64	15%	55%	60%	55%
128	10%	50%	60%	42.5%

Kemudian untuk data yang sudah dibersihkan menghasilkan akurasi yang lebih tinggi dengan epoch yang lebih sedikit, dapat dilihat di tabel 8.

Tabel 8. Tabel hasil akurasi dataset 2

bs/epoch	5	10	15	25	50
8	80%	72.5%	75%	82.5%	82.5%
16	70%	77.5%	82.5%	85%	85%
32	42%	82.5%	80%	82.5%	72.5%
64	30%	77.5%	75%	82.5%	77.5%
128	15%	70%	65%	65%	85%

Dengan epoch 5, hasil akurasi tertinggi adalah batch size 8, kemudian untuk epoch 10 adalah batch size 32, dan terakhir epoch 15, 25, 50 batch size 16 menghasilkan akurasi tertinggi. Kami mengambil salah satu hasil akurasi tertinggi pada tabel 8 untuk dijadikan perhitungan pada tabel 9 Confusion Matrix sebagai berikut :

Tabel 9. Tabel confusion matrix

n = 40	Aktual positif	Aktual negatif
Prediksi positif	TP : 34	FP : 6
Prediksi negatif	FN : 0	TN : 0

$$\begin{aligned}
 \text{Akurasi} &= (TP + TN) / (TP + FP + FN + TN) \\
 &= (34 + 0) / (34 + 6 + 0 + 0) \\
 &= 0.85 * 100\% \\
 &= 85\%
 \end{aligned}$$

$$\begin{aligned}
 \text{Presisi} &= TP / (TP + FP) \\
 &= 34 / (34 + 6) \\
 &= 0.85 * 100\% \\
 &= 85\%
 \end{aligned}$$

$$\begin{aligned}
 \text{Recall} &= TP / (TP + FN) \\
 &= 34 / (34 + 0) \\
 &= 1 * 100\% \\
 &= 100\%
 \end{aligned}$$

$$\begin{aligned}
 \text{F1-Score} &= (2 * \text{Recall} * \text{Presisi}) / (\text{Recall} + \text{Presisi}) \\
 &= (2 * 1 * 0.85) / (1 + 0.85) \\
 &= 0.918 * 100\% \\
 &= 91.8\%
 \end{aligned}$$

Dari kedua hasil akurasi pada tabel 7 dan tabel 8, bisa dilihat bahwa akurasi tertingginya tidak berubah yaitu 85%, akan tetapi untuk mencapai akurasi tertinggi, kedua dataset itu membutuhkan epoch yang berbeda. Pada tabel 7, untuk mencapai akurasi 85% membutuhkan epoch 50 dengan batch size 16, dibandingkan pada tabel 8 yang membutuhkan epoch 25 dengan batch size 16 juga, selebihnya rata – rata akurasi lebih tinggi pada tabel 8 yang dimana menggunakan dataset kedua.

5 Kesimpulan

Kesimpulan yang dapat diambil berdasarkan hasil penelitian ini adalah Model arsitektur AlexNet dapat mengenali sapi dengan akurasi tertinggi adalah 80% dengan epoch sebanyak 50 dan waktu lebih dari 1 jam dibandingkan dengan model AlexNet yang sudah disesuaikan menghasilkan akurasi 85%

dengan epoch lebih sedikit, yaitu 25 epoch dalam waktu pelatihan 435 detik atau setara dengan 7 menit 15 detik, sistem identifikasi ini bisa dibuat dengan perangkat spesifikasi minimum, yang membutuhkan komputer / laptop dengan ram 4 GB, penyimpanan disk lokal 128GB, membutuhkan akses internet dan akun google untuk bisa terhubung dengan google colab. selain itu kecepatan pelatihan dataset dipengaruhi oleh model dan juga oleh dukungan hardware yang baik, dataset yang sudah dicrop berpengaruh penting dalam peningkatan akurasi pengenalan sapi pada sistem.

Referensi

- [1] A I Awad, "From classical methods to animal biometrics: A review on cattle identification and tracking," *Computers and Electronics in Agriculture*, vol 123, 2016
- [2] M Vlad, R A Parvulet, M S Vlad et al, "A survey of livestock identification systems," in *Proceedings of the 13th WSEAS International Conference on Automation and Information*,(ICAI12) Iasi, Romania: WSEAS Press, 2012.
- [3] D Petrovska-Delacretaz, A Edwards, J Chiassoli, G Chollet, and D S ' Pilliod, "A reference system for animal biometrics: Application to the northern leopard frog," in *2014 1st International Conference on Advanced Technologies for Signal and Image Processing (ATSIP) IEEE*, 2014
- [4] A Johnston and D Edwards, "Welfare implications of identification of cattle by ear tags," *Veterinary Record*, vol 138, no 25, pp 612–614, 1996.
- [5] G H Collins, S L Petersen, C A Carr, and L Pielstick, "Testing vhf/gps collar design and safety in the study of free-roaming horses," *PloS one*, vol 9, no 9, p e103189, 2014.
- [6] Gatc, J, "Image segmentation of cattle muzzle using region merging statistical technic", *Journal of Computer Science and Technology*, 12(3), 11-20, 2015.
- [7] Arrofiqoh, E N, & Harintaka, H., "Implementasi Metode Convolutional Neural Network Untuk Klasifikasi Tanaman Pada Citra Resolusi Tinggi" *Geomatika*, 2018.
- [8] Abror, Z F, & Zaki, "Klasifikasi Citra Kebakaran dan Non Kebakaran Menggunakan Convolutional Neural Network", *Jurnal Ilmiah Teknologi dan Rekayasa*, 24(2), 1-10, 2019
- [9] Bello, Rw, Talib, AZH, Mohamed, ASAB, "Deep learning-based architectures for recognition of cow using cow nose image pattern", *Gazi University Journal of Science* 33, 831–844, 2020
- [10] Sanjel, A B, "Non-invasive muzzle matching for cattle identification using deep learning *Journal of Animal Science*", 99(3), 123-145, 2023
- [11] Krizhevsky, A, Sutskever, I, Hinton, GE "ImageNet classification with deep convolutional neural networks," *Adv Neural Inf Process Syst (NIPS)* 2012
- [12] Xiaobing Han, Yanfei Zhong, Liqin Cao, & Liangpei Zhang "Pre-Trained AlexNet Architecture with Pyramid Pooling and Supervision for High Spatial Resolution Remote Sensing Image Scene Classification," *Wuhan University, China*, 2017
- [13] Giri Wahyu Wiriasto, Misbahuddin, A. Sjamjiar Rachman, Muhamad Syamsu Iqbal, Djul Fikry Budiman, Lalu Ahmad Syamsul Irfan Akbar, Bagi Hartawan, and Eko Prasetyo Wicaksono. "Studi Pengembangan Sistem Pengenalan Individu Sapi Berbasis Biometrik Muzzle Menggunakan Model Mobilenetv2", *Journal of Computer Science and Informatics Engineering* 8, no. 1 (2024): 84-93. <https://doi.org/10.29303/jcosine.v8i1.594>.

- [14] Ali Shojaeipour, Greg Falzon, Paul Kwan, Nooshin Hadavi, Frances C. Cowley, dan David Paul. "Automated Muzzle Detection and Biometric Identification via Few-Shot Deep Transfer Learning of Mixed Breed Cattle", *Agronomy* 11, no. 11 (2021): 2365. <https://doi.org/10.3390/agronomy11112365>.
- [15] Bello, Rw, Talib, AZH, Mohamed, A.S.A., Olubummo, D.A., Ootobo, F.N., 2020a. "Image-based Individual Cow Recognition using Body Pattern", (IJACSA) *International Journal of Advanced Computer Science and Applications*, Vol. 11, No. 3, 2020.
- [16] Achour, B., Belkadi, M., Filali, I., Laghrouche, M., Lahdir, M., "Image Analysis For Individual Identification And Feeding Behaviour Monitoring Of Dairy Cows Based On Convolutional Neural Networks (CNN) ", *Biosystems Engineering* 198, 31–49, 2020.
- [17] Yang, Z., Xiong, H., Chen, X., Liu, H., Kuang, Y., Gao, Y., "Dairy cow tiny face recognition based on convolutional neural networks", in *Chinese Conference on Biometric Recognition*, Springer. pp. 216–222, 2019
- [18] Guan, H., Motohashi, N., Maki, T., Yamaai, T., "Cattle identification and activity recognition by surveillance camera". *Electronic Imaging* 2020, 174–1, 2020.
- [19] Andrew, W., Greatwood, C., Burghardt, T., "Visual localisation and individual identification of holstein friesian cattle via deep learning", in *Proceedings of the IEEE International Conference on Computer Vision Workshops*, pp. 2850–2859, 2017.
- [20] Wang, X., Cheng, X., Chen, Z., Xu, F., "A method for individual identification of dairy cows based on deep learning", in: *Proceedings of the 2020 2nd International Conference on Robotics, Intelligent Control and Artificial Intelligence*, pp. 186–191, 2020.
- [21] Zhou, Hao, and Jiashi Feng. "Overfitting Mechanism and Avoidance in Deep Neural Networks", arXiv preprint arXiv:1901.06566, 2019.
- [22] Auliaddina, Siti, and Toni Arifin. "Penggunaan data augmentasi dan hyperparameter tuning dalam klasifikasi jenis batik menggunakan model CNN." *SISTEMASI: Jurnal Sistem Informasi*, vol. 13, no. 1, 2024, pp. 114-128. <https://doi.org/10.32520/stmsi.v13i1.3395>.