

Identifikasi Jenis Burung Berdasarkan Bentuk Paruh dengan Menggunakan Metode Convolutional Neural Network

Putri Iara Diandra Siregar

Fakultas Ilmu Komputer dan Teknologi Informasi, Program Studi Teknik Informatika, Universitas Budi Darma, Medan, Indonesia
Email: putriiarasrg04@gmail.com

Abstrak—Perkembangan teknologi berkembang begitu cepat memungkinkan banyaknya aplikasi yang dibuat untuk memudahkan pekerjaan manusia, Sistem pendeteksian yang akan dikembangkan saat ini berdasarkan jenis suatu objek. Data yang berupa gambar dapat memberikan beberapa informasi seperti jenis, usia atau informasi lainnya. Burung merupakan anggota kelompok hewan bertulang belakang (vertebrata) yang memiliki bulu dan sayap, oleh karena itu deteksi jenis burung sangat dibutuhkan dilihat dari banyaknya jenis burung. Sehingga dibuatlah suatu sistem yang dapat mengidentifikasi jenis burung berdasarkan bentuk paruhnya dengan menggunakan metode Convolutional Neural Network (CNN). Tahapan-tahapan yang dilakukan sebelum identifikasi jenis burung berdasarkan bentuk yaitu proses pengolahan citra, grayscale, dan thresholding. Dengan teknologi tersebut maka seseorang dapat melihat spesies atau jenis burung beserta informasi lainnya berdasarkan paruhnya saja. Penelitian ini akan memanfaatkan kamera smartphone yang digunakan untuk mendeteksi objek yang dikhususkan hanya pada paruh burung saja untuk melihat spesies/jenisnya untuk dideteksi.

Kata Kunci: Convolutional Neural Network; Computer Vision; Pre-Processing

Abstract—Technological developments are developing so fast that it allows many applications to be made to facilitate human work. The detection system that will be developed at this time is based on the type of an object. Data in the form of images can provide some information such as type, age or other information. Birds are members of a group of vertebrate animals (vertebrates) that have feathers and wings, therefore detection of bird species is very much needed considering the number of bird species. The steps taken before identifying bird species based on shape are image processing, grayscale, and thresholding. With this technology, one can see species or types of birds along with other information based on their beaks. This study will utilize a smartphone camera that is used to detect objects that are specifically only on the beak of birds to see the species/type to be detected.

Keywords: Convolutional Neural Network; Computer Vision; Pre-Processing

1. PENDAHULUAN

Pada saat ini di zaman era modernisasi ilmu pengetahuan dan teknologimendorong manusia untuk melakukan otomatisasi dan digitalisasi sebagai perangkat-perangkat manual. Seiring dengan perkembangan teknologi pada organisasi, perusahaan dan tempat-tempat lainnya yang menggunakan rancangan untuk mendeteksi suatu bentuk benda secara teliti serta akurat seperti otak manusia yang di peragakan sebagai perangkat komputer, sehingga dengan adanya teknologimaka pendeteksi bentuk benda manual ini dapat digantikan dengan sistem pendeteksi bentuk yang otomatis. Teknologi tersebut merupakan suatu bidang dari *computer vision* yang memanfaatkan teknologi pada kamera. Dengan adanya teknologi tersebut pada pengolahan citra (*image processing*) maka objek apapun bisa dideteksi dalam bentuk digital dan akan diolah kemudian akan didapatkan informasi dari objek yang sudah dideteksi.

Pengolahan citra (*image processing*) ialah suatu teknik mengolah citra yang mentransformasikan citra masukan menjadi citra lain yang bertujuan untuk keluaran memiliki kualitas yang baik dibandingkan kualitas citra masukan. Pengolahan citra itu sendiri sangat bermanfaat, sebagai alat untuk meningkatkan kualitas citra[1], menghilangkan ketidaksempurnaan pada citra serta mengidentifikasi objek, penggabungan dengan bagian citra yang lain. Dengan teknologi ini, maka di bentuk suatu aplikasi yang dapat menangkap suatu objek yang ada di depan kamera bisa mengidentifikasi bentuk objek serta melakukan tracking objek secara *real-time*.

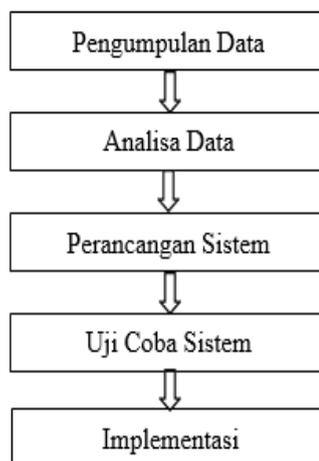
Sistem pendeteksian pada benda akan dikembangkan saat ini berdasarkan jenis sebuah objek. Data yang berupa *image* (gambar) bisa memberikan beberapa informasi berupa jenis burung. Burung merupakan salah satu anggota kelompok hewan bertulang belakang yang memiliki bulu dan sayap serta memiliki jenis yang sangat banyak di alam, oleh karena itu deteksi jenis burung sangat dibutuhkan dari banyaknya jenisnya. Gambar burung yang akan diidentifikasi pada penelitian ini diambil tepat dari bagian paruhnya, agar terlihat beberapa perbedaan dari masing-masing jenis burung.

Sangat banyak aplikasi yang berkembang salah satunya adalah aplikasi pengenalan jenis dari sebuah citra atau yang lebih sering disebut OCR (*Optical Character Recognitions*)[2]. kebanyakan hasil penelitian yang berupa pengolahan sinyal citra digital yang berupa deteksi gambar ataupun tulisan dengan menggunakan berbagai macam metode dan teknologi diantaranya *K-Nearest Neighbor Algorithm* dan *Convolutional Neural Network*. Dari berbagai metode ada kelebihan serta kekurangannya masing-masing. Sedangkan pada algoritma *Convolutional Neural Network* ialah salah satu metode yang cukup dan mungkin sangat efektif untuk diterapkan sebagai pengenalan pada citra digital[3]. Metode pengenalan sangat sederhana dan bisa diterapkan tanpa harus melakukan proses ekstraksi ciri terlebih dahulu, namun begitu memiliki hasil pengenalan yang cukup bahkan sangat akurat[4]. Dilaporkan akan diteliti tentang pengolahan citra digital khususnya pada bidang pengenalan pola untuk menganalisa jenis Burung yang diambil berdasarkan bentuk Paruhnya. Sistem ini memakai metode identifikasi *Convolutional Neural Network* (CNN) yang nantinya dapat mengetahui tingkat akurasi untuk mengidentifikasi jenis pada burung.

2. METODOLOGI PENELITIAN

2.1 Kerangka Kerja Penelitian

Kerangka penelitian merupakan suatu bentuk kerangka yang dapat digunakan sebagai pendekatan dalam memecahkan masalah. Kerangka kerja dalam penelitian ini bertujuan untuk menguraikan semua tahapan-tahapan kegiatan yang dilaksanakan pada waktu penelitian agar sesuai dengan tujuan yang telah ditentukan. Secara garis besar. Adapun tahapan-tahapan dalam model penelitian ini meliputi pengumpulan data, analisa data, perancangan sistem, uji coba sistem, Implementasi sistem. Adapun kerangka kerja yang penulis lakukan dapat dilihat pada gambar 1. berikut:



Gambar 1. Kerangka Kerja Penelitian

2.2 Computer Vision

Computer Vision adalah suatu proses transformasi atau perubahan dari data yang berasal dari kamera video maupun foto/gambar kedalam sebuah hasil keputusan ataupun sebuah presentasi yang baru, dimana hasil dari kegiatan transformasi tersebut memiliki kepentingan untuk mencapai suatu tujuan. Data yang dimasukkan kedalam kegiatan transformasi tersebut memungkinkan untuk memiliki beberapa informasi yang terkonstektual seperti halnya sebuah foto/gambar yang didalamnya terdapat berbagai objek. Dengan demikian akan didapatkan keputusan-keputusan yang akan diambil pada gambar, misalnya berupa “adakah telapak tangan seseorang pada gambar tersebut?” atau “siapa sajakah orang yang terdapat pada foto tersebut?”. Adapun perubahan kedalam presentasi yang baru seperti perubahan gambar menjadi *grayscale* atau juga pemotongan objek pada gambar [5]

Berbeda halnya dengan manusia yang memiliki pemikiran untuk meneliti, memahami, dan membandingkan informasi pada objek secara langsung dengan informasi dari pengalaman-pengalaman yang didapatkan selama bertahun-tahun hidup di dunia.

2.3 Pengolahan Citra Digital (*Digital Image Processing*)

Proses perubahan citra menjadi citra digital disebut digitalisasi. Digitalisasi dapat dilakukan dengan menggunakan alat atau sensor seperti kamera, webcam scanner dan sebagainya. Pada umumnya citra digital berbentuk empat persegi panjang, dengan dimensi ukurannya dinyatakan sebagai lebar x tinggi.

Citra digital memiliki koordinat spasial, dengan tingkat kecerahan atau intensitas cahaya (skala keabu-abuan) yang memiliki numerik yang diskrit direpresentasikan dalam bentuk fungsi matematis $f(x,y)$ yang menyatakan intensitas cahaya pada titik (x,y) . Citra digital disimpan didalam berkas (*file*) dengan format tertentu. Format citra yang baku di lingkungan sistem operasi *Microsoft Windows* adalah berkas *bitmap* (bmp). Format bmp mempunyai kelebihan dari segi kualitas gambar apabila citra ditampilkan pada layar monitor, karena citra dalam format *BMP* umumnya tidak dimampatkan, sehingga tidak ada informasi yang hilang, walaupun akibatnya ukuran berkasnya relatif besar [6], [7].

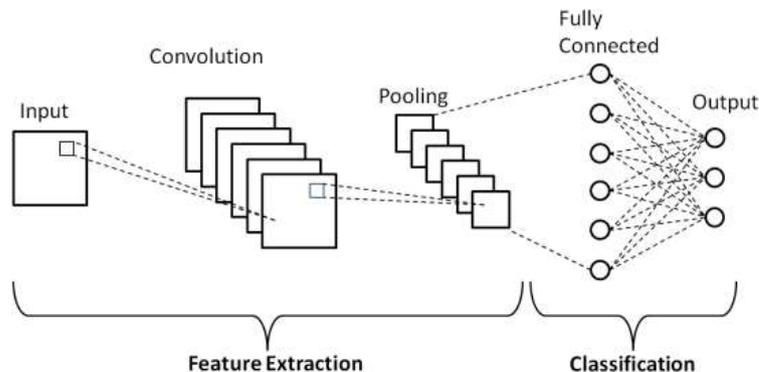
Citra dalam format bmp ada 3 macam yaitu citra biner, citra berwarna dan hitam putih (*grayscale*). Citra biner hanya memiliki dua nilai keabuan, 0 dan 1. Oleh karena itu, 1 *Bit* sudah cukup untuk merepresentasikan nilai piksel. Sedangkan citra yang lebih umum adalah citra berwarna. Adapun warna yang terlihat pada gambar dengan format bmp merupakan kombinasi dari tiga warna dasar yaitu *Red, Green, dan Blue* (RGB). Setiap piksel pada layar monitor disusun oleh tiga komponen warna tersebut. Kombinasi dari ketiga warna tersebut menghasilkan warna yang khas untuk satu piksel bersangkutan.

2.4 Convolutional Neural Network

Convolutional Neural Network (CNN) adalah pengembangan dari *Multilayer Perceptron* (MLP) yang didesain untuk mengolah data dua dimensi yang terdiri dari lapisan *input* dan *output*, serta beberapa lapisan tersembunyi [8], [9]. CNN menjadi sangat populer setelah memenangkan *ImageNet Large Scale Recognition Challenge ILSVRC 2012*. Dalam

makalah tersebut, mereka menggunakan lebih dari 600.000 neuron dan 7 lapisan tersembunyi untuk memberikan model data yang baik untuk menghindari *overfitting* [10].

Pada *CNN*, setiap neuron direpresentasikan dalam bentuk dua dimensi, tidak seperti *MLP* yang setiap neuron hanya berukuran satu dimensi. *CNN* termasuk dalam *Deep Neural Network* karena kedalaman jaringan yang tinggi. *Convolutional Neural Network (CNN)* merupakan salah satu jenis *Neural Network* yang berisi kombinasi beberapa layer yaitu *Convolutional layer*, *pooling layer* dan *fully connected layer* [11] seperti pada Gambar 2.

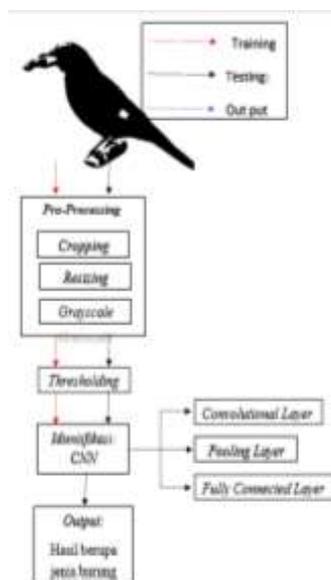


Gambar 2. Convolutional neural network

3. HASIL DAN PEMBAHASAN

3.1 Analisa Masalah dan Penerapan Metode

Analisa masalah merupakan tahapan penjelasan terhadap penyelesaian masalah. Adapun tahapan tersebut dapat dilihat berikut:



Gambar 3. Penerapan Metode

Pada gambar 3. ada beberapa tahapan yang akan digunakan untuk mengidentifikasi jenis burung berdasarkan paruh yaitu;

1. Input image

Input dari sistem ini adalah bentuk paruh burung yang diperoleh dari hasil capture citra burung yang diambil di lokasi alam bebas yang berupa jenis burung liar ataupun peternakan burung. Data citra dibagi menjadi 2 jenis, yaitu data training dan data testing. Hasil training disimpan dalam referensi CNN. Jumlah data yang digunakan untuk penelitian adalah 30 citra dan untuk data uji sebanyak 15 citra.

2. Pre-processing

Pada tahap ini akan dilakukan proses citra yaitu cropping untuk memotong citra, resizing yaitu untuk mengubah ukuran citra. Grayscale yang bertujuan untuk mengkonversi citra berwarna menjadi citra skala keabuan yang digunakan untuk mengurangi beban pada citra ketika di proses selanjutnya. Metode thresholding yaitu memisahkan pada perbedaan tingkat kecerahan atau gelap terangnya.

3. Identification

Pada tahap ini, bentuk citra jenis burung yang telah diproses pada tahap sebelumnya akan diidentifikasi. Untuk mengidentifikasi jenis burung berdasarkan bentuk paruh ada beberapa sisi dengan menggunakan metode Convolutional neural network (CNN).

4. Output

Output yang digunakan yaitu hasil identifikasi terhadap citra burung berdasarkan jenis paruh dimana hasilnya kita bisa mengetahui jenis burung itu sendiri.

5. Input image

Input dari system ini adalah bentuk paruh burung yang diperoleh dari hasil capture citra burung yang di ambil dilokasi alam bebas yang berupa jenis burung liar ataupun perternakan burung. Data citra dibagi menjadi 2 jenis, yaitu data training dan data testing. Hasil training diseimpa dalam refrence CNN. Jumlah data

6. Pre-processing

Pada tahap ini akan dilakukan proses citra yaitu cropping untuk memotong citra, resizing yaitu untuk mengubah ukuran citra. Grayscale yang bertujuan untuk mengkonversi citra berwarna menjadi citra skala keabuan yang di gunakan untuk mengurangi beban pada citra ketika di proses selanjutnya. Metode thresholding yaitu memisahkan pada perbedaan tingkat kecerahan atau gelap terangnya.

7. Identification

Pada tahap ini, bentuk citra jenis burung yang telah diproses pada tahap sebelumnya akan diidentifikasi. Untuk mengidentifikasi jenis burung berdasarkan bentuk paruh ada beberapa sisi dengan menggunakan metode Convolutional neural network (CNN).

8. Output

Output yang digunakan yaitu hasil identifikasi terhadap citra burung berdasarkan jenis paruh dimana hasilnya kita bisa mengetahui jenis burung itu sendiri, tahap selanjutnya yaitu identifikasi menggunakan metode Convolutional Neural Network (CNN) yang terdiri dari 3 layer yaitu Convolutinal layer, Max pooling layer dan fully connected layer. Proses ini bertujuan untuk melakukan identifikasi tahap objek.

3.1.1 Convolutional layer

Convolutional Layer melakukan oprasi konvolusi terhadap input ataupun output dari layer sebelumnya. Tujuan dilakukan konvolusi pada data citra adalah untuk mengekstrasi fitur dari citra input seperti gambar 3.9



Gambar 4. Contoh Citra input 5 x 5 pixel

Nilai idientitas citra lebih dari atau sama dengan nilai *thresholding* akan diubah jadi 1 (berwarna putih) sedangkan nilai intensitas citra yang kurang dari nilai *thresholding* akan diubah menjadi 0 (warna hitam). Perubahan lain *pixel* pada proses *thresholding* ditunjukkan pada gambar 5.

0	0	0	0	0	0	0	0
225	225	225	0	0	1	1	1
225	225	225	0	0	1	1	1
225	225	0	0	0			
225	225	0	0	0			

Gambar 5. Perubahan nilai *pixel* citra input dan *Feature Detector*

Hasil dari jumlah perkalian matriks antara matriks *feature detector* dan matriks pada citra input akan menghasilkan *feature map* seperti pada gambar 6. merupakan representasi *feature map* 3 x 3 *pixel*.

$C_{1,1}$	$C_{2,1}$	$C_{3,1}$
$C_{1,2}$	$C_{2,2}$	$C_{3,2}$
$C_{1,3}$	$C_{2,3}$	$C_{3,3}$

Gambar 6. *Feature Map*

Gambar 6. merupakan salah satu contoh citra input yang telah direpresentasikan ke dalam bentuk 5 x 5 piksel dengan perubahan nilai *pixel* pada proses *threshold* 3 x 3 sebagai *feature detector* proses konvolusi. Gambar 4.4

merupakan representasi *feature map* 3 x 3 *pixel*. Nilai *stride* yang digunakan dalam penelitian ini adalah 1 karena proses pengklasifikasian pada sistem ini juga berdasarkan bentuk. *Feature detector* bergerak dari sudut kiri atas dari gambar input dan bergeser ke kanan sebanyak 1 *pixel* hingga semua *pixel* dari baris pertama gambar input terlalui. Setelah semua baris pertama dari citra input dilalui, *feature map* akan turun ke baris selanjutnya hingga semua *pixel* yang dimiliki dilalui kemudian disimpan dalam matriks yang baru. Proses konvolusi dapat dilihat pada Gambar 7.

0	0	0	0	0
225	225	225	0	0
225	225	225	0	0
225	225	0	0	0
225	225	0	0	0
0	0	0	0	0
225	225	225	0	0
225	225	225	0	0
225	225	0	0	0
225	225	0	0	0
0	0	0	0	0
225	225	225	0	0
225	225	225	0	0
225	225	0	0	0
225	225	0	0	0
0	0	0	0	0
225	225	225	0	0
225	225	225	0	0
225	225	0	0	0
225	225	0	0	0
0	0	0	0	0
225	225	225	0	0
225	225	225	0	0
225	225	0	0	0
225	225	0	0	0
0	0	0	0	0
225	225	225	0	0
225	225	225	0	0
225	225	0	0	0
225	225	0	0	0
0	0	0	0	0
225	225	225	0	0
225	225	225	0	0
225	225	0	0	0
225	225	0	0	0
0	0	0	0	0
225	225	225	0	0
225	225	225	0	0
225	225	0	0	0
225	225	0	0	0
0	0	0	0	0
225	225	225	0	0
225	225	225	0	0
225	225	0	0	0
225	225	0	0	0
0	0	0	0	0
225	225	225	0	0
225	225	225	0	0
225	225	0	0	0
225	225	0	0	0
0	0	0	0	0
225	225	225	0	0
225	225	225	0	0
225	225	0	0	0
225	225	0	0	0

Gambar 7. Representasi proses konvolusi

Untuk setiap *pixel*, kita menghitung jumlah elemen perkalian antara matriks *feature detector* dan matriks pada gambar input. Hasil dari jumlah perkalian matriks tersebut akan menghasilkan *feature map*. Nilai dari *feature map* dihitung melalui Persamaan 2.3

$$C_{i,j} = A \times P1 + B \times P2 + C \times P3 + D \times P4 + E \times P5 + F \times P6 + G \times P7 + H \times P8 + I \times P9 \tag{1}$$

$$C1,1 = (0 \times 0) + (0 \times 0) + (0 \times 0) + (255 \times 1) = 1530$$

$$C2,1 = (0 \times 0) + (0 \times 0) + (0 \times 0) + (255 \times 1) + (255 \times 1) + (0 \times 1) + (255 \times 1) + (255 \times 1) + (0 \times 1) = 102$$

$$C3,1 = (0 \times 0) + (0 \times 0) + (0 \times 0) + (255 \times 1) + (0 \times 1) + (0 \times 1) + (255 \times 1) + (0 \times 1) + (0 \times 1) = 510$$

$$C1,2 = (255 \times 0) + (255 \times 0) + (255 \times 0) + (255 \times 1) + (0 \times 1) = 1275$$

$$C2,2 = (255 \times 0) + (255 \times 0) + (0 \times 0) + (255 \times 1) + (255 \times 1) + (0 \times 1) + (255 \times 1) + (0 \times 1) + (0 \times 1) = 765$$

$$C3,2 = (255 \times 0) + (0 \times 0) + (0 \times 0) + (255 \times 1) + (0 \times 1) = 255$$

$$C1,3 = (255 \times 0) + (255 \times 0) + (255 \times 0) + (255 \times 1) + (255 \times 1) + (0 \times 1) + (255 \times 1) + (255 \times 1) + (0 \times 1) = 1020$$

$$C2,3 = (255 \times 0) + (255 \times 0) + (0 \times 0) + (255 \times 1) + (0 \times 1) + (0 \times 1) + (255 \times 1) + (0 \times 1) + (0 \times 1) = 51$$

$$C3,3 = (255 \times 0) + (0 \times 0) + (0 \times 0) + (0 \times 1) = 0$$

Hasil dari jumlah perkalian matriks antara matriks gambar input dan *feature detector* tersebut akan menghasilkan *feature map*. Nilai *feature map* yang didapatkan dapat dilihat pada Gambar 8.

1530	1020	510
1275	765	255
1020	510	0

Gambar 8. Hasil *feature map*

3.1.2 Max Pooling Layer

Hasil dari proses konvolusi tersebut selanjutnya masuk ke proses *max pooling*. Tahap ini akan membagi *feature map* menjadi beberapa *grid* kecil, kemudian mengambil nilai maksimal dari setiap *grid* tersebut. *Max pooling* akan menghasilkan *output* yang lebih kecil dari *feature map* tetapi mengambil nilai maksimal dari matriks tersebut. Proses dari *Max pooling* berukuran 2 x 2 *pixel* direpresentasikan pada Gambar 9 dan hasil dari *max pooling* pada Gambar 10.

1530	1020	510	1530	1020	510
1275	765	255	1275	765	255
1020	510	0	1020	510	0
1530	1020	510	1530	1020	510
1275	765	255	1275	765	255
1020	510	0	1020	510	0

Gambar 9. Proses *Max Pooling*

Hasil nilai maksimum pada *Ma pooling* ditunjukkan pada gambar 4.8

1530	1020
1275	765

Gambar 10. Hasil *Max pooling*

Gambar 10. merupakan hasil dari *Max Pooling layer* dari Gambar 4.6. Matriks yang digunakan untuk *max pooling* berukuran 2×2 *pixel* dan mencari nilai maksimum yang ada pada matriks tersebut. Matriks 2×2 bergerak dari sudut kiri atas pada *feature map* dan bergeser ke matriks 2×2 selanjutnya sampai melalui semua nilai yang terdapat pada *feature map*. Hasil dari proses *max pooling* kemudian dimasukkan kedalam proses *flattening*.

3.1.2.1 Flattening

Hasil dari proses *max pooling* kemudian dimasukan kedalam proses *flattening*. Pada proses ini setiap hasil dari *max pooling layer* harus ditransformasikan menjadi *array* suatu dimensi sebelum dimasukan ke proses *fullyconnected layer*. Hasil dari proses *flattening* dapat dilihat pada gambar 11.

1530
1275
1020
765

Gambar 11. Hasil *flattening*

3.1.3 Fully Connected Layer

Pada tahap ini, dilakukan pembuatan model convolutional neural network antara lain menentukan jumlah hidden layer, epoch dan jumlah neuron.

3.1.3.1 Penentuan hidden layer

Pada penelitian ini, penentuan dari jumlah hidden layer yang optimal dilakukan dengan menggunakan cara identifikasi. Jumlah hidden layer yang digunakan pada penelitian ini adalah dari 50 hidden layer.

3.1.3.2 Penentuan epoch

Penentuan jumlah epoch yang optimal dilakukan dengan menggunakan cara identifikasi. epoch yang digunakan pada penelitian ini adalah dari 1000 epoch.

3.1.3.3. Penentuan jumlah neuron

Jumlah neuron pada hidden layer yang akan digunakan adalah 300×300 *pixel* yaitu 90.000 neuron. Pada penelitian ini, jumlah neuron yang digunakan pada *fully connected layer* adalah sebanyak 90.000 neuron.

```

double Min = 0;
int idxx = 0;
HasilTest3 = CNN.process(TestData);
for (int i = 0; i < HasilTest3.Length; i++)
{
    if (Min < HasilTest3[i])
    {
        Min = HasilTest3[i];
        idxx = i;
    }
}
textBox5.Text = CekKlasifikasi(idxx);

public String CekKlasifikasi(int idxx)
{
    String B = "";
    int suv = Convert.ToInt32(textBox12.Text);
    int mpv = suv + Convert.ToInt32(textBox11.Text);
    int sedan = mpv + Convert.ToInt32(textBox3.Text);

```

Gambar 12. Pseudocode proses Metode CNN

4. KESIMPULAN

Kesimpulan yang dapat diambil berdasarkan pengujian terhadap identifikasi jenis burung berdasarkan bentuk paruh dengan menggunakan metode *Convolutional Neural Network* (CNN) adalah sebagai berikut: Penggunaan metode

Convolutional Neural Network (CNN) dapat mengidentifikasi jenis burung berdasarkan bentuk paruh dengan nilai akurasi mencapai 94,44 %. Berdasarkan pengujian sistem, akurasi identifikasi jenis burung dipengaruhi oleh kondisi citra saat diuji. Akurasi terbaik adalah citra burung dalam keadaan terangan posisi paruh burung yang tepat, sedangkan terendah adalah dalam keadaan gelap dan posisi tidak tepat

REFERENCES

- [1] A. Yulianto, "Implementasi Teknologi Markerless Augmented Reality Berbasis Android sebagai Media Pengenalan Gedung-Gedung di FMIPA Universitas Lampung," pp. 95–102, 2015.
- [2] M. Gjoreski and G. Zajkovski, "Optical character recognition applied on receipts printed in Macedonian Language," no. August, pp. 59–62, 2014, doi: 10.13140/2.1.1632.4489.
- [3] M. A. Pangestu and H. Bunyamin, "Analisis Performa dan Pengembangan Sistem Deteksi Ras Anjing pada Gambar dengan Menggunakan Pre- Trained CNN Model," *J. Tek. Inform. dan Sist. Inf.*, vol. 4, pp. 337–344, 2018.
- [4] J. C. Russ, J. R. Matey, A. J. Mallinckrodt, and S. McKay, "The Image Processing Handbook," *Comput. Phys.*, vol. 8, no. 2, p. 177, 1994, doi:10.1063/1.4823282.
- [5] D. A. Latifah, B. Hidayat, and T. A. Wibowo, "Dan Deteksi Tepi Canny," 2011.
- [6] A. R. Putri, "Pengolahan Citra Dengan Menggunakan Web Cam Pada Kendaraan Bergerak Di Jalan Raya," *JIPi (Jurnal Ilm. Penelit. dan Pembelajaran Inform.)*, vol. 1, no. 01, pp. 1–6, 2016, doi: 10.29100/jipi.v1i01.18.
- [7] N. Wakhidah, "Deteksi Plat Nomor Kendaraan Bermotor Berdasarkan Area Pada Image Segmentation," *J. Transform.*, vol. 9, no. 2, p. 55, 2012, doi: 10.26623/transformatika.v9i2.58.
- [8] R. D. Kusumanto, A. N. Tompunu, and S. Pambudi, "Klasifikasi Warna Menggunakan Pengolahan Model Warna HSV Abstrak," *J. Ilm. Tek. Elektro*, vol. 2, no. 2, pp. 83–87, 2011.
- [9] Y. Xiao, "Vehicle Detection in Deep Learning," 2019, [Online]. Available: <http://arxiv.org/abs/1905.13390>.
- [10] T. F. Gonzalez, "Handbook of approximation algorithms and metaheuristics," *Handb. Approx. Algorithms Metaheuristics*, pp. 1–1432, 2007, doi: 10.1201/9781420010749.