

BAB II

LANDASAN TEORI

2.1 Penelitian Terdahulu

Berikut ini merupakan beberapa penelitian yang membahas pendeteksian kesegaran ikan menggunakan metode Machine Learning.

Penelitian pertama dari Mohammadi Lalabadi, Hosna Sadeghi, Morteza Mireei, Seyed Ahmad pada tahun 2019. Penelitian ini membahas penerapan, Kategorisasi kesegaran ikan dari mata dan fitur warna insang menggunakan jaringan saraf buatan multiklas dan mendukung mesin vector. Percobaan menggunakan dua alat pembelajaran mesin ANN dan SVM di antara kedua pengklasifikasi, ANN menghasilkan akurasi klasifikasi yang agak lebih baik[4].

Penelitian kedua berjudul “Implementasi HSV dan GLCM untuk Deteksi Kesegaran Ikan Bawal menggunakan Radial Basis Function Berbasis Android”. Dari 12 percobaan terdapat 6 percobaan benar dan 6 percobaan salah sehingga memiliki akurasi sebesar 50% [5].

Selanjutnya merupakan penelitian dari “Aplikasi Penentuan Tingkat Kesegaran Ikan Selar Berbasis Citra Digital Dengan Metode Kuadrat Terkecil”. Penelitian ini membahas bagaimana penentuan tingkat kesegaran ikan selar menggunakan pengolahan citra digital berdasarkan analisis perubahan warna mata ikan dengan metode kuadrat terkecil yaitu sebesar 81,333 % [6].

Yang keempat merupakan penelitian yang berjudul “Alat pendeteksi kesegaran ikan menggunakan metode K-Nearest Neighbor berdasarkan mata berbasis ATmega328 “. Penelitian ini membahas bagaimana metode K-Nearest Neighbor yang dimasukkan ke dalam program ATmega328. Hasilnya adalah persentase keberhasilan 73,33% [7].

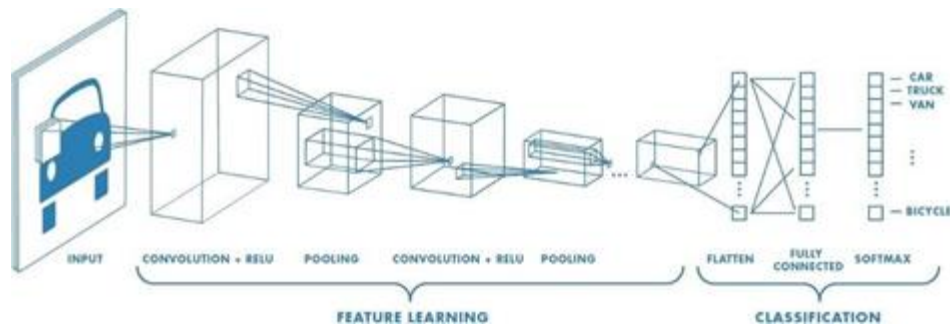
Penelitian terakhir membahas tentang “Klasifikasi kesegaran ikan bandeng dalam smart packaging berbasis image processing menggunakan metode fuzzy “.

Penelitian tersebut diambil kesimpulan yaitu pengolahan citra warna terbukti dapat mendeteksi nilai RGB pada *Polyaniline* dengan nilai hijau 82-184 dan nilai biru 67-174 [8].

2.2 Teori Umum

2.2.1 Convolutional Neural Network (CNN)

Penemuan CNN pertama kali dilakukan oleh Hubel dan Wiesel mengenai *virtual cortex* pada indera penglihatan kucing. Secara teknis, CNN adalah sebuah arsitektur yang dapat dilatih dan terdiri dari beberapa tahap, masukan (input) dan keluaran (output) [14]. Secara teknis, CNN melatih dan menguji setiap gambar masukan melalui serangkaian proses, yaitu *convolutional layer* diikuti oleh *pooling* untuk mengekstraksi fitur dari gambar masukan berturut-turut. Setelah operasi *pooling*, citra di-flatten dan kemudian dimasukkan ke proses *fully connected-layer* untuk melaksanakan tugas pengklasifikasian. Arsitektur yang menggambarkan proses-proses yang terjadi pada metode CNN ditampilkan pada Gambar 2.1[15].



Gambar 2.1 Ilustrasi Arsitektur CNN

1. Convolusi Layer

CNN (Convolutional Neural Network) menggunakan filter yang juga disebut sebagai kernel untuk mendeteksi fitur- fitur, seperti sudut yang terdapat di seluruh gambar. Filter hanyalah sebuah matriks nilai yang disebut bobot, dilatih untuk mendeteksi fitur tertentu. Filter bergerak di setiap bagian gambar untuk memeriksa apakah fitur yang dimaksudkan untuk dideteksi tersedia. Untuk memberikan nilai yang menunjukkan

seberapa tepat fitur yang tersedia tersebut, filter melakukan operasi konvolusi [16].

2. *Downsampling*

Downsampling berguna untuk mempercepat proses pelatihan dan mengurangi jumlah memori yang dikonsumsi oleh *network*. *Downsampling* yang umum digunakan adalah max pooling [16].

3. Fully-Connected Layer

Fully-Connected Layer dari neural network adalah proses dimana matriks masukan diflatten menjadi fitur vektor dan melewati jaringan neuron untuk memprediksi probabilitas keluaran [16]. Proses ini menggunakan ANN (Artificial Neural Network) yang terdiri atas 2 tahapan yaitu feedforward [17]:

$$z = \sum_{i,j} x_{ij} w_{ij} + b_i \quad (2.1)$$

$$\sigma(z) \begin{cases} \text{sigmoid activation} = \frac{1}{(1+e^{z_{ij}})} \\ \text{softmax activation} = \frac{e^{z_{ij}-\max(z_j)}}{\sum e^{z_{ij}-\max(z_j)}} \end{cases} \quad (2.2)$$

Keterangan:

- i = nilai baris matriks
- j = nilai kolom matriks
- z = jumlah bobot masukan
- w = *weight*
- b = bias
- σ = fungsi aktivasi

Untuk tahapan yang kedua yaitu *backpropagation*, persamaan-persamaan yang digunakan adalah sebagai berikut [18]:

$$z\delta_{ij} = \frac{1}{n} (a_{ij} - y_{ij}) w_{ij}^T \quad (2.3)$$

$$a\delta_{ij} = z\delta_{ij} \times \sigma^l(z) \quad (2.4)$$

$$\sigma^i(z) = (1 - \sigma(z)) \quad (2.5)$$

$$w_{ij} = w_{ij} - n x_{ij}^T a\delta_{ij} \quad (2.6)$$

$$b_{ij} = b_{ij} - n \sum a\delta_{ij} \quad (2.7)$$

Keterangan:

$z\delta$ = pencarian *gradient*

$a\delta$ = hasil perkalian *gradient* dengan aktivasi *derrivative*

η = *learning rate*

w = *weight*

b = bias

σ' = fungsi aktivasi *derivative*

2.2.2 Evaluasi

Confusion matrix adalah matriks untuk mengukur kinerja suatu klasifikasi yang memiliki 4 variabel yaitu TP (True Positive) berarti ketika sistem memprediksi positif dan hasilnya benar, TN (True Negative) berarti ketika sistem memprediksi negatif dan hasilnya benar, FP (False Positive) berarti ketika sistem memprediksi positif dan hasilnya salah, FN (False Negative) berarti ketika sistem memprediksi negatif dan hasilnya salah [19]. Ada beberapa perhitungan yang dapat digunakan sebagai pengujian pada sistem diantaranya [20]:

$$Accuracy = \frac{(TP+TN)}{(TP+FP+TN+FN)} \quad (2.8)$$

$$Error Rate = \frac{\text{Jumlah gambar yang salah diidentifikasi}}{\text{jumlah gambar}} \quad (2.9)$$

$$Precision = \frac{TP}{(TP+FP)} \quad (2.10)$$

$$Recall = \frac{TP}{(TP+FN)} \quad (2.11)$$

2.2.3 Citra Digital

Pengolahan citra digital (Digital Image Processing) adalah sebuah disiplin ilmu yang mempelajari tentang teknik-teknik mengolah citra. Citra yang dimaksud disini adalah gambar diam (foto) maupun gambar bergerak (yang berasal dari webcam). Sedangkan digital disini mempunyai maksud bahwa pengolahan citra/gambar dilakukan secara digital menggunakan komputer [3]. Secara matematis, citra merupakan fungsi kontinyu (continue) dengan intensitas cahaya pada bidang dua

dimensi. Agar dapat diolah dengan komputer digital, maka suatu citra harus dipresentasikan secara numerik dengan nilai-nilai diskrit. Representasi dari fungsi kontinu menjadi nilai-nilai diskrit disebut digitalisasi citra. Sebuah citra digital dapat diwakili oleh sebuah matriks dua dimensi $f(x, y)$ yang terdiri dari M kolom dan N baris, dimana perpotongan antara kolom dan baris disebut piksel (pixel = picture element) atau elemen terkecil dari sebuah citra.

$$f(x, y) \approx \begin{bmatrix} f(0,0) & f(0,1) & \dots & f(0, M-1) \\ f(1,0) & f(1,1) & \dots & f(1, M-1) \\ \vdots & \vdots & \ddots & \vdots \\ f(N-1,0) & f(N-1,1) & \dots & f(N-1, M-1) \end{bmatrix} \quad (2.12)$$

Suatu citra $f(x,y)$ dalam fungsi matematis dapat dituliskan sebagai berikut:

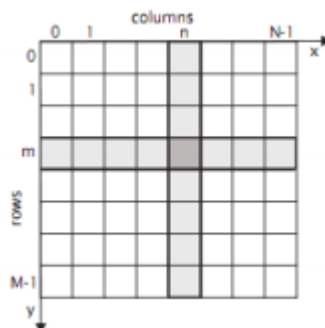
$$\begin{aligned} 0 &\leq x \leq M-1 \\ 0 &\leq y \leq N-1 \\ 0 &\leq f(x,y) \leq G-1 \end{aligned}$$

- dimana :
- M = jumlah piksel baris (row) pada array citra
 - N = jumlah piksel kolom (column) pada array citra
 - G = nilai skala keabuan (graylevel)

Besarnya nilai M , N dan G pada umumnya merupakan perpangkatan dari dua.

$$M = 2^m; N = 2^n; G = 2^k \quad (2.13)$$

dimana nilai m , n dan k adalah bilangan bulat positif. Interval $(0, G)$ disebut skala keabuan (grayscale). Besar G tergantung pada proses digitalisasinya. Biasanya keabuan 0 (nol) menyatakan intensitas hitam dan 1 (satu) menyatakan intensitas putih. Untuk citra 8 bit, nilai G sama dengan $2^8 = 256$ warna (derajat keabuan).



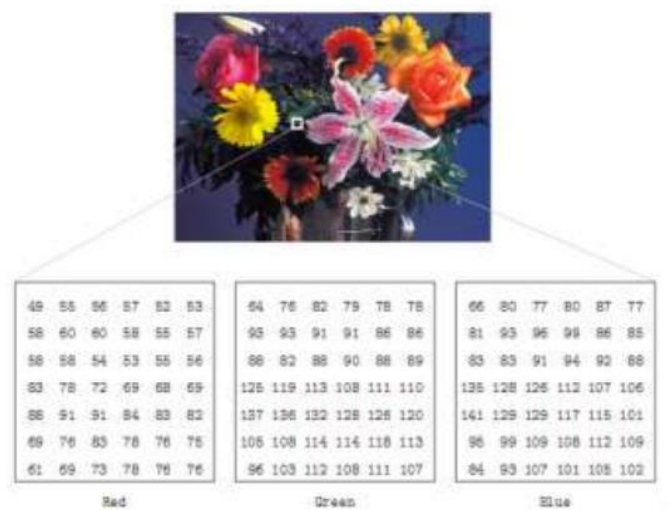
Gambar 2.2 Representasi citra digital dalam 2 dimensi

Obyek tertentu dapat dideteksi dengan menggunakan pengolahan citra digital ini. Salah satu metode yang digunakan adalah berdasarkan segmentasi warna. Normalisasi RGB adalah salah satu metode segmentasi warna yang memiliki kelebihan yaitu mudah, proses cepat dan efektif pada obyek traffic sign [9], maupun aplikasi untuk face detection [10].

2.2.3.1 Jenis Citra Digital

Pada aplikasi pengolahan citra digital pada umumnya, citra digital dapat dibagi menjadi 3, color image, black and white image dan binary image.

1. Color Image atau RGB (Red, Green, Blue). Pada color image ini masing-masing piksel memiliki warna tertentu, warna tersebut adalah merah (Red), hijau (Green) dan biru (Blue). Jika masing-masing warna memiliki range 0 - 255, maka totalnya adalah $255^3 = 16.581.375$ (16 K) variasi warna berbeda pada gambar, dimana variasi warna ini cukup untuk gambar apapun. Karena jumlah bit yang diperlukan untuk setiap pixel, gambar tersebut juga disebut gambar-bit warna. Color image ini terdiri dari tiga matriks yang mewakili nilai-nilai merah, hijau dan biru untuk setiap pikselnya, seperti yang ditunjukkan gambar 2.3.



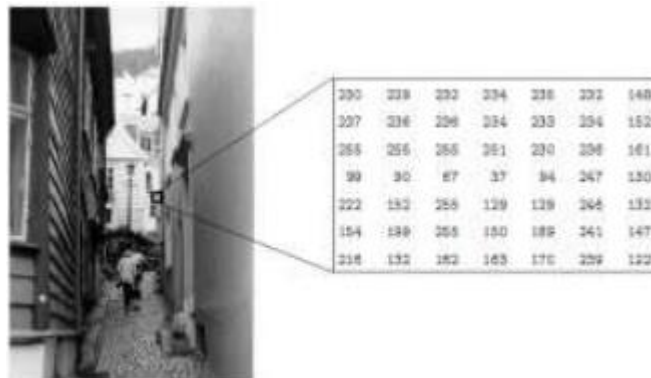
Gambar 2.3 Color Image

2. Black and White

Citra digital black and white (grayscale) setiap pikselnya mempunyai warna gradasi mulai dari putih sampai hitam. Rentang tersebut berarti bahwa setiap piksel dapat diwakili oleh 8 bit, atau 1 byte. Rentang warna pada black and white sangat cocok digunakan untuk pengolahan file gambar. Salah satu bentuk fungsinya digunakan dalam kedokteran (X-ray). Black and white sebenarnya merupakan hasil rata-rata dari color image, dengan demikian maka persamaannya dapat dituliskan sebagai berikut:

$$I_{BW}(x, y) = \frac{I_R(x, y) + I_G(x, y) + I_B(x, y)}{3} \quad (2.14)$$

Dimana $I_R(x, y)$ = nilai piksel *Red* titik (x,y), $I_G(x, y)$ = nilai piksel *Green* titik (x,y), $I_B(x, y)$ = nilai piksel *Blue* titik (x,y) sedangkan $I_{BW}(x, y)$ = nilai piksel *black and white* titik (x,y).



Gambar 2.4 Black and White (Grayscale)

3. Binary Image.

Setiap piksel hanya terdiri dari warna hitam atau putih, karena hanya ada dua warna untuk setiap piksel, maka hanya perlu 1 bit per piksel (0 dan 1) atau apabila dalam 8 bit (0 dan 255), sehingga sangat efisien dalam hal penyimpanan. Gambar yang direpresentasikan dengan biner sangat cocok untuk teks (dicetak atau tulisan tangan), sidik jari (finger print), atau gambar arsitektur. Binary image merupakan

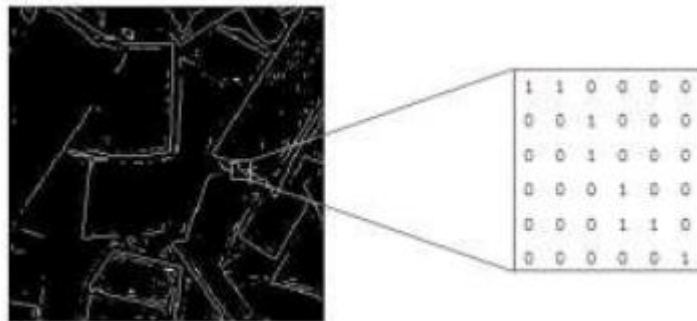
hasil pengolahan dari black and white image, dengan menggunakan fungsi sebagai berikut:

$$I_{Bin}(x, y) = \begin{cases} 0 & I_{BW}(x, y) < T \\ 255 & I_{BW}(x, y) \geq T \end{cases} \quad (2.15)$$

Dan dalam bentuk *floating point*

$$I_{Bin}(x, y) = \begin{cases} 0 & I_{BW}(x, y) < T \\ 1 & I_{BW}(x, y) \geq T \end{cases} \quad (2.16)$$

Dimana $I_B(x, y)$ = nilai piksel *Gray* titik (x,y), $I_B(x, y)$ = nilai piksel Binary titik (x,y), sedangkan T adalah nilai *threshold* [3][11].



Gambar 2.5 Binary Image

2.2.4 Android

Android adalah operating sistem (OS) berlisensi platform terbuka berbasis linux yang diperuntukan untuk mobile device seperti *smartphone* maupun tablet PC. Android OpenSource Project (AOSP) yang dibuat oleh Google. Beberapa produsen mobile menanamkan untuk menjadikan android sebagai sistem operasi. Berbagai keunggulan sistem operasi android mampu memenuhi kebutuhan informasi manusia secara cepat dan menarik. Android disematkan berbagai fitur sebagai fungsi utama [5].

2.3 Teori Program

2.3.1 Android Studio

Android studio adalah IDE (Integrated Development Environment) resmi untuk pengembangan aplikasi Android dan bersifat open source atau gratis. Peluncuran Android Studio ini diumumkan oleh Google pada 16 mei 2013 pada

event Google I/O Conference untuk tahun 2013. Sejak saat itu, Android Studio menggantikan Eclipse sebagai IDE resmi untuk mengembangkan aplikasi Android. Android studio sendiri dikembangkan berdasarkan IntelliJ IDEA yang mirip dengan Eclipse disertai dengan ADT plugin (Android Development Tools). Android studio memiliki fitur [7]:

1. Sistem Build berbasis Gradle yang fleksibel.
2. Emulator yang cepat dan kaya fitur.
3. Lingkungan terpadu sebagai tempat pengembangan aplikasi untuk semua perangkat android.
4. Menerapkan perubahan untuk melakukan push pada perubahan kode dan *resource* ke aplikasi yang sedang berjalan tanpa memulai ulang aplikasi.
5. Template kode dan integrasi Github untuk membantu membuat fitur aplikasi umum dan mengimport kode sampel.
6. Framework dan fitur pengujian yang lengkap.
7. Fitur link untuk merekam performa, kegunaan, kompatibilitas versi, dan masalah lainnya.
8. Mendukung C++ dan NDK.
9. Mendukung bawaan untuk Google Cloud Platform, yang memudahkan integrasi Google Cloud Messaging dan App Engine.
10. Halaman ini menyediakan fitur-fitur dasar Android Studio. Untuk ringkasan perubahan terbaru.

2.3.2 TensorFlow

TensorFlow adalah platform open source end-to-end untuk machine learning. Ini memiliki ekosistem alat, pustaka, dan sumber daya komunitas yang komprehensif dan fleksibel yang memungkinkan peneliti mendorong ML mutakhir dan pengembang dengan mudah membangun dan menerapkan aplikasi yang didukung ML.

2.3.3 Google Colab

Colaboratory, disingkat "Colab", memungkinkan Anda menulis dan mengeksekusi Python di browser Anda dengan beberapa keuntungan berikut:

1. Tidak memerlukan konfigurasi
2. Akses gratis ke GPU
3. Berbagi dengan mudah

2.3.4 Teachablemachine

Teachablemachine adalah web-based tool dalam membuat produk untuk membuat ML model dengan mudah dan cepat yang dibuat dengan menggunakan TensorFlow.js selain untuk Image Classification, teachablemachine berguna untuk Audio dan juga Pose.

1. **Tensorflow.js**: Anda bisa menggunakan Machine Learning di website dengan p5.js (mungkin akan ada blog tersendiri)
2. **Tensorflow**: Anda bisa mendownload model conversion type anda dengan **Keras**
3. **Tensorflow.life**: Cocok untuk project-project IoT.