

BAB II

TINJAUAN PUSTAKA DAN LANDASAN TEORI

2.1 TINJAUAN PUSTAKA

GLCM (*Gray Level Co-occurrence Matrix*) adalah salah satu teknik dalam pengolahan citra yang digunakan untuk menganalisis tekstur suatu citra. Matriks ini memberikan informasi tentang seberapa sering pasangan nilai piksel tertentu terjadi dalam suatu citra pada jarak dan arah tertentu, GLCM bekerja dengan menganalisis distribusi pasangan piksel dengan nilai intensitas tertentu yang berdekatan satu sama lain dalam arah tertentu (misalnya, 0° , 45° , 90° , dan 135°). Dari GLCM ini, berbagai fitur tekstur dapat dihitung, seperti *contrast*, *dissimilarity*, *homogeneity*, *energy*, *correlation*, dan ASM (*Angular Second Moment*), dengan kata lain, GLCM membantu kita memahami pola spasial dari intensitas piksel dalam sebuah citra.

Penelitian dengan klasifikasi telur utuh dan telur retak dengan metode GLCM digunakan untuk menganalisis perbedaan tekstur pada cangkang telur. Retakan pada cangkang telur menyebabkan perubahan pola tekstur yang dapat dideteksi dan diukur menggunakan fitur-fitur yang dihasilkan oleh GLCM. Ekstraksi fitur citra telur diolah menggunakan GLCM untuk menghasilkan fitur tekstur yang mencerminkan kondisi cangkang, apakah cangkang tersebut retak atau utuh. Fitur-fitur seperti *contrast*, *homogeneity*, dan *correlation* akan memiliki nilai yang berbeda pada telur retak dibandingkan dengan telur utuh. Setelah fitur-fitur tersebut diekstraksi, mereka dapat digunakan sebagai input untuk algoritma klasifikasi, seperti *Support Vector Machine* (SVM) atau *Convolutional Neural Network* (CNN). Algoritma ini kemudian dilatih untuk mengenali pola yang berbeda antara telur utuh dan retak berdasarkan fitur tekstur yang dihasilkan oleh GLCM.

Keunggulan penggunaan GLCM dalam klasifikasi telur retak digunakan untuk mendeteksi perbedaan tekstur yang mungkin tidak terlihat oleh mata manusia, terutama pada gambar dengan kualitas atau resolusi yang bervariasi. Ini membuat metode ini sangat efektif dalam aplikasi industri di mana inspeksi manual mungkin tidak praktis dan akurat. Sedangkan *Convolutional Neural Network* (CNN) adalah

salah satu arsitektur *deep learning* yang dirancang khusus untuk menangani data berbentuk grid, seperti citra (gambar). CNN sangat efektif untuk tugas-tugas seperti klasifikasi citra, deteksi objek, dan segmentasi citra karena mampu secara otomatis mengekstraksi fitur-fitur yang relevan dari data mentah tanpa memerlukan ekstraksi fitur manual yang kompleks. Lapisan konvolusi pada CNN adalah proses di mana filter (juga dikenal sebagai kernel) diterapkan pada citra untuk mendeteksi fitur-fitur tertentu, seperti tepi, sudut, atau tekstur. Filter ini secara sistematis melintasi seluruh citra, menghasilkan peta fitur (*feature map*) yang menyoroti keberadaan fitur yang terdeteksi di berbagai lokasi.

Setelah proses konvolusi, peta fitur seringkali dikurangi ukurannya melalui proses *pooling* (umumnya *max pooling*), yang mempertahankan fitur yang paling penting sekaligus mengurangi resolusi citra. *Pooling* membantu mengurangi jumlah parameter dan komputasi dalam jaringan, serta membuat model lebih tahan terhadap pergeseran kecil dalam citra. Pada tahap akhir, peta fitur yang dihasilkan oleh lapisan konvolusi dan *pooling* diratakan (*flattening*) dan dihubungkan ke satu atau lebih *lapisan fully connected*, di mana setiap neuron terhubung ke semua neuron di lapisan sebelumnya. Lapisan ini bertanggung jawab untuk melakukan klasifikasi akhir berdasarkan fitur yang telah diekstraksi. Aktivasi seperti ReLU (*Rectified Linear Unit*) sering digunakan setelah lapisan konvolusi untuk memperkenalkan non-linearitas, sehingga model dapat belajar fungsi yang lebih kompleks. Di lapisan terakhir, fungsi *softmax* digunakan untuk mengonversi output menjadi probabilitas kelas.

Dalam penelitian klasifikasi telur utuh dan telur retak, CNN digunakan untuk secara otomatis mengenali pola visual yang membedakan antara telur yang utuh dan telur yang retak. CNN sangat kuat dalam menangani variasi visual dalam data, seperti perubahan pencahayaan, orientasi, dan deformasi kecil dalam objek. Hal ini sangat berguna dalam klasifikasi telur karena kondisi pencahayaan dan sudut pengambilan gambar bisa sangat bervariasi. CNN secara otomatis mengekstraksi fitur-fitur visual dari citra telur, yang kemudian digunakan untuk membuat prediksi apakah telur tersebut utuh atau retak. Dengan arsitektur yang dirancang khusus untuk menangani data berbentuk grid, CNN mampu mengenali pola-pola yang

kompleks dan sering kali tidak terlihat oleh manusia, menjadikannya pilihan yang sangat baik untuk aplikasi di industri peternakan, termasuk kualitas telur.

Penelitian yang berkaitan dengan deteksi retakan kulit telur terutama berfokus pada analisis *frekuensi Fourier* dari respons berbasis getaran. Penelitian ini mengusulkan metodologi deteksi baru untuk retakan kulit telur menggunakan *transformasi wavelet* dan teknik *support vector machine* (SVM). Metodologi yang diusulkan terdiri dari sistem eksperimental dan sistem pemrosesan data. Sistem eksperimental digunakan untuk menghasilkan gaya tumbukan dan mengukur sinyal akustik, sedangkan sistem pemrosesan data mengekstrak ciri sinyal tepat waktu dan mengklasifikasikan sinyal berdasarkan algoritma SVM yang digunakan. Transformasi *wavelet* diperkenalkan dengan tujuan untuk mengatasi kelemahan dari analisis konvensional dan untuk menemukan ciri sensitif retakan baru berdasarkan informasi waktu dan frekuensi lokal dari sinyal.

Beberapa ciri berbasis wavelet diekstraksi melalui analisis distribusi energi koefisien transformasi *wavelet* untuk telur retak dan utuh. Ciri ini termasuk *first resonance scale* (FS), *s-coordinate* (SC) dan *corrected t-coordinate* (TC) dari *centroid* untuk domain skala waktu, dan *weighted standard deviation* (WD). Ciri yang digunakan beserta beberapa kombinasinya, digunakan untuk membangun pengklasifikasi SVM. Efek klasifikasi dari pengklasifikasi SVM dilakukan melalui tiga ciri yang terbaik dan disediakan pengklasifikasi SVM ciri yang ke empat. Skema deteksi berdasarkan empat pengukuran untuk setiap telur, contoh eksperimental mencapai tingkat deteksi retak tertinggi 98,9%, dan tingkat kesalahan terkecil 0,8% (Deng dkk., 2010).

Deteksi retakan yang dilakukan dalam penelitian pada kulit telur sulit dikenali karena bintik-bintik gelap di sekitarnya yang disorot pada permukaan telur di bawah penerangan lampu latar kemudian dilakukan metode baru untuk mengidentifikasi retakan berdasarkan *machine vision*. Setelah menganalisis karakteristik retakan pada citra telur di bawah penerangan lampu latar, operator LOG negatif (*Laplacian of Gaussian*) digunakan untuk secara efektif meningkatkan retakan pada citra telur. Kemudian algoritma ambang batas *Hysteresis* diadopsi untuk memperoleh ambang batas yang tepat, yang menghilangkan bintik-bintik gelap yang tidak relevan dalam

citra telur biner dan memastikan retakan. Indeks LFI (*Local Fitting Image*) yang ditingkatkan digunakan untuk membedakan wilayah retakan dari daerah yang salah (Guanjun dkk., 2019).

Metode dalam penelitian untuk mendeteksi cacat pada permukaan melengkung berfokus pada cacat telur unggas untuk identifikasi kerusakan pada cangkang akibat telur bercangkang tipis. Sistem penglihatan didasarkan pada deteksi cacat dengan memindai pola laser dari cahaya terstruktur dan pencitraan, menyoroti perubahan geometri sebagai akibat deformasi transisi laser yang dihasilkan dengan memindai permukaan telur. Kemudian, citra dianalisis untuk mendapatkan titik-titik yang berjarak sama di sepanjang kurva dan dievaluasi dengan membuat interpolasi kubik. Interpolasi memungkinkan ekstraksi karakteristik metrik deskriptif untuk mengamati perbedaan antara kurva, menggambarkan cacat dengan melakukan interposisi grafik. Informasi metrik yang diperoleh digunakan untuk mengklasifikasikan sampel yang rusak dengan mengembangkan algoritma menggunakan jaringan saraf tiruan, dilatih dengan database yang terdiri dari 200 citra, yang mencapai efisiensi 97,5% selama evaluasi 150 sampel telur. Teknik ini dapat diterapkan untuk mendeteksi telur yang bergelombang, berkerut, berjerawat, berbentuk aneh, dan tidak berbentuk (Coutiño., 2019).

Studi ini mengusulkan pendekatan baru untuk secara otomatis mendeteksi retakan kulit telur ini menggunakan jaringan saraf konvolusional (CNN). Model CNN dilatih pada tambalan citra yang diekstraksi dari citra telur dengan ukuran *batch* yang berbeda. Kesimpulan setelah pengujian menunjukkan bahwa model yang dilatih pada ukuran *batch* 64 memberikan hasil terbaik. Model ini juga mengungguli *support vector machine* (SVM), *classifier* dilatih pada ciri *histogram of oriented gradients* (HOG) dan *local binary pattern* (LBP) dengan *presisi*, *recall*, *akurasi*, *false positive rate* (FPR) dan *area under the curve* (AUC) sebesar 98,28%, 92,41%, 96,92%, 0,81% dan 96,53%, masing-masing dalam mengklasifikasikan patch citra. Selain itu, model CNN yang diusulkan melampaui model SVM dalam mengklasifikasikan citra telur dengan akurasi 95,38% (Botta dkk., 2022).

Penghitungan telur secara akurat adalah permintaan dasar dalam penetasan dari budidaya permasalahan penghitungan manual akan sangat memakan waktu dan

menimbulkan kesalahan dan sekarang masih diadopsi di sebagian besar peternakan. Banyak metode penghitungan telur otomatis tradisional tidak dapat memberikan akurasi yang cukup tinggi, dalam penelitian sebelumnya metode baru berbasis *convolutional neural network* (CNN) untuk penghitungan telur udang, dibandingkan dengan metode tradisional yang terutama didasarkan pada deteksi kontur atau segmentasi citra, metode yang diusulkan memanfaatkan regresi peta kepadatan dan lebih efisien dalam kasus terdistribusi padat.

Penggunaan metode pembuatan kumpulan data sintetis berdasarkan *Generative adversarial networks* (GAN) diusulkan untuk menghindari pelabelan manual yang memberatkan, dan memiliki aplikasi potensial dalam pembelajaran terawasi untuk penghitungan telur atau larva yang didistribusikan. Kemudian, dengan mempertimbangkan keuntungan dari tidak ada prapengolahan manual dan ekstraksi ciri berbutir halus di CNN, jaringan penghitungan telur udang (SECNet- *shrimp egg counting network*) berdasarkan *full convolutional regression network* (FCRN) diusulkan untuk mewujudkan penghitungan melalui regresi citra input ke dalam peta kepadatannya. Hasil pengujian menunjukkan akurasi penghitungan rata-rata dari SECNet yang diusulkan dapat mencapai 96,2% ketika SECNet dilatih sebelumnya pada kumpulan data sintetis dan disempurnakan pada kumpulan data yang dikumpulkan (J. Zhang dkk., 2021).

Penelitian lain yang menggunakan sistem real-time berdasarkan sistem berputar terus menerus untuk mendeteksi telur yang rusak Kombinasi yang digunakan adalah CNN dan BiLSTM yang telah dilatih sebelumnya diusulkan untuk klasifikasi telur. Model yang diusulkan memastikan akurasi 96,17%. Model yang diusulkan memperoleh kinerja yang unggul dibandingkan dengan metode-metode sebelumnya. Penelitian tersebut menggunakan sistem yang berputar terus menerus untuk memvisualisasikan semua permukaan telur. Dengan demikian, kondisi retakan yang mungkin terjadi pada setiap permukaan telur telah berhasil dipantau. Sistem yang diusulkan untuk mendeteksi telur yang kuat, ciri diekstraksi menggunakan model jaringan residual yang telah dilatih sebelumnya dan kemudian ciri yang diperoleh dimasukkan ke dalam memori jangka pendek dua arah (BiLSTM) (Turkoglu., 2021). Ekspansi metode pembelajaran mendalam baru-baru

ini telah menemukan penerapannya dalam deteksi penyakit tanaman, menawarkan metode yang baik dengan hasil yang sangat akurat. Penelitian ini menyajikan tinjauan yang bertujuan untuk mengidentifikasi penggunaan *Convolutional neural network* (CNN) dalam proses identifikasi dan klasifikasi (Abade dkk., 2021).

Berdasarkan penelitian sebelumnya kebaruan dari penelitian ini belum ada klasifikasi kualitas telur ayam petelur dengan parameter keretakan dan telur utuh menggunakan Metode *Gray Level Co-Occurrence Matrix - Convolutional Neural Network* sebagai komputasi berdasarkan jarak, sudut dan pencahayaan yang berbeda. Disertasi penelitian ini menghasilkan klasifikasi telur yang berkualitas baik dan yang buruk yang memudahkan peternak dalam memilih kualitas telur yang baik.

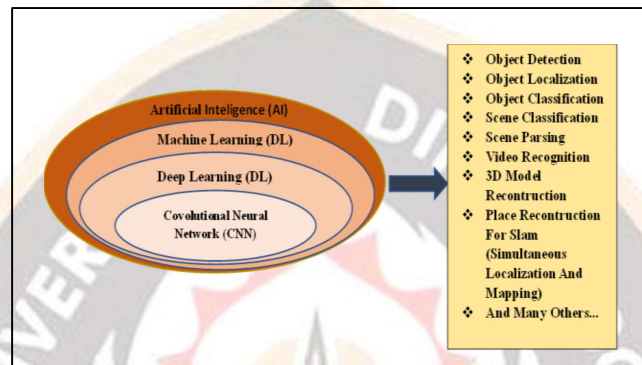
2.2 LANDASANTEORI

2.2.1 PEMBELAJARAN MENDALAM (*DEEP LEARNING*)

Pembelajaran mendalam adalah salah satu subkategori pembelajaran mesin yang banyak digunakan dalam berbagai domain kecerdasan buatan, dengan cara ini digunakan arsitektur hierarkis, abstraksi dan ciri tingkat tinggi dari data yang dipelajari. Pembelajaran mendalam adalah teknik berbasis jaringan saraf yang memiliki ciri unik untuk mengekstraksi ciri secara otomatis dan memilih yang terbaik (Nasiri dkk., 2020). Jaringan saraf konvolusional adalah kelompok algoritma pembelajaran mendalam yang paling terkenal dan umum digunakan karena penerapannya yang kuat dalam mengenali berbagai pola, terutama dalam menemukan objek dalam foto digital. Hubungan antara kecerdasan buatan, pembelajaran mesin, pembelajaran mendalam, dan jaringan saraf konvolusional digambarkan pada Gambar 2.1 (Adi dkk., 2022).

Pembelajaran mendalam memungkinkan model komputasi yang terdiri dari beberapa lapisan pemrosesan untuk mempelajari representasi data dengan berbagai tingkat abstraksi. Metode-metode ini telah meningkatkan *state-of-the-art* dalam pengenalan suara, pengenalan objek visual, deteksi objek dan banyak domain lainnya seperti penemuan obat dan genomik. Pembelajaran mendalam menemukan struktur rumit dalam kumpulan data besar dengan menggunakan algoritma

backpropagation untuk menunjukkan bagaimana mesin harus mengubah parameter internalnya yang digunakan untuk menghitung representasi di setiap lapisan dari representasi di lapisan sebelumnya. *Deep convolutional* telah menghasilkan terobosan dalam pemrosesan citra, video, ucapan, dan audio (LeCun dkk., 2015).



Gambar 2.1 Pembelajaran mendalam (Adi dkk., 2022)

2.2.2 JARINGAN SARAF KONVOLUSIONAL (*CONVOLUTIONAL NEURAL NETWORK - CNN*)

CNN adalah alat ekstraksi ciri yang menjanjikan yang secara bertahap menggantikan algoritma pembelajaran mesin tradisional. CNN tidak hanya dapat mengekstrak ciri yang paling kuat dan efektif tetapi juga memiliki kemampuan generalisasi yang kuat, yang tidak layak dengan metode pembelajaran mesin tradisional (Liu dkk., 2021). Beberapa tahun terakhir, para peneliti telah menerapkan CNN ke bidang lain, seperti pengenalan suara, pengenalan wajah, pengenalan objek, pemrosesan bahasa alami, analisis gelombang otak, dan sebagainya. Bidang-bidang ini berlanjut ke berbagai arah dan beberapa terobosan telah dilakukan. Dibandingkan dengan jaringan saraf biasa, CNN mengandung ekstraktor ciri, yang terdiri dari lapisan konvolusi dan lapisan *down-sampling*. Neuron hanya terhubung ke bagian neuron di lapisan atas, dan bagian neuron ini disebut bidang reseptif lokal. Lapisan konvolusional umumnya memiliki beberapa peta ciri, setiap peta ciri terdiri dari sejumlah neuron tertentu, dan bobot neuron dibagi antara peta ciri yang sama. Ciri terbesar dari pembagian bobot adalah untuk

mengurangi hubungan antara berbagai lapisan jaringan, mengurangi parameter jaringan, dan pada saat yang sama memainkan peran dalam mencegah *overfitting*. Secara umum, nilai awal kernel konvolusi dihasilkan secara acak di dalam proses pelatihan jaringan, bobot baru terus dipelajari dan diperbarui secara terus menerus sampai bobot yang wajar akhirnya dipelajari. *Down sampling*, juga disebut *pooling*, adalah proses konvolusi khusus. Oleh karena itu, CNN memiliki tiga ciri utama, yaitu bidang reseptif lokal, pembagian bobot dan *pooling*.

a. Bidang reseptif lokal

Setiap neuron pada CNN, dari lapisan tersembunyi terhubung ke area kecil dari lapisan input, dan setiap koneksi memiliki bobot parameter yang dapat dipelajari. Daerah ini disebut bidang reseptif lokal dari *neuron* lapisan tersembunyi.

b. Pembagian bobot

Bidang reseptif lokal dengan 25 piksel, ada bobot 5×5 untuk setiap neuron di lapisan tersembunyi. Pembagian bobot adalah bahwa bobot yang sesuai dengan neuron ini di lapisan tersembunyi adalah sama. Karena adanya pembagian bobot, jumlah parameter jaringan dan waktu pelatihan sangat berkurang.

c. *Pooling*

Lapisan *pooling* umumnya berada di belakang lapisan konvolusi, tujuannya adalah untuk memperkecil citra setelah konvolusi untuk mengurangi jumlah parameter.

Jaringan saraf konvolusional adalah kombinasi dari algoritma pembelajaran mendalam dan jaringan saraf tiruan, dan banyak digunakan dalam pemrosesan citra. Jaringan saraf konvolusional umumnya terdiri dari tiga bagian: lapisan input, lapisan tersembunyi dan lapisan output. Di antara lapisan tersebut, lapisan input adalah citra asli yang belum diproses, lapisan output adalah hasil dari mengklasifikasikan ciri, dan lapisan tersembunyi adalah lapisan neuron dengan struktur nonlinier *multi-layer* yang rumit, termasuk lapisan konvolusi dan lapisan sub-sampling. Jaringan saraf konvolusional mengekstraksi dan mengklasifikasikan ciri dalam lapisan tersembunyi. Optimalisasi lapisan konvolusional dan perceptron

lapisan tunggal dapat meningkatkan akurasi ekstraksi ciri dan mengoptimalkan efek klasifikasi.

Lapisan tersembunyi penting lainnya dari CNN adalah lapisan aktivasi, ketika memecahkan masalah yang lebih kompleks, fungsi aktivasi menambahkan faktor non linier ke dalam jaringan saraf sehingga, jaringan saraf dapat beradaptasi dengan lebih banyak masalah yang kompleks, fungsi aktivasi yang umum digunakan dan lebih mudah untuk dilatih dan seringkali mencapai kinerja yang lebih baik yaitu *rectified linear activation function* (ReLU) merupakan fungsi linier yang akan mengeluarkan input secara langsung jika positif dan jika tidak akan menghasilkan nol, persamaan fungsi ReLU didefinisikan dengan persamaan 2.1 :

$$f(x) = \max(0, x) \quad (2.1)$$

Keterangan :

F (x) = fungsi x

Lapisan yang sepenuhnya terhubung adalah beberapa lapisan terakhir di CNN dan bertindak sebagai pengklasifikasi di seluruh jaringan. Jika lapisan sebelumnya dari lapisan yang sepenuhnya terhubung juga merupakan lapisan yang terhubung sepenuhnya, maka 1×1 konvolusi digunakan. Jika lapisan sebelumnya dari lapisan sepenuhnya terhubung adalah lapisan konvolusional, konvolusi global $h \times w$ digunakan, dan variabel h dan variabel w adalah tinggi dan lebar dari hasil konvolusi lapisan sebelumnya. (Tian, 2020). Dari Gambar 2.2 tiga lapisan utama yang terdiri dari CNN adalah konvolusi, dan lapisan yang terhubung sepenuhnya dan keluaran berupa klasifikasi. *Convolutional Neural Network* (CNN) adalah jenis arsitektur jaringan saraf tiruan yang khususnya efektif dalam memproses data berstruktur grid, seperti gambar. CNN dikenal karena kemampuannya dalam menangkap pola spasial dalam data gambar menggunakan operasi konvolusi, beberapa poin penting tentang *Convolutional Neural Network*:

1. **Input Layer** : *Input Layer* adalah komponen awal yang penting dalam CNN.

Input Layer bertugas dalam menyiapkan data input untuk diproses oleh lapisan selanjutnya, sehingga memungkinkan CNN untuk mulai belajar dan mengenali pola dalam data visual. *Input layer* menerima data input dalam

bentuk citra atau data visual lainnya. Setiap input biasanya direpresentasikan sebagai matriks piksel, di mana setiap elemen matriks mewakili nilai intensitas piksel dari citra tersebut. Dimensi input dari *input layer* ditentukan oleh dimensi citra atau data visual yang diberikan. Misalnya, jika input adalah citra berwarna (RGB) dengan resolusi 128x128 piksel, maka dimensi inputnya akan menjadi (128, 128, 3), di mana 3 adalah kedalaman atau jumlah saluran warna (merah, hijau, biru).

Sebelum diteruskan ke layer berikutnya dalam CNN, data input akan mengalami proses *preprocessing*. *Preprocessing* ini bisa mencakup normalisasi nilai piksel, *resizing* citra ke ukuran yang diinginkan, atau *augmentasi* data untuk meningkatkan keanekaragaman dataset. *Input Layer* dalam CNN dapat menerima berbagai jenis data visual, termasuk citra berwarna (RGB), citra *grayscale*, atau bahkan data visual yang lebih kompleks seperti citra medis atau data dari sensor penginderaan. Layer ini dirancang untuk menangani berbagai jenis data visual dan mengekstrak fitur-fitur penting dari input tersebut. *Input Layer* dalam CNN tidak memiliki neuron atau bobot yang harus dipelajari. Fungsinya adalah hanya sebagai penerima data input dan meneruskannya ke layer berikutnya dalam arsitektur CNN.

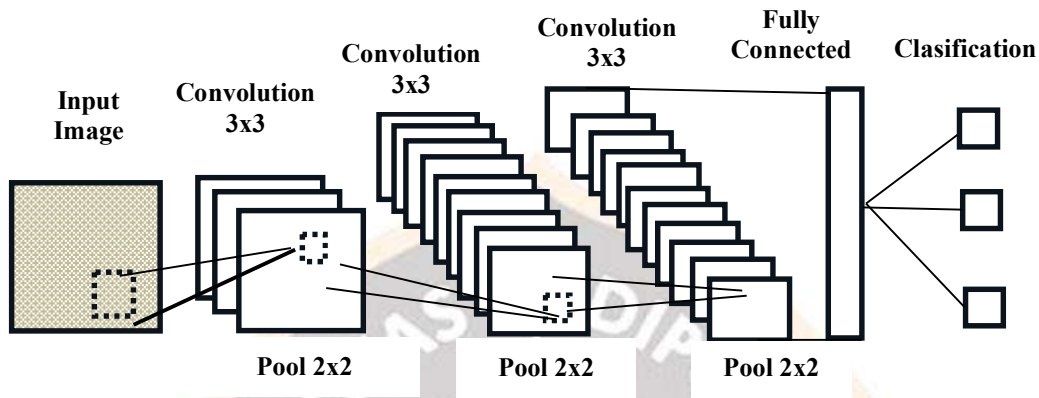
2. **Layer Konvolusi:** *Layer* konvolusi adalah komponen utama dalam CNN. Setiap *layer* konvolusi memiliki sejumlah filter konvolusi yang digunakan untuk mengekstraksi ciri-ciri penting dari input citra. Proses konvolusi melibatkan menerapkan filter ke seluruh citra untuk menghasilkan peta fitur yang menyoroti pola-pola tertentu dalam citra. Lapisan konvolusi (*convolutional layer*) ini bekerja dengan cara menggeser filter (*kernel*) kecil yang berisi bobot di atas citra untuk mendeteksi pola dan karakteristik spasial.
3. **Layer Pooling:** *Layer pooling* digunakan untuk mengurangi dimensi spasial dari peta fitur yang dihasilkan oleh layer konvolusi. *Pooling* biasanya dilakukan dengan operasi seperti *max pooling* atau *average pooling*, yang

membantu mengurangi ukuran data tanpa menghilangkan informasi penting. *Layer pooling* dalam Jaringan Saraf Konvolusi (CNN) juga dikenal sebagai *layer downsampling*, merupakan komponen penting dalam arsitektur Jaringan Saraf Konvolusi (CNN). Fungsinya adalah untuk mengurangi dimensi spasial dari peta fitur yang dihasilkan oleh lapisan konvolusi sebelumnya. Tujuan dari *layer pooling* adalah untuk mengurangi *overfitting*, mempercepat pelatihan, dan mengurangi jumlah parameter dalam model CNN.

Max Pooling yang memiliki ukuran 2x2 atau 3x3 pada lapisan input dibagi menjadi kelompok yang tidak tumpang tindih. Nilai maksimum diambil dari setiap kelompok, dan nilai-nilai maksimum tersebut digunakan sebagai representasi untuk wilayah tersebut dalam lapisan output. Operasi ini membantu dalam ekstraksi ciri-ciri yang paling dominan dari setiap wilayah, sehingga membantu dalam mempertahankan informasi penting dari citra asli. Dengan mengurangi dimensi spasial dan jumlah parameter, *layer pooling* memungkinkan model CNN untuk belajar fitur-fitur yang lebih abstrak dan kompleks dari gambar, serta meningkatkan kemampuan generalisasinya. Hal ini membuat *layer pooling* menjadi salah satu komponen kunci dalam arsitektur CNN yang efektif untuk tugas-tugas pengenalan gambar dan visi komputer lainnya.

4. **Layer Aktivasi:** Setelah setiap operasi konvolusi atau pooling, fungsi aktivasi diterapkan untuk memperkenalkan non-linearitas ke dalam model. Fungsi aktivasi yang umum digunakan adalah ReLU (*Rectified Linear Unit*), yang secara efektif meningkatkan kemampuan model untuk mempelajari pola-pola yang kompleks.

5. **Layer Fully Connected:** Setelah beberapa layer konvolusi dan pooling, outputnya kemudian disampaikan ke dalam satu atau beberapa *layer fully connected* (*dense layer*). Layer ini bertanggung jawab untuk mengubah fitur-fitur yang telah diekstraksi menjadi prediksi akhir atau representasi kelas dari input.



Gambar 2.2 Arsitektur model CNN (Botta dkk., 2022)

Di dalam dunia *Convolutional Neural Network* (CNN), terdapat berbagai library yang berperan penting dalam membangun, melatih, dan menjalankan model CNN.

1. **TensorFlow:** TensorFlow adalah salah satu library *deep learning* yang paling populer yang dikembangkan oleh Google. Ini menyediakan infrastruktur yang kuat untuk mengembangkan berbagai jenis model *deep learning*, termasuk CNN. *TensorFlow* juga memiliki TensorFlow Keras sebagai antarmuka tingkat tinggi yang mempermudah pembuatan dan pelatihan model CNN.
2. **Keras:** Keras adalah salah satu library *deep learning* tingkat tinggi yang berjalan di atas *TensorFlow*, *Theano*, atau *Microsoft Cognitive Toolkit (CNTK)*. Keras menyediakan antarmuka yang sederhana dan intuitif untuk membangun model *neural network*, termasuk CNN.
3. **OpenCV:** *OpenCV* adalah library pengolahan citra *open source* yang memiliki berbagai fungsi untuk memanipulasi dan menganalisis gambar. Ini sering digunakan dalam pengolahan pra-pemrosesan gambar sebelum input ke model CNN, serta untuk tugas-tugas deteksi objek atau segmentasi.
4. **NumPy:** *NumPy* adalah library *Python* yang mendasar untuk komputasi numerik. Ini menyediakan struktur data array multidimensi yang kuat yang sangat berguna untuk memanipulasi data input dan output dalam model CNN.

5. **Matplotlib:** *Matplotlib* adalah library visualisasi data Python yang sering digunakan untuk membuat grafik dan plot. Ini berguna untuk memvisualisasikan data, termasuk gambar, sehingga memungkinkan pengguna untuk memahami dan menganalisis hasil dari model CNN.
6. **Scikit-learn:** *Scikit-learn* juga dikenal sebagai *sklearn*, adalah salah satu *library Python* yang paling populer untuk *machine learning*. *Library* ini menyediakan beragam algoritma *machine learning* dan fungsi utilitas untuk tugas-tugas umum seperti klasifikasi, regresi, klastering, pemrosesan data, dan evaluasi model.
7. **Seaborn :** *Seaborn* adalah *library Python* yang digunakan untuk membuat visualisasi data statistik yang menarik dan informatif. Dibangun di atas *matplotlib*, *Seaborn* menyediakan antarmuka yang lebih tinggi untuk membuat grafik statistik yang estetik dan siap pakai.

2.2.3 GRAY LEVEL CO-OCCURRENCE MATRIX (GLCM)

Metode ekstraksi ciri GLCM adalah matriks yang menggambarkan frekuensi terjadinya dua piksel dengan intensitas tertentu pada jarak d dan orientasi sudut θ dalam citra. Ekstraksi ciri GLCM dilakukan dalam 4 arah sudut, yang masing-masing dengan interval 45° yaitu 0° , 45° , 90° , 135° . Ekstraksi ciri yang menggunakan analisis tekstur dilakukan dengan mengambil karakteristik skala abu-abu dari suatu objek yang membedakannya dari objek lain. Karakteristik yang diekstraksi ini termasuk kontras, korelasi, energi, dan homogenitas.

1. Kontras

Ciri kontras digunakan untuk menghitung tingkat perbedaan keabuan dalam suatu citra. Semakin besar perbedaan keabu-abuan, semakin tinggi kontrasnya. Sebaliknya, semakin tidak signifikan perbedaan keabuan antara dua piksel, semakin rendah kontrasnya, kontras didefinisikan dengan persamaan 2.2 :

$$\text{Kontras} = \sum_i \sum_j (i - j)^2 p(i, j) \quad (2.2)$$

Keterangan :

$p(i, j)$ = Probabilitas kemunculan pasangan nilai intensitas piksel (i, j) .

2. Korelasi

Korelasi menunjukkan betapa berkorelasinya piksel referensi ke tetangganya atas citra, korelasi didefinisikan dengan persamaan 2.3:

$$Korelasi = \sum_i \sum_j \frac{ij P_d(i,j) - \mu_x \mu_y}{\sigma_x \sigma_y} \quad (2.3)$$

Keterangan :

μ_x, μ_y = Rata-rata intensitas piksel di baris dan kolom pada GLCM.

σ_x, σ_y = Simpangan baku intensitas piksel di baris (*row*) dan kolom (*column*) pada GLCM.

3. Energi

Nilai energi menggambarkan tingkat distribusi keabuan dalam suatu citra, energi ditulis dengan persamaan 2.4:

$$Energi = \sum_i \sum_j P^2(i,j) \quad (2.4)$$

Keterangan :

$p(i, j)$ = Probabilitas kemunculan pasangan nilai intensitas piksel (i, j).

4. Homogenitas

Ciri homogenitas menghitung tingkat homogenitas kealahbanan dalam suatu citra. Nilai homogenitas lebih tinggi dalam citra dengan tingkat keabu-abuan yang hampir sama. Homogenitas dengan persamaan 2.5:

$$Homogeneity = \sum_i \sum_j \frac{P(i,j)^2}{1 + |i - j|} \quad (2.5)$$

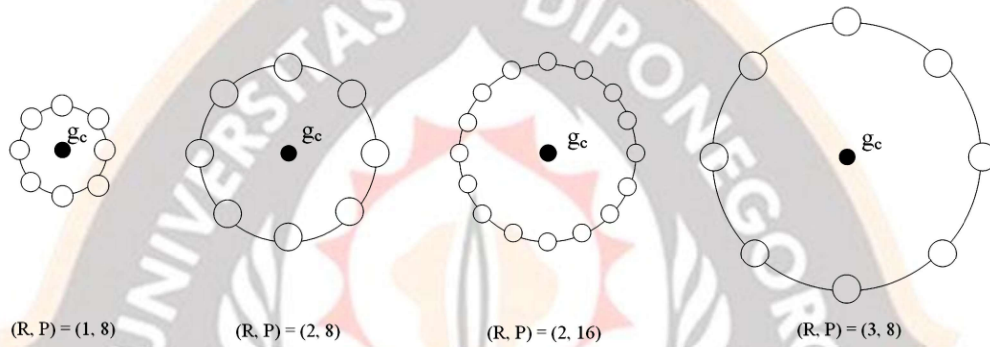
Keterangan :

$p(i, j)$ = Probabilitas kemunculan pasangan nilai intensitas piksel (i, j).

2.2.4 LOCAL BINARY PATTERN (LBP)

Local Binary Pattern (LBP) adalah metode deskripsi tekstur yang banyak digunakan dalam pengolahan gambar dan pengenalan pola. LBP bekerja dengan membandingkan nilai piksel di sekitar area lokal dan menghasilkan pola biner yang mewakili tekstur di area tersebut. LBP diusulkan pada tahun 1996 (Ojala dkk., 1996) dan ditingkatkan pada tahun 2002 (Ojala dkk., 2002). Sejak itu, karena

kesederhanaan dan efisiensinya, LBP telah banyak digunakan dalam klasifikasi tekstur dan pengenalan wajah, dan telah menjadi metode ekstraksi ciri mendasar. Seperti yang ditunjukkan pada Gambar. 1, LBP mengekstrak struktur tekstur dari skala yang berbeda R menggunakan template lingkungan melingkar. Template lingkungan melingkar didefinisikan oleh skala sampling R ($R > 0$) dengan P ($P > 0$) piksel lingkungan sampel g_p ($p = 0, 1, \dots, P - 1$) didistribusikan secara merata di sekitar keliling lingkaran.



Gambar 2.3. Template lingkungan melingkar dengan skala pengambilan sampel yang berbeda R dan jumlah piksel lingkungan sampel P

Umumnya, untuk mengekstrak fitur tekstur skala besar dengan lebih baik, jumlah piksel lingkungan sampel g_p dan skala pengambilan sampel R harus memenuhi hubungan $P = 8 \times R$. Piksel gambar tekstur dianggap sebagai piksel tengah g_c . Interpolasi bilinear digunakan untuk mendapatkan piksel lingkungan g_p yang tidak benar-benar ada dalam gambar tekstur. LBP diterapkan ke setiap piksel tengah dari gambar tekstur dan mengekstrak informasi fitur lokal dari setiap piksel tengah, yang terkait dengan piksel lingkungan. Dengan membandingkan piksel tengah g_c dan piksel lingkungan sampel yang sesuai g_p ($p = 0, 1, \dots, P - 1$), kita dapat memperoleh vektor perbedaan $d_p = [g_0 - g_c, g_1 - g_c, \dots, g_{P-1} - g_c]$, ($p = 0, 1, \dots, P - 1$). Nilai LBP didefinisikan dengan persamaan 2.6:

$$LBP_{P,R} = \sum_{p=0}^{P-1} 2^p \times S(g_p - g_c), s(x) = \begin{cases} 1, & x \geq 0 \\ 0, & \text{otherwise} \end{cases} \quad (2.6)$$

Keterangan :

P = jumlah piksel

R = skala pengambilan sampel

g_c = piksel tengah

g_p = piksel lingkungan sampel yang sesuai

Untuk menyederhanakan perhitungan dan mengurangi dimensi histogram, Ojala et al. mengklasifikasikan pola LBP menjadi pola seragam dan tidak seragam berdasarkan frekuensi kemunculannya. Parameter metrik seragam U , yang mewakili jumlah bitwise "0/1" dan "1/0" dalam pola biner, diusulkan sebagai kriteria untuk membedakan antara pola seragam dan tidak seragam. Rumus untuk U ditunjukkan dengan persamaan 2.7:

$$U(LBP_{P,R}) = \sum_{p=1}^{p-1} |S(g_p - g_c) - S(g_{p-1} - g_c)| + |S(g_{p-1} - g_c) - S(g_0 - g_c)| \quad (2.7)$$

Keterangan :

g_c = piksel tengah

g_p = piksel lingkungan sampel yang sesuai

Secara efektif, nilai parameter metrik seragam U mencerminkan frekuensi perubahan pola lokal, dengan nilai U rendah sesuai dengan informasi sinyal frekuensi rendah dan nilai U tinggi sesuai dengan informasi sinyal frekuensi tinggi. Karena informasi sinyal frekuensi rendah merupakan mayoritas gambar alami, pola seragam dengan nilai U rendah sesuai dengan struktur yang banyak terdapat pada gambar alami. Untuk mencapai kinerja klasifikasi yang lebih baik, berdasarkan operator LBP asli dan parameter metrik seragam U , Ojala et al. mengusulkan operator LBP seragam invarian rotasi, didefinisikan dengan persamaan 2.8 sebagai berikut :

$$LBP_{P,R} = \sum_{p=0}^{P-1} 2^p \times S(g_p - g_c), s(x) = \begin{cases} 1, & x \geq 0 \\ 0, & \text{otherwise} \end{cases} \quad (2.8)$$

Keterangan :

P = jumlah piksel

R = skala pengambilan sampel

g_c = piksel tengah

g_p = piksel lingkungan sampel yang sesuai

Untuk mencapai kinerja klasifikasi yang lebih baik, berdasarkan operator LBP asli dan parameter metrik seragam U, Ojala et al. mengusulkan operator LBP seragam invarian rotasi, didefinisikan dengan persamaan 2.9 sebagai berikut :

$$LBP_{P,R}^{riu2} = \begin{cases} \frac{\sum_{p=0}^{P-1} S(g_p - g_c)}{P + 1}, & U(LBP_{P,R}) \leq 2 \\ \text{otherwise} \end{cases} \quad (2.9)$$

Keterangan :

P = jumlah piksel

R = skala pengambilan sampel

g_c = piksel tengah

g_p = piksel lingkungan sampel yang sesuai

riu2 = *Rotation-Invariant Uniform* varian LBP yang dibuat untuk mengatasi masalah rotasi.

Prinsip Kerja LBP menggunakan pembagian area dengan langkah sebagai berikut:

1. Pertama, citra dibagi menjadi wilayah-wilayah kecil berukuran 3x3.
2. Perbandingan Piksel: setiap wilayah, nilai piksel pusat dibandingkan dengan nilai piksel tetangganya.
3. Pembentukan Pola Biner: Hasil perbandingan diubah menjadi pola biner, di mana nilai 1 diberikan untuk piksel yang lebih terang daripada piksel pusat, dan nilai 0 untuk piksel yang lebih gelap atau sama.

4. Histogram LBP: Pola biner dari semua wilayah digabungkan untuk menghasilkan histogram LBP, yang mewakili distribusi pola tekstur di seluruh citra.

2.2.5 GABOR FILTER

Filter Gabor adalah filter linier yang digunakan dalam pemrosesan gambar untuk deteksi tepi, klasifikasi tekstur, ekstraksi ciri, dan estimasi disparitas. Hal ini secara optimal terbatas sesuai prinsip ketidakpastian baik dalam frekuensi dan domain khusus yaitu $\Delta x \cdot \Delta \omega$ yang dekat dengan h , metrik ketidakpastian. Ini berarti bahwa Filter Gabor sangat selektif dalam kedua frekuensi dan posisi, sehingga mengakibatkan tajam tekstur deteksi batas. Paradigma Segmentasi terkait dengan Filter Gabor didasarkan pada model filter bank di mana beberapa Filter diterapkan serentak ke gambar input. Filter fokus pada berbagai tertentu frekuensi. Jika gambar masukan berisi dua wilayah tekstur yang berbeda, perbedaan frekuensi lokal antara daerah akan mendeteksi tekstur dalam satu atau lebih filter output sub-gambar. Fungsi Gabor dasar dapat melakukan dekomposisi ruang. Setiap Filter Gabor ditentukan oleh fungsi Gabor dasar. Karena spasial dan spasial frekuensi lokalisasi Filter Gabor secara luas digunakan untuk segmentasi tekstur. Filter Gabor adalah Gaussian yang dimodulasi oleh gelombang bidang Kernel konvolusi dapat dinyatakan dengan persamaan 2.10 sebagai berikut:

$$g(x, y) = \exp\left(-\left(\frac{a}{2\sigma}\right)^2 + \gamma^2\left(\frac{b}{2\sigma}\right)^2\right) * x \cos\left(\frac{2\pi}{\lambda}(x \cos \theta + y \sin \theta) + \phi\right) \quad (2.10)$$

Keterangan:

$$a = (x \cos \theta + y \sin \theta)^2$$

$$b = (-x \sin \theta + y \cos \theta)^2$$

σ = *bandwidth* menyatakan nilai efektif dari lebar suatu citra

λ = *lambda* menyatakan panjang gelombang suatu citra

θ = *theta* menyatakan sudut suatu citra

γ = *gamma* menyatakan tingkat kecerahan (*brightness*) suatu citra

ϕ = *phase* menyatakan bentuk suatu citra

Beberapa filter dapat didefinisikan dengan mengubah parameter λ , θ , σ , ψ dan γ .

Menghasilkan Fitur dengan Filter Gabor dengan beberapa cara, cara paling sederhana untuk menghasilkan fitur dengan Filter Gabor adalah dengan menggunakan bagian nyata atau bagian imajiner dari matriks respons sebagai vektor fitur. Fase matriks respons dapat diambil sebagai vektor fitur karena mengandung informasi relevan tentang tepinya. Amplitudo matriks respons dapat diambil sebagai vektor fitur karena mencakup spektrum frekuensi. Jumlah kuadrat respon panjang gelombang berbeda dengan orientasi yang sama dapat diambil sebagai vektor fitur. Ini mewakili energi lokal dalam arah tertentu.

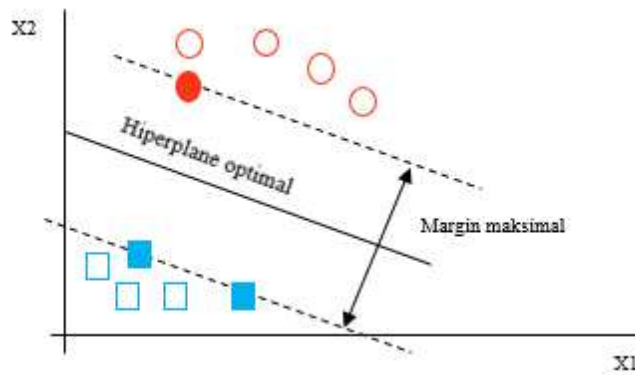
2.2.6 SUPPORT VECTOR MACHINE

Support Vector Machine (SVM) merupakan salah satu metode dalam *supervised learning* yang biasanya digunakan untuk klasifikasi (*Support Vector Classification*) dan regresi (*Support Vector Regression*). Dalam pemodelan klasifikasi, SVM memiliki konsep yang lebih matang dan lebih jelas secara matematis dibandingkan dengan teknik-teknik klasifikasi lainnya. SVM juga dapat mengatasi masalah klasifikasi dan regresi dengan *linear* maupun *non linear*.

SVM digunakan untuk mencari *hyperplane* terbaik dengan memaksimalkan jarak antar kelas. *Hyperplane* adalah sebuah fungsi yang dapat digunakan untuk pemisah antar kelas. Dalam 2-D fungsi yang digunakan untuk klasifikasi antar kelas disebut sebagai *line* whereas, fungsi yang digunakan untuk klasifikasi antar kelas dalam 3-D disebut *plane* similarly, sedangkan fungsi yang digunakan untuk klasifikasi di dalam ruang kelas dimensi yang lebih tinggi di sebut *hyperplane*.

Cara kerja SVM didasarkan pada SRM atau *Structural Risk Minimization* yang dirancang untuk mengolah data menjadi *Hyperplane* yang mengklasifikasikan ruang input menjadi dua kelas. Teori SVM diawali dengan pengelompokan kasus-kasus linier yang dapat dipisahkan dengan *hyperplane* dan dibagi menurut kelasnya. Konsep SVM diawali dengan masalah klasifikasi dua kelas sehingga membutuhkan set pelatihan positif dan negatif. SVM akan berusaha mendapatkan *hyperplane* (pemisah) sebaik mungkin untuk memisahkan kedua kelas dan memaksimalkan

margin kedua kelas tersebut. Berikut adalah visualisasi klasifikasi SVM dengan mencari *hyperplane* yang membedakan kedua kelas tersebut.



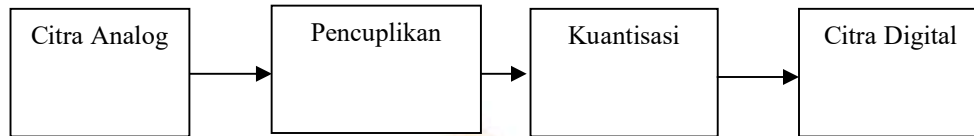
Gambar 2.4 Skema *Support Vector Machine*

2.2.7 PENGOLAHAN CITRA DIGITAL

Sebuah citra adalah visual merupakan representasi dari objek, orang, atau pemandangan. Sebuah citra digital adalah dua dimensi fungsi $f(x,y)$ yaitu proyeksi pemandangan 3 dimensi menjadi 2 dimensi bidang proyeksi, di mana x, y mewakili lokasi elemen citra atau piksel dan berisi nilai intensitas. Jika nilai x, y dan intensitas diskrit, maka bayangan tersebut dikatakan sebagai citra digital. Secara matematis, citra digital adalah representasi matriks dari citra dua dimensi menggunakan jumlah titik yang terbatas elemen sel, biasanya disebut sebagai piksel (elemen citra, atau kulit).

Setiap piksel diwakili oleh nilai numerik, untuk citra skala abu-abu, nilai tunggal yang mewakili intensitas piksel (biasanya dalam rentang $[0, 255]$) sudah cukup, untuk citra berwarna, tiga nilai (mewakili jumlah merah (R), hijau (G), dan biru (B)). Jika suatu citra hanya memiliki dua intensitas, maka citra tersebut disebut citra biner, dalam operasi pemrosesan citra, sebagian besar operasi dilakukan dicitra skala abu-abu. Untuk aplikasi pemrosesan citra berwarna, citra berwarna dapat didekomposisi menjadi komponen Merah, Hijau dan Biru dan masing-masing komponen adalah diproses secara independen sebagai citra skala abu-abu. Untuk diproses, citra yang diindeks dikonversi ke citra skala abu-abu atau citra warna

RGB untuk sebagian besar operasi sebagai prosesnya ditunjukkan pada Gambar 2.5.



Gambar 2.5 Proses konversi citra analog menjadi citra digital (Tyagi, 2018)

Citra digital memiliki tiga tingkat operasi pemrosesan citra yaitu :

1. Pemrosesan citra tingkat rendah: operasi sederhana pada citra (misalnya, perbaikan pada kontras, pengurangan kebisingan.) berada di bawah kategori ini, di mana keduanya masukan dan keluaran berupa citra.
2. Pemrosesan citra tingkat menengah: dalam kategori ini, operasi yang melibatkan ekstraksi atribut (misalnya, tepi, kontur, wilayah.).
3. Pemrosesan citra tingkat tinggi: kategori ini melibatkan citra yang memiliki kompleksitas dalam operasi pemrosesan yang terkait dengan analisis dan interpretasi konten yang berkaitan dengan pendukung keputusan.

Pengolahan citra memiliki beberapa metode dan algoritma. Yang paling sering digunakan dalam pemrosesan di antaranya:

1. *Binarization*/Binarisasi: Banyak tugas pemrosesan citra dapat dilakukan dengan mengkonversi citra berwarna atau citra skala abu-abu menjadi biner untuk menyederhanakan dan mempercepat sampai pengolahan. Konversi warna atau citra skala abu-abu ke citra biner hanya memiliki dua tingkat abu-abu (hitam dan putih) Dikenal sebagai binarisasi.
2. *Smoothing*/penghalusan: Teknik yang digunakan untuk mengaburkan atau menghaluskan detail objek dalam sebuah citra.
3. *Sharpening*/Penajaman: Teknik pemrosesan citra, dengan memperbaiki bagian tepi dan halus detail objek dalam citra ditingkatkan sehingga bisa dilihat dengan baik.

4. Penghapusan Noise dan *De-blurring*: Sebelum pemrosesan, jumlah *noise* yang masuk didalam citra dikurangi menggunakan filter penghilang *noise*. Teknik penghapusan citra terkadang bisa digunakan, tergantung pada jenis *noise* atau kabur pada citra.
5. Ekstraksi Tepi: Untuk menemukan berbagai objek sebelum menganalisis konten citra, dilakukan ekstraksi tepi.
6. Segmentasi: Proses pembagian citra digital ke dalam beberapa bagian. Segmentasi citra bertujuan untuk menyederhanakan penggambaran citra ke dalam bentuk yang lebih bermakna dan lebih mudah dianalisis

Berkenaan dengan cara penyimpanannya, citra digital dapat diklasifikasikan menjadi dua kategori: (1) Citra raster atau Bitmap, (2) Citra vektor, citra bitmap atau raster adalah array persegi panjang dari nilai atau piksel sampel. Citra-citra ini memiliki jumlah piksel yang tetap.

Di sisi lain, citra vektor disimpan dalam bentuk garis matematika dan kurva, informasi panjang garis, warna, ketipisan, disimpan dalam bentuk vektor. Citra-citra ini dapat ditampilkan dalam ukuran apa pun, resolusi apa pun, dan pada output apapun. Jenis berkas citra digital yang digunakan pada penelitian ini adalah JPEG, sebuah citra dalam bentuk yang paling sederhana mungkin hanya berisi dua intensitas, yaitu, hitam dan putih, dan hanya membutuhkan 1 bit untuk mewakili intensitas pada setiap piksel. Sistem pemrosesan citra digital yang lengkap terdiri dari berbagai elemen seperti akuisisi citra, penyimpanan, pemrosesan citra serta tampilan. Perangkat penginderaan digunakan untuk menangkap citra dengan cara menangkap citra dari benda yang ada dan mengubahnya menjadi bentuk digital.

Misalnya, digital kamera menggunakan intensitas cahaya yang ada pada benda dan mengubahnya menjadi bentuk citra digital. Elemen pemrosesan digunakan untuk melakukan berbagai operasi pada citra digital. Alat tersebut membutuhkan kombinasi perangkat keras dan perangkat lunak. Penyimpanan adalah bagian yang sangat penting dari sistem pengolahan citra. Ukuran berkas citra atau video sangat besar, citra 8-bit yang memiliki 1024 x 1024 piksel membutuhkan penyimpanan 1 megabyte ruang penyimpanan. Oleh karena itu, perangkat penyimpanan berukuran

besar diperlukan dalam sistem pemrosesan citra. Perangkat tampilan diperlukan untuk menampilkan citra. Ini bisa berupa komputer monitor, layar ponsel, proyektor atau printer (Tyagi, 2018).

2.2.8 OPTIMASI DIMENSI CITRA

Optimasi imensi citra merupakan merupakan proses mengubah ukuran citra menjadi lebih kecil atau lebih besar. Mengubah ukuran gambar adalah salah satu operasi dasar dalam pemrosesan gambar. Mengubah ukuran citra merupakan langkah umum dalam pra-pemrosesan citra sebelum memasukkannya ke dalam model atau algoritma pengolahan lebih lanjut. Pemilihan metode mengubah ukuran dan faktor perubahan ukuran harus dipertimbangkan dengan hati-hati untuk memenuhi kebutuhan aplikasi atau tugas tertentu. Mengubah ukuran citra biasanya dilakukan untuk beberapa tujuan:

1. Adaptasi Tampilan: Mengubah ukuran citra untuk menyesuaikan tampilan di layar atau media tertentu. Ini dapat melibatkan penyesuaian ukuran citra agar muat di dalam suatu ruang atau aplikasi tertentu.
2. Optimasi Kinerja: Citra sering diubah ukurannya untuk mengoptimalkan kinerja dalam proses analisis atau pemrosesan lebih lanjut. Ukuran citra yang lebih kecil memerlukan waktu dan sumber daya komputasi yang lebih sedikit.
3. Penghematan Ruang Penyimpanan: Mengurangi ukuran citra dapat menghemat ruang penyimpanan. Ini berguna dalam situasi di mana penyimpanan terbatas atau untuk mengurangi waktu transfer data.
4. Pengolahan Cepat: Beberapa algoritma atau metode pemrosesan citra dapat dijalankan lebih cepat pada citra berukuran kecil.

2.2.9 RUANG WARNA

RGB merupakan singkatan dari *red* (merah), *green* (hijau), dan *blue* (biru). RGB adalah salah satu ruang warna yang paling umum digunakan dalam pengolahan citra digital. Ruang warna RGB mewakili warna sebagai kombinasi dari tiga komponen tersebut. Setiap piksel pada citra RGB memiliki tiga nilai intensitas, masing-masing untuk komponen merah, hijau, dan biru. Nilai intensitas ini dapat

berkisar antara 0 hingga 255, di mana 0 mewakili tidak ada intensitas warna dan 255 mewakili intensitas warna yang maksimum.

Kombinasi nilai intensitas yang berbeda akan menghasilkan warna yang berbeda. Misalnya, piksel dengan nilai intensitas merah 255, hijau 0, dan biru 0 akan menghasilkan warna merah. Piksel dengan nilai intensitas merah 0, hijau 255, dan biru 0 akan menghasilkan warna hijau. Piksel dengan nilai intensitas merah 0, hijau 0, dan biru 255 akan menghasilkan warna biru.

2.2.10 AUGMENTASI

Augmentasi citra adalah proses memodifikasi atau memanipulasi suatu citra, sehingga citra asli dalam bentuk standar akan diubah bentuk dan posisinya. Tujuan dari augmentasi citra adalah untuk menambah jumlah data citra, sehingga model pembelajaran mesin dapat lebih terlatih dan akurat dalam mengenali objek atau pola yang terdapat pada citra. Pemilihan teknik augmentasi citra yang tepat tergantung pada jenis aplikasi yang akan digunakan. Misalnya, untuk aplikasi pengenalan objek, teknik rotasi, pergeseran, dan pembesaran dapat digunakan untuk meningkatkan ketahanan model terhadap perubahan posisi objek. Untuk aplikasi klasifikasi citra, teknik perubahan kontras dan kecerahan dapat digunakan untuk meningkatkan ketahanan model terhadap perubahan pencahayaan.

2.2.11 METODE EVALUASI

Metode evaluasi performa menggunakan *confusion matrix* di mana metode ini digunakan untuk mengevaluasi kinerja model klasifikasi. Sebuah *Confusion matrix* terdapat empat jenis hasil yang dihitung: *True Positive* (TP), *False Positive* (FP), *True Negative* (TN), dan *False Negative* (FN). *True Positive* (TP) adalah jumlah data yang diklasifikasikan sebagai kelas positif (atau label yang diinginkan) dan sebenarnya merupakan kelas positif. *False Positive* (FP) adalah jumlah data yang diklasifikasikan sebagai kelas positif (atau label yang diinginkan) namun sebenarnya bukan merupakan kelas positif. *True Negative* (TN) adalah jumlah data yang diklasifikasikan sebagai kelas negatif (atau label yang tidak diinginkan) dan sebenarnya merupakan kelas negatif. *False Negative* (FN) adalah jumlah data yang

diklasifikasikan sebagai kelas negatif (atau label yang tidak diinginkan) namun sebenarnya bukan merupakan kelas negatif.

Confusion matrix pada dasarnya memberikan informasi perbandingan hasil klasifikasi yang dilakukan oleh sistem (model) dengan hasil klasifikasi sebenarnya. *Confusion matrix* berbentuk tabel matriks yang menggambarkan kinerja model klasifikasi pada serangkaian data uji yang nilai sebenarnya diketahui. Gambar 2.4 dibawah ini merupakan *Confusion matrix* dengan 4 kombinasi nilai prediksi dan nilai aktual yang berbeda.

Nilai Prediksi	Nilai Sebenarnya	
	1 (Positif)	0 (Negatif)
	TP (True Positive)	FP (False Positive)
0 (Negatif)	FN (False Negative)	TN (True Negative)

Gambar 2.6 Tabel *Confusion matrix* (Valero-carreras dkk., 2023)

Kinerja dari *Confusion matrix* dapat dihitung dengan menggunakan persamaan akurasi, presisi, dan *recall*.

a. Akurasi

Accuracy menggambarkan seberapa akurat model dapat mengklasifikasikan dengan benar. Maka, *accuracy* merupakan rasio prediksi benar (positif dan negatif) dengan keseluruhan data. Dengan kata lain, *accuracy* merupakan tingkat kedekatan nilai prediksi dengan nilai aktual (sebenarnya). Nilai *accuracy* dapat diperoleh dengan persamaan 2.6.

$$Accuracy = \frac{TP+TN}{TP+TN+FP+FN} \quad (2.11)$$

b. Presisi

Precision menggambarkan tingkat keakuratan antara data yang diminta dengan hasil prediksi yang diberikan oleh model. Maka, *precision* merupakan rasio prediksi benar positif dibandingkan dengan keseluruhan hasil yang diprediksi positif. Dari semua kelas positif yang telah di prediksi dengan benar, berapa

banyak data yang benar-benar positif. Nilai *precision* dapat diperoleh dengan persamaan 2.7.

$$Precision = \frac{TP}{TP+FP} \quad (2.12)$$

c. *Recall*

Recall menggambarkan keberhasilan model dalam menemukan kembali sebuah informasi. Maka, *recall* merupakan rasio prediksi benar positif dibandingkan dengan keseluruhan data yang benar positif. Nilai *recall* dapat diperoleh dengan persamaan 2.8.

$$Recall = \frac{TP}{TP+FN} \quad (2.13)$$

d. *F1-score*

F1 Score adalah rata-rata harmonik dari presisi dan *recall*, dan sering digunakan sebagai metrik evaluasi yang lebih baik atau ketika kesalahan klasifikasi positif palsu dan negatif palsu memiliki konsekuensi yang tidak seimbang. Semakin tinggi nilai *F1 Score*, semakin baik kinerja model dalam memprediksi kelas target. Nilai *F1 Score* dapat diperoleh dengan persamaan 2.9.

$$F1\ Score = 2 \times \frac{Precision \times Recall}{Precision + Recall} \quad (2.14)$$

e. *Specificity*

Specificity adalah salah satu metrik evaluasi penting dalam analisis klasifikasi yang mengukur seberapa baik model dapat mengidentifikasi kasus negatif dari semua kasus negatif yang sebenarnya. *Specificity* juga dikenal sebagai *true negative rate*. Nilai *Specificity* dapat diperoleh dengan persamaan 2.10.

$$Specificity = \frac{TN}{TN+FP} \quad (2.15)$$

2.2.12 KUALITAS TELUR

Telur berkualitas adalah telur dengan cangkang yang utuh dan tidak retak merupakan sebagai indikator utama kualitas telur (Amer Eissa, 2009; De Ketelaere dkk., 2023; Ketelaere dkk., 2006). Kondisi cangkang sangat penting dalam menjaga keamanan pangan karena berfungsi sebagai penghalang fisik yang mencegah masuknya patogen seperti *Salmonella*. Penelitian menunjukkan bahwa

kerusakan pada cangkang dapat meningkatkan risiko kontaminasi dan mengurangi nilai gizi telur. Telur ayam yang berkualitas dan dapat dikonsumsi adalah telur yang belum mengalami proses pengayaan zat gizi atau fortifikasi, pendinginan, pengawetan, dan proses pengeraman, dilihat dari sisi cangkang memiliki bentuk, kehalusan, ketebalan, keutuhan, dan kebersihan yang bagus.

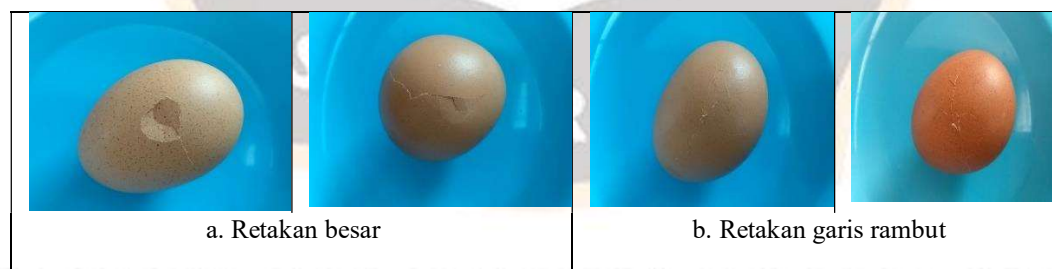
Tabel 2.1 Tingkatan Mutu Telur

No	Kondisi Cangkang	Keterangan
1	Bentuk	Normal
2.	Kehalusan	Halus
3	Ketebalan	Tebal
4	Keutuhan	Utuh
5	Kebersihan	Bersih

Sumber : (SNI 01-3926:2008, 2008).

Telur ayam konsumsi diklasifikasikan berdasarkan warna cangkang dan bobot telur, untuk warna cangkang sesuai dengan galurnya dan bobot kecil (< 50 g), sedang (50 g sampai dengan 60 g) dan besar (> 60 g) (SNI 01-3926:2008, 2008).

Telur yang mengalami cacat atau kurang berkualitas baik pada Gambar 2.7, dapat dilihat bahwa hal tersebut merupakan permasalahan yang ada pada sebuah telur, telur retak di kategorikan dalam retakan garis rambut, retakan bintang, atau retakan besar yang mengakibatkan lubang pada cangkang, penyebabnya di antaranya, stres, usia ayam tua, nutrisi yang tidak memadai (Alltech, 2018).



Gambar 2.7 Kualitas Telur bermasalah (Alltech, 2018)