

Implementasi Algoritma Convolutional Neural Network Untuk Klasifikasi Citra Kemasan Kardus Defect dan No Defect

Alan Antoni*, Tatang Rohana, Adi Rizky Pratama

Fakultas Ilmu Komputer, Teknik Informatika, Universitas Buana Perjuangan, Karawang, Indonesia

Email: Email: ^{1,*}if19.alanantoni@mhs.ubpkarawang.ac.id, ²tatang.rohana@ubpkarawang.ac.id, ³adi.rizky@ubpkarawang.ac.id

Email Penulis Korespondensi: ^{1,*}if19.alanantoni@mhs.ubpkarawang.ac.id

Submitted: 21/03/2023; Accepted: 31/03/2023; Published: 31/03/2023

Abstrak-Kemasan merupakan aspek penting sebuah produk, karena kemasan dapat mempengaruhi kualitas serta daya saing dari produk tersebut. Kemasan rusak dapat mengakibatkan kualitas produk menurun. Salah satu kemasan yang populer digunakan adalah kardus berjenis corrugated box. Untuk membedakan kemasan kardus defect dan no defect secara visual adalah terdapat sobekan, lubang serta penyok pada kemasan kardus defect. Sedangkan pada kemasan kardus no defect memiliki visual tidak terdapat sobekan, lubang ataupun penyok. Untuk mempermudah klasifikasi tersebut diperlukan teknologi yang dapat membedakan antara kemasan kardus defect dan no defect. Dalam penelitian ini total citra yang digunakan sebagai dataset adalah 1300 citra, yang kemudian dibagi menjadi 2 dengan persentase 80% untuk data latih dan 20% untuk data test. Dataset terlebih dahulu melalui tahap preprocessing sebelum digunakan. Preprocessing terdiri dari cropping, augmentation dan resizing. Serta dilakukan segmentasi menggunakan metode Grabcut. Kemudian dilakukan juga ekstraksi ciri menggunakan Local Binary Pattern (LBP). Penelitian ini menggunakan algoritma Convolutional Neural Network dengan jumlah convolution layer sebanyak 3 yakni sebesar 16,32 dan 64 dan optimizer Adam. Dilakukan 4 eksperimen dengan membedakan Hyperparameter epoch, input size citra serta besaran learning rate. Dari hasil penelitian menunjukkan model yang dihasilkan dengan Hyperparameter epoch sebesar 30, input size citra sebesar 300x300 serta learning rate sebesar 0,001 mendapatkan performa terbaik dengan nilai accuracy mencapai 95,77%, precision 96%, recall 96%, f1-score 96% dan loss sebesar 0.1478.

Kata Kunci: Kemasan Kardus; Convolutional Neural Network; Deep Learning; Local Binary Patern (LBP); Klasifikasi

Abstract-Packaging is an important aspect of a product, because packaging can affect the quality and competitiveness of the product. Damaged packaging can result in decreased product quality. One popular packaging used is corrugated cardboard type box. To visually distinguish defect and no defect cardboard packaging, there are tears, holes and dents on the defect cardboard packaging. Whereas the no defect cardboard packaging has a visual that there are no tears, holes or dents. To simplify the classification, technology is needed that can distinguish between defect and no-defect cardboard packaging. In this study the total images used as a dataset are 1300 images, which are then divided into 2 with a percentage of 80% for training data and 20% for test data. The dataset first goes through the preprocessing stage before being used. Preprocessing consists of cropping, augmentation and resizing. And also do the segmentation process using Grabcut method. Then feature extraction is also performed using Local Binary Pattern (LBP). This study uses the Convolutional Neural Network algorithm with a total of 3 convolution layers, namely 16.32 and 64 and the Adam optimizer. Four experiments were carried out by differentiating the hyperparameter epoch, the input image size and the learning rate. The results showed that the model produced with an epoch hyperparameter of 30, an input image size of 300x300 and a learning rate of 0.001 obtained the best performance with an accuracy value of 95.77%, 96% precision, 96% recall, 96% f1-score and loss of 0.1478.

Keywords: Cardboard Packaging; Convolutional Neural Networks; Deep Learning; Local Binary Pattern (LBP); Classification

1. PENDAHULUAN

Kemasan atau *packaging* merupakan aspek penting dalam sebuah produk, karena kemasan dapat mempengaruhi kualitas serta daya saing dari produk tersebut. Kemasan dapat memengaruhi konsumen membeli suatu produk [1]. Terdapat beberapa fungsi dari kemasan, diantaranya adalah melestarikan keamanan dan kualitas produk, memaksimalkan durasi masa kadaluarsa, bukti bahwa produk belum dirusak, identifikasi tanggal dan tempat pembuatan untuk pengendalian persediaan dan identifikasi potensi bahaya, serta informasi penting pada label terkait persiapan proses produksi, keamanan dan nilai gizi [2]. Kemasan yang populer digunakan adalah kardus berjenis *corrugated box*, dimana kardus ini digunakan sebagai wadah serta pelindung produk yang akan dipasarkan. Apabila kardus mengalami kerusakan maka dapat menurunkan kualitas produk yang ada didalamnya, serta apabila sampai ke konsumen maka dapat menimbulkan konsumen melakukan complain. Untuk itu industri manufaktur perlu melakukan inspeksi terhadap produk yang akan dikirimkan kepada konsumen. Inspeksi yang saat ini berjalan mayoritas masih dilakukan dengan cara manual. Keterbatasan pada penglihatan manusia yang terjadi akibat *human error* menjadi salah satu faktor produk dengan kemasan rusak bisa lolos terkirim ke konsumen.

Untuk membedakan kemasan kardus produk *defect* dan *no defect* secara visual adalah pada kemasan kardus produk *defect* terdapat sobekan, lubang serta penyok. Sedangkan pada kemasan kardus *no defect* memiliki visual yang berbanding terbalik dari kemasan kardus produk *defect*, yakni tidak terdapat sobekan, lubang ataupun penyok. Untuk mempermudah klasifikasi tersebut diperlukan teknologi yang dapat membedakan antara kemasan kardus *defect* dan *no defect*. Dengan pesatnya perkembangan teknologi, muncul metode baru pengolahan citra yakni metode *Deep Learning*. *Deep Learning* adalah arsitektur jaringan saraf komputasi dengan banyak *hidden layers* [3]. Tujuan penelitian ini adalah untuk dapat menemukan model yang baik dengan menerapkan algoritma *Convolutional Neural Network* sehingga dapat membuat model dengan akurasi yang tinggi dan dapat dimanfaatkan dalam klasifikasi kemasan kardus produk.

Terdapat beberapa penelitian sebelumnya yang menggunakan algoritma *Convolutional Neural Network* dalam melakukan berbagai klasifikasi, diantaranya dilakukan oleh Febian Fitra Maulana, dkk yang mengklasifikasi citra buah dengan jumlah kelas sebanyak 15 kelas menggunakan algoritma CNN. Pada penelitian tersebut model CNN yang dihasilkan memiliki akurasi sebesar 100% dan loss sebesar 0,012, dan pada pengujian model CNN menggunakan 45 sampel citra buah diperoleh akurasi sebesar 91,42 % [4]. Selanjutnya penelitian oleh Moh.Arie Hasan dkk. untuk klasifikasi penyakit pada daun anggur dengan menggunakan model CNN-VGG16, dimana dataset yang digunakan adalah data dari Kaggle dengan total 4000 citra daun anggur untuk empat kelas yang terdiri dari daun campak hitam, bercak daun, hawar daun dan daun sehat. Selain itu juga menggunakan citra yang diambil dari Google dengan jumlah 100 gambar untuk dijadikan data uji di luar dataset. Dan diperoleh hasil akurasi pelatihan model CNN adalah 99,50%. Pengujian dengan data uji memberikan akurasi 97,25%, sedangkan menggunakan data citra uji diluar dataset memberikan akurasi 95% [5]. Selain itu juga penelitian yang dilakukan oleh Dodi Efendi, dkk yang menggunakan algoritma CNN dengan arsitektur ResNet50 untuk mengklasifikasi citra daging sapi dan babi dengan jumlah data citra sebanyak 1000 citra untuk masing-masing kelas yakni daging babi, daging sapi dan daging oplosan. Dimana hasil dari penelitian menunjukkan dengan menggunakan optimizer SGD mendapatkan tingkat *accuracy* sebesar 97,83%, *precision* 97%, *recall* 97% serta *f1 score* 97% dengan kondisi *batch size* 32, *learning rate* 0.01 dan nilai *epoch* sebesar 50 [6]. Penelitian lain yang menggunakan CNN adalah penelitian yang dilakukan oleh Alif Kirana, dkk yang digunakan untuk pengenalan pola aksara sunda dengan empat jenis pengujian dan pengujian menggunakan citra font komputer mencapai akurasi tertinggi yakni 100% [7].

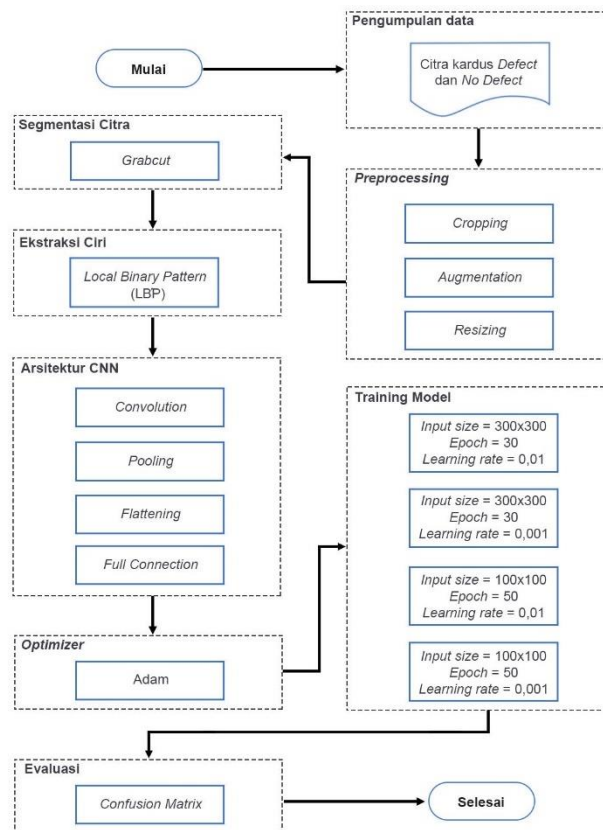
Kemudian penelitian lain yang menggunakan CNN adalah penelitian yang dilakukan oleh Rindi Kusumawardani, dkk dimana pada penelitian ini objek yang di klasifikasi adalah cacat pada kemasan kaleng. Percobaan dilakukan dengan membandingkan lima model jaringan, diantaranya *GoogLeNet*, *ShuffleNet*, *ResNet18*, *ResNet50* dan *ResNet101*. Hasil penelitian menunjukkan model arsitektur jaringan *ResNet50* dan *ResNet101* memberikan hasil terbaik dalam pengelompokan cacat terhadap kemasan kaleng dibandingkan dengan ketiga model jaringan lainnya dengan akurasi pengujian mencapai 95,56%. Nilai akurasi pengujian yang dimiliki kelima model diatas 90% [8]. Terdapat juga penelitian menggunakan CNN yang dilakukan oleh Ayu Ratna Juwita, dkk dimana CNN digunakan untuk klasifikasi citra batik karawang dengan langkah awal *preprocessing* dengan tujuan mengubah citra warna menjadi *grayscale*, kemudian dilakukan proses segmentasi pada citra menggunakan metode *otsu* untuk memisahkan citra pola batik dengan *background* kemudian di ekstraksi dengan metode *gray level co-occurrence matrix* dengan tujuan mendeteksi pola-pola dari batik. Pada penelitian tersebut data training yang digunakan adalah sebanyak 1094 citra latih dengan nilai akurasi untuk citra latih adalah 18,19%, dengan uji coba 344 citra batik dapat mengklasifikasikan 45 citra batik Karawang, 299 citra batik luar Karawang dengan nilai akurasi mencapai 18,60%, sedangkan hasil uji coba menggunakan citra batik karawang mencapai nilai akurasi mencapai 73,33 % [9]. Adapun penelitian yang menggunakan algoritma CNN yang digunakan untuk sortir botol minuman yang dilakukan oleh Inggis Kurnia Trisiawan, dkk dimana model CNN memperoleh hasil nilai akurasi prediksi mencapai 98,526%, serta mendapatkan rata-rata nilai akurasi mencapai 97,71% saat dilakukan pengujian menggunakan *10-fold cross validation* [10].

Berdasarkan tinjauan dari berbagai penelitian menggunakan algoritma *Convolutional Neural Network* (CNN) untuk berbagai klasifikasi dan mendapatkan hasil model dengan tingkat akurasi yang tinggi serta permasalahan yang telah diuraikan diatas, maka pada penelitian ini bertujuan untuk menggunakan *Convolutional Neural Network* (CNN) dalam membuat model untuk mengklasifikasi objek baru yakni kemasan kardus. Klasifikasi terdiri dari dua kelas yakni *defect* dan *no defect*. Arsitektur *Convolutional Neural Network* (CNN) yang dibangun memiliki 3 *Convolution layer* dan menggunakan *Optimizer* Adam serta akan menggunakan variasi *hyperparameter input size*, *epoch* dan *learning rate* berbeda untuk membuat model dengan akurasi terbaik yang akan membantu untuk mengklasifikasi kemasan kardus *defect* dan *no defect*.

2. METODOLOGI PENELITIAN

2.1 Tahapan Penelitian

Alur kerja penelitian yang digunakan pada penelitian ini yaitu dimulai dari pengumpulan dataset, *preprocessing data*, segmentasi, ekstraksi ciri, membangun arsitektur *Convolutional Neural Network*, Penggunaan *Optimizer* untuk memaksimalkan akurasi, *training model* yang terdiri dari 4 eksperimen dengan membedakan *hyperparameter* kemudian Evaluasi. Untuk dapat memudahkan dalam melakukan penyelesaian penelitian ini, maka dibuatlah tahapan penelitian. Diagram alir untuk tahapan pada penelitian ini dapat dilihat pada Gambar 1.



Gambar 1. Tahapan Penelitian

Pada Gambar 1 menampilkan penelitian diawali dengan pengumpulan data yakni berupa citra kemasan kardus *defect* dan *no defect*. Tahap selanjutnya yakni *preprocessing* dimana *preprocessing* terdiri dari 3 tahap diantaranya *cropping*, *augmentation* dan *resizing*. Setelah dilakukan *preprocessing* tahap berikutnya adalah melakukan segmentasi menggunakan metode *grabcut*, dan dilanjutkan dengan ekstraksi ciri menggunakan *Local Binary Pattern (LBP)*. Tahap selanjutnya adalah membuat arsitektur *Convolutional Neural Network (CNN)* yang terdiri dari proses *Convolution*, *Pooling*, *Flattening* dan *Full Connection* serta ditambahkan *Optimizer Adam* untuk memaksimalkan akurasi. Kemudian selanjutnya adalah tahap *training model* yang terdiri dari 4 eksperimen dengan *hyperparameter* yang berbeda. *Hyperparameter* yang dibedakan yakni besaran *input size*, *epoch* dan *learning rate*. Dan terakhir yakni tahap evaluasi, evaluasi dilakukan terhadap model yang dihasilkan dari proses *training* menggunakan *confusion matrix*.

2.2 Pengumpulan Data

Data yang digunakan dalam penelitian ini merupakan data foto kemasan kardus produk *defect* dan *no defect* yang berasal dari *warehouse finished goods* di salah satu industri manufaktur makanan dan minuman yang bertempat di kabupaten Karawang. Data yang diperoleh dari akuisisi dari sumber tersebut dan setelah melalui tahap *preprocessing* adalah sebanyak 650 foto kemasan kardus *defect* dan 650 foto kemasan kardus *no defect*. Sehingga total citra yang digunakan sebagai dataset adalah sebanyak 1300 citra. Berikut rincian jumlah dataset dapat dilihat pada Tabel 1.

Tabel 1. Jumlah dataset

Kelas	Jumlah
<i>Defect</i>	650
<i>No Defect</i>	650



Gambar 2. Contoh Citra Kemasan Kardus Defect



Gambar 3. Contoh Citra Kemasan Kardus *No Defect*

Gambar 2 merupakan sampel dataset citra kemasan kardus dengan kondisi *Defect* dan Gambar 3 merupakan sampel dataset citra kemasan kardus dengan kondisi *No Defect*. Kemudian dataset dibagi menjadi 2 bagian yang pertama digunakan sebagai dataset latih dan kedua sebagai dataset test. Tabel 2 merupakan persentase pembagian dataset.

Tabel 2. Pembagian dataset

Jenis	Kelas <i>Defect</i>	Kelas <i>No Defect</i>
Data latih	80%	80%
Data test	20%	20%

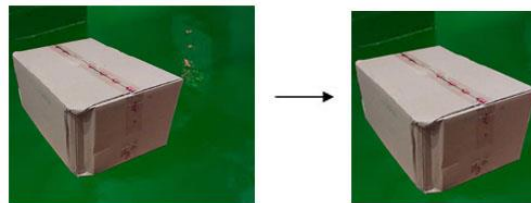
Dari persentase pembagian dataset yang ditunjukkan pada Tabel 2, maka didapatkan jumlah data latih untuk Citra Kemasan Kardus *Defect* dan *No Defect* masing-masing sebanyak 520 citra. Kemudian jumlah data *test* untuk Citra Kemasan Kardus *Defect* dan *No Defect* masing-masing sebanyak 130 citra.

2.3 Image Preprocessing

Setelah data berhasil dikumpulkan, tahap selanjutnya merupakan tahap *image preprocessing*. Tujuan dari tahap ini adalah agar menghasilkan citra dengan kualitas yang lebih baik untuk proses berikutnya. *Preprocessing* yang dilakukan meliputi *cropping*, *augmentation* dan *resizing*.

a. *Cropping*

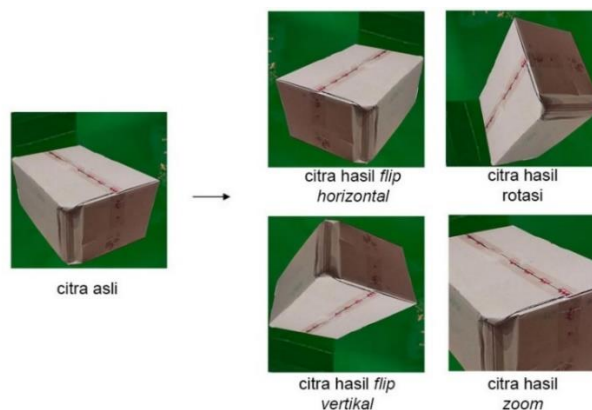
Tujuan dari *Cropping* adalah menyeleksi atau memfokuskan area yang ingin digunakan. Citra kardus pada tahap ini diseleksi dan di-*cropp* dengan ukuran 500x500 piksel.



Gambar 4. Proses *Cropping* Citra

b. *Augmentation*

Augmentasi adalah proses pengolahan data citra dengan memodifikasi data citra [11]. Fungsinya adalah menghindari adanya masalah *overfitting*, yakni dengan memperluas atau memperbanyak dataset secara artifisial [12]. Pada penelitian ini proses augmentasi data meliputi *flip horizontal*, *flip vertical*, *rotation* dan *zoom*.



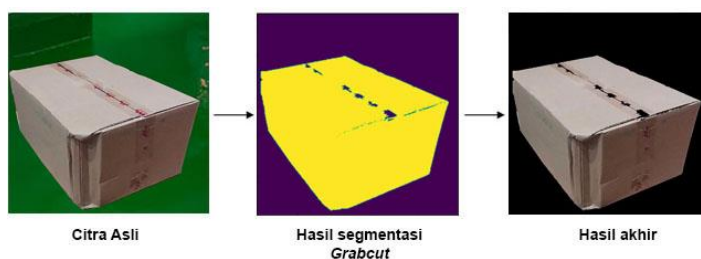
Gambar 5. Hasil *Augmentation*

c. *Resizing*

Selanjutnya adalah *resizing*, Tujuan dari *resize* adalah untuk efisiensi dan efektifitas yang terdapat pada sistem input CNN [13]. Pada proses ini citra di-*resize* menjadi ukuran 300x300 piksel dan *resize* kedua adalah 100x100. Dengan tujuan untuk menyamakan ukuran citra input kemasan kardus dan untuk mengetahui perbedaan tingkat akurasi dengan citra input yang berbeda.

2.4 Segmentasi Citra

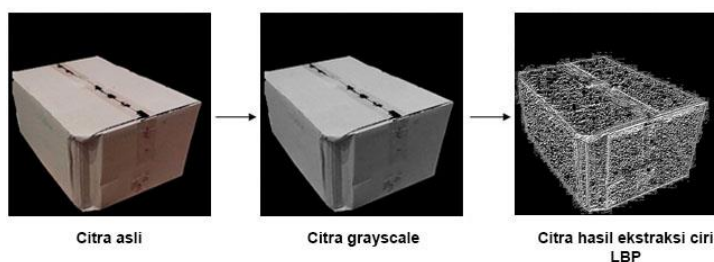
Dalam penelitian ini, segmentasi dilakukan menggunakan metode *Grabcut*. *Grabcut* merupakan metode yang digunakan untuk mengelompokkan bagian gambar yang termasuk kedalam kelompok *foreground* gambar agar terpisah dari *background* [14]. Dua tahap umum yang terdapat pada segmentasi citra berbasis *Grabcut* adalah inisiasi dan iterasi [15]. *Grabcut* memperkirakan distribusi warna antara *foreground* dan *background* dengan *Gaussian Mixture Model* (GMM) dan Inisiasi kotak yang digunakan sebagai daerah segmentasi.



Gambar 6. Proses segmentasi citra

2.5 Ekstraksi Ciri

Ekstraksi ciri merupakan salah satu proses dalam pengenalan citra, pada penelitian ini ekstraksi ciri digunakan untuk mendapatkan ciri atau informasi atau fitur dari citra kemasan kardus. Metode ekstraksi ciri yang digunakan adalah LBP (*Local Binary Pattern*), LBP pertama kali diperkenalkan oleh Timo Ojala. LBP merupakan operator yang berfungsi untuk menggambarkan fitur tekstur lokal dari sebuah citra [16]. Matriks yang digunakan LBP berukuran 3x3 dengan *threshold* merupakan nilai tengah dari suatu piksel. Cara kerja dari ekstraksi ciri LBP adalah dengan membandingkan nilai piksel pada pusat atau tengah dari citra dengan nilai piksel tetangganya. Nilai piksel di tengah atau pusat akan dikurangi dengan nilai pada piksel tetangganya. Jika nilai piksel yang dihasilkan dari pengurangan tersebut lebih atau sama dengan 0, maka akan diberi nilai 1. Jika nilai pikselnya kurang dari 0, maka akan diberi nilai 0. Selanjutnya menyusun 8 nilai biner tersebut searah jarum jam, kemudian diubah kedalam nilai desimal untuk menggantikan nilai piksel pada pusat atau tengah citra [17]. Gambar 8 merupakan proses dari ekstraksi ciri menggunakan LBP, citra hasil dari ekstraksi ciri ini yang kemudian dijadikan dataset untuk *training model*.



Gambar 7. Proses ekstraksi ciri

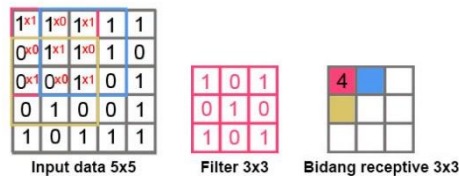
2.6 Arsitektur Convolutional Neural Network

Convolutional Neural Network merupakan salah satu kelas dari *deep learning*. CNN dapat melakukan pengenalan dan klasifikasi citra dengan mengimplementasikan jaringan saraf tiruan pada makhluk hidup. Metode CNN dapat diaplikasikan untuk pengenalan wajah, klasifikasi gambar, analisis dokumen, dsb [18]. Pada *Convolutional Neural Network* metode yang digunakan yaitu *backpropagation* (pembelajaran) dan *feedforward* (klasifikasi). Tahapan dalam membangun arsitektur *Convolutional Neural Network* diantaranya :

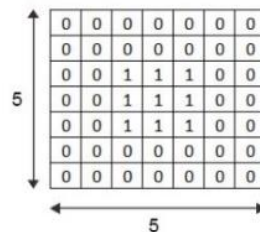
a. *Convolution*

Tujuan dari *Convolution* adalah untuk mendeteksi karakter dari sebuah objek seperti tepi, kurva atau warna. Beberapa parameter pada *layer* ini dapat dimodifikasi untuk mengubah sifat pada tiap *layer* misalnya ukuran filter, stride dan padding. Pada penelitian ini proses *Convolution* dilakukan sebanyak 3 kali proses pada arsitektur *Convolutional Neural Network* yang dibuat. Proses *Convolution* yang pertama menggunakan *layer filter* 16, kedua menggunakan *layer filter* 32 dan ketiga menggunakan *layer filter* 64. Dan dari ketiga proses *convolution* tersebut

menggunakan kernel 3x3. Kemudian untuk Fungsi aktivasinya menggunakan ReLU (*Rectified Linear Unit*). Proses stride diilustrasikan seperti pada Gambar 8 dan padding diilustrasikan pada Gambar 9.



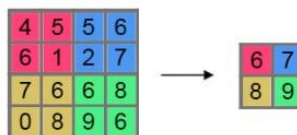
Gambar 8. Proses *Convolution* dengan stride 1



Gambar 9. Operasi zero padding 2 pada data 3x3

b. *Pooling*

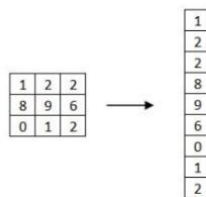
Pooling atau subsampling dipergunakan untuk pengurangan ukuran matriks. Terdapat dua jenis *pooling* yakni *average pooling* dan *max-pooling*. Pada penelitian ini proses *pooling* pada arsitektur *Convolutional Neural Network* yang dibuat menggunakan *max-pooling*. Perhitungan pada *max-pooling* ditunjukkan pada Gambar 10 berikut :



Gambar 10. Perhitungan *max-pooling*

c. *Flattening*

Setelah data *pooling* yang berupa array 2 dimensi, selanjutnya dikonversi menjadi data satu dimensi *single vector*. Ilustrasinya ditunjukkan pada Gambar 11 berikut:

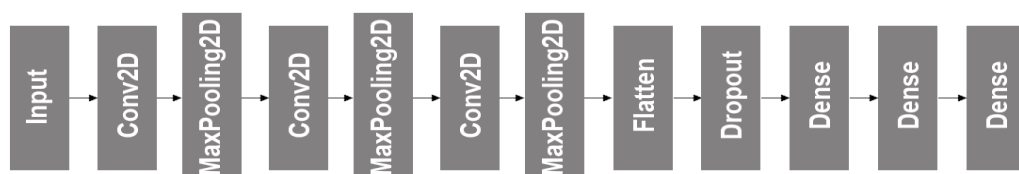


Gambar 11. Proses *flattening*

d. *Full Connection*

Tahap selanjutnya yakni proses dense, dimana proses dense dilakukan sebanyak 3 kali. Pada proses dense pertama 1024 *unit code*, kedua menggunakan 520 *unit node* sedangkan ketiga menggunakan jumlah *unit code* sesuai jumlah kelas yakni 2. Pada proses dense pertama dan kedua, *activation* yang digunakan adalah ReLU sedangkan pada proses dense yang ketiga menggunakan *activation softmax*. Selain proses dense, dilakukan juga proses dropout, dimana proses ini berfungsi untuk meregulasi parameter dengan tujuan menghindari *overfitting*. Proses dropout yang dilakukan adalah 1 kali.

Dari beberapa tahapan yang telah diuraikan diatas, berikut merupakan struktur arsitektur dari model yang digunakan dalam penelitian ini :



Gambar 12. Arsitektur model CNN

2.7 Optimizer

Tujuan dari digunakannya *Optimizer* adalah untuk memaksimalkan akurasi, meminimalkan kesalahan, dan menemukan bobot yang optimal. Pada saat proses *training* dilakukan, parameter atau bobot model diubah untuk meminimalkan kerugian supaya prediksi dapat seakurat mungkin. *Optimizer* yang digunakan dalam penelitian ini adalah Adam. *Optimizer* Adam adalah perkembangan dari algoritma *Stochastic Gradient Descent* (SGD) klasik dimana bobot network telah diperbarui [19]. Pada penggunaan *Optimizer* Adam, diperlukan adanya penentuan besar *learning rate* yang akan digunakan selama *training* dataset. *Learning rate* merupakan salah satu parameter *training* yang diatur untuk menghitung nilai koreksi bobot selama proses *training* [19]. Dalam penelitian ini *learning rate* yang digunakan adalah sebesar 0,01 dan 0,001.

2.8 Training Model

Proses *training* dilakukan dengan menggunakan algoritma CNN dan menggunakan tools Jupyter Notebook serta bahasa pemrograman Python 3.9. *Training* dilakukan dengan beberapa eksperimen untuk mendapatkan hasil akurasi terbaik. Eksperimen yang dilakukan yakni dengan membedakan *epoch*, *input size* citra serta besaran *learning rate*.

2.9 Evaluasi

Metode yang digunakan untuk mengevaluasi klasifikasi pada penelitian ini menggunakan *Confusion Matrix*. Data yang digunakan sebagai data uji adalah menggunakan data test. Tabel 3 merupakan ilustrasi dari *Confusion Matrix*.

Tabel 3. Ilustrasi *Confusion Matrix* [20]

	<i>Predicted Label : Yes</i>	<i>Predicted Label : No</i>
<i>True Label : Yes</i>	TP	FN
<i>True Label : No</i>	FP	TN

Nilai *True Negative* (TN) adalah data yang diklasifikasikan dengan benar sebagai negatif atau salah. *True Positive* (TP) adalah data yang diklasifikasikan dengan benar sebagai keluaran positif atau benar. *False Positive* (FP) adalah data yang diklasifikasikan dengan kurang tepat sebagai keluaran positif atau benar. *False Negative* (FN) adalah data yang diklasifikasikan dengan kurang tepat dengan keluaran negatif atau salah. Pada penelitian ini matriks yang digunakan adalah nilai *macro avg* atau nilai rata-rata dari *accuracy*, *precision*, *recall* dan *f1-score* untuk dua kelas yakni kelas *defect* dan kelas *no defect*. Untuk nilai dari *accuracy*, *precision*, *recall* dan *f1-score* dihitung dengan rumus 1, 2, 3 dan 4 [20].

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

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

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

$$F1 - Score (\%) = \frac{(2 \times recall \times precision)}{(recall+precision)} \quad (4)$$

3. HASIL DAN PEMBAHASAN

3.1 Hasil Training dan Evaluasi Model

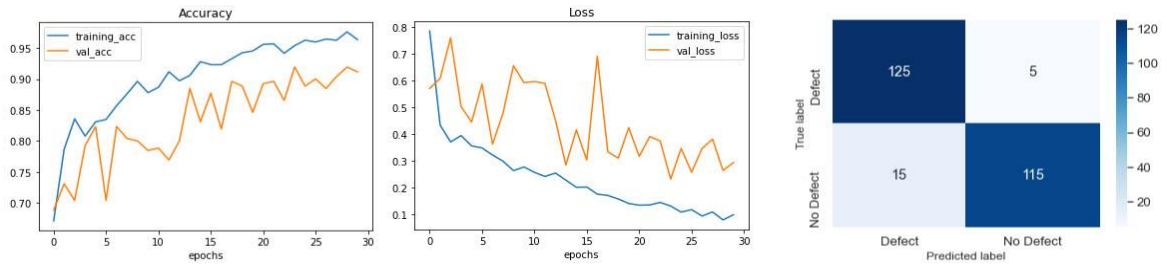
Hasil dari implementasi proses *training* adalah sebuah model python, dimana proses *training* dilakukan menggunakan algoritma *Convolutional Neural Network* (CNN) serta *Optimizer* Adam untuk meningkatkan akurasi. *Tools* yang digunakan adalah Jupyter Notebook dengan bahasa pemrograman Python 3.9. Untuk menghasilkan akurasi terbaik, dilakukan sebanyak 4 eksperimen yakni dengan membedakan *input size* citra, *epoch* dan *learning rate*. Untuk *hyperparameter* yang pertama yakni menggunakan *input size* 300x300, *epoch* 30 dan *learning rate* 0,01. Untuk *hyperparameter* kedua menggunakan *input size* 300x300, *epoch* 30 dan *learning rate* 0,001. *Hyperparameter* ketiga menggunakan *input size* 100x100, *epoch* 50 dan *learning rate* 0,01 dan untuk *hyperparameter* keempat menggunakan *input size* 100x100, *epoch* 50 dan *learning rate* 0,001. Tabel 4 merupakan rincian *hyperparameter* yang digunakan dalam penelitian ini :

Tabel 4. *Hyperparameter* yang digunakan

<i>Hyperparameter ke</i>	<i>Input Size</i>	<i>Epoch</i>	<i>Learning rate</i>
1	300x300	30	0,01
2	300x300	30	0,001
3	100x100	50	0,01

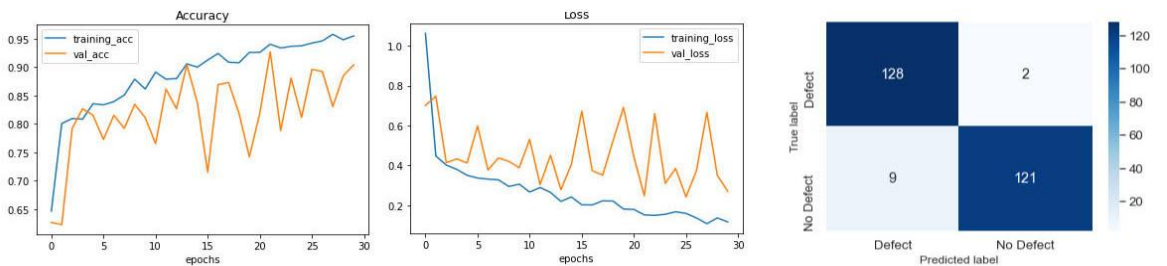
4	100x100	50	0,001
---	---------	----	-------

Setelah penentuan *hyperparameter* yang telah dilakukan sebelumnya, kemudian dilakukan proses *training* menggunakan *hyperparameter* tersebut. Setelah proses *training* selesai maka dilanjutkan dengan mengevaluasi model menggunakan *confusion matrix*. Dan berikut merupakan hasil dari proses *training* serta evaluasi model menggunakan *confusion matrix*.



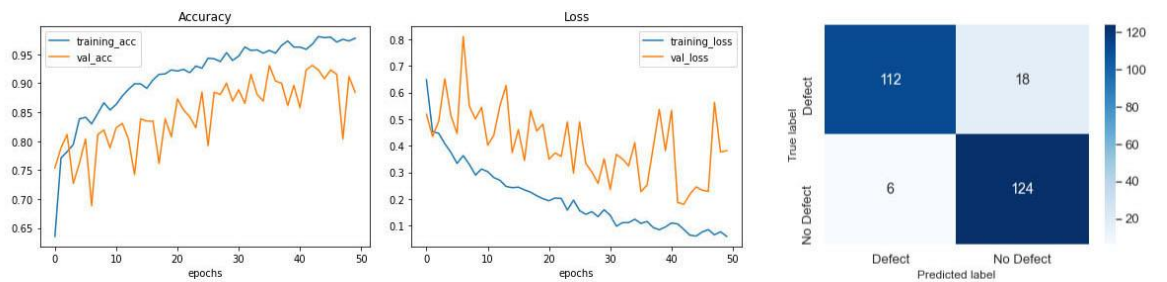
Gambar 13. Grafik Accuracy, Loss dan Confusion Matrix dengan Hyperparameter ke 1

Gambar 13 merupakan grafik *performance* serta hasil evaluasi *confusion matrix* dari proses training menggunakan *hyperparameter* ke 1 dimana *input size* 300x300, *epoch* 30 dan *learning rate* 0,01. Dan model yang dihasilkan pada training dengan *hyperparameter* tersebut menghasilkan nilai *accuracy* mencapai 92,31%, *precision* 93%, *recall* 92%, *f1-score* 92%, *loss* sebesar 0.2678.



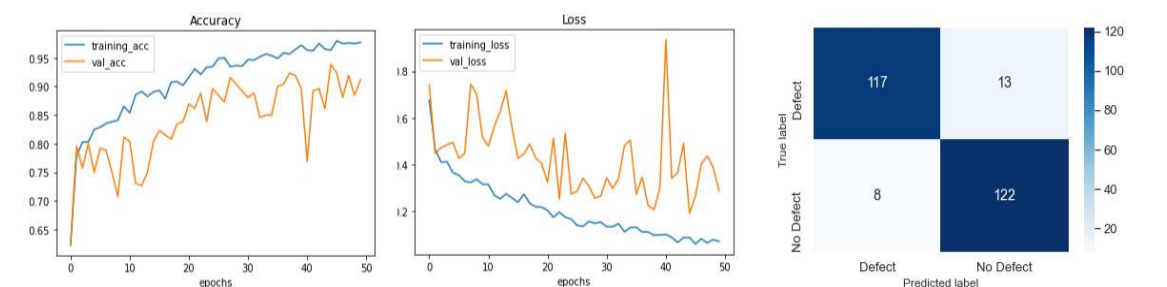
Gambar 14. Grafik Accuracy, Loss dan Confusion Matrix dengan Hyperparameter ke 2

Gambar 14 merupakan grafik *performance* serta hasil evaluasi *confusion matrix* dari proses training menggunakan *hyperparameter* ke 2 dimana *input size* 300x300, *epoch* 30 dan *learning rate* 0,001. Dan model yang dihasilkan pada training dengan *hyperparameter* tersebut menghasilkan nilai *accuracy* mencapai 95,77%, *precision* 96%, *recall* 96%, *f1-score* 96%, *loss* sebesar 0.1478.



Gambar 15. Grafik Accuracy, Loss dan Confusion Matrix dengan Hyperparameter ke 3

Gambar 15 merupakan grafik *performance* serta hasil evaluasi *confusion matrix* dari proses training menggunakan *hyperparameter* ke 3 dimana *input size* 100x100, *epoch* 50 dan *learning rate* 0,01. Dan model yang dihasilkan pada training dengan *hyperparameter* tersebut menghasilkan nilai *accuracy* mencapai 90,77%, *precision* 91%, *recall* 91%, *f1-score* 91%, *loss* sebesar 0.3136.



Gambar 16. Grafik Accuracy, Loss dan Confusion Matrix dengan Hyperparameter ke 4

Gambar 16 merupakan grafik *performance* serta hasil evaluasi *confusion matrix* dari proses training menggunakan *hyperparameter* ke 4 dimana *input size* 100x100, *epoch* 50 dan *learning rate* 0,001. Dan model yang dihasilkan pada training dengan *hyperparameter* ke 4 menghasilkan nilai *accuracy* mencapai 91,92%, *precision* 92%, *recall* 92%, *f1-score* 92%, *loss* sebesar 0.1861.

Tabel 5. Rangkuman *Performance*

Hyperparameter ke	Input Size	Epoch	Learning rate	Accuracy	Precision	Recall	F1-Score	Loss
1	300x300	30	0,01	92,31%	93%	92%	92%	0.2678
2	300x300	30	0,001	95,77%	96%	96%	96%	0.1478
3	100x100	50	0,01	90,77%	91%	91%	91%	0.3136
4	100x100	50	0,001	91,92%	92%	92%	92%	0.1861

Tabel 5 merupakan rangkuman *performance* dari 4 *hyperparameter* yang digunakan. *Hyperparameter* ke 2 yakni dengan *input size* 300x300, *epoch* 30 dan *learning rate* 0,001 memiliki performa terbaik yakni dengan nilai *accuracy* mencapai 95,77%, *precision* 96%, *recall* 96%, *f1-score* 96%, *loss* sebesar 0.1478. Disusul dengan *hyperparameter* ke 1 yakni *input size* 300x300, *epoch* 30 dan *learning rate* 0,01 yang menghasilkan nilai *accuracy* mencapai 92,31%, *precision* 93%, *recall* 92%, *f1-score* 92% dan *loss* sebesar 0.2678. Selanjutnya *hyperparameter* ke 4 yakni dengan *input size* 100x100, *epoch* 50 dan *learning rate* 0,001 menghasilkan *accuracy* mencapai 91,92%, *precision* 92%, *recall* 92%, *f1-score* 92% dan *loss* sebesar 0.1861. Dan *hyperparameter* yang menghasilkan *accuracy* terkecil yaitu *hyperparameter* ke 3 yakni dengan *input size* 100x100, *epoch* 50 dan *learning rate* 0,01 yang menghasilkan nilai *accuracy* 90,77%, *precision* 91%, *recall* 91%, *f1-score* 91% dan *loss* sebesar 0.3136.

4. KESIMPULAN

Dalam penelitian ini dilakukan klasifikasi citra kemasan kardus yang terdiri dari 2 kelas yaitu *defect* dan *no defect* dengan menggunakan dataset total sebanyak 1300 citra yang terdiri dari 650 citra untuk kelas *defect* dan 650 citra untuk kelas *no defect*. Dari total 1300 citra kemudian dibagi menjadi data latih dan data test dengan persentase sebesar 80% untuk data latih dan 20% untuk data test. Proses *training* dilakukan menggunakan algoritma *Convolutional Neural Network* (CNN) dengan 4 *hyperparameter* yang berbeda untuk menghasilkan model dengan akurasi terbaik. Hasil menunjukkan model dengan *hyperparameter* *input size* 300x300, *epoch* 30 dan *learning rate* 0,001 menghasilkan performa terbaik yakni dengan *accuracy* mencapai 95,77%, *precision* 96%, *recall* 96%, *f1-score* 96% dan *loss* sebesar 0.1478. Disusul dengan *hyperparameter* *input size* 300x300, *epoch* 30 dan *learning rate* 0,01 yang menghasilkan nilai *accuracy* mencapai 92,31%, *precision* 93%, *recall* 92%, *f1-score* 92% dan *loss* sebesar 0.2678. Selanjutnya dengan *hyperparameter* *input size* 100x100, *epoch* 50 dan *learning rate* 0,001 menghasilkan *accuracy* mencapai 91,92%, *precision* 92%, *recall* 92%, *f1-score* 92% dan *loss* sebesar 0.1861. Dan *hyperparameter* yang menghasilkan *accuracy* terkecil yaitu *input size* 100x100, *epoch* 50 dan *learning rate* 0,01 menghasilkan nilai *accuracy* 90,77%, *precision* 91%, *recall* 91%, *f1-score* 91% dan *loss* sebesar 0.3136. Dari hasil penelitian ini dapat dikatakan bahwa algoritma CNN dapat mengklasifikasi citra kemasan kardus dengan kategori *defect* dan *no defect* dengan baik serta penentuan *hyperparameter* *input size*, *epoch* serta *learning rate* dapat mempengaruhi nilai *accuracy* dari model yang dihasilkan.

REFERENCES

- [1] Ari Widiati, "Peranan Kemasan (Packaging) Dalam Meningkatkan Pemasaran Produk Usaha Mikro Kecil Menengah (Umkm) Di 'Mas Pack' Terminal Kemasan Pontianak," Jurnal Audit dan Akuntansi Fakultas Ekonomi (JAAKFE), vol. Vol. 8, pp. 67–76, 2019.
- [2] Kementrian Perdagangan Republik Indonesia, "Desain Kemasan Produk Makanan Olahan," 2017. [Online]. Available: <http://djpen.kemendag.go.id>
- [3] Tariq M. arif, "Synthesis Lectures on Mechanical Engineering," 2020.
- [4] F. Fitra Maulana and N. Rochmawati, "Klasifikasi Citra Buah Menggunakan Convolutional Neural Network," JINACS, vol. Volume 01, no. Nomor 02, pp. 104–108, 2019.
- [5] Moh. A. Hasan, Y. Riyanto, and D. Riana, "Grape leaf image disease classification using CNN-VGG16 model," Jurnal Teknologi dan Sistem Komputer, vol. 9, no. 4, pp. 218–223, Oct. 2021, doi: 10.14710/jtsiskom.2021.14013.
- [6] D. Efendi, J. Jasril, S. Sanjaya, F. Syafria, and E. Budianita, "Penerapan Algoritma Convolutional Neural Network Arsitektur ResNet-50 untuk Klasifikasi Citra Daging Sapi dan Babi," JURIKOM (Jurnal Riset Komputer), vol. 9, no. 3, p. 607, Jun. 2022, doi: 10.30865/jurikom.v9i3.4176.
- [7] A. Kirana, H. Hikmayanti, and J. Indra, "Pengenalan Pola Aksara Sunda dengan Metode Convolutional Neural Network," Scientific Student Journal for Information, Technology and Science, vol. 1, no. 2, pp. 95–100, 2020.
- [8] R. Kusumawardani and P. D. Karningsih, "Detection and Classification of Canned Packaging Defects Using Convolutional Neural Network," PROZIMA (Productivity, Optimization and Manufacturing System Engineering), vol. 4, no. 1, pp. 1–11, Mar. 2021, doi: 10.21070/prozima.v4i1.1280.
- [9] A. Ratna Juwita et al., "IDENTIFIKASI CITRA BATIK DENGAN METODE CONVOLUTIONAL NEURAL NETWORK," Buana Ilmu, no. Vol 6 No 1, pp. 192–208, 2021.



- [10] I. K. Trisiawan and Y. Yuliza, “Penerapan Multi-Label Image Classification Menggunakan Metode Convolutional Neural Network (CNN) Untuk Sortir Botol Minuman,” *Jurnal Teknologi Elektro*, vol. 13, no. 1, p. 48, Feb. 2022, doi: 10.22441/jte.2022.v13i1.009.
- [11] N. IBRAHIM et al., “Klasifikasi Tingkat Kematangan Pucuk Daun Teh menggunakan Metode Convolutional Neural Network,” *ELKOMIKA: Jurnal Teknik Energi Elektrik, Teknik Telekomunikasi, & Teknik Elektronika*, vol. 10, no. 1, p. 162, Jan. 2022, doi: 10.26760/elkomika.v10i1.162.
- [12] B. Hanin and D. Rolnick, “How to Start Training: The Effect of Initialization and Architecture,” *Adv. Neural Inf. Process. Syst.*, vol. 2018–Decem, no. NeurIPS, pp. 571–581, 2018.
- [13] P. Winardi and E. Setyati, “Identifikasi Jenis Daging dengan Menggunakan Algoritma Convolution Neural Network,” *Journal of Information System, Graphics, Hospitality and Technology*, vol. 3, no. 02, pp. 82–88, Dec. 2021, doi: 10.37823/insight.v3i02.178.
- [14] Adrian Rosebrock, “OpenCV GrabCut: Foreground Segmentation and Extraction,” *pyimagesearch.com*, 2020.
- [15] A. W. Setiawan, “PERBANDINGAN PRESKRINING LESI KULIT BERBASIS CONVOLUTIONAL NEURAL NETWORK: CITRA ASLI DAN TERSEGMENTASI,” *Jurnal Teknologi Informasi dan Ilmu Komputer (JTIK)*, vol. 8, no. 4, pp. 793–799, 2021, doi: 10.25126/jtiik.202184411.
- [16] N. Akmalia, P. Sihombing, and Suherman, “Skin Diseases Classification Using Local Binary Pattern and Convolutional Neural Network,” in *2019 3rd International Conference on Electrical, Telecommunication and Computer Engineering, ELTICOM 2019 - Proceedings*, Sep. 2019, pp. 168–173. doi: 10.1109/ELTICOM47379.2019.8943892.
- [17] M. Ezar, A. Rivan, and S. Devella, “PENGENALAN IRIS MENGGUNAKAN FITUR LOCAL BINARY PATTERN DAN RBF CLASSIFIER,” *Jurnal SIMETRIS*, vol. 11, no. 1, 2020.
- [18] M. F. Herlambang, A. N. Hermana, and K. R. Putra, “Pengenalan Karakter Huruf Braille dengan Metode Convolutional Neural Network,” *Systemic: Information System and Informatics Journal*, vol. 6, no. 2, pp. 20–26, Jan. 2021, doi: 10.29080/systemic.v6i2.969.
- [19] N. Rochmawati et al., “Analisa Learning rate dan Batch size Pada Klasifikasi Covid Menggunakan Deep learning dengan Optimizer Adam,” *JIEET (Journal Information Engineering and Educational Technology)*, vol. 05, no. 02, pp. 44–48, 2021.
- [20] K. Foyosal Haque, F. Farhan Haque, L. Gandy, and A. Abdelgawad, “Automatic Detection of COVID-19 from Chest X-ray Images with Convolutional Neural Networks,” in *Proceedings - 2020 International Conference on Computing, Electronics and Communications Engineering, iCCECE 2020*, Aug. 2020, pp. 125–130. doi: 10.1109/iCCECE49321.2020.9231235.