

IDENTIFIKASI KARAT PADA SLACK SUPPORT MENGUNAKAN METODE CONVOLUTIONAL NEURAL NETWORK (CNN)

Keristian Paonsen¹, Leonardus Sandy Ade Putra², Redi Ratiandi Yacoub³
d1021201023@student.untan.ac.id¹, leonardusandy@ee.untan.ac.id², rediyacoub@ee.untan.ac.id³
Universitas Tanjungpura

ABSTRAK

Logam merupakan unsur yang sangat tidak asing dikarenakan banyak benda – benda disekitar kita yang menggunakan unsur logam, material logam sudah sangat umum digunakan di kehidupan sehari - hari. Logam merupakan sebuah unsur kimia dengan sifat yang sangat kuat,liat, keras dan dapat menghantarkan listrik atau energi panas, logam berasal dari biji logam yang didapat dengan cara penambangan[1]. Karat merupakan salah satu ancaman terhadap benda logam karena dapat menyebabkan menurunnya kegunaan alat karena bersifat rapuh, mudah larut dan juga dapat bersifat racun. Karat menyebar bertahap pada permukaan logam seperti penyakit yang sangat menular, karat merupakan hasil dari korosi yang merupakan oksidasi suatu logam dengan proses pengkaratan yang merupakan proses elektrokimia besi (Fe) bertindak sebagai pereduksi dan oksigen (O₂) yang terlarut dalam air sebagai pengoksidasi[2]. Kerugian yang besar dari dampak pengkaratan sehingga mengharuskan adanya upaya – upaya pencegahan terjadinya karat, dengan prinsip melindungi besi dan menjauhkan dari penyebab terjadinya karat dilihat dari faktor yang memengaruhi proses pengkaratan besi[3]. Oleh karena itu dalam upaya mengatasi permasalahan tersebut, dampak pencegahan awal dapat dilakukan dengan melakukan citra digital, hal ini menjadi potensi pencegahan awal serta ke-efisiensi penggunaan teknologi. Penggunaan teknologi mempermudah akan sesuatu yang kita perlukan, maka penggunaan citra digital dapat membantu pada identifikasi suatu objek. Citra adalah suatu representasi (gambaran), kemiripan, atau imitasi dari suatu objek. Citra terbagi 2 yaitu citra yang bersifat analog dan ada citra yang bersifat digital. Citra analog adalah citra yang bersifat continue seperti gambar pada monitor televisi, foto sinar X, dan lain-lain. Sedangkan pada citra digital adalah citra yang dapat diolah oleh komputer. Citra dapat didefinisikan sebagai fungsi $f(x,y)$ berukuran M baris dan N kolom, dengan x dan y adalah koordinat spasial, dan amplitudo f di titik koordinat (x,y) dinamakan intensitas atau tingkat keabuan dari citra pada citra tersebut. [4].

Kata Kunci: Karat, Citra Analog, Citra Digital.

ABSTRACT

Metals are elements that are very familiar due to the abundance of objects around us that utilize metallic components. Metal materials are widely used in everyday life. Metals are chemical elements characterized by their strength, ductility, hardness, and their ability to conduct electricity or thermal energy. Metals originate from metal ores obtained through mining processes [1]. Rust is one of the major threats to metal objects, as it can reduce their usability due to its brittle nature, solubility, and potential toxicity. Rust spreads gradually on the surface of metals like a highly contagious disease. Rust is the result of corrosion, which is an oxidation process involving electrochemical reactions in which iron (Fe) acts as the reducing agent and dissolved oxygen (O₂) in water acts as the oxidizing agent [2]. The significant losses caused by corrosion demand preventive efforts to avoid rust formation. These efforts are based on the principle of protecting metal, especially iron, from exposure to the factors that cause corrosion [3]. To address this issue, early prevention can be achieved through digital imaging, which presents a promising solution for initial detection and improves technological efficiency. The use of technology simplifies various tasks, and thus, digital imaging can assist in object identification. An image is a representation, resemblance, or imitation of an object. Images are classified into two types: analog images and digital images. Analog images are continuous, such as those seen on television screens, X-ray

photographs, and so on. Meanwhile, digital images are images that can be processed by computers. A digital image can be defined as a function $f(x,y)$ of size M rows and N columns, where x and y are spatial coordinates, and the amplitude of f at coordinates (x,y) is referred to as the intensity or gray level at that point [4].

Keywords: Rust, Analog Image, Digital Image.

PENDAHULUAN

Logam merupakan unsur yang sangat tidak asing dikarenakan banyak benda – benda disekitar kita yang menggunakan unsur logam, material logam sudah sangat umum digunakan dikehidupan sehari hari. Logam merupakan sebuah unsur kimia dengan sifat yang sangat kuat,liat, keras dan dapat menghantarkan listrik atau energi panas, logam berasal dari biji logam yang didapat dengan cara penambangan[1]. Walaupun banyak benda sehari – hari kita gunakan mengandung logam benda yang mengandung unsur logam terdapat kelemahan yaitu mudah berkarat.

Karat merupakan salah satu ancaman terhadap benda logam karena dapat menyebabkan menurunnya kegunaan alat karena bersifat rapuh, mudah larut dan juga dapat bersifat racun. Karat menyebar bertahap pada permukaan logam seperti penyakit yang sangat menular, karat merupakan hasil dari korosi yang merupakan oksidasi suatu logam dengan proses pengkaratan yang merupakan proses elektrokimia besi (Fe) bertindang sebagai pereduksi dan oksigen (O_2) yang terlarut dalam air sebagai pengoksidasi[2].

Kerugian yang besar dari dampak pengkaratan sehingga mengharuskan adanya upaya – upaya pencegahan terjadinya karat, dengan prinsip melindungi besi dan menjauhkan dari penyebab terjadinya karat dilihat dari faktor yang memengaruhi proses pengkaratan besi[3]. Oleh karena itu dalam upaya mengatasi permasalahan tersebut, dampak pencegahan awal dapat dilakukan dengan melakukan citra digital, hal ini menjadi potensi pencegahan awal serta ke-efisiensi penggunaan teknologi. Penggunaan teknologi mempermudah akan sesuatu yang kita perlukan, maka penggunaan citra digital dapat membantu pada identifikasi suatu objek.

Citra adalah suatu representasi (gambaran), kemiripan, atau imitasi dari suatu objek. Citra terbagi 2 yaitu citra yang bersifat analog dan ada citra yang bersifat digital. Citra analog adalah citra yang bersifat continue seperti gambar pada monitor televisi, foto sinar X, dan lain-lain. Sedangkan pada citra digital adalah citra yang dapat diolah oleh komputer. Citra dapat didefinisikan sebagai fungsi $f(x,y)$ berukuran M baris dan N kolom, dengan x dan y adalah koordinat spasial, dan amplitudo f di titik koordinat (x,y) dinamakan intensitas atau tingkat keabuan dari citra pada citra tersebut. Citra digital merupakan representatif dari citra yang diambil oleh mesin dengan bentuk pendekatan berdasarkan sampling dan kuantisasi. Sampling menyatakan besarnya kotak – kotak yang disusun pada baris dan kolom. Sampling menyatakan besarnya nilai tingkat kecerahan yang dinyatakan[4].

METODE PENELITIAN

Pada pengerjaan penelitian ini, bahan penelitian merujuk pada literatur - literatur yang membahas penggunaan metode Convolutional Neural Network (CNN) seperti yang dibahas pada penelitian terdahulu dalam tinjauan pustaka dan juga citra pada slack support berkarat. Adapun tambahan penulis mengambil dari sumber – sumber seperti jurnal terdahulu dalam daftar pustaka.

HASIL DAN PEMBAHASAN

Tahapan Perancangan Pengambilan Data

Pada tahap ini dilakukan proses pengambilan data yang akan diolah dengan proses sebagai berikut.

1. Proses Pengambilan Citra.

Pada proses ini citra yang sudah diambil seperti gambar maka akan melalui proses pemisahan antara benda dan background dengan bantuan software lain.



Gambar 1 Citra Asli

2. Penggunaan Perangkat

Pada sub bab ini menampilkan kebutuhan perangkat yang digunakan dalam pengimplementasian metode yang digunakan dalam penelitian ini sebagai berikut.

Tabel 1 Penggunaan Perangkat

Perangkat Keras (Hardware)	1. AMD Ryzen 3 2.6GHZ 2. Samsung A16 (media pengambilan data)
Perangkat Lunak (Software)	1. Microsoft Windows 10 Home 64 bit 2. Matrix Laboratory 2018a 3. Website Roboflow 4. Website Google Collaboratory

Penjelasan Proses Program

Pada proses ini menjelaskan terkait kode program yang digunakan berdasarkan fungsinya serta hasil dari kode program tersebut

1. Proses Segmentasi dan Resizing

Pada proses ini tombol yang digunakan untuk proses segmentasi gambar menjadi 244x244 agar menyesuaikan dengan inputan proses klasifikasi pada CNN. Perintah program mengambil gambar dari setiap batch file data yang sudah tidak ada background, kemudian akan dilakukan proses resizing pada data yang diambil.

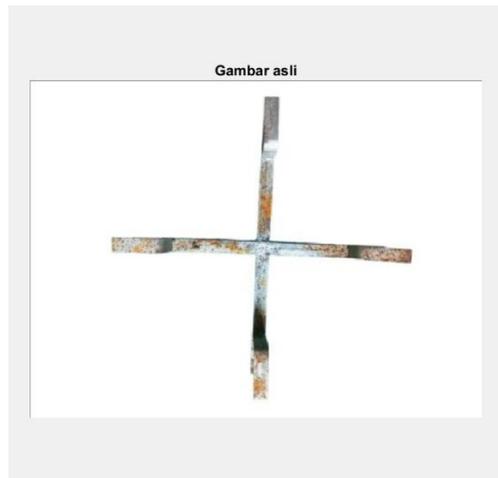
2. Program Perubahan Warna RGB to HSV

Pada proses ini gambar dikonversikan dari RGB menjadi ruang gambar HSV dengan pengkodean MatLab dengan hasil sebagai berikut:

Kode Program 1 Konversi Warna Asli ke HSV

```
% Mengonversi gambar ke ruang warna HSV
hsvImage = rgb2hsv(image);

% Untuk menampilkan hasil gambar
figure;
imshow (image)
title('Gambar asli')
```



Gambar 2 Citra asli sudah hilang background

Kode Program 2 Pengambilan Rentang Warna

```
Pengambilan rentang warna coklat dan oranye pada citra  
% Rentang warna coklat dalam HSV (Hue, Saturation, Value)  
brownMin = [0.05, 0.2, 0.2]; % Min: Hue, Saturation, Value  
brownMax = [0.15, 1, 1]; % Max: Hue, Saturation, Value  
  
% Rentang warna oranye dalam HSV (Hue, Saturation, Value)  
orangeMin = [0.05, 0.3, 0.3]; % Min: Hue, Saturation, Value  
orangeMax = [0.14, 1, 1]; % Max: Hue, Saturation, Value
```

3. Program Segmentasi Warna

Proses ekstraksi warna merupakan salah satu tahapan penting dalam pengolahan citra digital, khususnya pada aplikasi yang berkaitan dengan klasifikasi atau deteksi objek berdasarkan warna. Dalam konteks penelitian ini, proses ekstraksi warna dilakukan terhadap citra objek yang telah melalui tahap penghilangan latar belakang (background removal). Penghilangan latar belakang bertujuan untuk memfokuskan analisis hanya pada objek utama, yaitu permukaan logam atau besi yang mengalami korosi.

Warna yang menjadi fokus dalam ekstraksi ini adalah warna oranye dan coklat, yang secara visual umum diasosiasikan dengan warna karat. Tujuan dari ekstraksi kedua warna tersebut adalah untuk memisahkan area yang menunjukkan indikasi karat dari area permukaan logam yang masih bersih atau belum terkorosi. Dengan demikian, proses ini memungkinkan sistem hanya mengekstrak dan menganalisis bagian citra yang relevan, yaitu area yang mengandung karat.

Untuk melakukan ekstraksi warna, metode yang digunakan adalah pembuatan color mask atau masker warna. Masker warna merupakan teknik segmentasi berdasarkan nilai piksel tertentu dalam ruang warna yang telah ditentukan, seperti ruang warna HSV (Hue, Saturation, Value). Ruang warna HSV dipilih karena mampu memisahkan informasi warna (Hue) dari tingkat intensitas cahaya (Value), sehingga lebih efektif dalam mendeteksi warna spesifik dibandingkan dengan model warna RGB.

Tahapan proses ekstraksi warna pada citra meliputi:

1. Transformasi Citra ke Model Warna HSV
Citra RGB terlebih dahulu dikonversi ke ruang warna HSV untuk mempermudah pemisahan komponen warna.
2. Penentuan Rentang Warna Karat (Oranye dan Coklat)
Rentang nilai Hue, Saturation, dan Value yang merepresentasikan warna oranye dan coklat ditentukan secara empiris. Rentang ini disesuaikan agar mampu mencakup variasi warna karat yang muncul pada permukaan objek.
3. Pembuatan Masker Warna
Berdasarkan rentang warna yang telah ditentukan, dibuat sebuah masker biner yang menandai piksel-piksel yang berada dalam rentang warna karat sebagai 1 (putih) dan piksel lainnya sebagai 0 (hitam). Masker ini berfungsi sebagai filter untuk menyaring bagian citra yang mengandung warna karat.
4. Penerapan Masker pada Citra Asli
Masker warna kemudian diterapkan ke citra awal yang telah dihilangkan latar belakangnya. Hasilnya adalah citra baru yang hanya menampilkan area dengan warna karat, sementara bagian lain dari objek atau latar belakang tidak ditampilkan.

Proses ini sangat penting dalam rangkaian sistem klasifikasi tingkat karat, karena hanya dengan mengekstrak bagian citra yang mengandung warna karat, sistem dapat melakukan analisis lanjutan seperti perhitungan luasan karat, ekstraksi fitur, atau klasifikasi tingkat kerusakan dengan lebih akurat dan efisien. Penggunaan pendekatan berbasis warna juga mengurangi kompleksitas pemrosesan dan meningkatkan ketepatan dalam mendeteksi area korosi.

Kode Program 3 Mask Warna Coklat dan Oranye

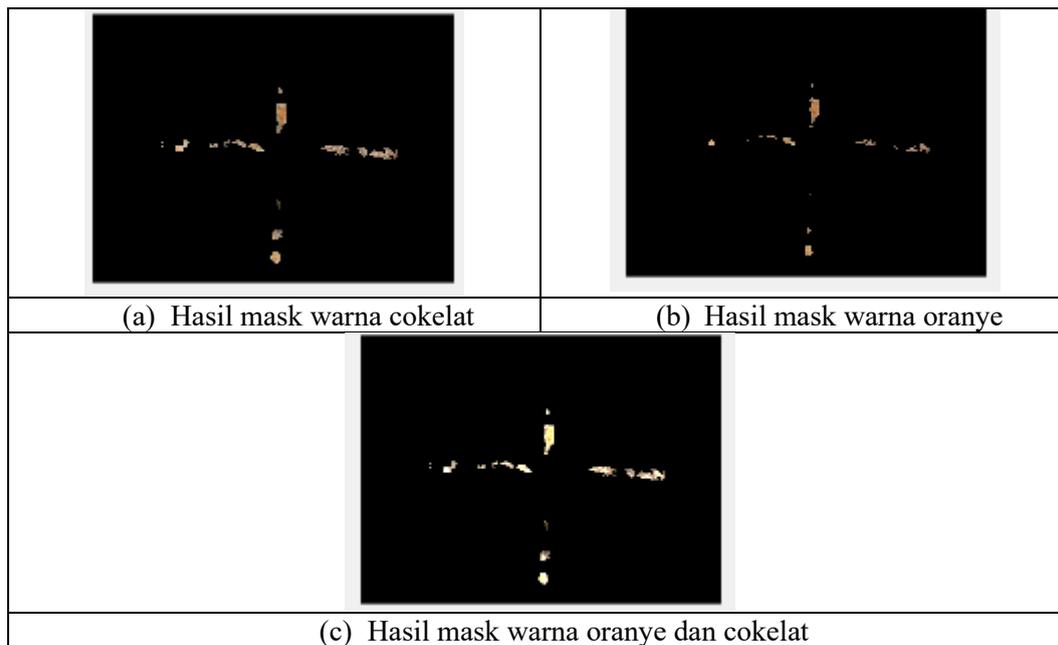
```
% Membuat mask untuk warna coklat
brownMask = (hsvImage(:,:,1) >= brownMin(1)) & (hsvImage(:,:,1) <= brownMax(1))
& ...
    (hsvImage(:,:,2) >= brownMin(2)) & (hsvImage(:,:,2) <= brownMax(2)) & ...
    (hsvImage(:,:,3) >= brownMin(3)) & (hsvImage(:,:,3) <= brownMax(3));

% Membuat mask untuk warna oranye
orangeMask = (hsvImage(:,:,1) >= orangeMin(1)) & (hsvImage(:,:,1) <=
orangeMax(1)) & ...
    (hsvImage(:,:,2) >= orangeMin(2)) & (hsvImage(:,:,2) <= orangeMax(2)) & ...
    (hsvImage(:,:,3) >= orangeMin(3)) & (hsvImage(:,:,3) <= orangeMax(3));

% Membuat gambar untuk menampilkan area coklat dengan warna RGB
brownArea = image;
brownArea(repmat(~brownMask, [1, 1, 3])) = 0; % Set selain coklat menjadi hitam

% Membuat gambar untuk menampilkan area oranye dengan warna RGB
orangeArea = image;
orangeArea(repmat(~orangeMask, [1, 1, 3])) = 0; % Set selain oranye menjadi hitam

% Menggabungkan area coklat dan oranye dalam satu gambar
combinedImage = brownArea + orangeArea; %
```



Gambar 3 Hasil Mask Warna

4. Konversi Warna HSV ke Grayscale

Konversi warna merupakan salah satu tahapan penting dalam pengolahan citra digital yang bertujuan untuk mengubah representasi suatu citra dari satu model warna ke model warna lainnya. Proses ini kerap kali diperlukan dalam berbagai tahapan analisis citra, seperti segmentasi objek, deteksi tepi, serta ekstraksi fitur. Salah satu bentuk konversi yang umum dilakukan dalam pengolahan citra adalah konversi dari model warna HSV (Hue, Saturation, Value) ke model citra grayscale (abu-abu).

Citra grayscale merupakan representasi citra digital yang hanya mengandung informasi mengenai tingkat intensitas cahaya atau kecerahan dari setiap piksel, tanpa menyertakan informasi mengenai warna. Dalam format 8-bit, nilai intensitas pada citra grayscale berkisar antara 0 hingga 255, di mana nilai 0 merepresentasikan warna hitam dan nilai 255 merepresentasikan warna putih. Konversi citra ke grayscale menjadi penting ketika informasi yang dibutuhkan dalam pengolahan citra lebih berfokus pada bentuk, struktur, atau tekstur suatu objek, dibandingkan dengan informasi warnanya.

Model warna HSV terdiri atas tiga komponen utama, yaitu:

1. Hue (H), yang menunjukkan rona atau jenis warna (misalnya merah, hijau, biru, dan sebagainya),
2. Saturation (S), yang menunjukkan tingkat kejenuhan warna atau seberapa murni warna tersebut,
3. Value (V), yang menunjukkan tingkat kecerahan atau intensitas cahaya dari warna tersebut.

Dalam proses konversi citra HSV ke grayscale, komponen Value (V) umumnya digunakan sebagai dasar utama. Hal ini disebabkan karena nilai Value secara langsung mewakili tingkat kecerahan suatu piksel, yang sejalan dengan konsep representasi dalam citra grayscale. Dengan mengekstrak komponen Value dari model HSV, dapat diperoleh citra grayscale yang menggambarkan seberapa terang atau gelap suatu piksel, tanpa mempertimbangkan unsur rona maupun kejenuhan warnanya.

Proses konversi ini sangat bermanfaat terutama dalam berbagai aplikasi pengolahan citra digital yang lebih menekankan pada analisis bentuk atau kontur objek, seperti pada

sistem deteksi tepi, sistem pengenalan pola, maupun aplikasi pemantauan berbasis visual (visual surveillance). Dalam konteks tersebut, informasi mengenai intensitas kecerahan sering kali lebih penting dibandingkan dengan informasi warna.

Selain itu, konversi dari citra warna ke grayscale juga berkontribusi terhadap efisiensi komputasi. Hal ini disebabkan karena citra grayscale hanya memiliki satu kanal data intensitas, sedangkan citra dalam model warna seperti HSV atau RGB memiliki tiga kanal. Dengan demikian, jumlah data yang harus diproses menjadi lebih sedikit, yang secara langsung dapat meningkatkan kecepatan pemrosesan, khususnya pada sistem yang bekerja secara real-time atau memiliki keterbatasan sumber daya komputasi.

Dengan memahami prinsip dasar konversi dari model warna HSV ke grayscale, proses ini dapat diimplementasikan secara optimal sesuai dengan kebutuhan aplikasi tertentu, baik dalam bidang computer vision, analisis citra medis, maupun dalam sistem pengenalan otomatis berbasis visual lainnya. Model warna HSV (Hue, Saturation, Value) memisahkan informasi warna (hue) dari intensitas (value), sehingga untuk mengubah citra HSV menjadi citra grayscale, komponen Value (V) biasanya digunakan sebagai dasar. Hal ini dikarenakan komponen Value mewakili tingkat kecerahan atau intensitas cahaya dari suatu piksel, yang sejalan dengan representasi dalam citra grayscale.

Kode Program 4 Konversi Grayscale

```
% Jika citra berwarna, konversi ke grayscale
if size(img, 3) == 3
    img = rgb2gray(img);
```



Gambar 4 Tampilan konversi citra grayscale

5. Program Potong Gambar

Pada program ini berfungsi dalam memotong gambar sehingga bagian yang tidak perlu dapat dibuang.

Kode Program 5 Program Potong Gambar

```
% Potong kiri
cout=img;
a=sum(cout);
b=find(a>0);
cout(:,1:b(1)-1)=[];
% Potong kanan
cout=fliplr(cout);
a=sum(cout);
b=find(a>0);
cout(:,1:b(1)-1)=[];
% Potong atas
cout=cout';
a=sum(cout);
b=find(a>0);
```

```

cout(:,1:b(1)-1)=[];
% Potong bawah
cout=fliplr(cout);
a=sum(cout);
b=find(a>0);
cout(:,1:b(1)-1)=[];
% Penyesuaian orientasi
cout=cout';
cout=flipud(cout);
cout=fliplr(cout);
else
cout=img;

```



Gambar 5 Citra yang sudah dipotong

6. Program Resizing

Resize merupakan salah satu proses dalam pengolahan citra digital yang bertujuan untuk mengubah dimensi (ukuran) suatu citra, baik dengan cara memperbesar (upscaling) maupun memperkecil (downscaling) jumlah piksel yang menyusun citra tersebut. Proses ini dilakukan dengan menyesuaikan resolusi citra tanpa memperhatikan konten atau informasi semantik yang terkandung di dalamnya.

Secara umum, resize digunakan untuk berbagai kebutuhan, seperti menyesuaikan ukuran citra agar sesuai dengan tampilan perangkat, mengurangi ukuran file untuk efisiensi penyimpanan dan transmisi data, hingga mempersiapkan dataset citra dalam proses pelatihan model machine learning. Dalam implementasinya, resize dilakukan dengan menggunakan metode interpolasi tertentu untuk menentukan nilai piksel baru berdasarkan piksel-piksel yang sudah ada. Program ini berfungsi untuk menyesuaikan gambar yang telah di segmentasi warna.

Kode Program 6 Resize

```

x10=imresize(cout,[244 244]);
imshow(x10)

figure;
imshow (cout)

```



Gambar 6 Hasil gambar yang sudah diresize menjadi 244x244

7. Proses Klasifikasi

Pada proses ini program yang telah dibuat menggunakan penerapan klasifikasi CNN yang akan memroses gambar dari inputan gambar yang sudah diekstraksi sehingga mendapatkan output dengan pengelompokan dari tingkat persentase karat pada *slack support*. GUI memanggil inputan yang tersedia dalam sebuah file, kemudian diproses klasifikasi dengan tombol proses dan menampilkan hasil pada figure tampilan program yang telah dibuat ditampilkan pada gambar beserta kodingan program.

Data Latih dan Data Uji

Pada sub bab ini terdapat data latih dan data uji sebagai berikut.

1. Data Latih

Data latih merupakan sekumpulan data yang digunakan untuk melatih model dalam mengenali pola atau karakteristik tertentu dari data masukan. Melalui data ini, model akan mempelajari hubungan antara fitur-fitur input dan output yang diharapkan, sehingga dapat menyusun parameter-parameter internal yang optimal. Selama proses pelatihan, model secara bertahap memperbaiki prediksi-prediksi yang dihasilkannya berdasarkan umpan balik dari data latih tersebut. Oleh karena itu, kualitas dan keragaman data latih sangat memengaruhi kemampuan model dalam menghasilkan prediksi yang akurat. Pada penelitian ini menggunakan 105 data latih yang dibagi menjadi 3 kelas seperti ringan, sedang dan berat.

Kode Program 7 Pemanggil Data Latih

```
Pemanggilan Data Latih
from roboflow import Roboflow
rf = Roboflow(api_key="rmZ06wAqMKcwx34JBSxj")
project = rf.workspace("keristian-paonsen-5sred").project("pendeteksian-karat-slack-support")
version = project.version(1)
```

Kode Program 8 Pelatihan/Training Pemodelan

```
Pelatihan/Training Pemodelan
results = model.train(data="/content/Karat-Boundary-1/data.yaml",
    epochs=100,
    imgsz=640,
    batch=8,
```

Metrik	Nilai	Penjelasan
Precision	92%	Dengan angka yang didapat untuk tingkat akurasi deteksi karat menunjukkan angka yang cukup tinggi.
Recall	98%	Menunjukkan bahwa model dapat mendeteksi dari seluruh objek karat yang seharusnya terdeteksi.
mAP0.5	95%	Nilai yang menunjukkan ketepatan model dalam seluruh kelas.
mAP0.5-0.95	94%	Nilai ketepatan dari prediksi yang lebih kecil dibandingkan Map 0.5

Gambar 7 Tabel Hasil Dari Data Latih

2. Data Uji

Data uji merupakan data yang digunakan untuk mengevaluasi kinerja model setelah proses pelatihan selesai. Data uji bersifat independen terhadap data latih dan tidak digunakan dalam proses pembelajaran model. Dengan demikian, pengujian menggunakan data uji memungkinkan penilaian objektif terhadap kemampuan model dalam menggeneralisasi atau memprediksi data baru yang belum pernah dikenali sebelumnya. Pada penelitian ini terdapat hasil pada data uji untuk akurasi sebagai berikut.

Tabel 2 Data Uji Karat

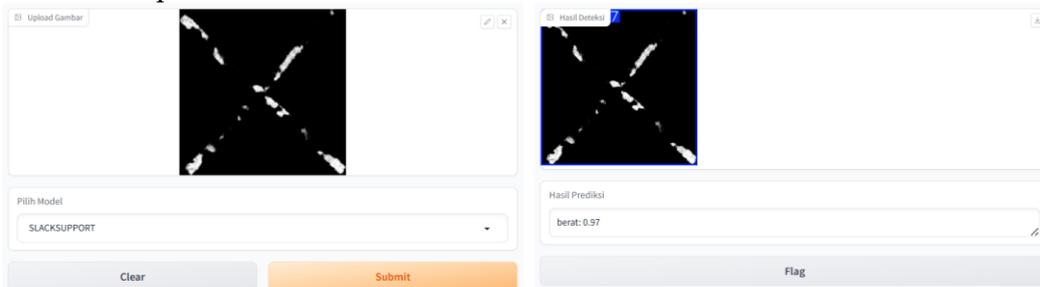
Data Karat	Jumlah Citra	Akurasi
Ringan	35	98%
Sedang	35	96%
Berat	35	98%
Total	105	293
Rata-rata		97,6%

Hasil Klasifikasi

Pada proses klasifikasi ini ditunjang dengan aplikasi *Google Colaboratory* sebagai media klasifikasi citra berkarat yang sudah di segmentasi pada aplikasi Matlab.

1. Proses Input Citra Hasil Segmentasi

Pada tahapan ini inputan citra hasil segmentasi akan diproses pada *Google Colaboratory* yang akan mengklasifikasikan jenis karat yang terdapat pada gambar hasil segmentasi citra pada Matlab.

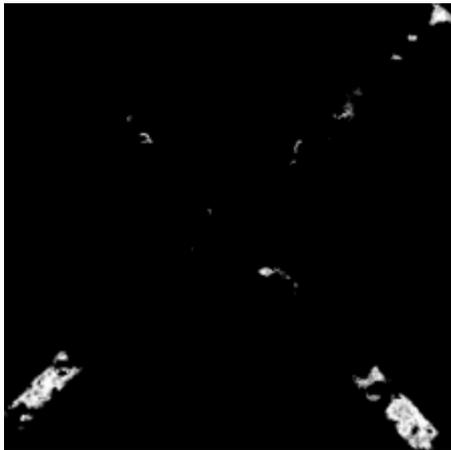


Gambar 8 Tampilan Inputan dan Output pada *Google Colaboratory*

2. Hasil Deteksi Karat

Karat pada benda besi seperti slack support umumnya dapat dilihat secara langsung namun dengan adanya program ini kita dapat menilai kelayakan besi tersebut sebagai alat ukur dalam kontrol kualitas sarana pekerjaan. Jika karat pada slack support sudah berlebihan dapat disayangkan karena akan memperpendek umur dari slack support tersebut, dengan pemeliharaan seperti pengecatan kembali akan tetapi jika karat yang sudah ada pada slack support sudah dikelas berat maka tidak akan maksimal dari segi pemakaian slack support tersebut. Dengan adanya hasil deteksi karat ini jika untuk kadar karat yang menutupi besi slacksupport dianggap tinggi maka slack support tersebut

dianggap sudah tidak layak pakai dan diperbaiki.



(a) Inputan Citra Sedang



(b) Ouput Citra Sedang

Gambar 9 Input dan Output Citra Karat Sedang

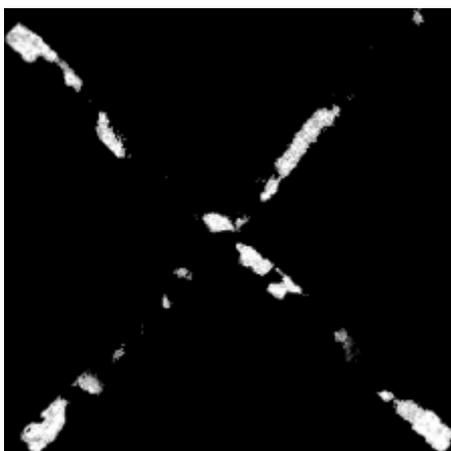


(a) Input Citra Ringan



(b) Output Citra Ringan

Gambar 10 Input dan Output Citra Karat Ringan



(a) Input Citra Berat



(b) Output Citra Berat

Gambar 11 Input dan Output Citra Karat Berat

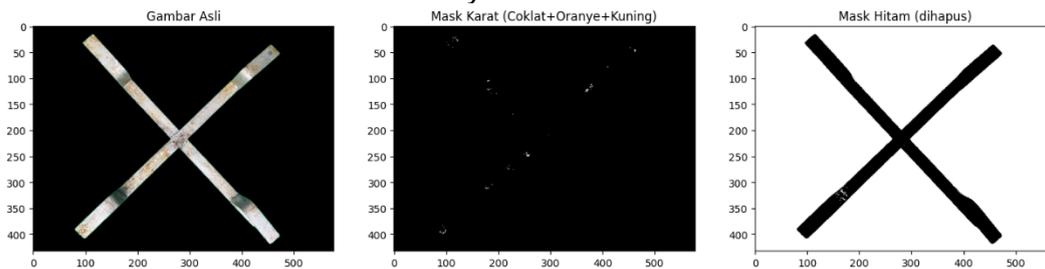
Pada gambar diatas terdapat citra input dan output hasil dari deteksi karat pada besi slack support dengan citra yang sudah di segmentasi, segmentasi sangat membantu dalam proses kelas gambar, dengan adanya segmentasi dapat memudahkan pandangan program pada proses klasifikasi dengan hilangnya background serta warna – warna lain yang tidak diperlukan dan hanya menyisakan warna karat saja.

Dengan proses masking untuk mengambil warna karat saja serta program potong gambar agar membuang area yang tidak diperlukan. Untuk output dari gambar diatas untuk akurasi dari citra ringan terdapat 99% yang menandakan tingkat keyakinan kelas untuk citra tersebut ringan, citra sedang juga mendapatkan tingkat akurasi sebesar 99% , namun untuk output hasil dari citra berat itu hanya 97% hal ini bisa dikarenakan tingkat karat pada citra berat bisa dianggap sedang karena untuk jumlah karat pada besi berat dan ringan itu tidak jauh berbeda atau ambiguitas dan disini pemrograman ini dapat berperan agar bisa memprediksi kelas karat pada slack support.

3. Hasil Persentase Karat

Pada sub-bab ini membahas citra karat menggunakan metode thresholding grayscale. Citra asli (kiri) menampilkan kondisi visual objek besi yang telah mengalami korosi sedangkan citra hasil deteksi (kanan) menunjukkan area putih yang teridentifikasi sebagai karat berdasarkan intensitas warna. Deteksi ini dilakukan secara otomatis untuk menghitung luas area berkarat. Persentase luas area yang terdeteksi sebagai karat dihitung dengan menggunakan rumus berikut:

$$\text{Persentase Karat} = \left(\frac{\text{Jumlah Piksel Karat}}{\text{Jumlah Total Piksel}} \right) \times 100\%$$

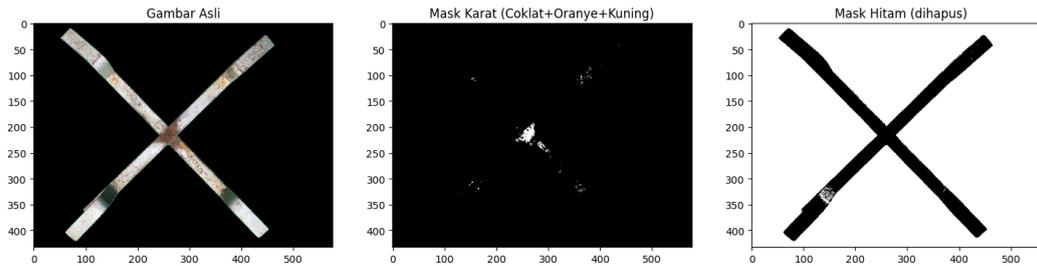


Gambar 12 Hasil Persentase Karat Contoh 1

Berdasarkan hasil pengolahan citra pada gambar di atas, diperoleh jumlah piksel warna karat sebanyak 179 piksel dari total 25.989 piksel yang terdapat dalam citra. Piksel putih tersebut menunjukkan area permukaan besi yang mengalami korosi ringan dan telah berhasil terdeteksi menggunakan metode thresholding berdasarkan intensitas warna coklat, oranye dan kuning serta mengabaikan warna hitam sebagai background. Nilai persentase area karat dihitung menggunakan persamaan:

$$\text{Persentase Karat} = \left(\frac{179}{25.989} \right) \times 100\% = 0,69\%$$

Persentase sebesar 0,69% menunjukkan bahwa area yang teridentifikasi sebagai karat sangat kecil dibandingkan dengan keseluruhan permukaan logam. Hal ini mengindikasikan bahwa kondisi korosi masih berada pada tahap awal atau ringan, di mana karat hanya menyebar di beberapa titik kecil pada permukaan besi. Hasil deteksi ini penting sebagai indikator awal dalam sistem monitoring korosi, agar tindakan pencegahan atau perawatan dini dapat dilakukan sebelum tingkat kerusakan bertambah parah. Metode pengolahan citra ini terbukti mampu mengidentifikasi keberadaan karat meskipun dalam jumlah yang sangat kecil dan tersebar.

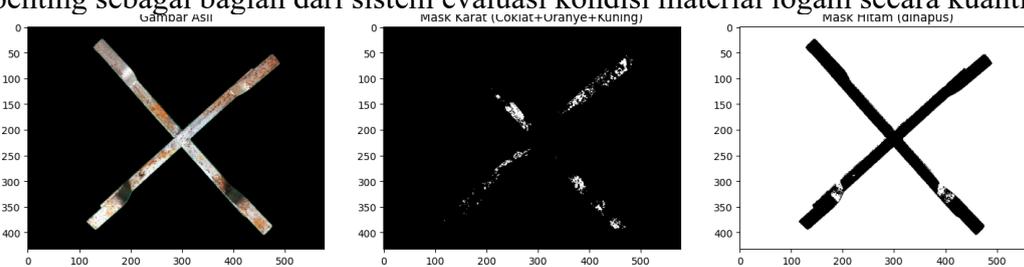


Gambar 13 Hasil Persentase Karat Contoh 2

Gambar di atas menunjukkan hasil deteksi area karat dengan tingkat keparahan sedang. Berdasarkan hasil segmentasi citra menggunakan metode thresholding, jumlah piksel putih yang terdeteksi sebagai area berkarat adalah 854 piksel, dari total 27.697 piksel dalam keseluruhan citra. Persentase luas area karat dihitung dengan rumus:

$$\text{Persentase Karat} = \left(\frac{854}{27697} \right) \times 100\% = 3,12\%$$

Persentase sebesar 3,12% ini menunjukkan bahwa karat telah menyebar ke beberapa area penting permukaan logam namun belum mendominasi secara keseluruhan. Karat tampak lebih merata dan mencakup lebih banyak wilayah dibandingkan kondisi karat sebelumnya, meskipun belum sebesar pada kondisi karat lainnya. Dengan demikian, hasil ini merepresentasikan tahap pertengahan dalam progresi korosi, di mana intervensi pemeliharaan masih memungkinkan untuk mencegah penyebaran karat lebih lanjut. Data ini penting sebagai bagian dari sistem evaluasi kondisi material logam secara kuantitatif.



Gambar 14 Hasil Persentase Karat Contoh 3

Berdasarkan hasil deteksi pada gambar di atas, teridentifikasi 2.832 piksel putih dari total 22.485 piksel dalam citra. Piksel putih tersebut merepresentasikan area permukaan besi yang mengalami korosi, yang berhasil disegmentasi menggunakan metode thresholding berbasis intensitas grayscale. Perhitungan persentase area karat menghasilkan nilai sebesar:

$$\text{Persentase Karat} = \left(\frac{2.832}{22.485} \right) \times 100\% = 12,60\%$$

Nilai 12,60% menunjukkan bahwa sebagian besar area penting pada permukaan logam telah mengalami kerusakan akibat karat. Citra hasil deteksi memperlihatkan distribusi karat yang lebih luas, menyebar di sepanjang bagian diagonal objek, serta tampak lebih padat dan merata dibandingkan pada kondisi contoh 1 dan contoh 2. Kondisi ini dapat dikategorikan sebagai tahap lanjut dari korosi, yang apabila tidak segera ditangani, dapat berdampak pada penurunan kekuatan struktural material. Oleh karena itu, hasil deteksi ini dapat digunakan sebagai dasar evaluasi bahwa objek perlu dilakukan tindakan lanjutan seperti penggantian, pelapisan ulang, atau perawatan khusus.

KESIMPULAN

Berdasarkan hasil penelitian yang telah dilakukan, maka dapat disimpulkan beberapa hal sebagai berikut:

- 1) Penelitian ini berhasil merancang dan mengimplementasikan sistem identifikasi

persentase karat pada benda logam jenis slack support. Sistem ini memanfaatkan pemrosesan citra digital yang dikombinasikan dengan teknik segmentasi warna dalam ruang warna HSV untuk mengekstrak area berkarat, dan CNN sebagai metode klasifikasi untuk menentukan tingkat keparahan karat.

- 2) Proses pengolahan citra meliputi: konversi citra RGB ke HSV, pembuatan masker warna untuk mengekstraksi warna oranye dan coklat (representasi warna karat), pemotongan area tidak relevan, resizing citra, serta konversi ke grayscale. Tahapan-tahapan ini memungkinkan sistem melakukan analisis area karat secara fokus dan efisien.
- 3) Sistem dikembangkan dengan menggunakan platform MATLAB untuk antarmuka pengguna (GUI) serta Google Colaboratory untuk pelatihan dan klasifikasi menggunakan CNN. Data latih yang digunakan sebanyak 105 citra yang dibagi ke dalam tiga kategori: ringan, sedang, dan berat. Evaluasi terhadap data uji menunjukkan tingkat akurasi klasifikasi rata-rata sebesar 97,6%, yang menunjukkan bahwa sistem memiliki performa deteksi yang tinggi.
- 4) Dengan adanya sistem identifikasi ini, proses kontrol kualitas terhadap slack support dapat dilakukan lebih objektif dan efisien. Hasil persentase karat dapat dijadikan sebagai parameter kelayakan untuk pemakaian ulang atau penggantian perangkat. Sistem ini juga dapat berperan sebagai solusi awal dalam upaya prediktif perawatan komponen logam di lingkungan kerja.

Saran

Berdasarkan hasil penelitian dan implementasi sistem identifikasi persentase karat pada slack support menggunakan metode Convolutional Neural Network (CNN), maka penulis menyampaikan beberapa saran yang dapat dijadikan pertimbangan untuk pengembangan lebih lanjut, yaitu:

- 1) Pengembangan dataset disarankan untuk memperluas jumlah dan variasi dataset citra, baik dari segi sudut pengambilan gambar, pencahayaan, maupun jenis karat. Hal ini bertujuan agar sistem dapat beradaptasi dan melakukan klasifikasi secara lebih akurat dalam kondisi nyata yang beragam.
- 2) Integrasi dengan Sistem Monitoring Otomatis dapat dikembangkan lebih lanjut dengan mengintegrasikan perangkat keras seperti kamera industri atau sistem pengambilan gambar otomatis di lingkungan pabrik atau pergudangan untuk proses inspeksi real-time.
- 3) Penelitian lanjutan dapat mengkaji arsitektur CNN yang lebih kompleks atau melakukan fine-tuning model pretrained (misalnya MobileNet, ResNet) guna meningkatkan akurasi dan efisiensi klasifikasi karat.
- 4) Penggunaan format citra yang lebih luas untuk meningkatkan fleksibilitas sistem, disarankan agar sistem dapat mendukung format citra lainnya seperti PNG, BMP, dan TIFF, tidak hanya JPEG/JPG, guna mengakomodasi kebutuhan pengguna yang lebih luas.
- 5) Optimasi komputasi dan kecepatan eksekusi untuk penggunaan di lingkungan industri, diperlukan optimasi pada proses pemrosesan citra dan klasifikasi agar sistem dapat bekerja lebih cepat tanpa mengorbankan akurasi.

DAFTAR PUSTAKA

- [1] Christin Devina. (2021). Pengertian Logam. Diakses 24 Mei 2024 dari Gramedia website <https://www.gramedia.com/literasi/pengertian-logam/>.
- [2] Wikipedia. (25 Agustus 2023). Karat. Diakses 24 Mei 2024 dari Wikipedia website <https://id.wikipedia.org/wiki/Karat>.

- [3] Sains Kimia. Ilmu Kimia Dalam Karat Besi dan Pencegahannya. Diakses 24 Mei 2024 dari Sains Kimia website <https://sainskimia.com/ilmu-kimia-dalam-karat-besi-dan-cara-pencegahannya/>.
- [4] Muchlisin Riadi. (21 April 2016). Pengolahan Citra Digital. Diakses 26 Mei dari KAJIANPUSTAKA website <https://www.kajianpustaka.com/2016/04/pengolahan-citra-digital.html>.”
- [5] A. Fatchurrachman and D. Udjulawa, “Identifikasi Penyakit Pada Tanaman Kopi Berdasarkan Citra Daun Menggunakan Metode Convolution Neural Network,” *Jurnal Algoritme*, vol. 3, no. 2, doi: 10.35957/algoritme.xxxx.
- [6] O. Virgantara Putra and J. Umami, Deteksi Jalan Berlubang Pada Citra Berkabut Menggunakan Convolutional Neural Network dan Dark Channel Prior. [Online]. Available: <https://www.researchgate.net/publication/356917170>
- [7] M. Hamsy Romario, E. Ihsanto, and T. Maya Kadarina, “Sistem Hitung Dan Klasifikasi Objek Dengan Metode Convolutional Neural Network,” vol. 11, no. 2, p. 108, 2020.
- [8] A. I. Saputra, I. Weni, U. Khaira, A. Info, and K. Kunci, “Implementasi Metode Convolutional Neural Network Untuk Deteksi Penyakit Pada Tanaman Kopi Arabika Melalui Citra Daun Berbasis Android,” vol. 4, no. 1, pp. 41–51, 2024, doi: 10.51454/decode.v4i1.231.
- [9] Y. Bili et al., “Perancangan Alat Pendeteksi Kematangan Buah Nanas Dengan Menggunakan Mikrokontroler Dengan Metode Convolutional Neural Network (CNN),” 2022. [Online]. Available: <http://ojs.fikom-methodist.net/index.php/METHOTIKA>
- [10] D. Dhelviana, T. Amelia, J. Sulaksono, and D. W. Widodo, “Program Studi Teknik Informatika,” 2023.
- [11] M. Sayyidin, H. #1, and I. Muhimmah, “Aplikasi Pendeteksi Tingkat Kematangan Pepaya menggunakan Metode Convolutional Neural Network (CNN) Berbasis Android,” 2024.
- [12] J. Vicky, F. Ayu, and B. Julianto, “Implementasi Pendeteksi Penyakit pada Daun Alpukat Menggunakan Metode CNN.”
- [13] Ratna Sulisyanti; FX Arinto Setyawan; Muhammad Komarudin, PENGOLAHAN CITRA; Dasar dan Contoh Penerapannya. 2016.
- [14] Pemrograman Matlab. Pengolahan Citra. Diakses 24 Mei 2024 dari Pemrograman Matlab website <https://pemrogramanmatlab.com/2017/07/26/pengolahan-citra-digital/>”
- [15] Advernesia. (21 September 2017). Apa Itu Matlab. Diakses 24 Mei 2024 dari Advernesia website <https://www.advernesia.com/blog/matlab/apa-itu-matlab/>.
- [16] Apa Itu Matlab. <https://www.mathworks.com/discovery/matlab-gui.html>.
- [17] Qolbiyatul Lina (2 Januari 2019). Apa Itu Convolutional Neural Network. Diakses 25 Mei 2024 dari Medium website <https://akademi-ai.medium.com/apa-itu-convolutional-neural-network-dan-cara-kerjanya-691464160e9a>.
- [18] Trivusi. (17 September 2022). Jenis – jenis Operasi Pooling pada Algoritma CNN. Diakses 26 Mei 2024 dari Trivusi website <https://www.trivusi.web.id/2022/07/jenis-jenis-pooling.html>.