

## DEEP LEARNING UNTUK MENDETEKSI OBJEK BUAH DENGAN ALGORITMA CONVOLUTIONAL NEURAL NETWORK (CNN) MENGGUNAKAN MATLAB

**Mahdalena, Laila Qadriah, cut lilis setiawati**

Universitas Jabal Ghafur, Program Studi Teknik Informatika

Email: [mahda2219@gmail.com](mailto:mahda2219@gmail.com),

### ABSTRACT

At present, technological developments are increasingly sophisticated. One of the technological problems at this time, especially in computer vision, which is being sought for a solution and continues to be developed, namely regarding the classification of objects in images or images in general. The problem that is being sought for a solution is how to duplicate human abilities in understanding and detecting information in the images they see, so that they can be applied to computers so that they can recognize image objects like humans. This is difficult to apply to computers because the detection of images has various differences for each image object, for example the object's point of view, differences in scale, lighting conditions, and so on. In this research will produce a system that is able to classify the types of bananas. The system to be built implements a neural network, namely the Convolutional Neural Network (CNN) using MATLAB. The results obtained in this study are, Image Processing using banana image data can detect what type of banana is input, the Matlab system has succeeded in recognizing the similarity of banana images to perform the computational process well with the Convolutional Neural Network (CNN) algorithm model. The level of similarity or accuracy value in the detection of banana images, have different levels of accuracy. The level of accuracy in each image reaches more than 50%, where the level of similarity is well read by the system.

Keywords: Convolutional Neural Network (CNN), banana image, system, Matlab

### ABSTRAK

Pada masa sekarang ini perkembangan teknologi semakin canggih. Salah satu permasalahan teknologi pada masa ini terutama pada visi komputer yang lagi dicari solusi dan terus dikembangkan yaitu tentang klasifikasi objek pada gambar atau citra secara umum. Permasalahan yang sedang dicari solusi adalah bagaimana cara menduplikasikan kemampuan manusia dalam memahami dan mendeteksi informasi pada gambar yang dilihatnya, agar dapat diterapkan ke komputer supaya dapat mengenali objek gambar seperti manusia. Hal tersebut sulit diterapkan pada komputer karena pendeteksian pada gambar memiliki berbagai perbedaan pada setiap objek gambar, misalnya sudut pandang objek, perbedaan skala, kondisi pencahayaan, dan sebagainya. Dalam penelitian ini nantinya akan menghasilkan sistem yang mampu mengklasifikasikan jenis-jenis pisang. Sistem yang akan dibangun ini menerapkan jaringan syaraf yaitu *Convolutional Neural Network* (CNN) dengan menggunakan MATLAB. Hasil yang didapatkan dalam penelitian ini yaitu, *Image Processing* dengan menggunakan data gambar pisang dapat mendeteksi jenis pisang apa yang diinput, sistem Matlab berhasil melakukan pengenalan kemiripan gambar pisang melakukan proses komputasi dengan baik model algoritma *Convolutional Neural Network* (CNN). Tingkat kemiripan atau nilai akurasi dalam pendeteksian gambar pisang, memiliki tingkatan akurasi yang berbeda. Tingkat akurasi pada setiap *image* mencapai 50% lebih, dimana tingkat kemiripan terbaca dengan baik oleh sistem.

Kata Kunci: *Convolutional Neural Network* (CNN), *image* pisang, sistem, Matlab

## PENDAHULUAN

Pada masa sekarang ini perkembangan teknologi semakin canggih. Salah satu permasalahan teknologi pada masa ini terutama pada visi komputer yang lagi dicari solusi dan terus dikembangkan yaitu tentang klasifikasi objek pada gambar atau citra secara umum. Permasalahan yang sedang dicari solusi adalah bagaimana cara menduplikasikan kemampuan manusia dalam memahami dan mendeteksi informasi pada gambar yang dilihatnya, agar dapat diterapkan kekomputer supaya dapat mengenali objek gambar seperti manusia. Hal tersebut sulit diterapkan pada komputer karena pendeteksian pada gambar memiliki berbagai perbedaan pada setiap objek gambar, misalnya sudut pandang objek, perbedaan skala, kondisi pencahayaan, dan sebagainya.

Menurut Hinton G. E., (2018) *Deep learning* sedang menjadi topik yang diteliti dan dikembangkan dengan sistem *machine learning*. Hal ini dikarenakan *deep learning* telah mencapai tingkat keberhasilan yang tinggi dalam mencapai tujuan komputer. *Deep learning* merupakan *machine learning* yang pembuatannya terinspirasi dari korteks manusia yang sistem kerjanya seperti jaringan saraf manusia yang nantinya akan bekerja pada *hidden layer* yang akan menklasifikasikan data inputan sesuai dengan apa yang dilihat dengan mata. Metode yang nantinya digunakan adalah metode *Convolutional Neural Network* (CNN). Metode CNN ini adalah metode yang dirancang untuk menutupi kesalahan dan kelemahan dari metode sebelumnya. Terdapat beberapa kelemahan dalam metode sebelumnya, tetapi dengan menggunakan model ini sejumlah parameter bebas nantinya dapat dikurangi dan perubahan bentuk gambar input seperti, translasi, rotasi, dan skala dapat diatasi permasalahannya.

Prinsip-prinsip dalam sistem pengenalan sebuah gambar pisang yaitu dengan membandingkan satu gambar pisang dengan database pisang lainnya, sehingga menghasilkan pendekatan dan kecocokan gambar pisang. Melihat perkembangannya,

masih ada beberapa masalah dalam proses pengenalan sebuah gambar, terutama pada gambar objek seperti pisang. Kondisi gambar pisang yang dijadikan *inputan* kedalam sistem nantinya akan mengalami beberapa permasalahan yang penting ketika mempengaruhi tingkat keakuratan sistem dalam menganalisa sebuah pisang, contohnya seperti pencahayaan, corak dalam pisang, dan perubahan atribut warna. Penelitian ini akan fokus terhadap pendeteksian gambar pisang.

Untuk menerapkan sistem kerja dari perangkat yang digunakan bagus, maka sebuah komputer atau sistem harus dapat mengontrol suatu objek dalam mengidentifikasi sebuah objek atau gambar. Kemampuan inilah yang nantinya akan ditiru oleh sistem komputer. Nantinya ketika sebuah komputer menerima inputan atau masukan dari luar yang berupa objek atau gambar kemudian akan diproses, diidentifikasi dan menghasilkan output atau keluaran yang berupa deskripsi objek dalam cerita.

Oleh karena itu pada penelitian ini akan dibuat sebuah program komputerisasi yang dapat mengenal dan mengidentifikasi sebuah objek yang berupa gambar pisang dengan menggunakan metode CNN yang pendekatannya secara *deep learning*, maka penelitian ini berjudul “*Deep Learning* untuk Mendeteksi Objek Buah dengan Algoritma *Convolutional Neural Network* (CNN) menggunakan MATLAB”.



## PERANCANGAN SISTEM

Pada implementasi ini peneliti akan memperlihatkan tampilan yang digunakan pada program dan cara menjalankannya, kemudian peneliti akan membahas mengenai fungsi-fungsi yang digunakan serta metode yang diterapkan. Dalam penelitian ini data merupakan aspek terpenting, dimana peneliti harus mengumpulkan data-data yang diperlukan untuk dapat mengetahui hasil penelitiannya. Berikut adalah penjelasan mengenai bahan dan data yang digunakan dalam penelitian ini, yaitu:

Dalam proses pengklasifikasian buah pisang, peneliti perlu mengumpulkan

banyak data tentang objek buah pisang untuk tujuan dapat mengklasifikasikan objek buah pisang. Objek buah pisang ini dikumpulkan dalam bentuk foto (*image*) yang mana terdiri dari 2 jenis buah pisang. Bentuk atau jenis buah pisang yang berupa foto (*image*) adalah buah pisang jenis raja dan mas yang disimpan dalam format JPG.

Tabel 3.1. Contoh Data

No.	Gambar	Jenis
1.		Mas
2.		Raja

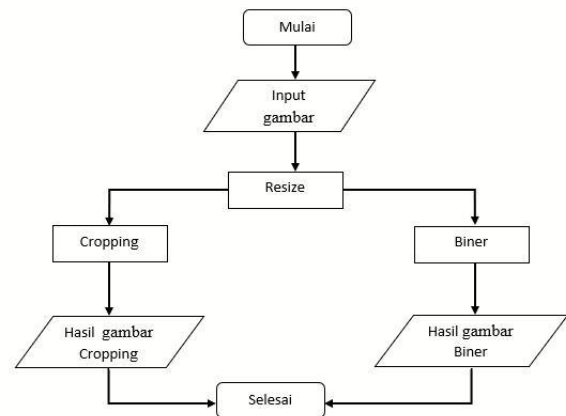
Jumlah data yang akan digunakan untuk mengkalasifikasi objek buah pisang adalah sebanyak 100 data foto pisang. Foto buah pisang yang diambil terdiri atas 2 jenis buah pisang, dimana setiap jenisnya akan diambil 50 foto sebagai data. Sehingga 2 x 50 maka didapat 100 data. Jumlah yang digunakan sebagai *testing* dan *training* sebanyak 90 dan sebagai data tunggal sebanyak 10.

Data buah pisang yang digunakan dalam klasifikasi jenis pisang diambil di beberapa pasar dan tempat-tempat jualan yang ada disekitaran Panteraja. Ada banyak pasar dan tempat-tempat jualan didaerah Panteraja, salah satunya pasar Panteraja.

Banyaknya data yang digunakan diperoleh dari cara mendapatkan dan mengumpulkan data tersebut. Data berupa foto buah pisang ini didapatkan dengan cara seurvey ke beberapa pasar, kemudian membeli buah pisang sesuai jenis yang

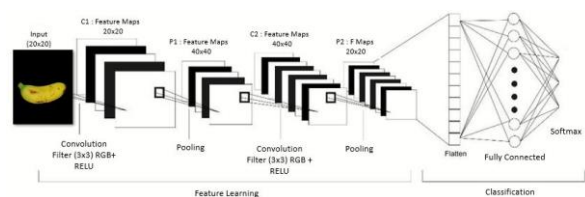
diperlukan. Lalu dilakukan pengambilan foto secara manual foto dengan kamera REDMI 4 dengan kamera 13 MP. Jarak pengambilan foto dilakukan setinggi 20 cm. Pengambilan data berupa foto buah pisang itu sendiri dilakukan setiap hari dengan membeli 1 sisir buah pisang setiap harinya. Biasanya 1 sisir berisi 10 buah pisang untuk 1 jenis saja dan jumlah data yang harus dikumpulkan ialah 100. Sehingga  $100/10 = 10$ , itu berarti jumlah data akan terkumpul kurang lebih selama 2 minggu.

Berikut adalah alur proses dari pengolahan data image pisang. Langkah awal untuk menentukan klasifikasi jenis buah pisang maka peneliti menggunakan aplikasi matlab. Setelah itu baru kemudian gambar pisang yang telah ada disamakan ukurannya, dengan menyamakan ini akan mempermudah melakukan ekstrasi ciri dan klasifikasi.



Gambar 3.1 Flowchart Pemrosesan Data Pisang

Pada gambar diatas, gambar pisang menjadi gambar inputan, yang telah diambil kemudian akan disamakan ukurannya dengan cara resize. Setelah di resize kemudian akan di cropping yang bertujuan untuk menghitung statistik warna pada citra. Sedangkan binerisasi dilakukan untuk mencari nilai nilai matrik warna yang ada pada gambar pisang yang telah diinput.



Gambar 3.2 Pemodelan Algoritma CNN

Pada gambar di atas memperlihatkan pemodelan jaringan pada proses *training* yang akan menghasilkan model yang sederhana tetapi akurat. Input gambar yang digunakan dalam penelitian ini berukuran sama dengan ukuran 20x20. Pemodelan jaringan CNN dapat dijelaskan sebagai berikut:

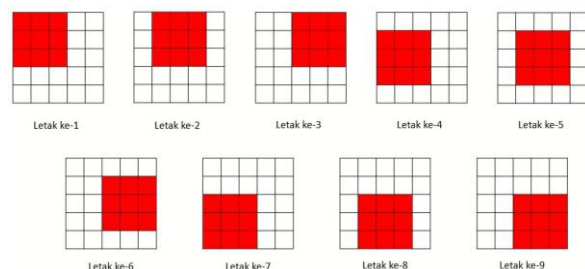
1. Tahap konvolusi yang pertama digunakan 3 lapisan layer dengan jumlah filter 20. Tahapan ini yaitu proses kombinasi antara dua buah matriks yang berbeda untuk menghasilkan nilai matriks yang baru. Setelah tahap konvolusi selesai, maka akan ditambahkan sebuah aktivasi baru yaitu RELU (*Retrified Linear Unit*). RELU ini berfungsi untuk mengubah nilai negatif menjadi nol (menghilangkan nilai negatif pada matriks hasil konvolusi). Nilai *padding* yang digunakan dalam tahapan konvolusi adalah 0, sehingga hasil ukuran dari konvolusi tetap sama yaitu 20x20.
2. Menurut Novianti Pradnyani, (2017) Tahapan *pooling* merupakan sebuah tahapan pengurangan ukuran matriks dengan memakai operasi *pooling*. *Pooling layer* terdiri atas sebuah *filter* yang ukuran nilai matriksnya secara bergantian akan bergeser pada area *feature maps*. Hasil dari tahapan *pooling* akan menghasilkan nilai matriks yang baru, karena dalam penelitian ini akan menggunakan aktivasi *maxpooling*. Cara kerja dari *maxpooling* adalah dengan mengambil nilai maksimum yang berdasarkan pergeseran karnelnya.
3. Tahap konvolusi yang kedua yaitu meneruskan hasil dari tahapan *pooling* pertama, yaitu dengan inputan matriks gambar yang berukuran 40x40 dengan jumlah *filter* 20. Pada tahapan ini juga menggunakan aktivasi RELU.
4. Tahapan selanjutnya yaitu *pooling* kedua, yang mana proses tahapannya sama dengan *pooling* yang pertama. Perbedaan antara keduanya hanya pada hasil akhir matriksnya. *Pooling* kedua

menghasilkan *output* yang berukuran 20x20.

5. Setelah itu masuk ke tahapan *flatten* atau *fully connected*. Tahapan ini hanya terdiri dari satu *hidden layer*. Tahapan *flatten* mengubah *output pooling layer* menjadi *vector*. Lalu selanjutnya tahapan *flatten* akan melakukan proses pendeteksian gambar.
6. Tahapan yang terakhir adalah penggunaan aktivasi fungsi *softmax*. Fungsi ini biasanya digunakan dalam deteksi *multiclass linear discriminant analysis* dan *multinomial logistic regression*.

### Proses Perhitungan Matriks secara Manual

Agar proses konvolusi dapat dipahami, maka peneliti akan mengambil sampel dari sebagian matriks yang ada pada *input image*. Ukuran *input image* yang digunakan sebesar 20x20 *pixel*. Maka dari itu peneliti hanya menggunakan sebagian matriks dari *input image* untuk dijadikan sampel yang akan digunakan pada proses konvolusi. Visualisasinya adalah sebagai berikut:

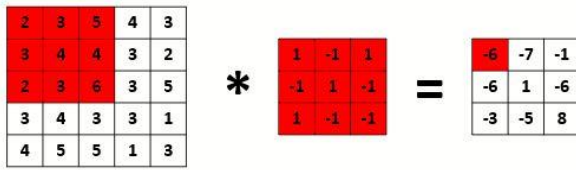


Gambar 3.3 Posisi karnel dalam konvolusi

Gambar di atas menunjukkan perhitungan dari *dot product* yang terdapat dalam proses konvolusi dengan ukuran karnelnya 3x3. Nilai *padding* yang digunakan adalah 1, agar terjadinya penambahan nilai 0 dinilai matriks inputan, yang nantinya berguna untuk menghasilkan nilai matriks *input* dan *output* yang sama.

Setelah proses konvolusi selesai, maka akan dilanjutkan dengan proses *pooling layer*. *Pooling layer* ini digunakan untuk menghilangkan nilai negatif yang ada pada

hasil matriks, dengan menambahkan aktivasi fungsi RELU (*Rectified Linear Unit*).



Gambar 3.4 Proses Konvolusi

Pada gambar 3.4 di atas, *stride* yang digunakan pada kernel adalah 1 dengan ukuran 3x3. *Stride* ini memperlihatkan pergeseran jumlah kernel dalam matriks yang bernilai satu. Nilai *padding* yang digunakan adalah 1, agar terjadinya penambahan nilai 0 dinilai matriks inputan, yang nantinya berguna untuk menghasilkan nilai matriks *input* dan *output* yang sama. Perhitungan dari *dot product* adalah sebagai berikut:

$$Position_1 = (2*1) + (3*(-1)) + (2*1) + (3*(-1)) + (4*1) + (3*(-1)) + (5*1) + (4*(-1)) + (6*(-1)) = -6$$

$$Position_2 = (3*1) + (4*(-1)) + (3*1) + (5*(-1)) + (4*1) + (6*(-1)) + (4*1) + (3*(-1)) + (3*(-1)) = -7$$

$$Position_3 = (5*1) + (4*(-1)) + (6*1) + (4*(-1)) + (3*1) + (3*(-1)) + (3*1) + (2*(-1)) + (5*(-1)) = -1$$

$$Position_4 = (3*1) + (2*(-1)) + (3*1) + (4*(-1)) + (3*1) + (4*(-1)) + (3*1) + (2*(-1)) + (5*(-1)) = -6$$

$$Position_5 = (4*1) + (3*(-1)) + (4*1) + (4*(-1)) + (6*1) + (3*(-1)) + (3*1) + (3*(-1)) + (3*(-1)) = 1$$

$$Position_6 = (4*1) + (6*(-1)) + (3*1) + (3*(-1)) + (3*1) + (3*(-1)) + (2*1) + (5*(-1)) + (1*(-1)) = -6$$

$$Position_7 = (2*1) + (3*(-1)) + (4*1) + (3*(-1)) + (4*1) + (5*(-1)) + (6*1) + (3*(-1)) + (5*(-1)) = -3$$

$$Position_8 = (3*1) + (4*(-1)) + (5*1) + (6*(-1)) + (3*1) + (5*(-1)) + (3*1) + (3*(-1)) + (1*(-1)) = -5$$

$$Position_9 = (6*1) + (3*(-1)) + (5*1) + (3*(-1)) + (3*1) + (1*(-1)) + (5*1) + (1*(-1)) + (3*(-1)) = 8$$

Setelah proses konvolusi selesai, maka akan dilanjutkan dengan proses *pooling layer*. *Pooling layer* ini digunakan untuk menghilangkan nilai negatif yang ada pada hasil matriks, dengan menambahkan aktivasi fungsi RELU (*Rectified Linear Unit*).

*Pooling* adalah pengurangan langkah dalam matriks nilai dengan menggunakan proses penggabungan operasi. Dalam proses *pooling* ini biasanya menggunakan metode *maxpooling*. Metode ini biasanya digunakan dalam penelitian yang berkaitan dengan *deep learning*. Penjelasan Kadir dalam jurnalnya menyatakan bahwa *pooling layer* merupakan bagian dari *feature learning* pada CNN yang berperan untuk mengurangi dimensi dari matriks hasil konvolusi (*convolved feature*). Tujuan dari penggunaan *pooling layer* adalah untuk mempercepat proses komputasi. Hal ini dapat terjadi karena setelah melewati *pooling layer*, parameter yang harus di-update semakin sedikit, sehingga resiko terjadinya *overfitting* semakin minim. Sama seperti pada *convolution layer*, *pooling layer* juga memiliki filter dengan ukuran tertentu yang akan melakukan proses sliding window terhadap matriks input. Terdapat dua jenis *pooling* yang sering digunakan, yakni *max pooling* dan *average pooling*.

Ukuran *pooling* yang dipakai pada penelitian ini berukuran 2x2 dengan penggunaan *stride* 1, dimana pergeseran kernel pada *input* matriks berjumlah satu.



Dalam proses *pooling* digunakan *maxpooling*, yang nantinya akan bergeser sesuai dengan *stride* dan ukurannya, yang bertujuan untuk mendapatkan nilai paling maksimum. Seperti pada gambar di atas *output* yang dihasilkan berasal dari nilai maksimum yang diambil dari matrik *feature map* hasil konvolusi. *Output* dari *maxpooling* berukuran 2x2.

Menurut Novianti Pradnyani, (2017) Proses *fully connected* adalah proses yang hasilnya masih dalam bentuk *multidimensional array*. Karena bentuk hasilnya masih dalam bentuk *array*, maka perlu ditambahkan fungsi *flatten*. Fungsi *flatten* ini akan membuat hasil dalam bentuk *array* menjadi bentuk *vector*, yang nantinya akan dapat digunakan sebagai *input* dari *fully connected layer*. Setelah penggunaan fungsi *flatten* ditambahkan, selanjutnya akan ditambahkan fungsi *dense*. Fungsi ini akan menambahkan *layer* pada *fully connected* yang akan dijadikan sebagai *layer classification*. *Fully connected layer* sering digunakan pada *multi layer perceptron* dengan tujuan untuk melakukan transformasi pada dimensi data agar data-data tersebut dapat diklasifikasikan secara linear. Input pada *fully connected layer* adalah data dari *feature learning*. Data dari proses *feature learning* yang merupakan input pada Tanpa *Dropout* dengan *Dropout 15 layer* ini adalah data dalam bentuk vektor, yang sebelumnya telah diproses oleh *flatten*.

Pada proses *classification* nantinya akan melakukan penentuan bagian-bagian mana yang akan dipilih pada setiap piksel akan dibentuk menjadi pola ataupun bentuk pisang yang sesuai dari *input image* yang telah ada. Dalam penelitian ini peneliti menguji gambar pisang dengan 2 jenis pisang, yaitu pisang mas dan pisang raja, yang nantinya apakah gambar *input image* yang pisang dapat dideteksi dengan baik.

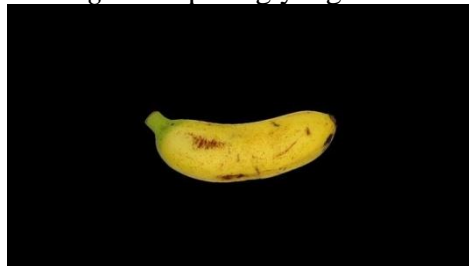
### IMPLEMENTASI SISTEM

Tahapan-tahapan dalam mengolah data pisang adalah sebagai berikut:

#### 1. Resizing Data

Setelah data *image* pisang terkumpul, maka tahapan selanjutnya adalah melakukan *resizing* terhadap semua *image*

ukuran pisang. Tahapan ini dilakukan untuk menyamakan ukuran *image* data pisang yang telah terkumpul. Ukuran *image* yang digunakan adalah (225, 400) 225 sebagai lebar dan 400 sebagai panjang. Berikut adalah *image* data pisang yang telah di *resize*:



Gambar 4.1 Pisang Mas yang telah di resize

#### 2. Cropping Image

Tahapan ini dilakukan setelah tahapan *resizing*. Dalam tahapan ini data *image* pisang akan diseleksi objeknya yang hanya memiliki ukuran warna RGB (*Red, Green, Blue*). Tahapan ini dilakukan supaya sistem dapat menentukan perbedaan.



Gambar 4.2 Pisang Mas yang telah di cropping

Data pisang yang digunakan penulis pada penelitian ini yakni terdiri dari 2 jenis pisang, jenis-jenis pisang tersebut ialah pisang mas dan pisang raja.



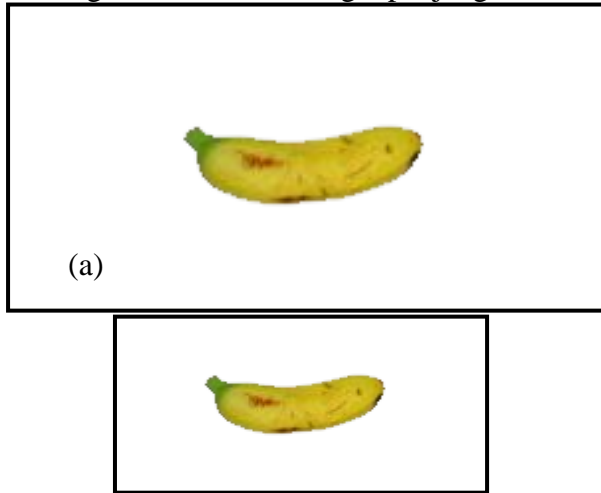
Gambar 4.3 Data Pisang Mas



Gambar 4.4 Data Pisang Raja

*Resizing Image Pisang*

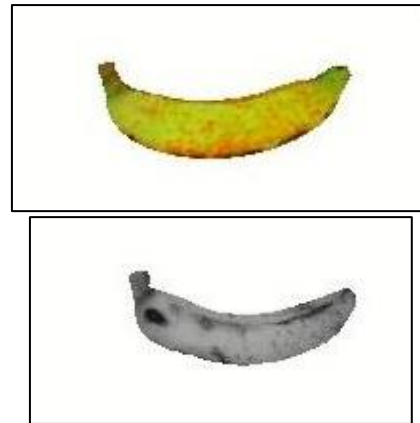
Tahap selanjutnya yakni *resizing* semua ukuran pisang. *Resizing* dilakukan untuk menyamakan semua ukuran pisang, sehingga nantinya data pisang mempunyai ukuran yang sama secara merata. Ukuran *image* pisang yang digunakan ialah [2, 4] cm, 2 sebagai lebar dan 4 sebagai panjang.



Gambar 4.5 (a) Sebelum diresize (b) Sesudah diresize

*Konversi citra RGB ke citra Grayscale*

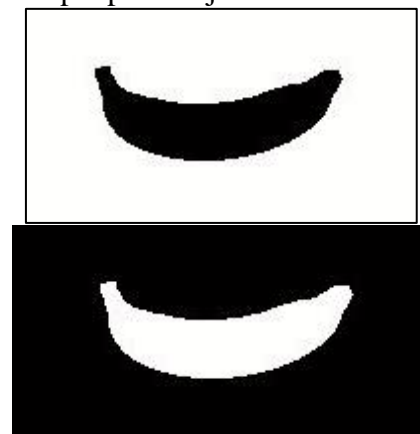
Citra RGB ini merupakan sebuah citra yang tersusun oleh tiga warna yaitu merah, hijau dan biru. Pada citra RGB 24-bit masing-masing warna memiliki nilai piksel sendiri, yang mana nilainya dari 0 sampai dengan 255. Jadi setiap piksel pada RGB memiliki nilai kombinasi dari R, G dan B yaitu sebanyak  $256 \times 256 \times 256 = 16.777.216$ . Sedangkan citra *grayscale* merupakan citra yang hanya memiliki satu warna saja. Nilai piksel pada citra *grayscale* adalah representasi dari sudut keabuan dimana nilai 0 menyatakan warna hitam sempurna dan nilai 255 warna putih sempurna. Konversi *image* ini dilakukan supaya *image* hanya memiliki satu warna nilai piksel saja.



Gambar 4.6 Konversi citra rgb ke citra grayscale

*Konversi citra Grayscale ke citra Biner*

Citra biner merupakan citra yang pikselnya memiliki bit sebesar 1, sehingga hanya memiliki dua nilai warna, yaitu nilai 0 (hitam) dan nilai 1 (putih). Konversi citra *grayscale* ke citra biner ini melalui proses *thresholding*. Dalam proses *thresholding* dibutuhkan suatu nilai *threshold* sebagai nilai pembatas konversi. Jika nilai piksel pada *image* lebih besar atau sama dengan nilai *threshold* maka akan dikonversikan menjadi 1. Sedangkan jika nilai piksel kurang dari nilai *threshold* maka akan di konversikan menjadi 0. Tahapan konversi ini dilakukan untuk memisahkan objek dengan *background* yang terdapat pada objek.



Gambar 4.7 Konversi citra grayscale ke citra biner

*Identifikasi Objek*

Tahapan identifikasi objek *image* ini dilakukan untuk input yang dimasukkan

nantinya dapat diketahui oleh sistem berapa luas dari objek *image* yang akan dibaca dan kemudian akan dideteksi. Pada pengukuran luas objek ini kita perlu menjalankan *method* yang nantinya dapat mengetahui luas objek.  
Data Uji

Data uji ini adalah data yang akan menjadi keluaran untuk pengujian dan nantinya akan menjadi keluaran hasil, setelah program untuk pemanggilan citra sesuai model selesai diprogram.

**Hasil Output**

Hasil *output* adalah hasil keluaran yang dilakukan oleh sistem. Dimana program yang telah dibuat telah berhasil dilakukan dan dijalankan. Pada pengolahan citra/*image* ini menampilkan hasil keluaran yang berupa Pisang Mas dan Pisang Raja. Berikut adalah hasil keluaran yang ditampilkan.

1. *Figure 1*



Gambar 4.8 Hasil Output 1

Pada gambar 4.8 hasil output 1, adalah menampilkan keluaran data gambar pisang yang telah dilatih agar dapat dibaca oleh sistem Matlab. Dimana hasil keluarannya adalah sistem Matlab berhasil mendeteksi gambar pisang Mas.

2. *Figure 2*



Gambar 4.9 Hasil Output 2

Pada gambar 4.9 hasil output 2, adalah menampilkan keluaran data gambar pisang yang telah dilatih agar dapat dibaca oleh sistem Matlab. Dimana hasil keluarannya adalah sistem Matlab berhasil mendeteksi gambar pisang Mas.

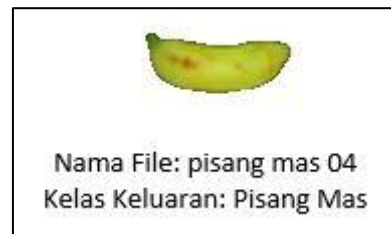
3. *Figure 3*



Gambar 4.10 Hasil Output 3

Pada gambar 4.10 hasil output 3, adalah menampilkan keluaran data gambar pisang yang telah dilatih agar dapat dibaca oleh sistem Matlab. Dimana hasil keluarannya adalah sistem Matlab berhasil mendeteksi gambar pisang Mas.

4. *Figure 4*



Gambar 4.11 Hasil Output 4

Pada gambar 4.11 hasil output 4, adalah menampilkan keluaran data gambar pisang yang telah dilatih agar dapat dibaca oleh sistem Matlab. Dimana hasil keluarannya adalah sistem Matlab berhasil mendeteksi gambar pisang Mas.

**Hasil Akurasi**

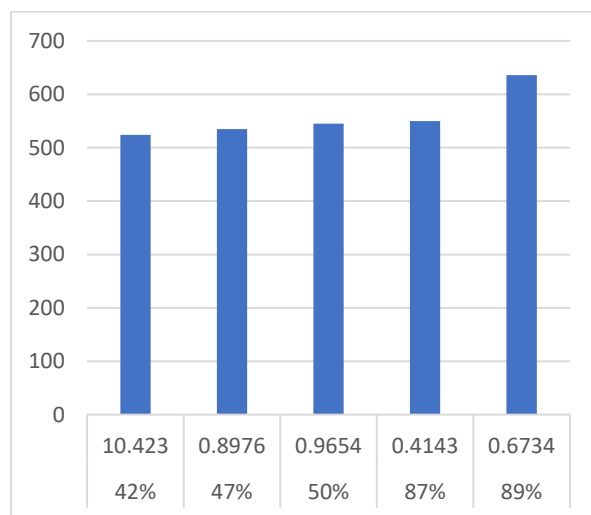
Hasil tingkat akurasi langsung dapat dilihat dari Matlab. Dimana nilai-nilai dari hasil matriks *image* pisang harus dianalisis kembali. Hasil ini akan sangat berpengaruh terhadap tingkat keberhasilan program yang telah dijalankan. Berikut adalah tabel hasil



akurasi dari masing-masing nilai *learning rate* yang telah dimasukkan.

Tabel 4.1 Hasil Akurasi

Accuracy Validation	Loss Validation	Time (Minute)
42%	0.9583	524
47%	0.1824	535
50%	0.0964	545
87%	0.9314	550
89%	0.0487	636



Gambar 4.16 Hasil Akurasi

Dari hasil grafik diatas, dapat dilihat bahwa tingkat akurasi tertinggi dicapai pada saat waktu 636 detik, dimana tingkat kegagalan yang dibaca oleh sistem dari gambar pisang yang diinput adalah sebesar 0.6734, dan tingkat akurasi yang dihasilkan sebesar 89%. Sedangkan pada waktu 524 detik, tingkat kegagalan yang dibaca oleh sistem dari gambar pisang yang diinput adalah sebesar 10.423, dan tingkat akurasi yang dihasilkan sebesar 42%. Jadi dari hasil tersebut dapat dilihat bahwa tingkat akurasi yan dapat dibaca oleh Matlab dengan inputan berupa gambar pisang adalah sebesar 89%.

## KESIMPULAN DAN SARAN

Dalam hasil penelitian tentang klasifikasi jenis buah pisang dengan *Image Processing* menggunakan algoritma *Convolutinal Neural Network* (CNN), dapat disimpulkan beberapa hal yaitu:

1. Perancangan sistem deteksi *image* pisang yang digunakan untuk mengolah data *image* pisang mas dan pisang raja adalah perancangan yang dilakukan secara *deep learning* dengan menggunakan metode *Convolutional Neural Network* (CNN).
2. Pengimplementasian algoritma CNN, *image* pisang harus melalui beberapa proses ekstrak data yaitu dari *image* rgb diubah dalam bentuk *grayscale*, kemudian diubah menjadi dalam bentuk biner. *Image* yang melalui beberapa tahapan pengubahan matrik tersebut berhasil dilakukan dengan mendapatkan nilai minimal 0 dan nilai maksimal 1.
3. Tingkat akurasi pada setiap *image* mencapai 50% lebih, dimana tingkat kemiripan terbaca dengan baik oleh sistem. Tingkat kemiripan atau nilai akurasi dalam pendeteksian gambar pisang, memiliki tingkatan akurasi yang berbeda. *Image Processing* dengan menggunakan data gambar pisang dapat mendeteksi jenis pisang apa yang diinput. Yang mana sistem Matlab dalam pengenalan kemiripan gambar pisang melakukan proses komputasi dengan model algoritma *Convotional Neural Network* (CNN).

Dari hasil yang telah diuji dalam penelitian ini, peneliti memberikan beberapa saran untuk penelitian yang lebih lanjut dalam konteks yang sama, yaitu sebagai berikut:

1. Menambahkan jumlah dataset dan menambahkan objek gambar lain untuk melatih model CNN yang ada serta dapat mencapai akurasi yang tinggi dengan menggunakan *software* Matlab.
2. Menambahkan objek gambar dengan objek yang banyak dan berbeda dan

dengan menggunakan tingkat level warna yang lebih banyak.

#### **DAFTAR PUSTAKA**

Eaton, John W, 2017. *Artificial Intelligence*. Binus University.

Nurfita Royani Darma, Ariyanto Gunawan. 2018. “*Implementasi Deep Learning Berbasis Tensorflow Untuk Pengenalan Sidik Jari*”. Program Studi Informatika Universitas Muhammadiyah Surakarta (UMS) Surakarta, Indonesia. Vol. 5 No. 2.

Permata Endi, Suherman Andri, Maulana Alief. 2017. *Klasifikasi Daun Tanaman The Obroma Cacao Menggunakan Metode Neural Network*. Vol. 2 No. 10 SENTIKA, Yogyakarta.

Pradnyani Novianti. 2017. “*Unconstrained face recognition in Machine Learning*”. Vol. 5 No. 26. Springer Science & Business Media.

Rismayanti . 2017. “*Komponen Convolutional Neural Network*”. Penerbit Andi. Hal 26-27, Vol II.

S. Jurgen. 2017. “*Deep Learning in Neural Networks : On Oeoverview*”. Vol: 61, Rev: II, No. 5, hal. 85-117.

S. Ravi. 2018. *A Study on Face Recognition Technique based on Eigenface*. Vol. 7 No. 4. Hal 57–62.

Samuel Arthur. 2018. *Machine Vision*. McGraw-Hill.