

BAB II

DASAR TEORI

2.1.KAJIAN PUSTAKA

Tabel 2.1 Rangkuman Kajian Pustaka

| No | Penulis | Tahun | Judul | Keterangan |
|----|--|-------|---|---|
| 1 | Sarirotul Ilahiyah dan Agung Nilogiri | 2018 | Implementasi Deep Learning Pada Identifikasi Jenis Tumbuhan Berdasarkan Citra Daun Menggunakan Convolutional Neural Network | Penelitian ini bertujuan untuk mengidentifikasi jenis tumbuhan berdasarkan citra daun menggunakan arsitektur AlexNet. Dataset yang digunakan diambil dari Neeraj Kumar yang memiliki 20 jenis tumbuhan dengan 100 citra daun yang mempunyai resolusi yang tinggi. Dalam tahap pengujian menggunakan cross validation dilakukan 3 kali percobaan dan dihitung akurasi. Hasilnya dalam 3 kali percobaan, nilai akurasi tertinggi terdapat pada fold-7 pada percobaan pertama dengan nilai akurasi mencapai 90% |
| 2 | Syamsul Ma'arif, Tatang Rohana, dan Kiki Ahmad Baihaqi | 2022 | Deteksi Jenis Beras Menggunakan Algoritma YOLOv3 | Penelitian ini bertujuan untuk membantu masyarakat terutama kaum milenial yang tidak dapat membedakan jenis-jenis beras. Terdapat 4 jenis beras yang digunakan dalam penelitian ini yaitu beras IR42, beras ketan, beras pera, dan beras pandan wangi. penelitian ini dimulai dengan pengambilan dataset berupa gambar beras yang bertempat di pasar Karawang. Dataset tersebut yang akan diuji menggunakan metode YOLOv3. Hasil yang didapat dari pengujian ini ketika dilakukan 12 kali pengujian, beras yang diletakan berjajar atau tidak bertumpukan nilai akurasi adalah 100%. Sedangkan untuk beras yang diletakan bertumpukan mendapatkan akurasi sebesar 60% |

Tabel 2.2 Penelitian Sebelumnya

| No | Penulis | Tahun | Judul | Keterangan |
|----|---|-------|--|---|
| 3 | Mitra Saputra Ardi, Abdullah, dan Usman | 2021 | Rancang Bangun Pendeteksi Kualitas Beras Menggunakan Metode <i>K-Nearest Neighbor</i> Berbasis Android | Tujuan dari penelitian ini untuk membantu menemukan solusi untuk uji visual kualitas beras yang masih dilakukan secara manual dan dinilai kurang efektif karena sering terjadi kesalahan yang diakibatkan oleh beberapa faktor. Solusi yang ditawarkan adalah dengan melakukan pengujian secara visual menggunakan citra digital. Citra ini yang nantinya akan diolah datanya dan diklasifikasikan kedalam 3 kelas yaitu beras dengan kualitas baik, sedang, dan buruk. Hasilnya, dari 10 percobaan yang dilakukan menggunakan k-NN didapatkan nilai akurasi untuk k=1 sebesar 85,55 %, k=3 sebesar 82,21 % dan k=5 sebesar 85,55 %. Dengan simpangan baku yaitu k=1 sebesar 5.37 %, k=3 sebesar 7.77 % dan k=5 sebesar 5.37 %. |
| 4 | Gansar Suwanto, Riza Ibnu Adam, dan Garno | 2021 | Identifikasi Citra Digital Jenis Beras Menggunakan Metode ANFIS dan Sobel | Identifikasi citra digital ini menggunakan dua metode yaitu ANFIS dan Sobel dengan 140 citra dimana 100 citra digunakan sebagai data <i>training</i> . Proses identifikasi dimulai dengan pengambilan nilai ciri menggunakan metode morfologi kemudian dibandingkan dengan metode Sobel. Sedangkan nilai ciri tekstur diambil menggunakan metode <i>grayscale</i> . Nilai ciri dari tekstur dan bentuk yang sudah didapatkan dilakukan pengelompokan berdasarkan jenis beras. Hasilnya setelah dilakukan 5 kali pengujian metode ANFIS dapat mengidentifikasi beras dengan nilai akurasi 85,2%. Sedangkan deteksi tepi pada sobel dapat mempengaruhi akurasi sebesar 3%. |
| 5 | Jumi, Utami Tri Sulistyorini, dan Azizah | 2019 | Identifikasi jenis beras melalui akurasi kemiripan fitur bentuk dan warna | Mengembangkan model identifikasi level kualitas dan jenis beras berdasarkan citra beras menggunakan fitur bentuk dan warna. |

Tabel 2.3 Penelitian Sebelumnya

| No | Penulis | Tahun | Judul | Keterangan |
|----|---------|-------|-------|------------|
|----|---------|-------|-------|------------|

| | | | | |
|---|--|------|---|---|
| 5 | Jumi, Utami Tri Sulistiyorini, dan Azizah | 2019 | Identifikasi jenis beras melalui akurasi kemiripan fitur bentuk dan warna | Proses pengolahan citra dimulai dengan <i>preprocessing</i> , ekstraksi, dan klustering. Hasil dari identifikasi jenis beras menggunakan fitur bentuk dan warna memperoleh akurasi diatas 80%. Dapat dikatakan bahwa hasil identifikasi mendapat nilai akurasi yang tinggi dan sistem dapat melakukan identifikasi dengan baik. |
|---|--|------|---|---|

2.2.DASAR TEORI

2.2.1. *Machine Learning*

Machine Learning merupakan serangkaian teknik yang dapat membantu dalam menangani dan memprediksi data yang sangat besar dengan cara mempresentasikan data-data tersebut dengan algoritma pembelajaran. Proses pembelajaran dalam *Machine Learning* merupakan suatu usaha dalam memperoleh kecerdasan, yang biasanya melalui dua yaitu proses latihan (*training*) dan pengujian (*testing*)[5].

Machine learning terbagi menjadi tiga kategori klasifikasi yaitu *Supervised Learning*, *Unsupervised Learning*, *Reinforcement Learning* dengan penjelasan :

1. *Supervised Learning*

Pada klasifikasi *Supervised Learning*, pembelajaran menggunakan masukan data yang telah diberi kelas atau label. Setelah itu model membuat prediksi sesuai data yang telah diberi label.

2. *Unsupervised Learning*

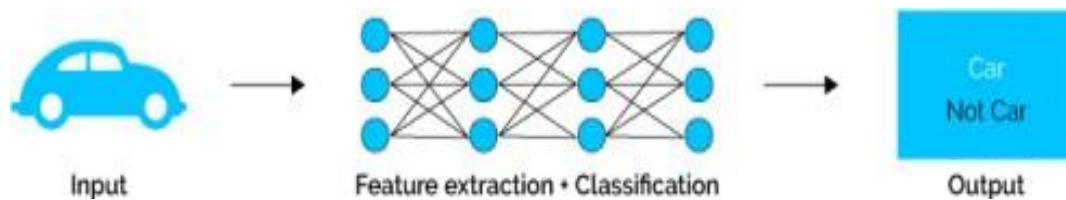
Penggunaan skenario *unsupervised learning*, pembelajaran menggunakan masukan data pembelajaran yang tidak diberi kelas atau label. Kemudian model mencoba untuk mengelompokan data berdasarkan karakteristik-karakteristik yang ditemui dari data masukan.

3. *Semi-Supervised Learning*

Pengklasifikasian *semi-supervised learning* menggabungkan sejumlah kecil data berlabel dengan sejumlah besar data yang tidak berlabel selama *training*. Pembelajaran semi-diawasi berada di antara pembelajaran tanpa pengawasan (tanpa data *training* yang berlabel) dan pembelajaran yang diawasi (dengan hanya data *training* yang berlabel) [27].

2.2.2. *Deep Learning*

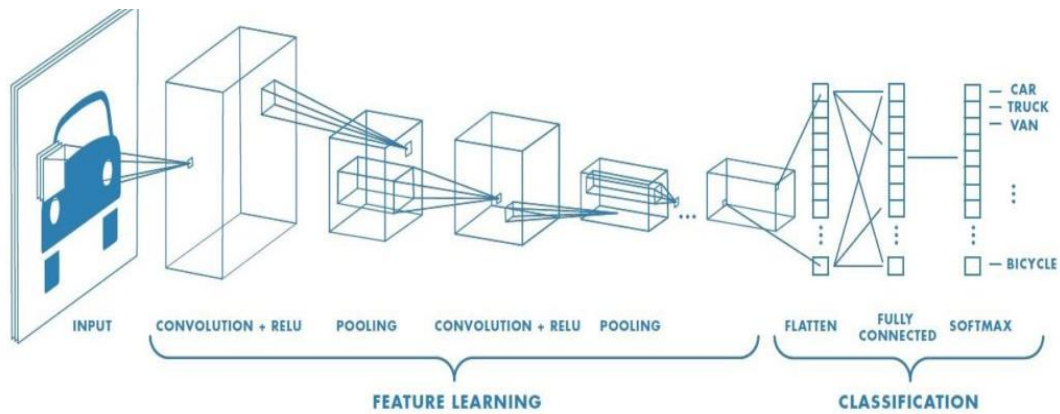
Deep Learning adalah salah satu cabang dari ilmu pembelajaran mesin (*machine learning*) yang terdiri dari algoritma pemodelan abstraksi tingkat tinggi pada data menggunakan sekumpulan fungsi transformasi *non-linear* yang ditata berlapis-lapis dan mendalam. *Deep Learning* sendiri pengaplikasiannya banyak digunakan untuk *speech recognition* pada ponsel pintar, analisa video dan citra, 12(dua belas) klasifikasi teks dan sebagainya. Teknik dan algoritma dalam *Deep Learning* dapat difungsikan untuk kebutuhan *Supervised Learning* (pembelajaran terarah), *unsupervised Learning* (pembelajaran tak terarah), dan *semi-Supervised Learning* (semi-terarah) dan juga struktur serta jumlah jaringan saraf pada algoritmanya sangat banyak hingga mencapai ratusan lapisan. Dalam pembangunan model terdapat 2(dua) istilah penting antara lain pelatihan dan pengujian. Pelatihan atau *training* merupakan sebuah proses konstruksi model dan untuk pengujian atau *testing* adalah sebuah proses menguji kinerja dari model hasil pembelajaran. Pelatihan biasanya menggunakan sebuah kumpulan data atau biasa disebut *dataset* yang berupa sampel data dalam statistika, dapat berupa citra [10].



Gambar 2.1 Arsitektur *Deep Learning* [10].

2.2.3. *Convolutional Neural Network*

Convolutional Neural Network (CNN) adalah pengembangan dari *Multilayer Perceptron (MLP)* yang di desain untuk mengolah data dua dimensi. *CNN* mampu mengenali informasi suatu objek seperti citra, teks, potongan suara dan sebagainya berupa data. Namun paling banyak digunakan pada bidang pemrosesan citra [11].

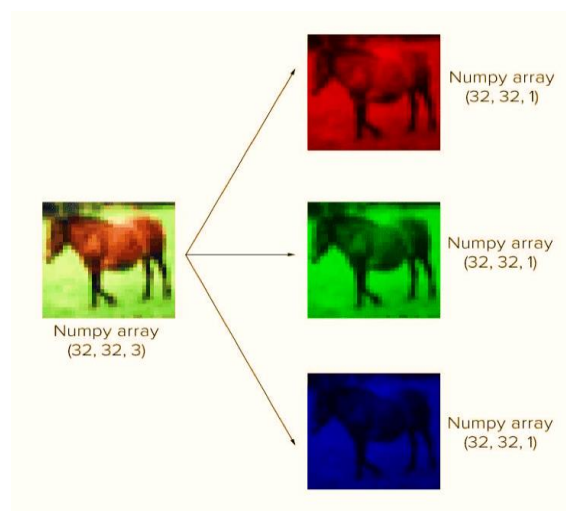


Gambar 2.2 Alur Kerja CNN [11].

Pada gambar 2.2 tentang alur kerja CNN dibagi menjadi 5 komponen utama *layer*(lapisan) antara lain :

1. *Input Layer*

Dalam lapisan *input layer* ini, lapisan yang masuk dapat berbentuk citra RGB (*Red, Green, Blue*) dengan ukuran sebesar 32×32 *pixel* yang sebenarnya itu merupakan sebuah *multidimensional array* dengan ukuran $32 \times 32 \times 3$ dan angka 3 terakhir tersebut adalah jumlah dari kanal. Contoh dari citra ditunjukkan pada Gambar 2.3.

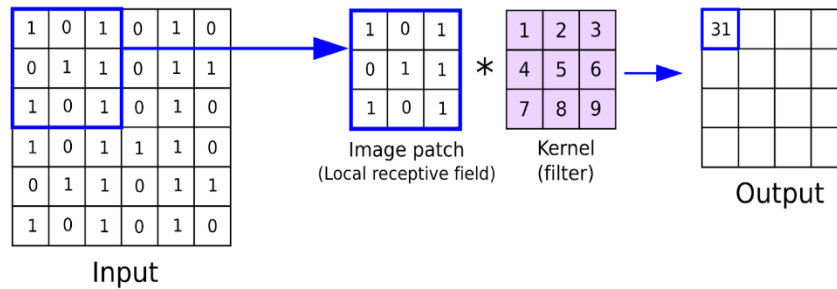


Gambar 2.3 Citra Input Layer [11].

2. *Convolution Layer*

Dalam lapisan ini merupakan lapisan pertama kali yang menerima citra langsung pada arsitektur. Dalam lapisan ini beroperasi sama seperti operasi konvolusi. Proses pemberian *filter* terhadap *Input* gambar yang dimasukkan ada dalam lapisan ini. *Filter* yang digunakan adalah ukuran tinggi, lebar, dan tebal karena *Input*nya berupa gambar. Nilai-nilai parameter *filter* tersebut dapat mempengaruhi hasil dari proses *learning* karena jika nilai yang diberikan semakin besar, maka parameter

latihnya akan semakin besar juga. Alur pada lapisan ini digambarkan seperti pada Gambar 2.4.



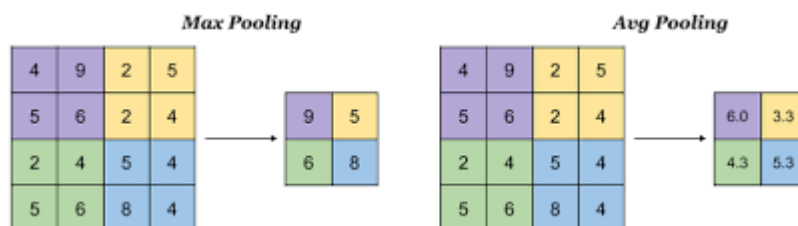
Gambar 2.4 Alur *Convolution Layer* [11].

3. *Activation Layer*

Lapisan ini merupakan lapisan dimana *feature map* dimasukkan kedalam fungsi aktivasi. Aktivasi berfungsi sebagai pengubah nilai-nilai pada *feature map* pada *range* tertentu sesuai dengan fungsi aktivasi yang digunakan. Tujuan dilakukannya untuk meneruskan nilai yang menampilkan fitur dominan yang berasal dari citra yang akan masuk ke lapisan selanjutnya.

4. *Pooling Layers*

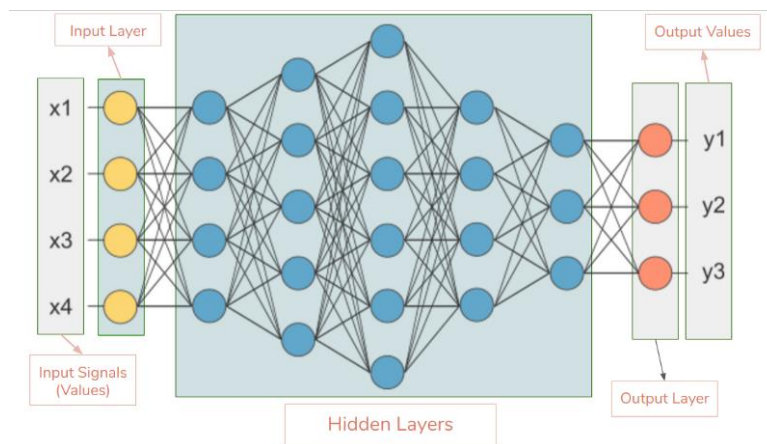
Pada lapisan ini masukkan yang berasal dari lapisan aktivasi akan dikurangi parameternya. Lapisan ini juga bisa disebut sebagai *subsampling* atau *downsampling* yang mengurangi dimensi dari *feature map* tanpa menghilangkan informasi penting di dalamnya. Proses dalam lapisan ini cukup sederhana, pertama akan ditentukan ukuran *down sampling* yang akan digunakan pada *feature map*, misal 2x2 kemudian akan dilakukan proses *pooling* pada *feature map*. Proses dari *pooling* ada beberapa macam seperti *max pooling*, *mean pooling*, dan *sum pooling* [11]. Contoh proses *pooling* ditunjukkan pada Gambar 2.5.



Gambar 2.5 Contoh Proses pooling layer [12].

5. *Fully Connected Layer*

Pada *layer* ini data satu dimensi dari proses sebelumnya digunakan sebagai *Input* dan akan diolah sehingga dapat dilakukan klasifikasi terhadap data tersebut. Di dalam lapisan ini terdapat neuron-neuron yang saling terhubung satu sama lain. Pada proses ini, sebelum masuk kedalam proses *fully connected layer*, *feature map* yang dihasilkan dari proses *pooling layer* akan melalui proses *reshape (flatten)* terlebih dahulu sehingga akan menghasilkan sebuah vektor yang nantinya digunakan untuk masukkan pada proses *fully connected layer* [11].. Gambaran dari proses yang terjadi pada *layer* ini dapat dilihat pada gambar 2.3 di bawah ini.



Gambar 2.6 Proses *Fully Connected Layer* [13].

2.2.4. *Confusion matrix*

Confusion matrix merupakan salah satu metode yang digunakan untuk mengukur kinerja suatu metode klasifikasi. *Confusion Matrix* mengandung informasi yang membandingkan hasil klasifikasi yang dilakukan oleh sistem dengan hasil klasifikasi yang sebenarnya.

| | | Predicted | | |
|--------|--------------|--|---|--|
| | | Negative (0) | Positive (1) | |
| Actual | Negative (0) | True Negative TN | False Positive FP (Type I error) | Specificity $= \frac{TN}{TN + FP}$ |
| | Positive (1) | False Negative FN (Type II error) | True Positive TP | Recall, Sensitivity, True positive rate (TPR) $= \frac{TP}{TP + FN}$ |
| | | Accuracy $= \frac{TP + TN}{TP + TN + FP + FN}$ | Precision, Positive predictive value (PPV) $= \frac{TP}{TP + FP}$ | F1-score $= 2 \times \frac{Recall \times Precision}{Recall + Precision}$ |

Gambar 2.7 Confusion Matrix [14].

Penjelasan gambar 2.7 dimulai dari *True Positive* (TP) adalah kasus di mana model klasifikasi yang dibuat dengan benar diprediksi positif. *False Negative* (FN) adalah kasus di mana model klasifikasi diprediksi salah, tetapi sebenarnya positif. Ini juga dianggap sebagai kesalahan Tipe II. *False Positif* (FP) adalah positif palsu kasus dimana model klasifikasi diprediksi positif, tetapi sebenarnya negatif. Ini juga dianggap sebagai kesalahan Tipe I. *True Negative* (TN) adalah kasus di mana model klasifikasi memprediksi negatif dengan benar. Setelah hasil dari masalah klasifikasi telah diterima, matrix klasifikasi dapat menghitung nilai Sensitivitas, Spesifisitas, Akurasi, Nilai Prediktif Negatif, dan Presisi. Dimana sensitivitas yang biasa disebut *True Positif Rate* atau *Recall* adalah rasio prediksi benar positif dibandingkan dengan keseluruhan data yang benar positif. Spesifisitas juga dikenal sebagai *True Negative Rate* merupakan kebenaran memprediksi negatif dibandingkan dengan keseluruhan data negatif. Akurasi merupakan rasio prediksi benar (positif dan negatif) dengan keseluruhan data. Nilai Prediktif Negatif merupakan hasil yang diberi label dengan benar sebagai salah. Presisi merupakan rasio prediksi benar positif dibandingkan dengan keseluruhan hasil yang diprediksi positif. *F1-Score* merupakan perbandingan rata-rata presisi dan *recall* yang dibobotkan $F1\ Score = 2 * (Recall * Precision) / (Recall + Precision)$ [15].

2.2.5. Visual Geometri Grup-16

VGG16 merupakan model *CNN* yang memanfaatkan *convolutional layer* dengan spesifikasi *convolutional filter* yang kecil (3×3). Dengan ukuran *convolutional filter*

tersebut, kedalaman *neural network* dapat ditambah dengan lebih banyak lagi *convolutional layer*. Hasilnya, model *CNN* menjadi lebih akurat daripada model-model *CNN* sebelumnya. Model *VGG16* mempunyai 19 *layer* yang terdiri dari 16 *convolutional layer* dan 3 *fully-connected layer* [16].

2.2.6. *Mobilenetv2*

Mobilenetv2 merupakan salah satu arsitektur *CNN* yang dapat digunakan untuk mengatasi kebutuhan akan *computing resource* berlebih. Seperti namanya, *Mobile*, para peneliti dari *Google* membuat arsitektur *CNN* yang dapat digunakan untuk ponsel. Perbedaan mendasar antara arsitektur *MobileNet* dan arsitektur *CNN* pada umumnya adalah penggunaan lapisan konvolusi dengan ketebalan *filter* yang sesuai dengan ketebalan dari gambar masukan. *MobileNet* membagi konvolusi menjadi *depthwise convolution* dan *pointwise convolution* [17].

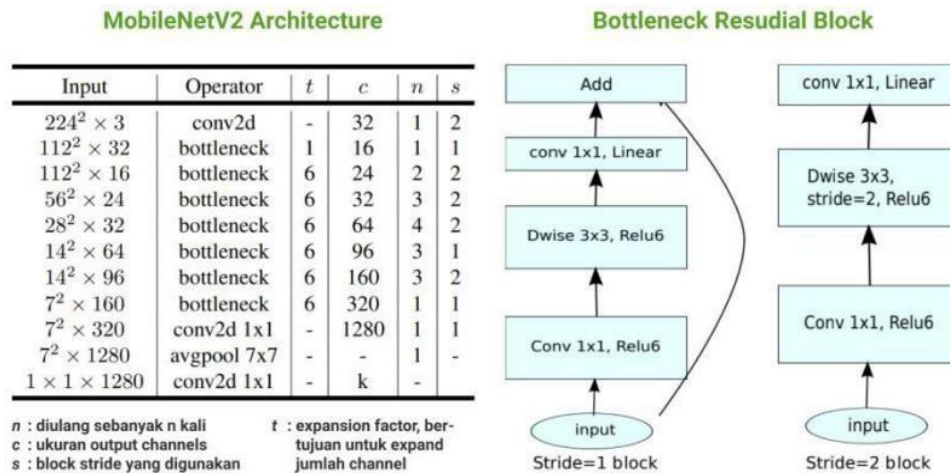
Perbandingan jumlah komputasi antara *normal convolution* dibandingkan dengan *depthwise separable convolution* :

1. *Normal Convolution*

Ada 256 *filter* berukuran $5 \times 5 \times 3$ yang bergerak sebanyak delapan kali delapan jumlah pergeseran *filter* 5×5 dari kiri atas ke kanan bawah terhadap *Input image* 12×12). Yang mempunyai arti ketika proses konvolusi $256 \times 5 \times 5 \times 3 \times 8 \times 8 = 1.228.800$ total perkalian yang dilakukan.

2. *Depthwise Separable Convolution*

Dalam tahap *depthwise convolution* terdapat 3(tiga) *filter* $5 \times 5 \times 1$ yang bergeser sebanyak delapan kali delapan dengan arti $3 \times 5 \times 5 \times 1 \times 8 \times 8 = 4.800$ yang kemudian, dalam *pointwise convolution* terdapat 256 *filter* $1 \times 1 \times 3$ yang bergeser sebanyak delapan kali delapan juga memiliki arti $256 \times 1 \times 1 \times 3 \times 8 \times 8 = 49.152$ yang jika ditotal perkalian yang dilakukan saat proses konvolusi pada *depthwise separable convolution* ini yaitu $4.800 + 49.152 = 52.952$. Dapat disimpulkan bahwa jumlah 52.952 hanya kurang lebih sekitar 22,7% dari 1.228.800 pada *normal convolution*.



Gambar 2.8 Arsitektur *MobileNetV2* [11].

MobileNetV2 merupakan perkembangan lebih lanjut dari *MobileNet* dimana ditambahkan fitur baru yaitu sebuah blok dalam arsitektur yang didalamnya menggunakan *depthwise separable convolution* [11].

2.2.7. Beras

Beras adalah salah satu komoditas penting bagi Indonesia. Hal ini mengingat hampir seluruh masyarakat Indonesia mengkonsumsi beras sebagai makanan pokoknya. Itu sebabnya Indonesia merupakan konsumen pangan dengan bahan pangan beras terbesar. Selain itu, beras sangat berpengaruh bagi perekonomian Indonesia karena lebih dari 60 % penduduk Indonesia berprofesi sebagai petani penghasil beras. Dengan demikian beras tidak hanya dibutuhkan untuk dikonsumsi tetapi juga merupakan sumber pendapatan dan penyerapan tenaga kerja [18].

2.2.8. Beras IR 64

Beras IR 64 merupakan salah satu jenis varietas padi sawah yang memiliki bentuk tegak, tinggi sekitar 115-126 cm. Gabahnya berbentuk ramping dan panjang dan berwarna kuning bersih. Padi jenis ini cocok ditanam di daerah lahan sawah irigasi dataran rendah sampai sedang. Beras IR 64 adalah jenis beras yang pulen jika dimasak menjadi nasi karena memiliki kadar *amilosa* sebanyak 23%. Bobot beras per seribu butir beras IR64 adalah 24,1 gram [19].

2.2.9. Beras Basmati

Beras basmathi bersal dari negara india/pakistan, beras basmathi berasal dari bahasa sanskerta “basmathi” artinya berarti harum atau wangi dalam bahasaindia ia juga mempunyai maksud ”soft rice” yaitu lembut. Seperti pada umumnya beras, beras basmati juga mempunyai dua jenis, bersa putih dan beras coklat ,Setelah dimasak menjadi nasi barulah keunikan beras ini terlihat. Beras ini akan melar memanjang butiran, sedikit pera, mudah terurai butiran berasnya dan aromanya sangat harum, beras ini memang aromanya sangat harum. Keistimewaan lainnya, butiran atau buliran beras ini panjang dan kecil melebihi ukuran beras yang biasanya agak pendek dan bulat. Berdasarkan uji laboratorium, beras basmati mengandung 0.09 bagian 2-*acetyl-1-pyrroline* per juta, yaitu 12 kali lebih dari konsenstrasi pada beras biasa. Karena secara alami wangi makan beras ini tak bisa ditandingi aromanya dengan beras lain [20].

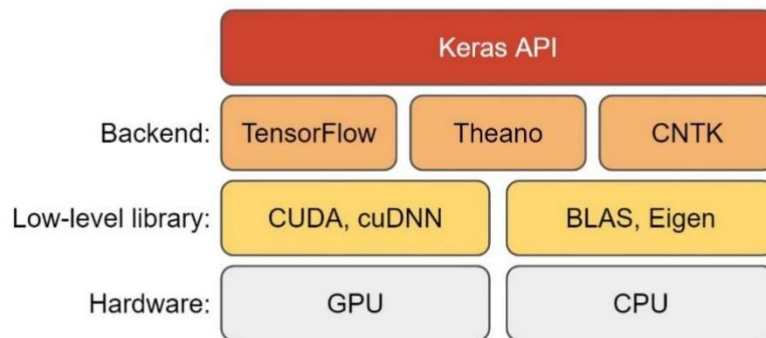
2.2.10. Beras Ketan Hitam

Beras ketan hitam (*Oryza sativa glutinosa*) merupakan salah satu varietas padi yang termasuk dalam *family Graminae*. Butir beras sebagian besar terdiri dari zat pati sekitar 80-85% yang terdapat dalam endosperma yang tersusun oleh granula-granula pati yang berukuran 3-10 *milimikron*. Beras ketan juga mengandung vitamin (terutama pada bagian *aleurone*), mineral dan air. Dari komposisi kimiawinya diketahui bahwa karbohidrat penyusun utama beras ketan adalah pati. Pati merupakan karbohidrat *polimer glukosa* yang mempunyai dua struktur yakni *amilosa* dan *amilopektin* [21].

2.2.11. Library Keras dan Tensorflow

Library keras merupakan *library* yang digunakan untuk menyederhanakan model dari kerangka *Deep Learning*. Keras dijalankan diatas framework *TensorFlow*. *Tensorflow* merupakan perpustakaan perangkat lunak yang dikembangkan untuk pembelajaran mesin dan penelitian jaringan syaraf tiruan. *Tensorflow* adalah *framework* yang berkaitan erat dengan *Deep Learning*. *Tensorflow* dapat digunakan dalam mendeteksi gambar ataupun suara. Salah satu jenis metode yang dapat digunakan dengan *Tensorflow* yaitu *Convolutional Neural Network(CNN)*. *Convolutional Neural Network(CNN)* merupakan *algoritma Deep Learning* yang sangat populer karena dapat menghilangkan kebutuhan untuk ekstraksi fitur secara

manual, dapat dilatih untuk tugas pengenalan data baru, dan juga memiliki beberapa model [22].



Gambar 2.9 Keras dan Tensorflow [23].

