

## **BAB III**

### **METODE PENELITIAN**

#### **3.1 Objek dan Subjek Penelitian**

Pada penelitian ini, objek yang diteliti adalah model CNN yang dilatih dengan skenario pelatihan yaitu perbedaan jumlah *convolutional layer*, jumlah *epoch*, jumlah *dataset*, serta perbandingan pembagian data latih dan data validasi. Total keseluruhan dari model CNN yang dilatih dan dianalisis adalah 48 model.

Sedangkan subjek dari penelitian ini adalah citra makanan tradisional khas Daerah Istimewa Yogyakarta yaitu bakpia, growol, mata kebo, tiwul, dan jadah.

#### **3.2 Alat dan Bahan Penelitian**

Adapun alat dan bahan pada penelitian ini terbagi menjadi dua bagian yaitu perangkat keras dan perangkat lunak. Berikut adalah perangkat keras yang digunakan:

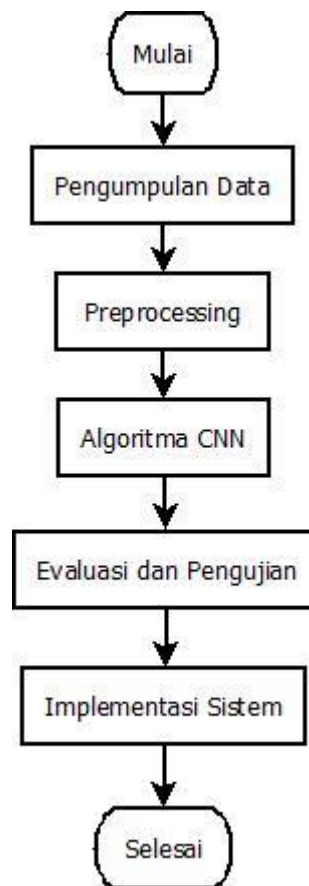
1. *Processor*: Intel(R) Core™ i7-4700 CPU @ 2.40 GHz 2.40 GHz
2. RAM: 8 GB (7.82 *usalbe*)
3. *Storage*: 1 TB HDD
4. VGA: NVIDIA GEFORCE 745M

Sedangkan perangkat lunak yang digunakan adalah:

1. Sistem operasi *Windows 10*
2. Anaconda Navigator
3. Jupiter Notebook
4. Bahasa pemrograman *python*

#### **3.3 Diagram Alir Penelitian**

Penelitian ini dimulai dengan pengumpulan data, *preprocessing*, penerapan algoritma CNN, evaluasi dan pengujian, dan model yang memiliki akurasi tertinggi maka dilanjutkan dengan pengimplementasian sistem. Proses tersebut dapat digambarkan dalam *flowchart* sebagai berikut:



Gambar 3.1 Diagram Alur Penelitian

Berdasarkan Gambar 3.1, maka alur penelitian ini adalah sebagai berikut:

a. Pengumpulan Data

Pada tahap ini, data berupa gambar makanan tradisional khas Yogyakarta yaitu bakpia, growol, mata kebo, tiwul dan jadah diambil dari dua sumber yang berbeda yaitu dari pengambilan secara manual dengan kamera digital dan dari hasil pencarian internet.

Pengambilan gambar dengan menggunakan kamera digital dilakukan dari dua sudut yang berbeda yaitu  $45^\circ$  dan  $90^\circ$ , dan dari setiap sudut tersebut diambil 25 gambar yang berbeda. Sehingga dihasilkan 50 gambar untuk setiap jenis makanan, karena terdapat 5 jenis makanan maka total keseluruhan gambar adalah 250 gambar.

Sedangkan data yang bersumber dari pencarian internet, didapatkan dengan melakukan pencarian gambar makanan tradisional Yogyakarta yaitu

bakpia, growol, jadah, mata kebo, dan tiwul pada mesin pencari *Google*, kemudian mendownloadnya dengan menggunakan *Fatkun AI Downloader*. Sebelum mendownload gambar, dilakukan proses pemilihan terlebih dahulu untuk memastikan bahwa gambar yang akan didownload sesuai dengan kriteria yang diinginkan yaitu gambar makanan tradisional Yogyakarta tanpa pembungkus.

Data yang bersumber dari pengambilan secara manual berjumlah 250, sedangkan data yang bersumber dari pencarian internet memiliki jumlah yang beragam. Oleh karena itu, setelah kedua data dari sumber yang berbeda tersebut disatukan maka jumlah data pada setiap kelas menjadi beragam. Data yang telah disatukan kemudian akan memasuki tahap *preprocessing data*.

#### b. *Preprocessing*

Tahap *preprocessing* merupakan tahap pengolahan data sebelum masuk ke dalam algoritma. *Preprocessing* yang dilakukan pada penelitian ini adalah *resize* ukuran data, memperbanyak jumlah data dengan augmentasi serta membagi *dataset* ke dalam dua kelompok yaitu data latih dan data validasi.

Pada proses *resize* ukuran, data berupa gambar akan diubah ukurannya menjadi seragam yaitu  $125 \times 125$  *pixel* dengan menggunakan program pada *python*. Proses *resize* ukuran gambar dilakukan untuk menghemat waktu pelatihan data, hal tersebut telah dibuktikan pada penelitian [24], dimana pelatihan data yang menggunakan data berukuran yang lebih kecil memerlukan waktu yang lebih sedikit dibandingkan dengan pelatihan data dengan ukuran data yang besar. Setelah didapatkan data dengan ukuran yang seragam, kemudian data tersebut diletakkan pada satu folder.

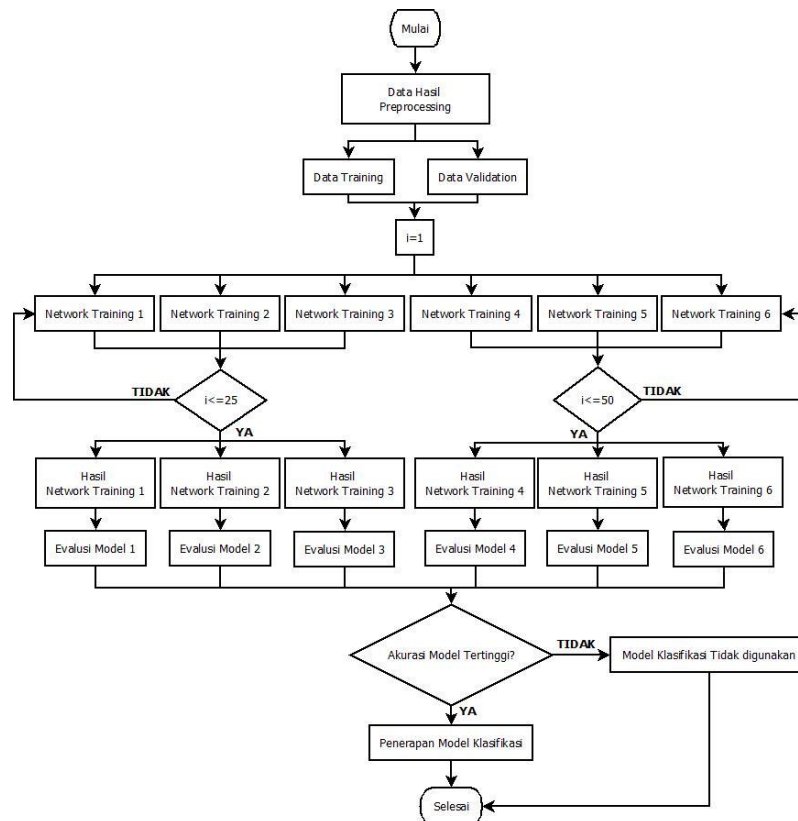
Selain proses *resize* ukuran, pada tahap *preprocessing* ini menerapkan proses augmentasi yaitu untuk memperbanyak jumlah data serta varian data. Proses augmentasi data dilakukan dengan merotasi secara acak, memperbesar, memotong, mengubah ukuran, membalik, merotasi  $90^\circ$ , dan merotasi  $270^\circ$ .

Pada akhir tahap *preprocessing*, didapatkan dua *dataset* yaitu *dataset* asli serta *dataset* augmentasi. Di mana *dataset* asli merupakan *dataset* yang hanya

melalui proses *resize* ukuran, sedangkan *dataset* augmentasi merupakan *dataset* yang melalui proses *resize* ukuran serta augmentasi. Kedua *dataset* tersebut kemudian dibagi menjadi dua bagian yaitu data latih serta data validasi dengan 4 perbandingan berbeda yaitu 60:40, 70:30, 80:20, dan 90:10. Pemilihan keempat skenario pembagian data tersebut dikarenakan pada penelitian [14], telah membuktikan bahwa semakin banyak rasio data latih pada skenario pembagian data, semakin tinggi pula akurasi model yang didapatkan, dimana model yang dilatih menggunakan *dataset* citra tomat. Sehingga pada penelitian ini diterapkan skenario perbandingan perbandingan data seperti pada penelitian [14] yaitu 60:40, 70:30, 80:20, namun dilatih dengan menggunakan *dataset* citra makanan tradisional Yogyakarta. Selain itu terdapat pula penelitian [35] yang menerapkan skenario perbandingan pembagian data 90:10 dengan hasil akurasi model sebesar 93%.

c. Algoritma *Convolutional Neural Network*

Pada penelitian ini algoritma yang digunakan adalah *Convolutional Neural Network* (CNN), dengan tahap awal yaitu memanggil *dataset* berupa data latih dan data validasi, kemudian proses *training* data dengan menggunakan *network training* dan dengan *epoch* yang telah ditentukan, evaluasi model, serta yang terakhir model dengan akurasi tertinggi akan digunakan sebagai pengujian dengan data *testing*. Adapun alur kerja algoritma *Convolutional Neural Network* adalah sebagai berikut:



Gambar 3.2 Alur Pembangunan Model Klasifikasi

Berdasarkan Gambar 3.2, maka alur algoritma CNN dimulai dengan memanggil *dataset* yaitu *dataset* asli dan *dataset* augmentasi yang sebelumnya telah dibagi kedalam dua kelompok yaitu data latih dan data validasi pada saat *preprocessing*. Pada penelitian ini terdapat 4 skenario pembagian data yaitu pembagian data dengan perbandingan data latih dan data validasi 60:40, 70:30, 80:20, dan 90:10. Kemudian dilanjutkan dengan perancangan *training network*. Adapun *training network* yang digunakan pada penelitian ini adalah:

Tabel 3.1 Training Network

No	Nama Training Network	Convolutional Layer	Pooling Layer	Fully Connected Layer
1	Training Network 1	3	3	2
2	Training Network 2	5	4	2
3	Training Network 3	7	5	2

Pemilihan konfigurasi *training network* pada Tabel 3.1, didasarkan pada penelitian [10], dimana penelitian tersebut membandingkan hasil akurasi dari penggunaan 1-7 *convolutional layer* dan model dengan hasil akurasi tinggi adalah model yang menerapkan 4-7 *convolutional layer*. Ketiga *training network* tersebut nantinya akan dilatih dengan menggunakan 25 *epoch* dan 50 *epoch*. Pemilihan kedua jumlah *epoch* tersebut didasarkan pada penelitian [24], yang telah membuktikan bahwa semakin banyak *epoch* semakin tinggi pula hasil akurasi model yang didapatkan, namun semakin lama pula proses pelatihan data. Namun pada penelitian ini hanya digunakan 25 *epoch* dan 50 *epoch* agar dapat menghemat proses pelatihan data. Pada penelitian ini terdapat dua *dataset*, empat skenario pembagian data, tiga *training network* serta dua macam *epoch* sehingga terdapat 48 pelatihan model.

*Training network* terdiri atas beberapa bagian dan pada umumnya terdiri dari *convolutional layer*, fungsi aktivasi, *maxpooling layer*, dan *fully connected layer* dengan jumlah tertentu. Adapun perhitungan manual dari setiap bagian *training network* adalah sebagai berikut:

a. *Convolutional layer*

Sebuah citra  $f(x, y)$  dengan ukuran 4x4 dan sebuah filter  $g(x, y)$  dengan ukuran 2x2

8	6	4	2
5	7	7	3
3	5	6	2
6	4	7	8

$f(x, y)$

0	1
1	-1

$g(x, y)$

Operasi konvolusi antara citra  $f(x, y)$  dan filter  $g(x, y)$  adalah dengan menggunakan persamaan  $h(x, y) = f(x, y) * g(x, y)$ , sehingga dapat diilustrasikan sebagai berikut

- 1) Filter ditempatkan pada sisi paling kiri, kemudian dilakukan perhitungan konvolusi sebagai berikut

$$h(x, y) = f(x, y) * g(x, y)$$

$$h(x, y) = (8 * 0) + (6 * 1) + (5 * 1) + (7 * (-1))$$

$$h(x, y) = 4$$

Sehingga didapatkan ilustrasi sebagai berikut

8	6	4	2
5	7	7	3
3	5	6	2
6	4	7	8

$f(x, y)$

0	1
1	-1

$g(x, y)$

4		

- 2) Filter bergeser satu *pixel* ke kanan, kemudian dilakukan perhitungan konvolusi sebagai berikut

$$h(x, y) = f(x, y) * g(x, y)$$

$$h(x, y) = (6 * 0) + (4 * 1) + (7 * 1) + (7 * (-1))$$

$$h(x, y) = 4$$

Sehingga didapatkan ilustrasi sebagai berikut

8	6	4	2
5	7	7	3
3	5	6	2
6	4	7	8

$f(x, y)$

0	1
1	-1

$g(x, y)$

4	4	

- 3) Filter bergeser sebanyak 1 *pixel* ke kanan, kemudian dilakukan perhitungan konvolusi sebagai berikut

$$h(x, y) = f(x, y) * g(x, y)$$

$$h(x, y) = (4 * 0) + (2 * 1) + (7 * 1) + (3 * (-1))$$

$$h(x, y) = 6$$

Sehingga didapatkan ilustrasi sebagai berikut

8	6	4	2
5	7	7	3
3	5	6	2
6	4	7	8

$f(x, y)$

0	1
1	-1

$g(x, y)$

4	4	6

- 4) Filter ditempatkan pada sisi paling kiri dan bergeser 1 *pixel* ke bawah, kemudian dilakukan perhitungan konvolusi sebagai berikut

$$h(x, y) = f(x, y) * g(x, y)$$

$$h(x, y) = (5 * 0) + (7 * 1) + (3 * 1) + (5 * (-1))$$

$$h(x, y) = 5$$

Sehingga didapatkan ilustrasi sebagai berikut

8	6	4	2
5	7	7	3
3	5	6	2
6	4	7	8

$f(x, y)$

0	1
1	-1

$g(x, y)$

0	4	6
5		

- 5) Filter bergeser kembali sebanyak 1 *pixel*, kemudian dilakukan perhitungan konvolusi sebagai berikut

$$h(x, y) = f(x, y) * g(x, y)$$

$$h(x, y) = (7 * 0) + (7 * 1) + (5 * 1) + (6 * (-1))$$

$$h(x, y) = 6$$

Sehingga didapatkan ilustrasi sebagai berikut



8	6	4	2
5	7	7	3
3	5	6	2
6	4	7	8

$f(x, y)$

0	1
1	-1

$g(x, y)$

0	4	6
5	6	

- 6) Filter bergeser ke kanan sebanyak 1 *pixel*, kemudian dilakukan kembali perhitungan konvolusi sebagai berikut

$$h(x, y) = f(x, y) * g(x, y)$$

$$h(x, y) = (7 * 0) + (3 * 1) + (6 * 1) + (2 * (-1))$$

$$h(x, y) = 7$$

Sehingga didapatkan ilustrasi sebagai berikut

8	6	4	2
5	7	7	3
3	5	6	2
6	4	7	8

$f(x, y)$

0	1
1	-1

$g(x, y)$

0	4	6
5	6	7

- 7) Filter ditempatkan pada sisi paling kiri bawah, kemudian dihitung operasi konvolusi sebagai berikut

$$h(x, y) = f(x, y) * g(x, y)$$

$$h(x, y) = (3 * 0) + (5 * 1) + (6 * 1) + (4 * (-1))$$

$$h(x, y) = 7$$

Sehingga didapatkan ilustrasi sebagai berikut

8	6	4	2
5	7	7	3
3	5	6	2
6	4	7	8

$f(x, y)$

0	1
1	-1

$g(x, y)$

0	4	6
5	6	7
7		

- 8) Filter kembali bergeser ke kanan sebanyak 1 *pixel*, kemudian dilakukan perhitungan konvolusi sebagai berikut

$$h(x, y) = f(x, y) * g(x, y)$$

$$h(x, y) = (5 * 0) + (6 * 1) + (4 * 1) + (7 * (-1))$$

$$h(x, y) = 3$$

Sehingga didapatkan ilustrasi sebagai berikut

8	6	4	2
5	7	7	3
3	5	6	2
6	4	7	8

$f(x, y)$

0	1
1	-1

$g(x, y)$

0	4	6
5	6	7
7	3	

- 9) Filter bergeser pada posisi akhir yaitu sisi paling kanan bawah, kemudian dilakukan perhitungan konvolusi sebagai berikut

$$h(x, y) = f(x, y) * g(x, y)$$

$$h(x, y) = (6 * 0) + (2 * 1) + (7 * 1) + (8 * (-1))$$

$$h(x, y) = 1$$

Sehingga didapatkan ilustrasi sebagai berikut

8	6	4	2
5	7	7	3
3	5	6	2
6	4	7	8

$f(x, y)$

0	1
1	-1

$g(x, y)$

0	4	6
5	6	7
7	3	1

#### b. Fungsi aktivasi ReLU

ReLU memiliki persamaan sebagai berikut:

$$f(x) = \max(0, x) = \begin{cases} x_i, & \text{if } x_i \geq 0 \\ 0, & \text{if } x_i < 0 \end{cases}$$

Dengan persamaan tersebut seluruh data yang melewati fungsi aktivasi ReLU akan memiliki nilai positif. Diketahui data citra hasil konvolusi adalah sebagai berikut

0	4	6
5	6	7
7	3	1

Data citra tersebut tidak memiliki nilai negatif sehingga jika dikenakan persamaan ReLU akan memiliki hasil yang sama seperti data awal. Hal tersebut dikarenakan setiap  $x_i$  adalah bilangan yang lebih dari atau sama dengan 0.

c. *Maxpooling*

Diketahui data citra hasil konvolusi dan aktivasi ReLU adalah sebagai berikut

0	4	6
5	6	7
7	3	1

Proses *pooling* ini menggunakan ukuran  $2 \times 2$  dan *stride* 1, sehingga *window* akan bergeser 1 *pixel* ke kanan dan mengambil nilai tertinggi sebagai hasilnya. Adapun proses tersebut adalah sebagai berikut:

- 1) *Window* ditempatkan pada sisi paling kiri atas kemudian diambil nilai terbesar sebagai hasil dari *maxpooling*.

0	4	6	6	
5	6	7		
7	3	1		

- 2) *Window* bergeser ke kanan sebanyak satu *pixel* kemudian diambil nilai terbesar sebagai hasil dari *maxpooling*.

0	4	6	6	7
5	6	7		
7	3	1		

- 3) *Window* kembali ditempatkan pada sisi paling kiri namun bergeser satu *pixel* ke bawah kemudian diambil nilai tertinggi.

0	4	6	6	7
5	6	7	7	
7	3	1		

- 4) *Window* bergeser sebanyak satu *pixel* ke kanan kemudian diambil nilai tertinggi.

0	4	6	6	7
5	6	7	7	7
7	3	1		

#### d. Evaluasi dan Pengujian

Tahap evaluasi dan pengujian merupakan tahap untuk mengetahui performa model yang telah melalui tahap *training*. Pada tahap ini model *Convolutional Neural Network* yang telah dievaluasi dengan menggunakan *Confusion Matrix*. Adapun contoh evaluasi menggunakan *Confusion Matrix* adalah sebagai berikut:

Tabel 3.2 *Confusion Matrix*

		<i>Predicted Class</i>		
		Yes	No	Total
<i>Actual Class</i>	Yes	150	30	180
	No	20	50	70
	Total	170	80	250

Menghitung nilai akurasi model untuk mengukur performa yang dihasilkan oleh model.

$$\begin{aligned}
 Accuracy &= \frac{TP + TN}{P + N} \\
 &= \frac{150 + 50}{180 + 70} \\
 &= \frac{200}{250} \\
 &= 0,8
 \end{aligned}$$

Menghitung nilai *error* untuk mengukur tingkat kesalahan model dalam mengklasifikasi.

$$\begin{aligned}
 Error\ Rate &= \frac{FP + FN}{P + N} \\
 &= \frac{20 + 30}{180 + 70} \\
 &= \frac{50}{250} \\
 &= 0,2
 \end{aligned}$$

Menghitung *precision* untuk menentukan ketepatan model dalam mengklasifikasi.

$$\begin{aligned}
 Precision &= \frac{TP}{TP + FP} \\
 &= \frac{150}{150 + 20} \\
 &= \frac{150}{170} \\
 &= 0,88
 \end{aligned}$$

Menghitung *recall* untuk melihat seberapa besar tingkat kesalahan dalam prediksi.

$$\begin{aligned}
 Recall &= \frac{TP}{TP + TN} \\
 &= \frac{150}{150 + 50} \\
 &= \frac{150}{200} \\
 &= 0,75
 \end{aligned}$$

#### e. Implementasi Sistem

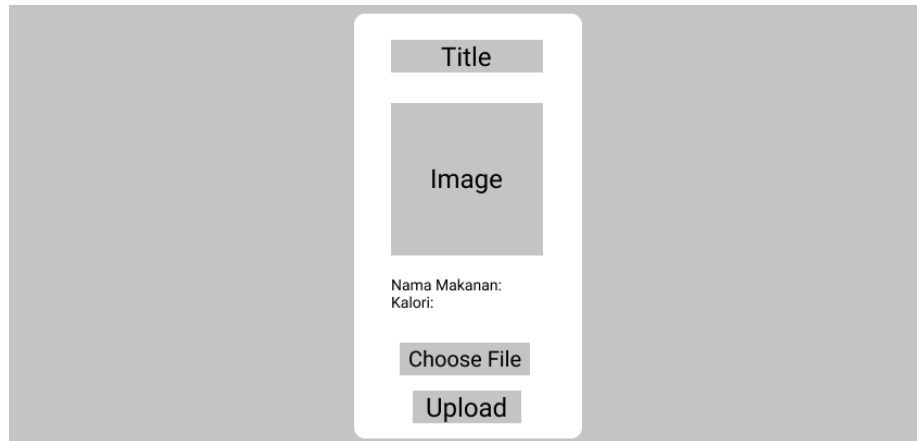
Tahap ini merupakan pembuatan sistem klasifikasi citra makanan tradisional Yogyakarta untuk menentukan kalori. Implementasi sistem ini menggunakan *flask web*, dimana proses awalnya adalah dengan menyimpan model yang telah dibuat menjadi file dengan format HDF5 atau .h5. Model yang telah disimpan tersebut kemudian diterapkan untuk klasifikasi data baru. Ketika user meng-*input*-kan suatu citra ke dalam sistem, maka sistem akan melakukan proses klasifikasi. Hasil dari klasifikasi tersebut selanjutnya akan diarahkan menuju *database* yang berisi tabel kalori. Adapun tabel kalori pada *database* adalah sebagai berikut:

Tabel 3.3 Kandungan Kalori

No	Nama Makanan	Kalori (kkal)	Takaran Saji
1	Bakpia	78,09	25 gram
2	Growol	284,3	175 gram
3	Mata kebo	105,7	45 gram
4	Tiwul	315,99	150 gram
5	Jadah	86,40	35 gram

Pada database, data berupa kalori serta takaran saji yang sesuai dengan hasil klasifikasi sebelumnya akan diambil untuk selanjutnya ditampilkan pada halaman web sebagai hasil dari klasifikasi dan penentuan kalori makanan tradisional Yogyakarta.

Pada perancangannya, sistem web memiliki tampilan berupa *form* untuk menginputkan data gambar. Adapun *mockup* dari sistem tersebut adalah sebagai berikut:



The mockup shows a white form centered on a gray background. The form contains the following elements from top to bottom: a 'Title' input field, an 'Image' placeholder area, two text labels 'Nama Makanan:' and 'Kalori:', a 'Choose File' button, and an 'Upload' button.

Gambar 3.3 *Mockup* Sistem