



Hak cipta dan penggunaan kembali:

Lisensi ini mengizinkan setiap orang untuk menggubah, memperbaiki, dan membuat ciptaan turunan bukan untuk kepentingan komersial, selama anda mencantumkan nama penulis dan melisensikan ciptaan turunan dengan syarat yang serupa dengan ciptaan asli.

Copyright and reuse:

This license lets you remix, tweak, and build upon work non-commercially, as long as you credit the origin creator and license it on your new creations under the identical terms.

Bab 3.

ANALISIS DAN PERANCANGAN SISTEM

3.1 Metode Penelitian

Berikut ini adalah tahapan – tahapan yang dilakukan dalam pelaksanaan penelitian ini.

1. Telaah Literatur

Dalam tahap ini, akan dilakukan studi mengenai proses – proses yang akan dilakukan, yaitu membaca paper yang tersedia, membuat flowchart, melakukan proses penelitian hingga penetapan projek yang akan dipakai. Setelah menimbang lebih lanjut, penulis memilih teknik CNN dengan input gambar sebagai dasar penelitian tersebut.

2. Pengambilan Data

Data yang digunakan adalah data foto skincare yang ada di internet dan diubah menjadi dataset baru. Gambar yang dipakai dalam berbentuk jpg berjumlah 1000 gambar yang dibagi menjadi 5 folder. Data tersebut akan digunakan dalam penelitian tersebut dengan class adalah nama merek skincare.

3. Perancangan dan Pembuatan Sistem

Pada tahap ini, dilakukan perancangan program untuk penelitian tersebut, dibuatkan model yang akan dipakai dalam penelitian ini. Bahasa yang digunakan adalah Python dengan menggunakan aplikasi Google Colab.

4. Evaluasi Model

Dalam tahapan ini, dilakukan evaluasi dari penerapan model tersebut yang akan menjadi patokan untuk laporan tersebut. Di dalam evaluasi akan dicari accuracy dan loss. Evaluasi ini akan menjadi telaan dalam menulis laporan skripsi.

5. Penulisan Laporan Penelitian

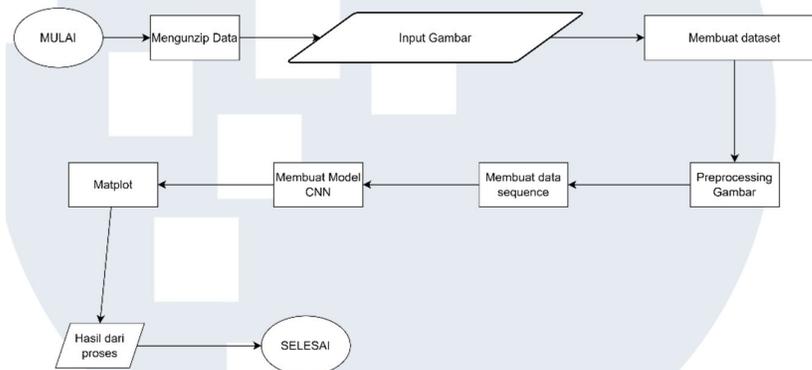
Setelah semua proses telah dilakukan, akan dilakukan penulisan laporan tentang hasil dari penelitian yang telah dilakukan serta menarik

kesimpulan dari penelitian tersebut. Serta menarik kesimpulan dan memberi saran yang sesuai dengan penelitian tersebut

3.2 Perancangan Aplikasi

Langkah – langkah dalam perancangan aplikasi deteksi CNN untuk gambar skincare adalah sebagai berikut

(a) Flowchart



Gambar 3-1 Flowchart Proses CNN

Dalam gambar yang ada di atas adalah proses flowchart yang akan dipakai dalam penelitian ini. Terlihat dari proses dengan mengunzip dataset yang sudah dibuat sebelumnya, lalu menginput gambar yang sudah diunzip. Kemudian membuat dataset yang akan menjadi train dan valid dataset. Lalu melakukan preprocessing gambar, dilanjutkan membuat data sequence dan model CNN, dan terakhir melakukan matplotlib untuk plotting accuracy dan val_loss. Dataset berbentuk gambar tersebut dipilih dengan ketentuan berdasarkan merek yang ada di pasaran.

(b) Pembuatan Dataset dan Unzip data

Pada proses awal ini, penulis membuat dataset dari awal dengan mencari gambar sebanyak 1000 gambar yang akan dibagi menjadi 5 folder. Folder pada nama merek skincare tersebut akan menjadi classes yang akan dipakai. Setelah semua proses tersebut, data tersebut akan dizip dengan nama CSV.zip. Nantinya file yang ada dalam file zip tersebut akan diekstrak ke folder yang ada di google colab '/content/data'. Data dari 5 folder berdasarkan nama merek skincare, yang

merupakan brand – brand terkenal yang sering dipakai orang Indonesia. Berikut ini adalah gambar contoh folder yang dipakai dalam penelitian ini.



Gambar 3-2 Gambar Data Folder dalam CSV.zip

Berikut ini adalah tabel nama folder beserta jumlah gambar yang ada dalam CSV.zip. Data tersebut merupakan data yang akan dipakai dalam penelitian tersebut. Gambar yang ada dalam folder tersebut konsisten berjumlah 10 file dalam setiap folder

Tabel 3-1 Tabel isi file dalam penelitian ini

Nama Merek	Jumlah File	Contoh Gambar
Azarine	200	
Nivea	200	

<p>Pond's</p>	<p>200</p>	
<p>Somethinc</p>	<p>200</p>	
<p>Wardah</p>	<p>200</p>	

Setelah semua sudah dizip dan diupload, selanjutnya file akan diunzip dan dilakukan pembagian data dengan menggunakan metode mapping.

```

data = data.map(lambda x,y: (x/255, y))

train_size = int(len(data)*0.5)
val_size = int(len(data)*0.2)
test_size = len(data) - train_size - val_size

train_size

16

train = data.take(train_size)
val = data.skip(train_size).take(val_size)
test = data.skip(train_size+val_size).take(test_size)

```

Gambar 3-3 Potongan Kode Pembagian Dataset

Seperti yang diperlihatkan pada gambar yang ada di atas, langkah pertama dalam pembuatan dataset adalah proses mapping. Sintaks `data = data.map(lambda x, y: (x/255, y))` menunjukkan mapping ke dataset, dimana `x` adalah input dan `y` adalah label class.

Fungsi lambda `lambda x, y: (x/255, y)` membagi input `x` dengan skala 255. Sintaks ini biasanya dilakukan saat normalisasi gambar dengan pixel values antara 0 dan 1.

Selanjutnya, dilakukan penghitungan ukuran data untuk pembagian dataset dengan hasil dari pembagian mapping data. Kode yang ada di bawah ini adalah sintaks pembagian size data dalam dataset ini.

```
train_size = int(len(data)*0.5)
val_size = int(len(data)*0.2)
test_size = len(data) - train_size - val_size
```

Dalam potongan kode ini, `train_size` dihitung sebagai 50% (0,5) dari total panjang kumpulan data. `val_size` dihitung sebagai 20% (0,2) dari total panjang kumpulan data. `test_size` dihitung dengan mengurangkan `train_size` dan `val_size` dari total panjang dataset.

Perhitungan ini membagi dataset menjadi tiga bagian untuk train, validasi, dan test. Train dataset akan berisi 50% data, val dataset akan berisi 20% data, dan data sisanya akan digunakan untuk test dataset.

3.3 Membuat Model CNN

Perancangan sistem klasifikasi pada penelitian ini dilakukan dengan beberapa model state-of-the-art yang telah dipilih penulis untuk mengklasifikasi dataset. Penelitian ini menggunakan dua model pre-trained yaitu MobileNetV3-Small dan CNN biasa untuk hasil yang lebih praktis. Penulis memilih MobileNetV3-Small karena menggunakan algoritma NetAdapt dan Network Architecture Search (NAS) untuk mendeteksi data tersebut. [10]. Penulis ingin mencoba untuk memberikan kenaikan performa pada model tersebut. Berikut ini

adalah simulasi dari MobileNetV3. Tugas pada kedua model tersebut bertujuan untuk mendeteksi merek dan menentukan seberapa cepat dalam memproses data tanpa kehilangan banyak accuracy yang signifikan.

Berikut ini adalah parameter yang dipakai dalam penggunaan CNN secara Tradisional dalam gambar tersebut.

```
# Create a sequential model
model = Sequential()

# Add the first convolutional layer with 32 filters, a 3x3 kernel, and 'relu' activation
model.add(Conv2D(16, (3, 3), activation='relu', input_shape=(256, 256, 3)))

# Add a max pooling layer with a 2x2 pool size
model.add(MaxPooling2D(pool_size=(2, 2)))

# Add another convolutional layer with 64 filters and a 3x3 kernel
model.add(Conv2D(32, (3, 3), activation='relu'))

# Add another max pooling layer
model.add(MaxPooling2D(pool_size=(2, 2)))

# Flatten the feature maps
model.add(Flatten())

# Add a fully connected layer with 128 neurons
model.add(Dense(128, activation='relu'))

# Add the output layer with the appropriate number of classes
model.add(Dense(1, activation='softmax'))

# Compile the model
model.compile(optimizer='adam', loss='categorical_crossentropy', metrics=['accuracy'])
```

Gambar 3-4 Parameter yang dipakai dalam MobileNetV3 small Version

Dense layer (1, softmax) digunakan untuk mengeluarkan output hasil klasifikasi menjadi 1 kelas dengan bantuan fungsi softmax. Fungsi softmax akan menyediakan probabilitas untuk setiap 1 kelas output.

Penggunaan hyperparameter pada masing-masing model juga akan disamakan. Berikut ini adalah tabel hyperparameter yang digunakan serta fungsi parameter tersebut pada penelitian ini.

Tabel 3-2 Tabel Hyperparameter

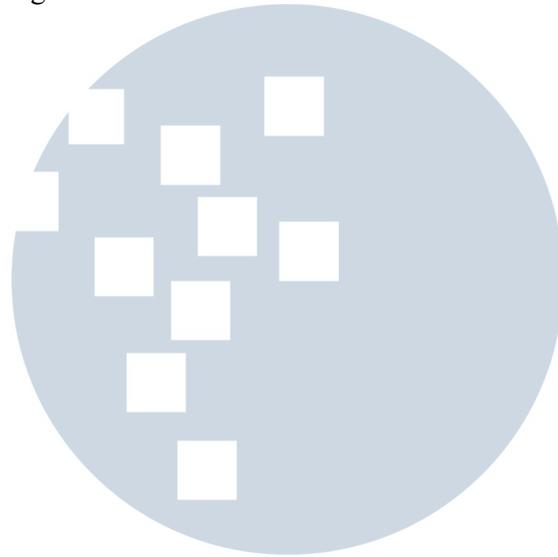
Hyperparameter	Value	Fungsi dan Alasan
Optimizer	Adam	Fungsi Adam Optimizer untuk menemukan nilai minimum lokal dari sebuah fungsi. Adam kombinasi antara RMSprop dan Stochastic

		Gradient Descent dengan momentum. Pemilihan optimizer Adam juga dilihat dari performa yang dihasilkan dari penelitian sebelumnya
Loss	Categorical Crossentropy	Fungsi Categorical Crossentropy Loss digunakan untuk tugas biner dan multiclass. Jenis Cross Entropy ini membutuhkan label untuk dikodekan sebagai kategori..
Epoch	20	Untuk membuat seluruh dataset melalui proses training pada jaringan saraf dalam sekali putaran. Untuk penelitian ini, penulis menetapkan 20 epoch dalam setiap model.

3.4 Evaluasi Model

Evaluasi metrik pada model memiliki peran penting dalam menganalisa performa model yang digunakan, baik untuk mengukur kinerja, membandingkan antara model satu dengan model yang lain, mengoptimalkan model, dan membantu penulis dalam mengambil kesimpulan uji model. Penting untuk memilih evaluasi metrik yang sesuai dengan penelitian yang dilakukan. Berikut evaluasi metrik yang penulis gunakan dalam penelitian ini.

- Penggunaan grafik pembelajaran atau performa untuk melihat kinerja model baik dari sisi kinerja train data maupun validation data. Sumbu x akan menampilkan jumlah epoch yang dipakai saat proses pelatihan. Sumbu y akan menampilkan hasil akurasi dan loss dari hasil pelatihan yang dilakukan. Tujuan penggunaan grafik ini untuk melihat stabilitas dan generalisasi model.



UMN
UNIVERSITAS
MULTIMEDIA
NUSANTARA