**DOKUMEN CD-5**



**KLASIFIKASI JENIS BATIK**

**MENGGUNAKAN MACHINE LEARNING**

**BERBASIS APLIKASI**

Oleh :

Aulia Chusnyriani Sani Z / 1101194043

I Gusti Ngurah Rejski A. P / 1101190017

Nada Fauzia Reviana / 1101194198

Rahmawati Hidayah / 1101194070

**PRODI S1 TEKNIK TELEKOMUNIKASI**

**FAKULTAS TEKNIK ELEKTRO UNIVERSITAS TELKOM BANDUNG**

**2023**

**Dokumentasi Produk Capstone Design**

Lembar Pengesahan Dokumen

|  |  |  |
| --- | --- | --- |
| Judul Capstone Design | : | Klasifikasi Jenis Batik Menggunakan  Machine Learning Berbasis Aplikasi |
| Jenis Dokumen | : | Usulan Gagasan / Desain Konsep Solusi |
| Nomor Dokumen | : | FTE-CD-5 |
| Nomor Revisi | : | …. (sesuaikan dengan jumlah revisi) |
| Tanggal Pengesahan | : | dd/mm/yyy |
| Fakultas | : | Fakultas Teknik Elektro |
| Program Studi | : | S1 Teknik Telekomunikasi |
| Jumlah Halaman | : | …. |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Data Pemeriksaan dan Persetujuan | | | | |
| Ditulis Oleh | Nama | : Aulia Chusnyriani Sani  Zulkarnaen | Jabatan | : Mahasiswa |
| NIM | : 1101194043 | Tanda Tangan |  |
| Nama | : I Gusti Ngurah Rejski A. P | Jabatan | : Mahasiswa |
| NIM | : 1101190017 | Tanda Tangan |  |
| Nama | : Nada Fauzia Reviana | Jabatan | : Mahasiswa |
| NIM | : 1101194198 | Tanda Tangan |  |
|  | Nama | : Rahmawati Hidayah | Jabatan | : Mahasiswa |
|  | NIM | : 1101194070 | Tanda Tangan |  |
| Diperiksa Oleh | Nama | : Nur Ibrahim S.T, M.T. | Jabatan | : Pembimbing 1 |
| Tanggal | : | Tanda Tangan |  |
| Nama | : R Yunendah Nur Fuadah, S.T,  M.T. | Jabatan | : Pembimbing 2 |
| Tanggal | : | Tanda Tangan |  |
|  | Nama | : Nor Kumalasari Caecar  Pratiwi S.T, M. T. | Jabatan | : Pembimbing 3 |
|  | Tanggal | : |  |  |
| Disetujui Oleh | Nama | : | Jabatan | : Penguji 1 |
| Tanggal | : | Tanda Tangan |  |
| Nama | : | Jabatan | : Penguji 2 |
| Tanggal | : | Tanda Tangan |  |

**DAFTAR REVISI DOKUMEN CAPSTONE DESIGN 5**

|  |  |  |  |
| --- | --- | --- | --- |
| Versi,  Tanggal | Revisi | Perbaikan yang dilakukan | Halaman Revisi |
| 1, 22 Maret  2022 | Latar belakang diperjelas akar masalahnya | Ditambahkan data-data yang  mendukung masalah yang diangkat | 3,4 |
| Spesifikasi model mohon dibuat lebih detail | Spesifikasi model sudah ditambahkan detail yang lebih lengkap antara lain:   1. *Flowchart* sistem 2. Fungsi setiap *node* pada sistem | 13,14 |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

**DAFTAR ISI**

# Pengantar

## Ringkasan Isi Dokumen

Dokumen Capstone Design-5 dirancang menjadi beberapa bagian, yaitu pengantar, implementasi sistem, analisis pengerjaan implementasi sistem, hasil akhir, dan Lampiran. Dalam Pengantar terdapat bagian ringkasan isi dokumen, tujuan adanya penulisan dan aplikasi, referensi dan daftar singkatan. Pada bagian implementasi sistem yang berisikan mengenai pengujian dalam setiap sub-sistem yang dikerjakan.Untuk klasifikasi ini maka terdapat sub-sistem bagian pre-processing, ekstrasi fitur, Klasifikasi dan juga *Mobile Application*. Terdapat analisis pengerjaan hasil tiap sub-sistem yang sudah dilakukan dipengujian. Hasil output dari pengerjaan dokumen Capstone Design-5 yaitu dengan berbentuk aplikasi. Adapaun nama dari Mobile Application adalah “BatiQu” yang didukung menggunakan software Android Studio. Serta yang terakhir terdapat lampiran yang berisikan terkait rujukan yang dapat ditambahkan sebagai informasi pendukung.

## Tujuan Penulisan dan Aplikasi/Kegunaan Dokumen

1. Penulisan dokumen Capstone Design-4 ini ditujukan untuk memenuhi kewajiban pada kelas Proposal Tugas Akhir.
2. Merancang, menganalisis dan mengimplementasikan sistem kerja dari klasifikasi jenis Batik menggunakan aplikasi software mobile apps (android).
3. Melakukan klasifikasi jenis batik dengan metode CNN.
4. Membantu masyarakat agar lebih banyak mengenal jenis ragam Batik Indonesia.
5. Melakukan analisis hasil sehingga dapat di identifikasi berdasarkan motif dan jenis Batik Indonesia serta melakukan analisis pada accuracy dan loss berdasarkan parameter metode yang digunakan dalam Machine Learning.

## Referensi

## Daftar Singkatan

|  |  |
| --- | --- |
| **Singkatan** | **Arti** |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |

# Pengujian Performa

Parameter performasi adalah proses yang digunakan sebagai tolak ukur dari penilaian dan mengevaluasi dari kinerja suatu sistem melalui beberapa sub-sistem yang telah dirancang sebelumnya. Tujuan utama dalam pengujian adalah untuk memastikan perancangan yang telah dibuat dapat berjalan sesuai dengan kebutuhan. Untuk mengetahui kualitas sistem dapat dilihat dari pengujian performa yang akan dilakukan yaitu pada bagian pre-processing, Feature Extraction dan Klaisifkasi.

## Pengujian Performa *Pre-Processing*

### Langkah Pengujian

Tahap pre-processing data merupakan awal suatu proses yang dilakukan dengan memiliki tujan untuk mendapatkan data citra yang lebih baik dan lebih banyak yang nantinya data output tersebut akan dilanjutkan pada tahap berikutnya. Berikut merupakan penjelasan dari parameter

1. Rescale

*Rescale* merupakan parameter untuk mengubah intensitas piksel dalam gambar. Dimana dalam nilai setiap piksel dalam gambar memiliki rentang 0 sampai dengan 255. Nilai 0 merupakan warna hitam dan nilai 255 merupakan warna putih. *Rescale* digunakan untuk melakukan normalisasi piksel dalam gambar dengan membagi setiap piksel dengan 1/255 serta dapat memudahkan dalam proses training data.

1. Brigthness

*Brightness* merupakan parameter yang digunakan untuk mengatur rentang kecerahan dengan menghasilkan variasi gambar dengan tingkat kecerahan yang berbeda-beda. Pada ‘*brightness*’ yang digunakan memiliki *tupple*. Dimana *tupple* berisikan dua nilai yaitu (0.1, 1.5), maka dapat dijelaskan bahwa kecerahan gambar bervariasi antara 50% sampai dengan 150% dari kecerahan asli data gambar yang dimasukkan.

1. Rotation

*Rotation* merupakan parameter yang berguna untuk mengubah rotasi gambar secara acak. Tujuan dari rotation untuk membantu dalam meningkatkkan variasi citra gambar guna mempermudah dalam pembelajaran model selanjutnya. Jika ‘*rotation*’ bernilai 0.1 yang berarti 10% maka menunjukkan gambar dapat diputar dalam rentang -10 hingga 10 derajat secara acak.

1. Width

*Width* merupakan parameter yang digunakan untuk melakukan pergeseran posisi pada lebar gambar. Pergeseran dilakukan pada gambar acak dimana gambar tersebut akan diubah posisinya ke kiri atau kekanan. Contohnya, jika menggunakan ‘*width*’ bernilai 0.2 maka citra gambar digeser secara acak hingga 20% dari lebar total dari gambar sebelumnya.

1. Zoom

Zoom merupakan parameter yang digunakan untuk memperbesar dan memperkecil citra gambar secara acak. Jika menggunakan zoom\_range dengan nilai 0.1, maka gambar akan diperbesar sejauh 10% . Penggunaan zoom pada Klasifikasi ini berbentuk nilai float tunggal.

1. Horizontal

*Horizontal* merupakan parameter yang berguna untuk membalik gambar secara horizontal dengan acak. Pada tahapan *horizontal* jika diatur sebagai *True*, maka gambar akan di balik secara *horizontal* dengan kemungkinan 50% yang berarti gambar juga bisa tidak terbalik secara sempurna. Namun, jika *horizontal* bernilai *False* maka dapat dipastikann bahwa fungsi *horizontal* tidak akan digunakan atau tidak akan dibalik.

### Hasil Pengujian

* + - 1. **Pengaruh Rotation terhadap sistem**

Tahap pertama yang dilakukan dalam pengujian rotation pada sistem ini adalah mengubah nilai rotation. Nilai rotation yang akan digunakan ada tiga yaitu 0.1, 0.2 dan 0.25 dengan menggunakan arsitektur Mobilenet. Setiap rotation akan menggunakan parameter berupa target size berukuran 300 X 300, dengan batch size 64. Hasil Rotation yang terbaik akan digunakan pada pengujian selanjutnyan dengan arstitektur yang akan diujia pada pengujian selanjutnya. Adapun hasil Rotation terbaik dapat dilihat pada tabel

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Rotation | Train Accuracy | Train Loss | Validation Accuracy | Validation Loss | Test Accuracy |
| 0.1 | 95,21% | 0,1778 | 58,33% | 1,2548 | 66,67% |
| 0.2 | 91,87% | 0,2223 | 59,17% | 1,4802 | 63,33% |
| 0.25 | 87,92% | 0.2985 | 60,83% | 1,4468 | 66,67% |

Pada tabel dijelaskan bahwa hasil pengujian rotation pada augmentasi data, jika nilai dari rotation semakin kecil maka nilai dari test akurasi menjadi naik dengan akurasi train yang bagus. Dari hasil pengujian rotation di tabel hasil akurasi terbaik ada pada rotation dengan nilai 0,1 yang mana akurasi dicapai sebesar 95,21% dari data train dengan akurasi test bernilai 66,67% pada kolom berwarnakan biru. Dengan nilai rotation 0,1 menunjukkan bahwa model mampu mempelajari dengan baik pada rotation sehingga dapat lanjut ke pengujian selanjutnya.

* + - 1. Pengaruh Zoom terhadap sistem

Tahap kedua yang dilakukan dalam pengujian zoompada sistem ini adalah mengubah nilai zoom dengan tiga nilai pengujian yaitu 0.1, 0.2 dan 0.25 dengan menggunakan arsitektur Mobilenet. Tahapan ini dilakukan setelah mendapatkan hasil dari pengujian sebelumnya. Setiap zoom akan menggunakan parameter berupa target size berukuran 300 X 300, dengan batch size 64. Hasil dari pengujian width yang terbaik akan digunakan pada pengujian selanjutnyan dengan arstitektur yang akan diuji pada pengujian selanjutnya. Adapun hasil Rotation terbaik dapat dilihat pada tabel

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Zoom | Train Accuracy | Train Loss | Validation Accuracy | Validation Loss | Test Accuracy |
| 0.1 | 95,21% | 0,1778 | 58,33% | 1,2548 | 66,67% |
| 0.2 | 88,54% | 0,3518 | 67,50% | 1,4311 | 56,67% |
| 0.25 | 90,21% | 0,2879 | 62,50% | 1,3210 | 73,33% |

Menjelaskan pada tabel diatas, bahwa tahap kedua pada pengujian ini akan menggunakan zoom bernilai 0,25 yang dimana memiliki akurasi pada train sebesar 90,21% dan memiliki nilai pada akurasi test sebesar 73,33%. Dapat disimpulkan bahwa tahap pengujian kedua sudah bisa dilanjutkan ke tahap pengujian selanjutnya untuk nantinya dapat memasukin pada pengujian ekstrasi fitur dan juga Klasifikasi pada penelitian ini.

* + - 1. PengaruhWidth terhadap sistem

Tahapan terakhir dari pengujian augmentasi ini adalah dengan mengubah nilai zoom yang bermemiliki tiga nilai pengujian yaitu 0.1, 0.2 dan 0.25 menggunakan arsitektur Mobilenet. Tahapan pengujian dilakukan setelah mendapatkan hasil terbaik dari width dan rotation. Pengujian zoom akan menggunakan parameter berupa target size berukuran 300 X 300, dengan batch size 64. Hasil dari pengujian zoom yang terbaik akan digunakan pada pengujian selanjutnyan dengan arstitektur yang akan diuji pada pengujian selanjutnya. Adapun hasil zoom terbaik dapat dilihat pada tabel

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Width | Train Accuracy | Train Loss | Validation Accuracy | Validation Loss | Test Accuracy |
| 0.1 | 95,21% | 0,1778 | 58,33% | 1,2548 | 66,67% |
| 0.2 | 83,75% | 0,4548 | 64,17% | 1,3976 | 50% |
| 0.25 | 83,54% | 0,4758 | 68,33% | 1,3285 | 68,33% |

Pada tabel diatas, menjelaskan bahwa tahap ketiga pada pengujian ini akan menggunakan width bernilai 0,1 yang dimana memiliki akurasi pada train sebesar 95,21% dan memiliki nilai pada akurasi test sebesar 66,76% Sehingga dapat disimpulkan bahwa tahap pengujian ketiga sudah bisa dilanjutkan ke pengujian ekstrasi fitur dan juga Klasifikasi pada penelitian ini. Untuk dapat memastikan bahwa Klasifikasi pada penelitian ini berhasil dijalankan.

## Pengujian Performa *Feature Extraction*

*Feature extraction* memiliki beberapa parameter yang bisa diuji dan dibandingkan untuk mengetahui parameter mana yang dapat menghasilkan performa yang paling optimal. Parameter dari *feature extraction* tersebut di antaranya *weight*, ‘include\_top’, ‘input\_shape’, dan *pooling layer.*

1. Parameter *weights*

*Weights* merupakan parameter yang digunakan untuk mengatur penggunaan bobot *pre-trained* yang digunakan pada model klasifikasi yang digunakan.

Langkah pengujian :

Skenario pengujian parameter *weights* dilakukan dengan menggunakan tiga jenis *weights* yaitu *imagenet, noisy-student*, dan *none* atau *random*. Pengujian akan dijalankan dengan kondisi nilai *epoch*=100 dan akan diamati hasil akurasi dan loss dari setiap jenis *weights* yang dijalankan.

1. Parameter ‘include\_top’

‘include\_top’ merupakan parameter yang mengatur kondisi output yang terhubung dengan *layer fully connected* (*top layer*). Kondisi ini diatur dengan menulis argumen ‘include\_top=True’ untuk menyertakan *top layer* dengan output, dan ‘include\_top=False’ untuk menghilangkan *top layer* tersebut.

Langkah pengujian :

Skenario pengujian parameter ‘include\_top’ dilakukan dengan menggunakan dua jenis argumen yaitu ‘include\_top=False’ dan ‘include\_top=True’. Pengujian akan dijalankan dengan kondisi nilai *epoch*=100 untuk melihat perbandingan dari hasil akurasi dan lossnya.

1. Parameter ‘*input\_shape’*

‘input\_shape’ merupakan parameter yang digunakan untuk mengatur dimensi dari *input* citra yang digunakan. Untuk nilai *input shape* yang ingin digunakan dapat diatur secara bebas sesuai keinginan.

Langkah pengujian :

Skenario pengujian parameter ‘input\_shape dilakukan dengan menggunakan nilai (300, 300, 3), (600, 600, 3), (900, 900, 3). Pengujian akan dijalankan dengan kondisi nilai *epoch*=100 untuk melihatperbandingan dari hasil akurasi dan lossnya.

1. Parameter *pooling layer*

*Pooling layer* merupakan parameter yang mengatur jenis operasi pooling yang berfungsi untuk mengurangi dimensi spasial sehingga nilai-nilai fitur yang dihasilkan *layer* konvolusi dapat digabung menjadi satu nilai/dimensi.

Langkah pengujian :

Skenario pengujian parameter *pooling layer* dilakukan dengan menggunakan dua jenis metode yaitu *max pooling* dan *average pooling*. Pengujian akan dijalankan dengan kondisi nilai *epoch=*100 dan akan menggunakan beberapa nilai *pool size* yang berbeda-beda. Dari pengujian tersebut akan dilihat perbandingan dari hasil akurasi dan lossnya.

### Hasil Pengujian *Feature Extraction* pada Arsitektur ResNet152 V2

#### Parameter Weights

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Weights | Train Accuracy | Train Loss | Validation Accuracy | Validation Loss | Test Accuracy |
| ImageNet | 99,7912% | 0,0083 | 76,5957% | 2,5510 | 81,6666 % |
| Noisy-Student |  |  |  |  |  |
| None/Random |  |  |  |  |  |

#### Parameter ‘include\_top’

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Include\_top | Train Accuracy | Train Loss | Validation Accuracy | Validation Loss | Test Accuracy |
| True | 77,0354% | 0,7043 | 63,3333% | 0,8552 | 64,9999% |
| False | 99,7912% | 0,0083 | 76,5957% | 2,5510 | 81,6666 % |

#### Parameter ‘input\_shape’

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Input shape | Train Accuracy | Train Loss | Validation Accuracy | Validation Loss | Test Accuracy |
| (300, 300, 3) | 99,7912% | 0,0083 | 76,5957% | 2,5510 | 81,6666 % |
| (600, 600, 3) |  |  |  |  |  |
| (900,900, 3) |  |  |  |  |  |

#### Parameter Pooling

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Pooling Layer | Train Accuracy | Train Loss | Validation Accuracy | Validation Loss | Test Accuracy |
| Max Pooling |  |  |  |  |  |
| Average Pooling |  |  |  |  |  |
| None | 99,7912% | 0,0083 | 76,5957% | 2,5510 | 81,6666 % |

### Hasil Pengujian *Feature Extraction* pada Arsitektur MobileNet V1

#### Parameter Weights

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Weights | Train Accuracy | Train Loss | Validation Accuracy | Validation Loss | Test Accuracy |
| ImageNet | 45,625% | 2,553 | 37,5% | 2,8325 | 26,6666% |
| Noisy-Student |  |  |  |  |  |
| None/Random | 16,6666% | 1,7917 | 16,6666% | 1,7917 | 16,6666% |

#### Parameter ‘include\_top’

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Include\_top | Train Accuracy | Train Loss | Validation Accuracy | Validation Loss | Test Accuracy |
| True | 17,2916% | 1,8256 | 20,8333% | 1,8290 | 8,3333% |
| False | 44,9999% | 1,9969 | 37,5% | 2,3646 | 51,6666% |

Pada tabel hasil pengujian ‘include\_top terlihat bahwa secara umum hasil akurasi dan loss terbaik didapatkan ketika argumen ‘include\_top=False’ dengan nilai akurasi tes mencapai 51,6666% yang ditandai dengan warna biru.

#### Parameter ‘input\_shape’

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Input shape | Train Accuracy | Train Loss | Validation Accuracy | Validation Loss | Test Accuracy |
| (300, 300, 3) | 44,9999% | 1,9969 | 37,5% | 2,3646 | 51,6666% |
| (600, 600, 3) | 39,1666% | 2,8535 | 31,6666% | 4,0261 | 30,2517% |
| (900,900, 3) |  |  |  |  |  |

#### Parameter Pooling Layer

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Pooling Layer | Train Accuracy | Train Loss | Validation Accuracy | Validation Loss | Test Accuracy |
| Max Pooling |  |  |  |  |  |
| Average Pooling |  |  |  |  |  |
| None | 45,625% | 2,553 | 37,5% | 2,8325 | 26,6666% |

## Pengujian Performa *Classification*

Tahap klasifikasi merupakan proses pengelompokan suatu citra ke dalam kelas-kelas tertentu. Pengujian ini memiliki cara kerja yaitu dataset batik yang telah dikumpulkan akan diklasifikasikan sesuai kelasnya. Hal ini didukung menggunakan perancangan metode *Convolutional Neural Network* atau CNN dengan arsitektur ResNet152 V2. Beberapa hyperparameter dan parameter yang digunakan di antaranya *optimizer*, *learning rate, callbacks*, *batch size*, *epoch.*

1. Optimizer

Optimizer merupakan algoritma yang digunakan untuk memaksimalkan bobot model agar menekan *loss* yang didapatkan antara nilai keluaran dengan nilai dari neuron. Optimizer yang akan diimplementasikan adalah optimizer Adam.

Langkah pengujian:

Skenario pengujian pada optimizer dilakukan dengan menggunakan tiga jenis *optimizer* yaitu Adam*,* Adamax, dan RMSprop.

1. Learning Rate

Learning rate adalah salah satu hyperparameter yang digunakan untuk membantu saat proses *training.* Learning rate sendiri dapat diimplementasikan untuk menghitung nilai ketepatan bobot citra atau objek gambar. Pemilihan pada nilai learning rate mempengaruhi kecepatan dan kestabilan dalam pelatihan model.

Langkah pengujian: Skenario pengujian pada learning rate dilakukan dengan menggunakan tiga jenis nilaiyaitu 0.01, 0.001, 0.0001.

1. Callbacks

*Callbacks* digunakan untuk menyimpan dan membatasi hasil akurasi terbaik saat melakukan proses *training*. Fungsi callbacks diterapkan dengan fungsi *checkpointer* yang memiliki kesamaan peran yaitu menyimpan hasil model saat pelatihan dengan performa terbaik.

1. Batch Size

*Batch size* bertugas untuk menentukan jumlah data pelatihan dalam setiap iterasi saat proses pelatihan. Nilai dari batch size dapat memengaruhi tingkat akurasi yang akan dihasilkan nantinya.

Langkah pengujian: Skenario pengujian pada batch size dilakukan dengan menggunakan tiga jenis nilaiyaitu 16, 32*,* dan 64.

1. Epoch

*Epoch* dapat menentukan jumlah iterasi saat pemrosesan model untuk mengetahui tingkat akurasi yang akan dicapai menghasilkan nilai yang baik atau tidak. Nilai *epoch* juga dapat memengaruhi pembelajaran dari model dan dapat dievaluasi keoptimalan kerjanya.

Langkah pengujian: Skenario pengujian pada epoch dilakukan dengan menggunakan tiga jenis nilaiyaitu 50, 100*,* dan 150.

### Hasil Pengujian *Classification* pada Arsitektur ResNet152 V2

#### Optimizer

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Width | Train Accuracy | Train Loss | Validation Accuracy | Validation Loss | Test Accuracy |
| Adam | 95.42% | 0.1504 | 83.33% | 1.6369 | 80% |
| Adamax | 97.5% | 0.2706 | 81.67% | 0.9788 | 75% |
| RMSprop | 97.92% | 0.1596 | 82.5% | 1.1782 | 83.33% |

* + - 1. Learning Rate

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Width | Train Accuracy | Train Loss | Validation Accuracy | Validation Loss | Test Accuracy |
| 0.01 | 98.96% | 0.0334 | 83.33% | 1.0615 | 81.67% |
| 0.001 | 99.79% | 0.0082 | 82.5% | 1.3790 | 86.67% |
| 0.0001 | 97.92% | 0.1596 | 82.5% | 1.1782 | 83.33% |

* + - 1. Batch Size

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Width | Train Accuracy | Train Loss | Validation Accuracy | Validation Loss | Test Accuracy |
| 16 |  |  |  |  |  |
| 32 | 99.79% | 0.0082 | 82.5% | 1.3790 | 86.67% |
| 64 |  |  |  |  |  |

* + - 1. Epoch

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Width | Train Accuracy | Train Loss | Validation Accuracy | Validation Loss | Test Accuracy |
| 50 |  |  |  |  |  |
| 100 |  |  |  |  |  |
| 150 |  |  |  |  |  |

## Pengujian Performa *Mobile Application*

*Mobile application* memiliki beberapa parameter yang diuji untuk melihat apakah dari parameter yang diujikan dapat berjalan dengan lancar. Parameter yang dapat diujikan yaitu dari fitur utama *Mobile Application* yaitu parameter kamera, *gallery* dan halaman informasiserta dari fitur tambahan yaitu *About Us*

### Pengujian pada Fitur Utama Aplikasi

#### Pengujian Tombol Kamera

Untuk pengujian tombol kamera pada aplikasi dilakukan pada Android Studio dan handphone. Langkah pengujian sebagai berikut :

1. Buka *Android Studio*
2. Buka Project yang telah dibuat sebelumnya
3. Tekan tombol play untuk run pada emulator yang disediakan oleh *Android Studio*
4. Tekan tombol Mulai untuk memulai aplikasi
5. Tekan tombol kamera dan mulai potret dari simulasi yang telah disediakan oleh *Android Studio*
6. Gambar yang telah dipotret akan ditampilkan kembali pada *ImageView* aplikasi
7. Setelah melakukan simulasi pada emulator berjalan, selanjutnya dapat dilakukan *debugging* pada handphone
8. Debugging pada handphone dilakukan menggunakan kabel USB sebagai perantara antara *Android Studio* dengan handphone

#### Hasil Pengujian

Berikut pengujian yang telah dilakukan untuk tombol kamera

#### Pengujian Tombol Gallery

Untuk pengujian tombol *gallery* pada aplikasi dilakukan pada Android Studio dan handphone. Langkah pengujian sebagai berikut :

1. Buka Android Studio
2. Buka Project yang telah dibuat sebelumnya
3. Tekan tombol play untuk run pada emulator yang disediakan oleh *Android Studio*
4. Tekan tombol Mulai untuk memulai aplikasi
5. Tekan tombol galeri dan mulai pilih gambar dari galeri yang telah disediakan oleh Android Studio
6. Gambar yang telah dipilih akan ditampilkan kembali pada ImageView aplikasi
7. Setelah melakukan simulasi pada emulator berjalan, selanjutnya dapat dilakukan debugging pada handphone
8. *Debugging* pada handphone dilakukan menggunakan kabel USB sebagai perantara antara Android Studio dengan handphone

#### Hasil Pengujian

Berikut pengujian yang telah dilakukan untuk tombol galeri

#### Pengujian Halaman Informasi

Untuk pengujian tombol *predict* pada aplikasi dilakukan pada Android Studio dan handphone. Pengujian ini dilakukan setelah melakukan pengambilan gambar untuk selanjutnya di klasifikasikan oleh *Machine Learning*. Langkah pengujian sebagai berikut :

1. Buka Android Studio
2. Buka Project yang telah dibuat sebelumnya
3. Tekan tombol play untuk run pada emulator yang disediakan oleh Android Studio
4. Tekan tombol Mulai untuk memulai aplikasi
5. Tekan tombol kamera atau galeri dan mulai pilih gambar
6. Dari gambar yang dipilih akan langsung di proses oleh Machine Learning dan selanjutnya akan diarahkan pada halaman informasi yang dibuat sesuai kelas batik.
7. Setelah melakukan simulasi pada emulator berjalan, selanjutnya dapat dilakukan debugging pada handphone
8. Debugging pada handphone dilakukan menggunakan kabel USB sebagai perantara antara Android Studio dengan handphone

#### Hasil Pengujian

Pengujian dilakukan dalam dua versi yaitu versi dengan menggunakan model resnet dan versi dengan menggunakan model mobilenet. Hasil pengujian dari dua versi tersebut dapat dilihat pada table di bawah ini.

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| No | Jenis Batik | Resnet 152 v2 | | | | |
| 1 | 2 | 3 | 4 | 5 |
| 1 | Kawung | Kawung | Kawung | Kawung | Kawung | Kawung |
| 2 | Parang | Parang | Parang | Parang | Parang | Parang |
| 3 | Tambal | Tambal | Tambal | Tambal | Tambal | Parang |
| 4 | Nitik | Nitik | Nitik | Ceplok | Nitik | Nitik |
| 5 | Megamendung | Megamendung | Megamendung | Megamendung | Megamendung | Megamendung |
| 6 | Ceplok | Ceplok | Ceplok | Ceplok | Ceplok | Kawung |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| No | Jenis Batik | MobileNet | | | | |
| 1 | 2 | 3 | 4 | 5 |
| 1 | Kawung | Ceplok | Kawung | Kawung | Kawung | Kawung |
| 2 | Parang | Parang | Megamendung | Parang | Parang | Parang |
| 3 | Tambal | Tambal | Nitik | Tambal | Tambal | Megamendung |
| 4 | Nitik | Ceplok | Nitik | Parang | Nitik | Nitik |
| 5 | Megamendung | Megamendung | Megamendung | Megamendung | Megamendung | Megamendung |
| 6 | Ceplok | Kawung | Ceplok | Megamendung | Tambal | Kawung |

Pengujian dilakukan lima kali dengan gambar yang diambil dari kamera dan galeri. Pengujian pertama dan kedua dilakukan jika gambar diambil dari internet melalui galeri dan gambar tersebut sudah dilatih sebelumnya oleh *Machine Learning*, dilanjutkan dengan pengujian ketiga dan keempat menggunakan gambar yang diambil dari kamera melalui galeri. Untuk pengujian yang terakhir menggunakan gambar yang diambil dari internet namun gambar tersebut belum dilatih oleh *Machine Learning*.

Pada dua model yang diuji pada aplikasi, dapat dilihat bahwa model MobileNet banyak memiliki kesalahan dalam menebak gambar batik yang diberikan. Pada model Resnet terdapat sedikit kesalahan pada batik Tambal dan batik Ceplok namun hal itu dikarenakan pada gambar batik tambal terdapat corak batik parang yang mendominasi sehingga gambar tersebut terbaca sebagai batik parang. Untuk batik ceplok, terdapat jenis ceplok kawung dan hal tersebut dapat mempengaruhi hasil jenis batik yang dikeluarkan.

### Pengujian pada Fitur Tambahan Aplikasi

#### Pengujian Tombol About Us

Untuk pengujian tombol *About Us* pada aplikasi dilakukan pada Android Studio dan handphone. Langkah pengujian sebagai berikut :

1. Buka Android Studio
2. Buka Project yang telah dibuat sebelumnya
3. Tekan tombol play untuk run pada emulator yang disediakan oleh Android Studio
4. Tekan tombol *About Us* yang terletak pada bawah tombol Mulai
5. Muncul tampilan antarmuka dari *About Us*

#### Hasil Pengujian

Berikut pengujian yang telah dilakukan untuk tombol *About Us*

# Analisis Hasil Pengujian

## Analisis Hasil pengujian Pre-processing

Hasil pengukuran dengan menggunakan arsitektur Mobilenet pada pengujian awal *pre-processing*. Menjelaskan bahwa augmentasi yang digunakan untuk mendapatkan hasil yang bagus pada akurasi yaitu ketika rotation bernilai 0,1 dengan zoom yang bernilai 0,25 dan juga width yang bernilai 0,1 Hasil akurasi test tersebut menggunakan kombinasi dengan karena dengan target size berukuran 300 X 300, dengan batch size 64. Maka dengan itu nilai yang sudah diuji didapatkan tingkat akurasi pada data training mencapai lebih dari 90% dan tingkat akurasi pada test menghasilkan nilai lebih dari 60%. Hal ini menunjukkan bahwa citra dengan augmentasi bekerja dengan baik pada arsitektur Mobilenet . Namun, arsitektur ini akan diuji Kembali pada ekstrasi ciri dan akan mengeluarkan hasil pada Klasifikasi model sistem yang dibuat.

## Analisis Hasil *Feautre Extraction*

## Analisis Hasil Pengujian *Mobile Application*

### Analisis Hasil Pengujian Kamera

Pada pengujian ini, kamera dalam aplikasi berhasil dijalan kan dan bisa melakukan proses pengambilan gambar menggunakan kamera handphone

### Analisis Hasil Pengujian Galeri

Pada pengujian ini, galeri dalam aplikasi berhasil dijalan kan dan bisa melakukan proses pengambilan gambar menggunakan galeri hp baik dari penyimpanan internal maupun hasil *download* gambar

### Analisis Hasil Pengujian Halaman Informasi

Dari dua versi yang telah diuji, dapat dilihat bahwa model Resnet 152 dapat menebak tiap jenis batik 100% benar. Namun kelemahan dari model ini adalah proses klasifikasi yang memakan waktu sehingga terkesan bahwa aplikasi ini tidak dapat berjalan semestinya. Dibandingkan dengan model Mobilenet yang memiliki kecepatan dalam proses klasifikasi, model ini masi tidak dapat menebak beberapa gambar jenis batik yang di ambil.

### Analisis Hasil Pengujian *About Us*

Pada pengujian ini, tampilan antarmuka *About Us* yang dibuat telah tampil pada aplikasi yang dibuat

# Lampiran

# Berikut merupakan lampiran biaya yang dikeluarkan untuk pembuatan box pada data uji: