



UNIVERSITAS DIPONEGORO

**ANALISIS PERBANDINGAN KINERJA *DEEP NEURAL NETWORK*
MENGUNAKAN MODEL XCEPTION DAN DENSENET PADA
KLASIFIKASI RAS KUCING**

TUGAS AKHIR

**Diajukan sebagai salah satu syarat untuk memperoleh gelar
Sarjana Teknik**

**RIO JULIAN AZIS PRATAMA
21120116120011**

**FAKULTAS TEKNIK
DEPARTEMEN TEKNIK KOMPUTER
SEMARANG
JUNI 2020**

HALAMAN PENGESAHAN

Tugas Akhir ini diajukan oleh

Nama : Rio Julian Azis Pratama
NIM : 21120116120011
Jurusan/Program Studi : Teknik Komputer
Judul Tugas Akhir : Analisis Perbandingan Kinerja *Deep Neural Network*
Menggunakan Model Xception dan DenseNet pada
Klasifikasi Ras Kucing.

Telah berhasil dipertahankan di hadapan Tim Penguji dan diterima sebagai bagian persyaratan yang diperlukan untuk memperoleh gelar Sarjana Teknik pada Jurusan/Program Studi Teknik Komputer, Fakultas Teknik, Universitas Diponegoro.

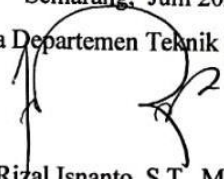
TIM PENGUJI

Pembimbing I : Dr. R. Rizal Isnanto, S.T., M.M., M.T.
Pembimbing II : Yudi Eko Windarto, S.T., M.Kom.
Ketua Penguji : Kurniawan Teguh Martono, S.T., M.T.
Anggota Penguji : Dania Eridani, S.T., M.Eng



Semarang, Juni 2020

Ketua Departemen Teknik Komputer



Dr. R. Rizal Isnanto, S.T., M.M., M.T.


NIP. 197007272000121001

HALAMAN PERNYATAAN ORISINALITAS

Tugas Akhir ini adalah hasil karya saya sendiri, dan semua sumber baik yang dikutip maupun yang dirujuk telah saya nyatakan dengan benar

Nama : Rio Julian Azis Pratama

NIM : 21120116120011

Tanda Tangan : 

Tanggal : Juni 2020

**HALAMAN PERNYATAAN PERSETUJUAN PUBLIKASI
TUGAS AKHIR UNTUK KEPENTINGAN AKADEMIS**

Sebagai civitas akademika Universitas Diponegoro, saya yang bertanda tangan di bawah ini:

Nama : Rio Julian Azis Pratama
NIM : 21120116120011
Jurusan/Program Studi : Teknik Komputer
Fakultas : Teknik
Jenis Karya : Tugas Akhir

demi pengembangan ilmu pengetahuan, menyetujui untuk memberikan kepada Universitas Diponegoro **Hak Bebas Royalti Noneksklusif** (*Non-exclusive Royalty Free Right*) atas karya ilmiah saya yang berjudul :

Analisis Perbandingan Kinerja *Deep Neural Network* Menggunakan Model Xception dan DenseNet pada Klasifikasi Ras Kucing.

beserta perangkat yang ada (jika diperlukan). Dengan Hak Bebas Royalti/Noneksklusif ini, Universitas Diponegoro berhak menyimpan, mengalihmedia/formatkan, mengelola dalam bentuk pangkalan data (*database*), merawat dan memublikasikan tugas akhir saya selama tetap mencantumkan nama saya sebagai penulis/pencipta dan sebagai pemilik Hak Cipta.

Demikian pernyataan ini saya buat dengan sebenarnya.

Dibuat di : Semarang

Pada Tanggal : Juni 2020

Yang menyatakan



Rio Julian Azis Pratama

KATA PENGANTAR

Puji syukur Alhamdulillah ke hadirat Allah SWT yang telah melimpahkan rahmat-Nya sehingga Penulis dapat menyelesaikan laporan Tugas Akhir yang berjudul “**Analisis Perbandingan Kinerja *Deep Neural Network* Menggunakan Model Xception dan DenseNet pada Klasifikasi Ras Kucing**”.

Laporan Tugas Akhir ini merupakan salah satu syarat untuk menyelesaikan studi di Departemen Teknik Komputer Fakultas Teknik Universitas Diponegoro. Diharapkan penyusunan laporan Tugas Akhir ini dapat memberikan manfaat dalam bidang Pendidikan.

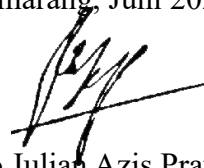
Dalam penyusunan laporan Tugas Akhir ini Penulis banyak mendapatkan dukungan, doa bimbingan dan dorongan dari berbagai pihak. Oleh karenanya, melalui kesempatan ini Penulis menyampaikan ucapan terima kasih kepada:

1. Bapak Dr. R. Rizal Isnanto, S.T., M.M., M.T. selaku Ketua Departemen Teknik Komputer dan juga selaku dosen pembimbing I, yang telah memberikan petunjuk serta bimbingan dalam pembuatan Tugas Akhir.
2. Bapak Yudi Eko Windarto, S.T., M.Kom. selaku dosen pembimbing II, yang telah memberikan petunjuk serta bimbingan dalam pembuatan Tugas Akhir.
3. Ibu Ike Pertiwi Windasari, S.T., M.T. selaku dosen Koordinator Tugas Akhir, yang telah memberikan petunjuk serta bimbingan pelaksanaan rangkaian kegiatan Tugas Akhir.
4. Bapak dan Ibu dosen Jurusan Teknik Komputer yang telah memberikan ilmu pengetahuan dan dorongan untuk terus berkarya.
5. Kedua orang tua dan keluarga yang selalu mendoakan dan mendukung penulis dalam menyelesaikan Tugas Akhir.
6. Sahabat-sahabat Penulis selama melaksanakan studi di Departemen Teknik Komputer yang selalu siap mendukung dan membantu Penulis setiap saat, yaitu Faisal, Alvin, Hisyam, Shodiq, Khoderi, Usman, Kemal yang selalu menguatkan dan saling membantu selama penulis mengerjakan Tugas Akhir ini.

7. Enjela Pratiwi yang selalu memberikan semangat, dan selalu memberikan motivasi kepada Penulis untuk segera menyelesaikan Tugas Akhir ini.
8. Teman-teman Teknik Komputer, khususnya angkatan 2016 yang selalu mendukung dan memberikan semangat kepada Penulis.
9. Staf Tata Usaha Departemen Teknik Komputer yang telah bekerja dengan baik.
10. Serta semua pihak yang tidak dapat disebutkan satu persatu yang telah membantu hingga terselesaikannya Tugas Akhir ini.

Penulis menyadari bahwa laporan Tugas Akhir ini masih perlu perbaikan, kritik, saran dan masukan di masa yang akan datang demi sempurnanya Laporan Tugas Akhir ini. Penulis berharap laporan Tugas Akhir ini dapat bermanfaat bagi Penulis maupun bagi orang banyak. Akhir kata Penulis mengucapkan terima kasih.

Semarang, Juni 2020



Rio Julian Azis Pratama

DAFTAR ISI

HALAMAN JUDUL	i
HALAMAN PENGESAHAN	ii
HALAMAN PERNYATAAN ORISINALITAS	iii
HALAMAN PERNYATAAN PERSETUJUAN PUBLIKASI.....	iv
KATA PENGANTAR.....	v
DAFTAR ISI.....	vii
DAFTAR GAMBAR	ix
DAFTAR TABEL.....	xiii
ABSTRAK.....	xiv
ABSTRACT.....	xv
BAB I PENDAHULUAN.....	1
1.1 Latar Belakang	1
1.2 Rumusan Masalah	2
1.3 Tujuan Penelitian.....	2
1.4 Batasan Masalah.....	2
1.5 Manfaat Penelitian.....	3
1.6 Sistematika Penulisan.....	3
BAB II KAJIAN PUSTAKA.....	5
2.1 Kajian Hasil Penelitian Terdahulu	5
2.2 Landasan Teori.....	6
2.2.1 Artificial Intellegence	6
2.2.2 Deep Learning.....	7
2.2.3 Convolutional Neural Network (CNN).....	8
2.2.4 Keras dan Tensorflow	16
2.2.5 <i>Transfer Learning</i>	17
2.2.6 Pembelajaran Terawasi (<i>Supervised Learning</i>)	17
2.2.7 Regularisasi dan Dropout.....	18
2.2.8 Fine-Tuning.....	18
2.2.9 Adam.....	19
2.2.10 Kucing.....	20

2.2.11	Xception	21
2.2.12	DenseNet-201	22
2.2.13	Flask	23
BAB III	METODOLOGI PENELITIAN	25
3.1	Populasi dan Sampel	25
3.2	Variabel dan Definisi Operasional Variabel	25
3.3	Jenis dan Sumber Data	26
3.4	Metode Analisis Data	26
3.5	Tahapan Penelitian	26
3.5.1	Kajian Pustaka	26
3.5.2	Pengumpulan Data dan Pengolahan Data	26
3.5.3	Perancangan Model	29
3.5.4	Pelatihan Model	36
3.5.5	Pengujian Model	42
3.5.6	Implementasi pada Aplikasi Web	47
BAB IV	ANALISIS HASIL PENGUJIAN	49
4.1.	Skema Himpunan Data	49
4.2.	Parameter Laju Pembelajaran (<i>Learning Rate</i>)	60
4.3.	Parameter Jumlah <i>Epoch</i>	72
4.4.	Skema Fine-Tuning	84
4.5	Implementasi pada Aplikasi Web	95
BAB V	KESIMPULAN DAN SARAN	98
5.1	Kesimpulan	98
5.2	Saran	99
DAFTAR PUSTAKA		100

DAFTAR GAMBAR

Gambar 2. 1 Ekstraksi Ciri.....	8
Gambar 2. 2 Konsep dari Peta Respon	9
Gambar 2. 3 Cara Kerja Konvolusi.....	10
Gambar 2. 4 Lokasi Valid untuk Jendela Berukuran 3x3 dalam 5x5 Peta Ciri	10
Gambar 2. 5 <i>Padding</i> 5x5 Peta Ciri Masukan	11
Gambar 2. 6 Konvolusi 3x3 dengan 2x2 Langkah	11
Gambar 2. 7 Konsep berbagi Bobot (<i>Weight-sharing</i>).....	12
Gambar 2. 8 Contoh <i>pooling</i>	12
Gambar 2. 9 Konvolusi dan <i>Pooling</i>	13
Gambar 2. 10. Konvolusi dan <i>Pooling</i>	13
Gambar 2. 11. Contoh dari <i>Confusion Matrix</i>	14
Gambar 2. 12. Contoh dari Multi-kelas <i>Confusion Matrix</i>	15
Gambar 2. 13 Deep-Learning Software and Hardware Stack.....	17
Gambar 2. 14 Ilustrasi klasifikasi biner	18
Gambar 2. 15 Contoh Citra berbagai Ras Kucing (Maine Coon, Persian, Siamese)	21
Gambar 2. 16 Xception: <i>Depthwise Separable Convolution</i>	22
Gambar 2. 17 Arsitektur DenseNet.....	23
Gambar 3.1 Tahapan Penelitian	27
Gambar 3.2 Augmentasi Gambar.....	29
Gambar 3.3 (a) Rancangan Arsitektur Model Xception dengan masukan Gambar Ukuran 224x224.....	30
Gambar 3.3 (b) Rancangan Arsitektur Model Xception dengan masukan Gambar Ukuran 224x224.....	31
Gambar 3.4 (a) Rancangan Arsitektur Model DenseNet-201 dengan Masukan Berukuran 224x224.....	33
Gambar 3.4 (b) Rancangan Arsitektur Model DenseNet-201 dengan Masukan Berukuran 224x224.....	34
Gambar 3.5 Lapisan Terhubung-Penuh pada Model Xception.....	37
Gambar 3.6 Lapisan Terhubung-Penuh pada Model DenseNet-201	38
Gambar 3.7 Jumlah Parameter pada Model Xception	39
Gambar 3.8 Jumlah Parameter pada Model DenseNet-201	39
Gambar 3.9 Contoh Hasil Pengujian menggunakan <i>Confusion Matrix</i>	45
Gambar 3.10 Perancangan Antarmuka Aplikasi.....	47
Gambar 3.11 <i>Use Case Diagram</i> Aplikasi Klasifikasi Ras Kucing	48
Gambar 4.1 Grafik tahap Pelatihan untuk Skema 1 pada Model Xception	50
Gambar 4.2 <i>Confusion Matrix</i> Skema 1 pada Model Xception.....	51
Gambar 4.3 Hasil Klasifikasi Skema 1 pada Model Xception	51

Gambar 4.4 Grafik tahap Pelatihan untuk Skema 2 pada Model Xception	52
Gambar 4.5 <i>Confusion Matrix</i> Skema 2 pada Model Xception	52
Gambar 4.6 Hasil Klasifikasi Skema 2 pada Model Xception	53
Gambar 4.7 Grafik tahap Pelatihan untuk Skema 3 pada Model Xception	53
Gambar 4.8 <i>Confusion Matrix</i> Skema 3 pada Model Xception	54
Gambar 4.9 Hasil Klasifikasi Skema 3 pada Model Xception	54
Gambar 4.10 Grafik tahap Pelatihan untuk Skema 1 pada Model DenseNet-201	55
Gambar 4.11 <i>Confusion Matrix</i> Skema 1 pada Model DenseNet-201	56
Gambar 4.12 Hasil Klasifikasi Skema 1 pada Model DenseNet-201	56
Gambar 4.13 Grafik tahap Pelatihan untuk Skema 2 pada Model DenseNet-201	57
Gambar 4.14 <i>Confusion Matrix</i> Skema 2 pada Model DenseNet-201	57
Gambar 4.15 Hasil Klasifikasi Skema 2 pada Model DenseNet-201	58
Gambar 4.16 Grafik tahap Pelatihan untuk Skema 3 pada Model DenseNet-201	58
Gambar 4.17 <i>Confusion Matrix</i> Skema 3 pada Model DenseNet-201	59
Gambar 4.18 Hasil Klasifikasi Skema 3 pada Model DenseNet-201	59
Gambar 4.19 Grafik tahap Pelatihan untuk lr=0,01 pada Model Xception	61
Gambar 4.20 <i>Confusion Matrix</i> untuk lr=0,01 pada Model Xception	61
Gambar 4.21 Hasil Klasifikasi untuk lr=0,01 pada Model Xception	62
Gambar 4.22 Grafik tahap Pelatihan untuk lr=0,001 pada Model Xception	62
Gambar 4.23 <i>Confusion Matrix</i> untuk lr=0,001 pada Model Xception	63
Gambar 4.24 Hasil Klasifikasi untuk lr=0,001 pada Model Xception	64
Gambar 4.25 Grafik tahap Pelatihan untuk lr=0,0001 pada Model Xception	64
Gambar 4.26 <i>Confusion Matrix</i> untuk lr=0,0001 pada Model Xception	65
Gambar 4.27 Hasil Klasifikasi untuk lr=0,0001 pada Model Xception	65
Gambar 4.28 Grafik tahap Pelatihan untuk lr=0,01 pada Model DenseNet-201 ..	66
Gambar 4.29 <i>Confusion Matrix</i> untuk lr=0,01 pada Model DenseNet-201	68
Gambar 4.30 Hasil Klasifikasi untuk lr=0,01 pada Model Xception	68
Gambar 4.31 Grafik tahap Pelatihan untuk lr=0,001 pada Model DenseNet-201	68
Gambar 4.32 <i>Confusion Matrix</i> untuk lr=0,001 pada Model DenseNet-201	69
Gambar 4.33 Hasil Klasifikasi untuk lr=0,001 pada Model DenseNet-201	69
Gambar 4.34 Grafik tahap Pelatihan untuk lr=0,0001 pada Model DenseNet-201	70
Gambar 4.35 <i>Confusion Matrix</i> untuk lr=0,0001 pada Model DenseNet-201	71
Gambar 4.36 Hasil Klasifikasi untuk lr=0,0001 pada Model DenseNet-201	71
Gambar 4.37 Grafik tahap Pelatihan untuk <i>epoch</i> =20 pada Model Xception	73
Gambar 4.38 <i>Confusion Matrix</i> untuk <i>epoch</i> =20 pada Model Xception	74
Gambar 4.39 Hasil Klasifikasi untuk <i>epoch</i> =20 pada Model Xception	74
Gambar 4.40 Grafik tahap Pelatihan untuk <i>epoch</i> =50 pada Model Xception	75

Gambar 4.41 Confusion Matrix untuk <i>epoch</i> =50 pada Model Xception.....	75
Gambar 4.42 Hasil Klasifikasi untuk <i>epoch</i> =50 pada Model Xception	76
Gambar 4.43 Grafik tahap Pelatihan untuk <i>epoch</i> =80 pada Model Xception	76
Gambar 4.44 Confusion Matrix untuk <i>epoch</i> =80 pada Model Xception.....	77
Gambar 4.45 Hasil Klasifikasi untuk <i>epoch</i> =80 pada Model Xception	78
Gambar 4.46 Grafik tahap Pelatihan untuk <i>epoch</i> =20 pada Model DenseNet-201	79
Gambar 4.47 Confusion Matrix untuk <i>epoch</i> =20 pada Model DenseNet-201	80
Gambar 4.48 Hasil Klasifikasi untuk <i>epoch</i> =20 pada Model DenseNet-201	80
Gambar 4.49 Grafik tahap Pelatihan untuk <i>epoch</i> =50 pada Model DenseNet-201	81
Gambar 4.50 Confusion Matrix untuk <i>epoch</i> =50 pada Model DenseNet-201	81
Gambar 4.51 Hasil Klasifikasi untuk <i>epoch</i> =50 pada Model DenseNet-201	82
Gambar 4.52 Grafik tahap Pelatihan untuk <i>epoch</i> =80 pada Model DenseNet-201	82
Gambar 4.53 Confusion Matrix untuk <i>epoch</i> =80 pada Model DenseNet-201	83
Gambar 4.54 Hasil Klasifikasi untuk <i>epoch</i> =80 pada Model DenseNet-201	83
Gambar 4.55 Grafik tahap Pelatihan untuk 10% Lapisan Aktif pada Model Xception	85
Gambar 4.56 Confusion Matrix untuk 10% Lapisan Aktif pada Model Xception	86
Gambar 4.57 Hasil Klasifikasi untuk 10% Lapisan Aktif pada Model Xception	86
Gambar 4.58 Grafik Tahap Pelatihan untuk 50% Lapisan Aktif pada Model Xception	87
Gambar 4.59 Confusion Matrix untuk 50% Lapisan Aktif pada Model Xception	87
Gambar 4.60 Hasil Klasifikasi untuk 50% Lapisan Aktif pada Model Xception	88
Gambar 4.61 Grafik tahap Pelatihan untuk 90% Lapisan Aktif pada Model Xception	88
Gambar 4.62 Confusion Matrix untuk 90% Lapisan Aktif pada Model Xception	89
Gambar 4.63 Hasil Klasifikasi untuk 90% Lapisan Aktif pada Model Xception	89
Gambar 4.64 Grafik tahap Pelatihan untuk 10% Lapisan Aktif pada Model DenseNet-201	90
Gambar 4.65 Confusion Matrix untuk 10% Lapisan Aktif pada Model DenseNet-201.....	91
Gambar 4.66 Hasil Klasifikasi untuk 10% Lapisan Aktif pada Model DenseNet-201.....	91
Gambar 4.67 Grafik tahap Pelatihan untuk 50% Lapisan Aktif pada Model DenseNet-201	92

Gambar 4.68 Confusion Matrix untuk 50% Lapisan Aktif pada Model DenseNet-201.....	92
Gambar 4.69 Hasil Klasifikasi untuk 50% Lapisan Aktif pada Model DenseNet-201.....	93
Gambar 4.70 Grafik tahap Pelatihan untuk 90% Lapisan Aktif pada Model DenseNet-201	93
Gambar 4.71 Confusion Matrix untuk 90% Lapisan Aktif pada Model DenseNet-201.....	94
Gambar 4.72 Confusion Matrix untuk 90% Lapisan Aktif pada Model DenseNet-201.....	94
Gambar 4.73 Tampilan Awal Aplikasi	96
Gambar 4.74 Pilih Gambar	96
Gambar 4.75 Tampilan setelah Gambar dipilih.....	96

DAFTAR TABEL

Tabel 3.1 Definisi Operasional Penelitian	25
Tabel 3.2 Arsitektur DenseNet-201 dengan Ukuran Masukan Berukuran 224x224. Catatan : Untuk setiap lapisan “conv” pada tabel mewakili urutan lapisan BN-ReLU-Conv2D.....	36
Tabel 3.3 Jumlah data sesuai skema pembagian himpunan data (70:30)	42
Tabel 3.4 Jumlah data sesuai skema pembagian himpunan data (80:20)	43
Tabel 3.5 Jumlah data sesuai skema pembagian himpunan data (90:10)	43
Tabel 3.5 Jumlah data sesuai skema pembagian himpunan data 90:10 (lanjutan)	44
Tabel 3.6 Tabel Parameter untuk Mengukur Kinerja pada Model	45
Tabel 4.1 Hasil Uji Percobaan pada model Xception berdasarkan Skema Himpunan Data	50
Tabel 4.2 Hasil Uji Percobaan pada model DenseNet-201 berdasarkan Skema Himpunan Data	55
Tabel 4.3 Hasil Uji Percobaan pada model Xception berdasarkan Paramater Laju Pembelajaran	60
Tabel 4.4 Hasil Uji Percobaan pada model DenseNet-201 berdasarkan Paramater Laju Pembelajaran	66
Tabel 4.5 Hasil Uji Percobaan pada model Xception berdasarkan Jumlah <i>Epoch</i> 72	
Tabel 4.6 Hasil Uji Percobaan pada model DenseNet-201 berdasarkan Jumlah <i>Epoch</i>	78
Tabel 4.7 Hasil Uji Percobaan pada model Xception berdasarkan Skema Fine-Tuning	85
Tabel 4.8 Hasil Uji Percobaan pada model DenseNet-201 berdasarkan Skema Fine-Tuning	90

ABSTRAK

Perkembangan teknologi saat ini sudah sampai pada tahap dimana komputer dapat mengenali objek dalam gambar secara otomatis. Berbagai model dan algoritma terus dikembangkan agar komputer dapat mengenali objek di dalam gambar secara lebih baik. Salah satu metode yang terbukti menghasilkan hasil terbaik (state-of-the-art) adalah teknik Convolutional Neural Network atau lebih dikenal dengan ConvNet atau CNN. Akan tetapi, CNN memiliki kekurangan dalam pengembangannya yang masih terhitung mahal secara waktu dan perangkat keras yang digunakan. Namun, kekurangan tersebut dapat diatasi dengan model-model CNN yang sudah mengalami proses latih sebelumnya atau biasa disebut pre-trained model. Tujuan utama pada penelitian ini yaitu menganalisis perbandingan kinerja pre-trained CNN model, yaitu Xception dan DenseNet dalam melakukan klasifikasi terhadap ras kucing. Penelitian ini juga menjelaskan bagaimana pendekatan deep learning, seperti CNN, yang dapat membedakan suatu objek dalam gambar.

Aplikasi yang digunakan dalam penelitian ini adalah Jupyter Notebook, Google-Colab, dan Flask dengan Google Chrome sebagai peramban (web browser). Sedangkan untuk bahasa pemrograman yang digunakan yaitu Python dengan framework Keras dan Tensorflow. Tahapan penelitian dimulai dari tahapan pendahuluan yang memuat penentuan topik, identifikasi dan perumusan masalah, dan menentukan metodologi penelitian. Kemudian dilakukan kajian pustaka, yaitu pengumpulan materi guna mendukung penelitian ini. Selanjutnya dilakukan pengumpulan data dan pengolahan data. Pengolahan data memuat prapengolahan data, perancangan model, pelatihan model dan pengujian model. Setelah selesai mengumpulkan dan mengolah data, dilakukan implementasi aplikasi ke dalam web yang berfungsi untuk memprediksi gambar kucing dari direktori lokal komputer.

Setelah menguji kedua model tersebut, diperoleh kesimpulan bahwa model memiliki hasil kinerja yang berbeda pada setiap percobaannya. Hasil percobaan menunjukkan bahwa secara umum model DenseNet-201 memiliki kinerja yang lebih baik dalam melakukan klasifikasi gambar ras kucing daripada model Xception.

Keywords: *Pre-trained Model, Xception, DenseNet, Ras Kucing, CNN, Klasifikasi.*

ABSTRACT

The recent development of technology has made it possible for computers to recognize objects in an image automatically. Various models and algorithms continue to be developed, so that computers can recognize objects in an image more better. One technique that is proven to produce the best results (state-of-the-art) is the Convolutional Neural Network technique, or better known as ConvNet or CNN. However, CNN has shortcomings in its development which is expensive in time and the hardware. These deficiencies can be overcome by CNN models that have previously trained on a large dataset, commonly referred to as pre-trained models. The main objective of this research is to analyze the performance of pre-trained CNN models, specifically Xception and DenseNet-201 in classifying cat breeds. Moreover, this research also explains how the deep learning approaches, such as CNN, can distinguish an object in an image.

The model are built using Jupyter notebook, Google-Colab, and Flask with Google Chrome as a browser (web browser). Whereas the programming language used is Python with Keras and Tensorflow frameworks. The stages of the research start from the preliminary stage which includes determining the topic, identifying and formulating the problem, and determining the research methodology. Then a literature review is conducted, i.e. gathering material to support this research. The next step are collecting data and processing data. Data processing includes several stages like data processing, designing the model, training the model and testing the model. After completing data collection and data processing, this trained model was then developed into a web-based cat breeds classification application.

After testing the two models, it was concluded that the model has different performance results in each experiment. The experimental results show that overall the DenseNet-201 model has better performance in classifying cat breeds images than the Xception model.

Keywords: *Pre-trained Model, Xception, DenseNet, CNN, Cat Breed, Classification*