MANIPAL INSTITUTE OF TECHNOLOGY
BENGALURU
(A constituent unit of MAHE, Manipal)

Image Classification Using CNN

A Report submitted to
Manipal Academy of Higher Education
in partial fulfilment of the requirement for the award of the degree of

BACHELOR OF TECHNOLOGY

in

Computer Science & Engineering

Submitted by
Devashish Pandey
225805224

DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

<OCTOBER> 2025

CHAPTER 1 – INTRODUCTION

A Convolutional Neural Network (CNN) is a specific class of deep learning models that are particularly powerful for image classification and visual recognition tasks. CNNs are inspired by the biological visual cortex and are designed to automatically and adaptively learn spatial hierarchies of features through backpropagation.

The objective of this project was to **implement and evaluate a Convolutional Neural Network model** capable of classifying images of *cats and dogs* accurately. The project explores data preprocessing, model construction, training, and optimization techniques to enhance accuracy and reduce overfitting.

The major steps involved include:

- Understanding and preprocessing image data
- Designing and constructing a CNN model
- Training and evaluating performance
- Applying optimization methods like batch normalization, dropout, and learning rate decay

This project demonstrates the relevance of CNNs in computer vision, emphasizing their potential for real-world applications such as object detection and autonomous robotics.

CHAPTER 2 – BACKGROUND THEORY

Convolutional Neural Networks are a form of feedforward artificial neural network that use convolution operations instead of general matrix multiplication in at least one of their layers.

The CNN architecture typically comprises:

1. **Convolutional Layers** – extract spatial features using filters.
2. **Pooling Layers** – reduce spatial dimensions while retaining essential information.
3. **Fully Connected Layers** – perform classification based on extracted features.
4. **Activation Functions** – introduce non-linearity (ReLU used here).

CNNs have revolutionized computer vision due to their ability to automatically detect edges, shapes, and objects with minimal manual feature engineering.

CHAPTER 3 – METHODOLOGY

The project workflow consists of the following steps:

3.1 Understanding and Pre-processing Data

The dataset consisted of labeled images of cats and dogs. Each image was resized to uniform dimensions to ensure consistent feature extraction. Preprocessing involved:

- Reading images from the dataset
- Resizing and normalizing pixel values
- Converting images from string to float
- Appending each processed image with its label

3.2 Model Creation

The CNN model was designed with **five convolution blocks**, each followed by a **max pooling layer**, and a **fully connected layer** with 256 units activated using **ReLU**.

The model was compiled using:

- **Loss function:** Binary Cross Entropy
- **Optimizer:** ADAM

The use of ReLU accelerates convergence due to its simple non-linear characteristics.

CHAPTER 4 – IMPLEMENTATION DETAILS & RESULT ANALYSIS

4.1 Model Optimization Techniques

To improve model performance and avoid overfitting, several optimization techniques were applied:

- **Batch Normalization:** Standardized layer inputs for each mini-batch to stabilize learning and reduce training epochs.
- **Dropout:** Applied 0.2–0.3 dropout to randomly deactivate neurons and prevent overfitting.
- **Learning Rate Decay:** Gradually reduced the learning rate from 1e−3 to 1e−5 across epochs for better convergence.

The model was trained on an **NVIDIA GeForce 940MX GPU** using various dataset sizes and hyperparameters.

4.2 Results

Sr. No	No. of Images	Image Size	Epochs	Training Accuracy (%)	Validation Accuracy (%)
1	5,000	120	5	52.79	69.68
3	10,000	120	5	73.00	60.27
4	10,000	140	5	76.53	48.43
5	10,000	200	20	98.00	88.00
6	15,000	140	5	79.60	43.04
7	15,000	150	5	83.66	35.18

Sr. No	No. of Images	Image Size	Epochs	Training Accuracy (%)	Validation Accuracy (%)
8	15,000	150	10	95.07	81.25
9	18,000	110	15	97.59	90.44

Table 1: Training Accuracy and Validation Accuracy

As dataset size and epochs increased, accuracy improved. However, overfitting was observed in higher training sets, where training accuracy rose while validation accuracy stagnated. This indicates the model's difficulty in generalizing to unseen data.

CHAPTER 5 – CONCLUSION AND FUTURE SCOPE

Conclusion

The project successfully implemented a Convolutional Neural Network for binary image classification (cats vs. dogs). The system achieved a training accuracy of up to 97.59% and validation accuracy of around 90.44%, demonstrating CNN's strong capability in feature extraction and classification.

Github link:

https://github.com/Devashish221/Image-Classification-using-CNN.git