NAAN MUTHALVAN

**EDUNET AND SKILLS BUILD**

**Experience Based Project Learning**

**-**

**Oracle (E2324)**

**MACHINE MASTERMIND: TEACHING A COMPUTER TO**

**SEE: AN IMAGE CLASSIFICATION ADVENTURE**

**DEPARTMENT OF**

**INFORMATION TECHNOLOGY**



**6127 - THE KAVERY ENGINEERING COLLEGE MECHERI,** **SALEM-636453.**

**MACHINE MASTERMIND: TEACHING A COMPUTER TO**

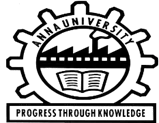
**SEE: AN IMAGE CLASSIFICATION ADVENTURE**

**A PROJECT REPORT**  *Submitted by*

**E.SHANMUGAPRIYA-(REG NO:612722205107)**

**In partial fulfillment of for the awarded of the degree**

# INFORMATION TECHNOLOGY



**6127- THE KAVERY ENGINEERING COLLEGE MECHERI**

**SALEM-636453**

**DEPARTMENT OF**

**BACHELOR OF TECHNOLOGY IN**

# INFORMATION TECHNOLOGY

**ABSTRACT**

**MACHINE MASTERMIND: TEACHING A COMPUTER TO**

**SEE:**  **AN IMAGE CLASSIFICATION ADVENTURE**

Teaching a computer to see involves using computer vision techniques to enable machines to interpret and understand visual information from the word around them. These include tasks like object recognition, image classification and scene understanding, often achieved through deep learning algorithm trained on vast amount of labeled data .

Image Classification nowadays is used to narrow the gap between the computer vision and human vision so that the images can be recognized by machines in the same way as we humans do. It deals with assigning the appropriate class for the given image.

Machine mastermind explores the application of Artificial intelligence (AI) techniques to the classic board game mastermind. The game involves one player, the codemaker, creating a secret code, and the other player, attempting guess the code within a limited number of turns. This paper investigates various AI strategies, including brute-force search algorithms, heuristic approaches, and machine learning techniques, to develop an efficient and effective codebreaking algorithm.

**ARTIFICIAL** **INTELLIGENCE** **IN** **DISEASE** **DIAGNOSIS**:

Artificial Intelligence (AI) has

revolutionized numerous fields, with healthcare being one of the most impacted. One of the most promising applications is in disease diagnosis through image classification.

1. \*\*Data Collection\*\*: Gathering a large dataset of labeled medical images.
2. \*\*Data Preprocessing\*\*: Cleaning and preparing images for analysis.
3. \*\*Model Selection\*\*: Choosing an appropriate AI model architecture.
4. \*\*Training\*\*: Teaching the model to recognize patterns by feeding it labeled images.
5. \*\*Validation\*\*: Testing the model on a separate set of images to evaluate its performance.

AI-driven image classification holds great promise for improving disease diagnosis, offering faster, more accurate, and cost-effective solutions. By addressing the challenges and ethical considerations, we can develop AI systems that not only enhance clinical outcomes but also gain the trust of healthcare professionals and patients alike. The journey of teaching a computer to see is not just a technological adventure but a crucial step towards a future where AI plays a central role in healthcare.

## Machine Mastermind: Teaching a Computer to See – An Image Classification Adventure

**Framework for AI in Disease Detection**

**Modeling:**

The integration of artificial intelligence (AI) into disease detection has opened new frontiers in medical diagnostics. This framework outlines the essential steps and methodologies for implementing AI in image classification, particularly for disease detection.

1. \*\*Data Collection\*\*

\*\*Medical Imaging Modalities\*\*: Collect images from various sources such as MRI, CT scans, Xrays, and histopathology slides.

1. \*\*Data Preprocessing\*\*

* \*\*Normalization\*\*: Standardize image sizes and color scales.
* \*\*Architecture\*\*: Choose architectures like VGG16, ResNet, or Inception for their proven efficacy in image classification tasks.

4. \*\*Model Training\*\*

\*\*Training Configuration\*\*

- \*\*Hyperparameters\*\*: Tune hyperparameters such as learning rate, batch size, and number of epochs.

5. \*\*Model Evaluation\*\*

\*\*Metrics\*\*

- \*\*Accuracy\*\*: Measure the overall correctness of the model.

AI disease detection through image classification involves a systematic approach encompassing data collection, preprocessing, model training, and continuous improvement. By following this framework, healthcare practitioners can leverage AI to enhance diagnostic accuracy and ultimately improve patient outcomes.

**SYSTEM REQUIREMENTS:**

**HARDWARE REQUIREMENTS:**

* + A computer with a decend processor (e.g., Intel i5 or higher, AMD Ryzen 5 or higher)

* + Sufficient RAM(8GB or more is recommended).

* + A dedicated GPU(NVIDIA GeForce GTX

1060 or higher, or an equivalent AMD GPU) for faster training if working with deep learning model.

**SOFTWARE REQUIREMENTS:**

* Python programming language and it’s libraries such as TensorFlow, PyTorch, or Keeps for deep learning.

* Image processing libraries like OpenCV for prepossessing images.

* Jupiter Notebook or any IDE for writing and testing code.

**DATASETS**:

Labeled dataset containing images relevant to the classification task.

Common datasets include CIFAR-10,

CIFAR-100, MINISTER, or I

ImageNet.

**OPTIONAL**:

* Cloud computing resources like

AWS,

Google Cloud Platform, or Microsoft Azure for faster computation and training on large datasets.

* Additional hardware like GPUs or TPUs for faster processing, especially for large a datasets and complex models.

**OCUMENTATION AND TUTORIALS:**

* Access to documentation and tutorials for the chosen libraries and frameworks to understand their functionalities and how to implement them effectively for image classification tasks.

**PATIENCE AND PERSEVERANCE:**

* Learning and fine-tuning machine learning models can be iterative and time-consuming, so patience and Perseverance are essential.

**TOOLS REQUIREMENTS:**

1.Dataset

2.Framework

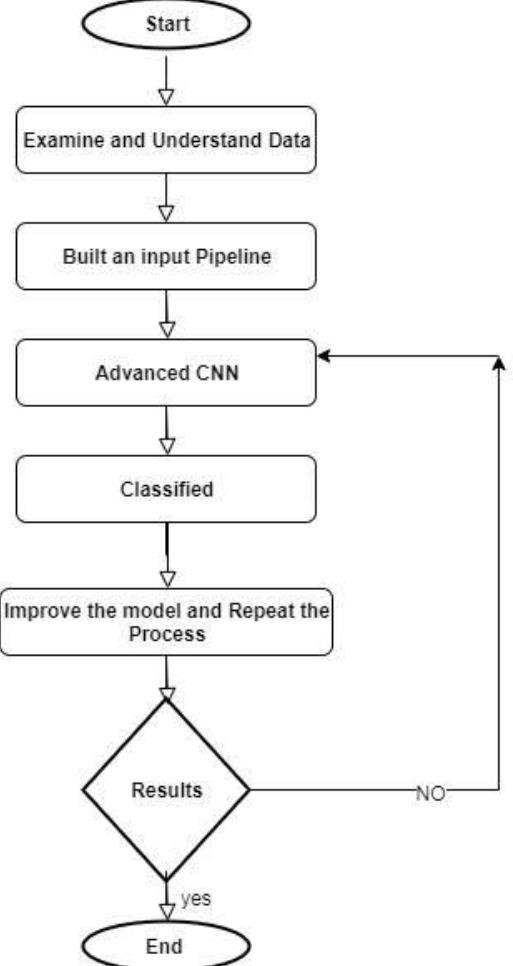
3.Model Architecture

4.Hardware

5.Data prepossessing tools

6.Training Environment

**Flow Chart :**



**Code Implementation**:

Sure! Here is a sample code for an image classification task using a Convolutional Neural Network (CNN) with TensorFlow and Keras. This code demonstrates how to train a model to classify images from a dataset like CIFAR10, a common benchmark dataset that consists of 60,000 32x32 color images in 10 classes.

### Step 1: Import Libraries

```python

Import tensorflow as tf

From tensorflow.keras import datasets, layers, models Import matplotlib.pyplot as plt

```

### Step 2: Load and Preprocess the

Data

```python

# Load CIFAR-10 dataset

(train\_images, train\_labels), (test\_images, test\_labels) = datasets.cifar10.load\_data()

# Normalize pixel values to be between 0 and 1

Train\_images, test\_images = train\_images / 255.0, test\_images /

255.0

```

### Step 3: Define the CNN Model

```python

Model = models.Sequential()

# First Convolutional Layer

Model.add(layers.Conv2D(32, (3, 3), activation=relu, input\_shape=(32, 32,

3)))

Model.add(layers.MaxPooling2D((2,

2)))

# Second Convolutional Layer

Model.add(layers.Conv2D(64, (3, 3), activation=’relu’))

Model.add(layers.MaxPooling2D((2,

2)))

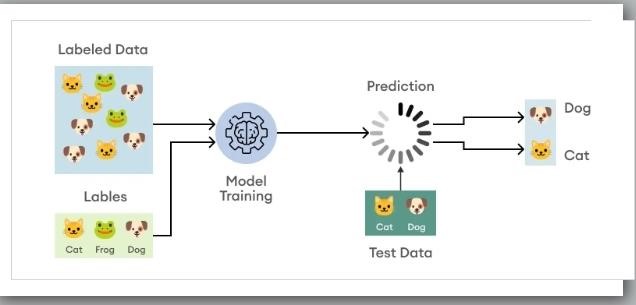
# Third Convolutional Layer

Model.add(layers.Conv2D(64, (3, 3), activation=’relu’))

# Flatten the output and feed it into a dense layer Model.add(layers.Flatten())

Model.add(layers.Dense(64, activation=’relu’))

**OUTPUT:**



**Conclusion:**

Image classification using CNN has revolutionized the field of computer vision, enabling accurate recognition of objects within images. With its ability to automatically learn and extract complex features, CNNs have become a powerful tool for various applications.