Internship Report

MKU-RUSA

Particulars	Report Details
Name	PRIYADHARSHANA M
Department	Computer Science
College/University	Government Arts college,Melur
Course	M.Sc(computer science)
Mobile Number	6369598603
Email ID	priyadharshanam12@gmail.com
Joining Date	23.05.2024
List of Objectives	Basic image processing tools,CNN using image classification,Thresholding functions tools,extract character from an image using threshold,Contour detection,water _shed segmentation,region_base segmentation,Canny_edge detection,use sift and histogram to find key point matches between images,object detection using already existing model And use linear regression to predict mobile price.
List the Other Learnings during this internship period	Numpy,pandas,matplotlib,tkinder,torchvision transformation,Torchvision model,seaborn,sklearn,tensorflow,keras and Basic image processing, segmentation techniques,cnn,linear regression, Object detection and sift-histogram.
How this experience would be useful for your career development	This will be help my final year project And support my career in future. This is the first time I learn image processing and machine learning concept through this internship. Thank you for gave me this opportunity.

WEEK1

S.NO	TASKS
1	Learn and create basic image processing tools
2	Learn CNN and implement image classification
3	Learn threshold and create tools to apply function for various threshold functions for various images

WEEK2

S.NO	TASKS
4	Use threshold function to extract character from an image
5	Learn and implement contour segmentation technique
6	Learn and implement water_shed segmentation technique

WEEK3

S.NO	TASKS
7	Learn and implement region_base segmentation technique
8	Learn and implement canny_edge _detection segmentation technique

WEEK 4

S.NO	TASKS
9	Learn and implement shift feature and histogram to find keypoint matches between images
10	Object detection using already existing model like alexnet
11	Learn linear regression and predict mobile price

ABSTRACT

During my internship, I acquired foundational skills in image processing, segmentation techniques, Convolutional Neural Networks (CNNs), linear regression, and object detection using pre-trained models. My journey began with mastering basic image processing methods, essential for enhancing image quality and preparing data for subsequent analysis. I then explored basic segmentation techniques to divide digital images into meaningful segments, aiding in focused analysis of specific image regions.

In the realm of machine learning, I studied CNNs, which are pivotal for image recognition and classification tasks. Through practical projects, I built and trained CNN models, learning about their architecture and the significance of convolutional layers for feature extraction. I also delved into linear regression, a fundamental technique for predictive analysis, and its applications within machine learning.

This comprehensive learning experience provided me with a robust foundation in image processing, segmentation, and machine learning, equipping me with the skills necessary for advanced studies and future projects in these fields.

INDEX

INTRODUCTION

My internship provided a profound opportunity to explore and understand key concepts in image processing and machine learning. Throughout this period, I focused on acquiring a solid foundation in basic image processing techniques, segmentation methods, Convolutional Neural Networks (CNNs), linear regression, and object detection using pre-trained models.

The journey began with basic image processing, where I learned various methods to enhance image quality, which is essential for accurate analysis and preprocessing in computer vision tasks. This foundational knowledge was crucial for my subsequent work in image segmentation, a technique that involves partitioning digital images into distinct regions for more focused and effective analysis.

TOOLS AND TECHNOLOGIES

Throughout my internship, I utilized a wide array of tools and technologies to tackle various tasks related to image processing, machine learning, and object detection. These tools enabled the practical application of theoretical concepts and facilitated a deeper understanding of the field. The key tools and technologies I worked with include:

1. **Basic Image Processing Tools:**
    - **OpenCV (cv2):** An open-source computer vision library used extensively for image processing tasks such as filtering, edge detection, and enhancement.
    - **PIL (Python Imaging Library) / Pillow:** A Python library for opening, manipulating, and saving image files.
2. **Convolutional Neural Networks (CNN) for Image Classification:**
    - **TensorFlow:** An open-source machine learning library developed by Google, used for building and training CNN models.
    - **Keras:** A high-level neural networks API written in Python, running on top of TensorFlow, for constructing and training CNNs.
3. **Thresholding Techniques:**
    - **OpenCV:** Used for applying global, adaptive, and Otsu's thresholding to preprocess images and segment regions of interest.
4. **Character Extraction from Images using Thresholding:**
    - **Tesseract OCR:** An open-source Optical Character Recognition engine used in conjunction with thresholding techniques for extracting characters from processed images.
5. **Segmentation Techniques:**
    - **SIFT (Scale-Invariant Feature Transform):** Utilized for detecting and describing local features in images.
    - **Histogram Matching:** Used to find keypoint matches between images by comparing color distributions.
6. **Object Detection using Pre-trained Models:**
    - **AlexNet:** Employed for object detection tasks, utilizing its pre-trained model to identify and classify objects within images.

- **torchvision & PyTorch:** Libraries used for deep learning tasks, including loading and using pre-trained models.

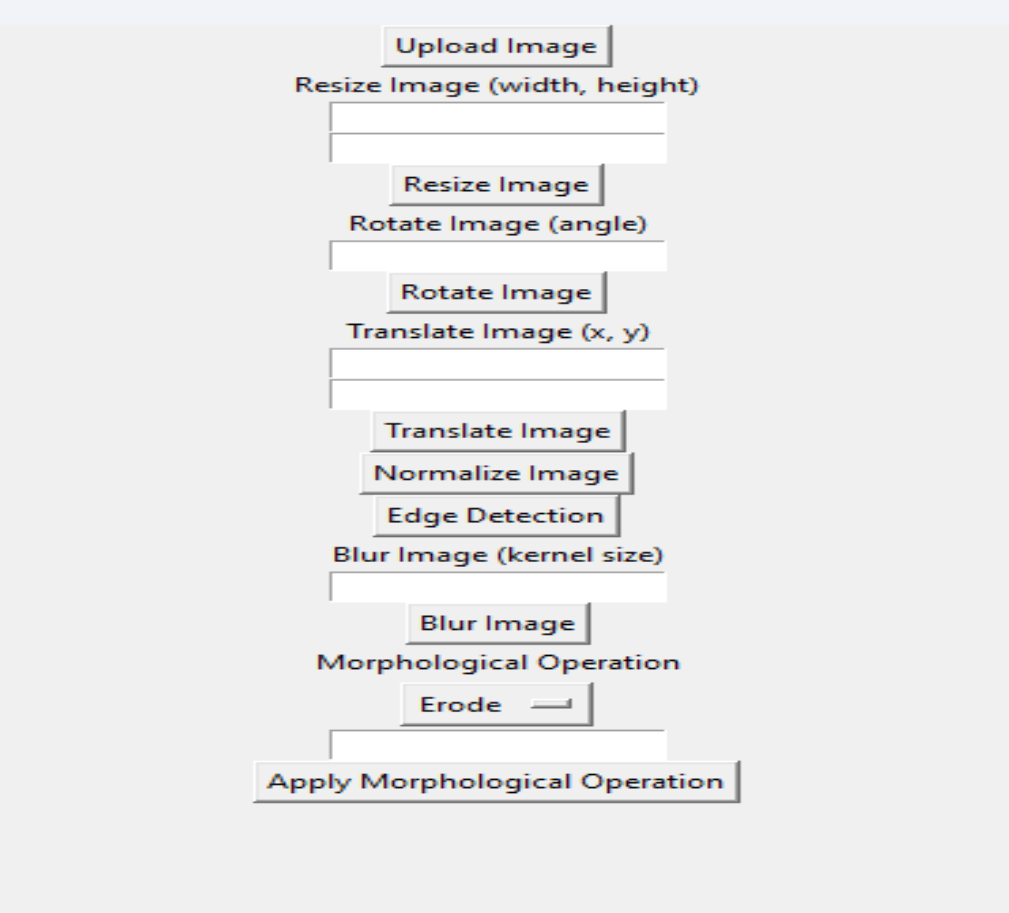
7. **Linear Regression for Predictive Analysis:**
    - **scikit-learn (sklearn):** A machine learning library in Python used for implementing linear regression to predict mobile prices based on various features.
8. **Data Manipulation and Visualization:**
    - **NumPy:** A library for numerical computing in Python, used for handling arrays and matrices of image data.
    - **Pandas:** A data manipulation library, essential for data preprocessing and analysis.
    - **Matplotlib:** A plotting library used for creating static, animated, and interactive visualizations.
    - **Seaborn:** A data visualization library based on Matplotlib, used for making statistical graphics.
9. **Graphical User Interface (GUI) Development:**
    - **Tkinter:** A standard Python interface to the Tk GUI toolkit, used for developing simple graphical user interfaces for various applications.

These tools and technologies were instrumental in my learning journey, allowing me to effectively apply image processing and machine learning techniques to solve real-world problems.

TASK DESCRIPTION

1.BASIC IMAGE PROCESSING TOOLS

The objective of this project is to develop a comprehensive basic image processing tool that empowers users to perform fundamental image manipulation tasks. This tool will include functionalities such as resizing, rotating, normalizing, morphological operations, and blurring, allowing users to enhance and modify images effectively. By providing an intuitive and user-friendly interface, this tool aims to support educational, research, and practical applications, serving as a foundational platform for learning and applying essential image processing techniques.



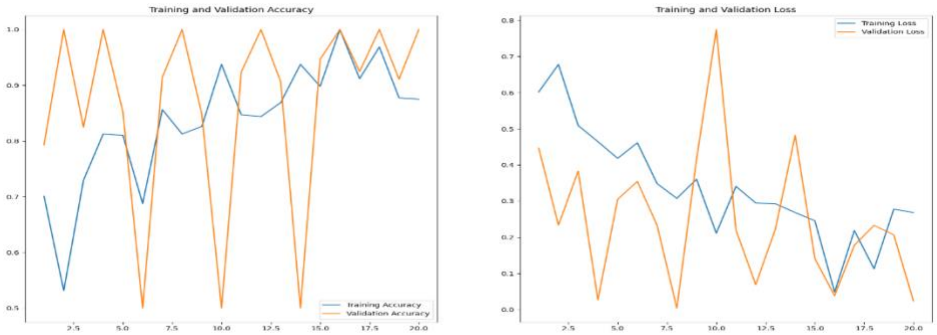
2.IMAGE CLASSIFICATION USING CNN

Objective: Develop a convolutional neural network (CNN) model for accurate and efficient classification of cat and dog images. The goal is to achieve high classification accuracy on a diverse dataset, leveraging deep learning techniques to automatically learn discriminative features from images. The model should be optimized for real-world applications, capable of handling varying image qualities and scenarios, while ensuring robust performance across different breeds and orientations of cats and dogs.

```
Found 1222 images belonging to 2 classes.
Found 1090 images belonging to 2 classes.
C:\Users\HP\PycharmProjects\pythonProject1\venv\lib\site-packages\keras\src\layers\convolutional\base_conv.py:107: User
  super().__init__(activity_regularizer=activity_regularizer, **kwargs)
2024-06-21 14:32:03.052705: I tensorflow/core/platform/cpu_feature_guard.cc:210] This TensorFlow binary is optimized to
To enable the following instructions: AVX2 FMA, in other operations, rebuild TensorFlow with the appropriate compiler f
Model: "sequential"
```

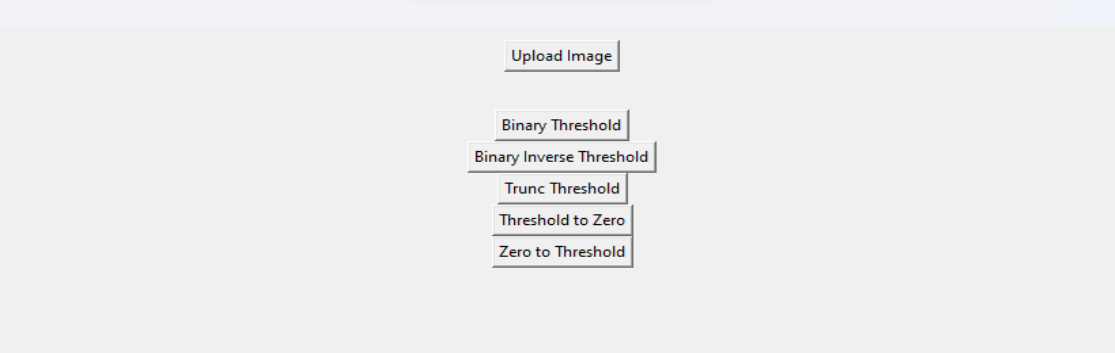
Layer (type)	Output Shape	Param #
conv2d (Conv2D)	(None, 148, 148, 32)	896
max_pooling2d (MaxPooling2D)	(None, 74, 74, 32)	0
conv2d_1 (Conv2D)	(None, 72, 72, 64)	18,496
max_pooling2d_1 (MaxPooling2D)	(None, 36, 36, 64)	0
conv2d_2 (Conv2D)	(None, 34, 34, 128)	73,856
max_pooling2d_2 (MaxPooling2D)	(None, 17, 17, 128)	0
conv2d_3 (Conv2D)	(None, 15, 15, 128)	147,584
max_pooling2d_3 (MaxPooling2D)	(None, 7, 7, 128)	0
flatten (Flatten)	(None, 6272)	0
dense (Dense)	(None, 512)	3,211,776
dropout (Dropout)	(None, 512)	0
dense_1 (Dense)	(None, 1)	513

```
 Total params: 3,453,121 (13.17 MB)
 Trainable params: 3,453,121 (13.17 MB)
 Non-trainable params: 0 (0.00 B)
Epoch 1/20
```



3.THRESHOLDING FUNTIONS TOOLS

Objective: Create a threshold functions tool program to facilitate efficient thresholding operations on digital images. The program aims to provide intuitive user interface for selecting and applying various thresholding techniques such as global, local, and adaptive thresholds. It should support visualization of thresholded results in real-time and allow users to adjust parameters for optimal segmentation and preprocessing of images in diverse applications like biomedical imaging and document analysis.



4.USE THRESHOLD TO EXTRACT CHARACTERS FROM AN IMAGE

Objective: Develop a thresholding-based character extraction method to accurately isolate text regions from images. The objective is to implement thresholding techniques, both global and adaptive, to effectively separate foreground characters from background noise.

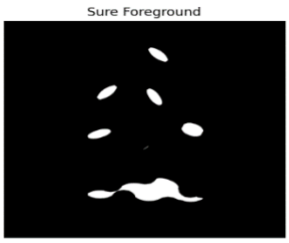
a 

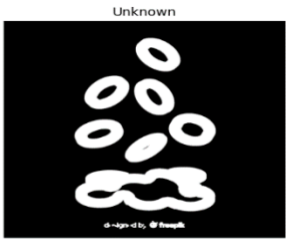
5.CONTOUR DETECTION SEGMENTATION TECHNIQUE

Objective: Develop an efficient contour detection algorithm to accurately identify and outline objects in digital images. The objective is to implement advanced computer vision techniques such as edge detection, morphological operations, and contour tracing to robustly extract boundaries of objects regardless of shape, size, or orientation. The system should be optimized for real-time performance and capable of handling noisy or complex backgrounds, facilitating applications in object recognition, medical imaging, autonomous navigation, and industrial quality inspection.

6.WATER_SHED SEGMENTATION TECHNIQUE

Implement watershed segmentation algorithm to achieve precise and automatic partitioning of digital images into regions of interest. The goal is to develop a robust system capable of accurately delineating boundaries between objects, even in the presence of overlapping or touching structures. The system should utilize gradient-based markers and flooding techniques to facilitate applications in medical imaging, satellite imagery analysis, and microscopy, enabling efficient and accurate segmentation for further quantitative analysis and object counting in various scientific and industrial domains.






designed by freepik

7.REGION_BASE SEGMENTATION TECHNIQUE

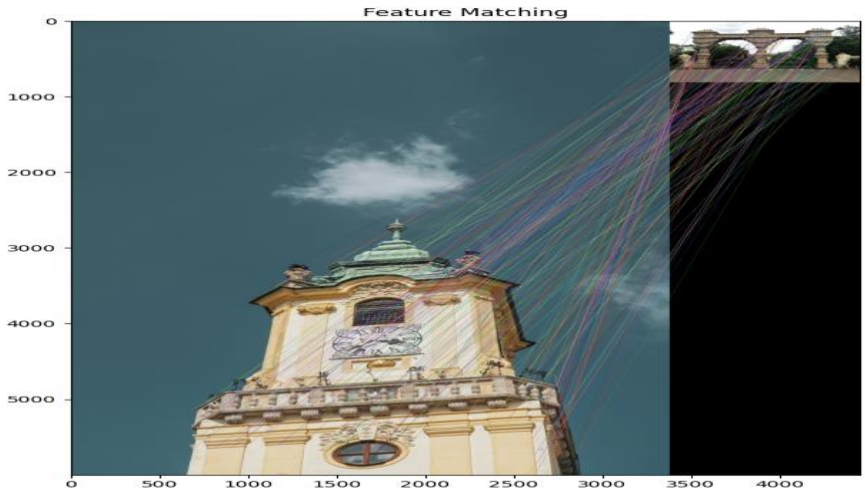
Objective: Develop a region-based segmentation technique to accurately partition digital images into meaningful regions of interest. The goal is to employ advanced algorithms such as region growing, split and merge, or graph-based methods to delineate distinct objects or areas based on color, texture, or intensity characteristics. The system should be optimized for handling complex image data, robustly segmenting objects with irregular shapes and varying sizes across different domains such as medical imaging, remote sensing, and industrial quality control. The objective includes enhancing automation and accuracy in image analysis tasks such as object recognition, scene understanding, and semantic segmentation for diverse applications.

8.CANNY_EDGE_DETECTION SEGMENTATION TECHNIQUE

Objective: Implement the Canny edge detection algorithm to accurately identify edges and boundaries in digital images. The goal is to develop a robust system that effectively detects edges with minimal noise and precise localization, utilizing techniques such as gradient calculation, non-maximum suppression, and hysteresis thresholding. The system should be optimized for real-time performance and capable of handling varying image qualities and complexities, enabling applications in object detection, image registration, autonomous driving, and medical imaging for enhanced feature extraction and analysis.

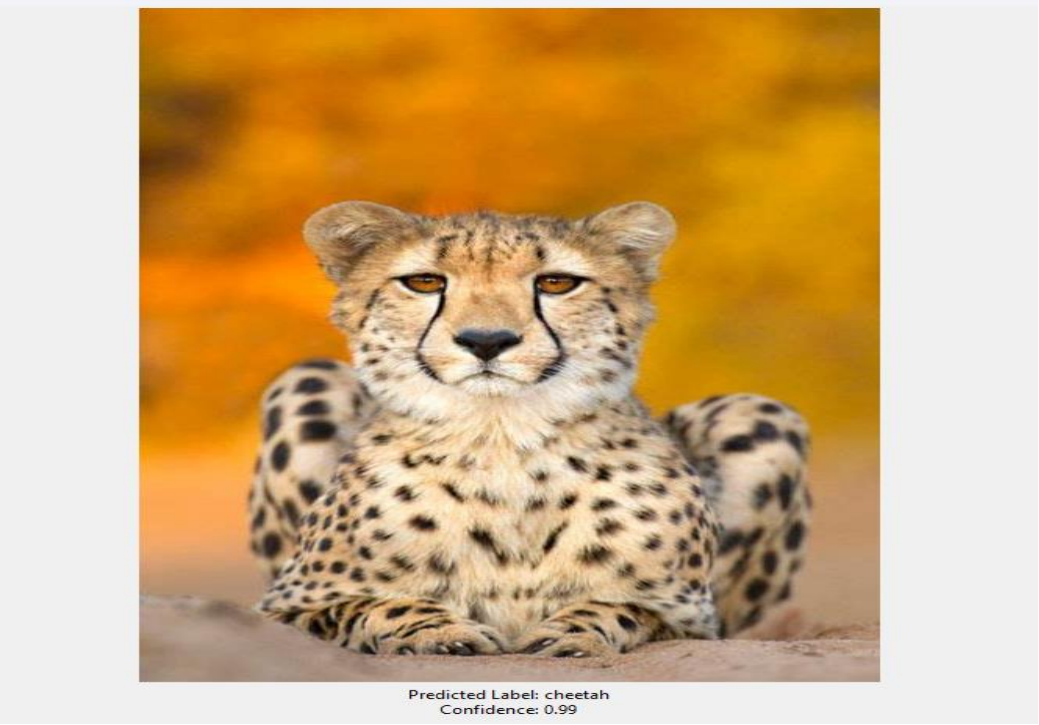
9.USING SIFT FEATURE AND HISTOGRAM ALGORITHM TO FIND KEYPOINT MATCHES BETWEEN IMAGES

Objective: Develop a robust image matching system using SIFT (Scale-Invariant Feature Transform) features and histogram algorithms to accurately identify corresponding keypoints between images. The goal is to implement feature extraction, descriptor generation, and matching techniques to achieve reliable alignment and recognition of objects across different viewpoints and lighting conditions. The system aims to enhance the accuracy and efficiency of image retrieval, object recognition, and panoramic stitching applications by leveraging distinctive local features and robust histogram-based matching strategies for comprehensive image analysis and visualization.



10.OBJECT DETECTION USING ALREADY EXISTING MODEL LIKE ALEXNET

Objective: Implement object detection using a pre-existing model like AlexNet to accurately localize and classify objects within digital images. The goal is to fine-tune the model for detection tasks by integrating region proposal techniques and bounding box regression. The system aims to achieve high detection accuracy across diverse object categories while optimizing computational efficiency, enabling applications in real-time surveillance, autonomous driving, and industrial inspection for improved object localization and recognition performance.

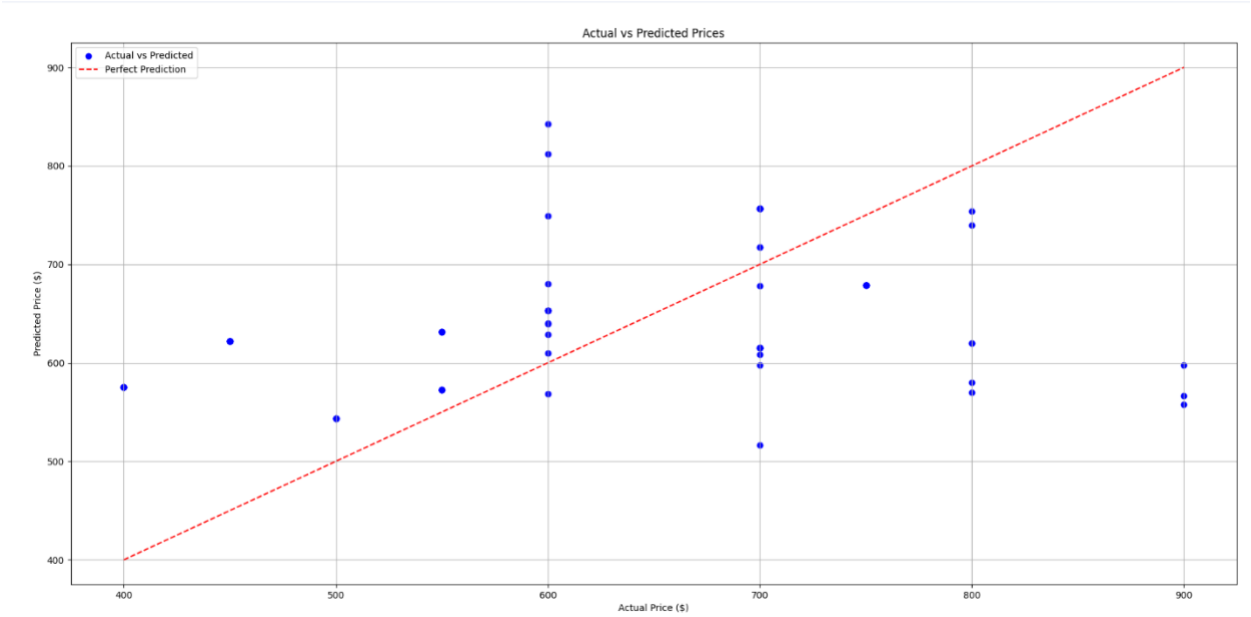


11.USING LINEAR REGRESSION TO PREDICT MOBILE PRICE

Objective: Develop a linear regression model to predict mobile prices based on relevant features such as specifications, brand, and market trends. The goal is to create an accurate and interpretable pricing model that can generalize well to unseen data, enabling stakeholders in the mobile industry to make informed pricing decisions. The system aims to optimize feature selection, handle data preprocessing effectively, and evaluate model performance rigorously to enhance pricing strategy formulation and market competitiveness.

```
C:\Users\HP\PycharmProjects\pythonProject1\venv\Scripts\python.exe C:\Users\HP\PycharmProjects\pythonProject1\CAT_DOG_CLASS\linearregression_final.py
Columns in dataset: Index(['Screen Size (inches)', 'RAM (GB)', 'Storage (GB)',
       'Battery Capacity (mAh)', 'Camera Quality (MP)', 'Price ($)'],
      dtype='object')
X_train shape: (204, 5)
X_test shape: (88, 5)
y_train shape: (204,)
y_test shape: (88,)
R-squared score: 0.07068251520957014
Predicted Price of the New Mobile Phone: [552.80889158]

Process finished with exit code 0
```



ACHIEVEMENTS

- Learn basic image processing and created tools for image processing such as resize,rotate,translate,normalize,morphological and blur.
- Learn CNN and have done image classification using CNN
- Learn segmentation techniques and it's function such as threshold ,contour,water_shed, region_base and canny_edge_detection.And created tools for threshold functions such as binary,binary_inverse,trunc,zero to thresh and thresh to zero.
- Learn object detection using already existing model such as alexnet.
- Learn linear regression and use linear regressing to predict mobile price.

SKILLS DEVELOPED

During my internship, I gained extensive experience in image processing, machine learning, and object detection by leveraging a diverse set of tools and technologies. I utilized fundamental image processing tools like OpenCV and PIL/Pillow for tasks such as filtering and edge detection. For advanced tasks like image classification using Convolutional Neural Networks (CNNs), I employed TensorFlow and Keras, while also utilizing OpenCV for thresholding techniques and character extraction using Tesseract OCR. For segmentation tasks, I utilized SIFT for feature detection and histogram matching for keypoint matching between images. Object detection tasks were accomplished using the pre-trained AlexNet model integrated with torchvision and PyTorch. For predictive analysis, I applied linear regression using scikit-learn to predict mobile prices based on various features. Additionally, I employed NumPy, Pandas, Matplotlib, and Seaborn for data manipulation, visualization, and GUI development using Tkinter. These tools collectively enhanced my understanding and proficiency in applying theoretical concepts to practical challenges in the fields of computer vision and machine learning.

CONCLUSION

My internship provided me with a comprehensive toolkit in image processing, machine learning, and object detection, facilitated by a range of essential tools and technologies. I adeptly utilized OpenCV and PIL/Pillow for foundational image processing tasks, while TensorFlow and Keras empowered me to construct and train Convolutional Neural Networks (CNNs) for image classification. I applied advanced techniques such as thresholding with OpenCV and character extraction using Tesseract OCR, demonstrating proficiency in segmentation methods like SIFT and histogram matching for image analysis. Object detection tasks were efficiently handled using pre-trained models like AlexNet integrated with torchvision and PyTorch. Additionally, I developed predictive models using scikit-learn for linear regression to forecast mobile prices based on diverse features. My proficiency in data manipulation and visualization using NumPy, Pandas, Matplotlib, and Seaborn, coupled with GUI development using Tkinter, underscored my ability to translate theoretical knowledge into practical solutions. This internship significantly enhanced my expertise in leveraging cutting-edge tools and technologies to solve complex challenges in computer vision and machine learning domains.

REFERENCES

- **BASIC IMAGE PROCESSING -** https://www.geeksforgeeks.org/image-processing-in-python/
- **CAT AND DOG IMAGE CLASSIFICATION-** https://www.kaggle.com/akrsnv https://www.analyticsvidhya.com/blog/2021/06/beginner-friendly-project-cat-and-dog-classification-using-cnn/
- **THRESHOLD-** https://www.geeksforgeeks.org/python-thresholding-techniques-using-opencv-set-1-simple-thresholding/
- **SHIFT-** https://www.analyticsvidhya.com/blog/2019/10/detailed-guide-powerful-sift-technique-image-matching-python/
- **WATER SHED SEGMENTATION TECHNIQUE-** https://www.geeksforgeeks.org/image-segmentation-with-watershed-algorithm-opencv-python/
- **REGION_BASE_SEGMENTATION-** https://www.analyticsvidhya.com/blog/2019/04/introduction-image-segmentation-techniques-python/
- **CANNY_EDGE_DETECTION-** https://www.analyticsvidhya.com/blog/2019/04/introduction-image-segmentation-techniques- python/
- **LINEAR REGRESSION FOR MOBILE PRICE PREDICTION-** https://www.kaggle.com/datasets/vishwajeetvijaykale/mobile-price-prediction-very-beginner-friendly https://www.analyticsvidhya.com/blog/2022/02/learn-mobile-price-prediction-through-four-classification-algorithms/

- **OBJECT DETECTION USING ALREADY EXISTING MODEL-** https://www.analyticsvidhya.com/blog/2020/08/top-4-pre-trained-models-for-image-classification-with-python-code/
  https://raw.githubusercontent.com/anishathalye/imagenet-simple-labels/master/imagenet-simple-labels.json

Signature of the Intern with Date

Signature, Date, and Stamp of the Supervising Faculty