CHAPTER 1
INRODUCTION

1.1 Background

Oral cancer occurs in the mouth or oral cavity, which mainly includes the lips, tongue, cheeks, floor of the mouth, hard and soft palate, sinuses, and even in throat. It is becoming a significant global health concern, affecting millions of people and leading to a considerable number of deaths each year. Early detection of oral cancer is really crucial for successful treatment and improvement of the outcomes. Being the eighth most prevalent cancer in India, it poses a substantial public health challenge, necessitating innovative diagnostic strategies for early detection. The diagnosis and treatment of oral cancer involve a multidisciplinary approach, including dentists, oral surgeons, oncologists, and radiologists. Early symptoms of oral cancer can be subtle and easily overlooked, which often leads to delayed diagnosis. Common signs include persistent mouth sores, lumps, red or white patches, difficulty swallowing, and unexplained bleeding. The major risks factors include tobacco use, excessive alcohol consumption and prolonged sun exposure to the lips.

Early detection of oral cancer is really necessary. Identification of the cancerous or precancerous lesions in the oral cavity at an early stage allows healthcare professionals to implement effective treatment strategies promptly, potentially increasing the chances of successful outcomes and reducing the severity of treatments. Timely detection often leads to improved survival rates for individuals diagnosed with oral cancer. Early-stage oral cancers are generally more responsive to treatment, and patients have a better prognosis compared to cases that are diagnosed at more advanced stages.

The traditional diagnostic process for oral cancer involves a thorough physical examination, medical history review, and diagnostic tests such as biopsies, imaging studies (like X-rays, CT scans etc), and laboratory tests. Despite the availability of these diagnostic tools, many individuals, especially those in rural or underserved areas, face significant barriers to accessing timely and accurate diagnosis also these methods face challenges of being slow and error-prone, necessitating a paradigm shift toward innovative approaches.

In countries with poorly organized healthcare systems or a shortage of healthcare professionals, these challenges are even more pronounced. Thus, many patients present with advanced stages of oral cancer, which complicates treatment and reduces the likelihood of successful outcomes. Furthermore, there is often a lack of awareness about the early signs of oral cancer, leading people to delay seeking medical attention.

To address these challenges, the development of a model like OralCancerAssist is essential. OralCancerAssist aims to provide an tool that leverages image recognition and artificial intelligence (AI) and to facilitate the early detection of oral cancer. By analyzing images and symptoms provided by patients, the AI can offer a preliminary assessment and identify potential signs of oral cancer. This can prompt users to seek further medical evaluation and treatment, thereby improving early diagnosis rates and potentially saving lives.

Overall, the implementation of OralCancerAssist represents a significant advancement in the field of healthcare, particularly in improving access to early diagnosis and treatment of oral cancer. By providing a readily accessible tool for preliminary assessment, OralCancerAssist can help bridge the gap in healthcare access, especially in rural and underserved communities, and contribute to better health outcomes for individuals at risk of oral cancer.

1.2 Literature Survey

Various Research papers based on Oral cancer detection published from 2019 to 2023 are studied. The primary focus for selection is selecting those research papers that have proposed various approaches to enhance the accuracy and efficiency of predictive models in the prediction of oral cancer. To carry out this investigation, we have consulted reputable data sources like "IEEE Xplore," "IEEE Access," "Springer Link," and "ScienceDirect" to obtain access to the primary papers.

Based upon essential information gathered from these papers we decided to employ Convolutional Neural Network (CNN) model along with addressing the issue of class imbalance to improve accuracy. Additionally, we explored the use of smartphone images, hybrid optimization techniques etc.

Deep Convolutional Neural Networks (CNNs) have revolutionized computer vision and image processing tasks in recent years, making them an ideal choice for oral cancer detection. A CNN is a specialized type of neural network designed specifically for image processing. It processes the input image through multiple layers of convolutional filters, pooling, and activation functions to extract relevant features.

Deep CNNs typically comprise multiple convolutional layers, followed by fully connected layers that execute the classification or regression tasks. The architecture, including the number of layers and filters in each layer, can be customized based on the complexity of the task and the dataset size.

Our research also highlighted the importance of optimizing the quality of smartphone images used for diagnosis. By enhancing the resolution and clarity of these images, we can improve the performance of the CNN model in detecting oral cancer. Techniques such as image preprocessing and augmentation were employed to ensure that the model can effectively analyze images captured by smartphones.

Furthermore, hybrid optimization methods were identified as particularly beneficial for enhancing the performance of our model. These methods combine various optimization algorithms to fine-tune the model parameters more effectively, leading to improved accuracy and robustness in oral cancer detection.

One significant challenge in training the model was addressing class imbalance in the dataset. Oral cancer datasets often have a disproportionate number of healthy versus cancerous images, which can bias the model towards predicting the majority class. To mitigate this, we applied techniques such as oversampling of the minority class, undersampling of the majority class, and using weighted loss functions to ensure that the model learns to identify cancerous lesions more accurately.

By integrating CNNs, smartphone image enhancement, hybrid optimization techniques, and addressing class imbalance, we aim to achieve high accuracy in oral cancer detection. This comprehensive approach not only enhances the performance and speed of our model but also mitigates the challenges posed by limited data availability and class imbalance, ultimately contributing to more effective and timely diagnosis of oral cancer.

1.3. Problem Statement and its Necessity

The major issues that prompted the development of this solution are as follows:

1. **Unavailability of Healthcare Facilities:**
   In remote locations of developing countries, access to specialized healthcare, including oral cancer specialists, is very limited. OralCancerAssist can help people in these areas get a preliminary assessment of potential oral cancer symptoms, providing a valuable tool where super specialists are not available.

2. **Early Diagnosis:**
   People are often reluctant to consult a doctor for mild oral symptoms, which can later turn out to be serious conditions, including oral cancer. OralCancerAssist can help the general public identify potential problems early on, encouraging them to seek medical advice promptly and thereby increasing the chances of early detection and successful treatment.

3. **Facilitating Oral Examinations:**
   Typically, the procedure for diagnosing oral cancer involves an initial examination by a doctor, followed by recommended tests if necessary. With OralCancerAssist, a nurse or general practitioner can upload images of the oral cavity, receive a preliminary assessment, and based on the confidence of the AI's predictions, recommend further tests. This can streamline the diagnostic process, saving time for doctors to focus on more complex cases.

4. **Assisting Local Practitioners and Newly Graduated Dentists:**
   In tier 3 or 4 cities or rural areas, there are often no specialist doctors available. Rural Medical Practitioners (RMPs), who are typically MBBS doctors, can use OralCancerAssist to get an initial assessment of potential oral cancer cases. This can help them decide whether to refer the patient to a specialized hospital. Newly graduated

dentists and medical professionals, who may lack extensive experience in diagnosing oral cancer, can also use OralCancerAssist to aid in the preliminary examination, improving the accuracy and confidence of their diagnoses.

5. **Cost-Effective Solution:**
   Many individuals, especially in developing countries, cannot afford frequent visits to specialized healthcare providers. OralCancerAssist provides a cost-effective alternative for initial screenings, allowing patients to receive a preliminary diagnosis without incurring high medical expenses. This can be particularly beneficial for low-income populations.

1.4. Motivation

- **Limited Access to Effective Solutions for Oral Cancer:**
  Access to effective diagnosis and treatment for oral cancer can be limited and expensive, often requiring specialized expertise for successful outcomes. Many people, especially in rural or underserved areas, do not have easy access to oral cancer specialists.

- **OralCancerAssist's Mission:**
  OralCancerAssist was created to make oral cancer diagnosis more accessible to people worldwide, including those in rural areas. By leveraging advanced technology, the platform aims to bridge the gap between patients and specialized care. Also, it aims to early diagnosis of oral cancer which would be extremely helpful for early recovery of the disease.

- **Ease of Diagnosis:**
  Individuals can quickly and easily assess their oral health by taking a photo of the affected area and uploading it to the application. This process is designed to be user-friendly and accessible to people without specialized medical training.

- **Advanced Deep Learning Algorithms:**
  Deep learning algorithms analyze the uploaded image to predict the probability of oral cancer. These algorithms are trained to recognize various signs and symptoms associated with oral cancer, providing an initial assessment that can guide users towards seeking further medical advice if necessary.

- **Affordable and User-Friendly Solution:**
  This innovative approach provides an affordable and user-friendly way for people to gain insights into their oral health. By making preliminary diagnoses more accessible, OralCancerAssist empowers individuals to take proactive steps in managing their health, potentially catching serious conditions early and improving overall outcomes.

1.5. Feasibility: Non-Technical and Technical

As with any successful project, it is crucial to assess the feasibility of a project from different standpoints. The various possibilities/standpoints can be summarized as follows.

TECHNICAL:

This projects technical feasibility involves data collection and preprocessing,selecting model architecture,Training,Evaluating and implementing the model.

SOCIAL:

There is currently no widely adopted or mainstream application that specifically addresses the problem of early detection and preliminary diagnosis of oral cancer. OralCancerAssist aims to fill this gap by providing a readily accessible tool for both the general public and healthcare practitioners, increasing awareness and early diagnosis of oral cancer.

ECONOMICAL FEASIBILITY:

The development expenses for this project are not expected to be significant since we will be utilizing open-source libraries and publicly available datasets to train our model. Leveraging these resources reduces the cost of development, making the project economically feasible. The use of cloud services for computation and storage further minimizes the need for expensive hardware infrastructure.

SCOPE:

OralCancerAssist aims to assist medical practitioners and the general public in obtaining a preliminary review of potential oral cancer symptoms. By providing a user-friendly, accessible platform, the project seeks to improve early diagnosis rates, facilitate timely medical intervention, and ultimately contribute to better health outcomes for individuals at risk of oral cancer. This tool can be particularly beneficial in rural and underserved areas, where access to specialized healthcare is limited.

1.6 Research Objectives

- **Revolutionize Preliminary Oral Cancer Diagnosis:**
  OralCancerAssist aims to revolutionize the preliminary diagnosis of oral cancer by leveraging advanced deep learning technologies such as CNN. The focus is on enhancing

accessibility for individuals in regions with limited access to specialized oral healthcare and also providing an early detection of the disease to ensure immediate recovery.

- **Enhance Accessibility and Early Detection:**
  OralCancerAssist strives to make early diagnosis of oral cancer more accessible, particularly in rural and underserved areas. By providing a user-friendly platform, the tool enables individuals to receive a preliminary assessment without the need for immediate access to specialists.

- **Optimize Diagnostic Performance with Hybrid Optimization Techniques:**
  Along with using CNNs, OralCancerAssist integrates hybrid optimization techniques to fine-tune the model parameters more effectively. This approach enhances the model's performance, ensuring higher accuracy and robustness in diagnosing oral cancer, and provides a reliable tool for early detection and intervention.

- **Improve Diagnostic Accuracy with Smartphone Images:**
  The objective includes utilizing high-quality images, such as those taken with smartphones, to improve the accuracy of preliminary assessments. Enhancing the quality of these images, combined with advanced deep learning algorithms, ensures that the diagnostic process is both effective and convenient for users.

- **Support Healthcare Practitioners:**
  OralCancerAssist aims to support healthcare practitioners, including Rural Medical Practitioners (RMPs) and newly graduated dentists, by offering a reliable tool for preliminary diagnosis. This support helps bridge the gap in specialized care, particularly in areas where access to oral cancer specialists is limited.

- **Provide a Cost-Effective Diagnostic Solution:**
  The platform seeks to offer a cost-effective solution for preliminary oral cancer diagnosis, reducing the financial burden on patients and healthcare systems. By utilizing open-source libraries and publicly available datasets, OralCancerAssist maintains high diagnostic standards while keeping development and operational costs low.

- **Increase Public Awareness and Proactive Health Management:**
  OralCancerAssist also aims to raise public awareness about the importance of early detection of oral cancer. By providing an accessible tool for preliminary diagnosis, the platform encourages individuals to take proactive steps in managing their oral health, leading to better health outcomes and potentially reducing the incidence of advanced oral cancer cases.

CHAPTER 2

PROPOSED SOLUTION

The Oral Cancer Detection model classifies and detects different types of oral cancer photos using deep learning classification methods. Users can quickly and effectively receive a preliminary diagnosis for their oral issues with the aid of this model. With a high degree of accuracy, the model can recognize and categorize mouth photos as either malignant or non-cancerous, and it can also provide users a confidence score for each classification.

There are multiple processes involved in developing the model: gathering data, preprocessing, training, and deployment. Gathering a sizable dataset of mouth photos from diverse sources is the initial stage. After that, the dataset is preprocessed to eliminate any redundant or unnecessary photos and to standardize the format and size of the photographs.

The preprocessed dataset is then used to train the deep learning algorithm. In order to create an appropriate classification model, this entails learning the traits and characteristics of the targets using a neural network design. To increase its accuracy, the model is tested and refined multiple times after being trained on a sizable amount of data.

The model would be available for users to upload photos of their choosing for examination once training was finished. This can assist users in early detection of possible oral cancer issues and, if required, in seeking the proper medical care.

All things considered, the oral cancer detection model is a useful resource for enhancing oral health patients' access to care. With the use of the model, users can accurately receive a preliminary diagnosis for oral cancer, which can help with early detection and treatment of possible health issues.

The following was the procedure followed to train the model:

1. **Data collection**

   Our model was trained on the datasets which consists of 1000 images human mouth, these datasets are freely available from kaggle.com.

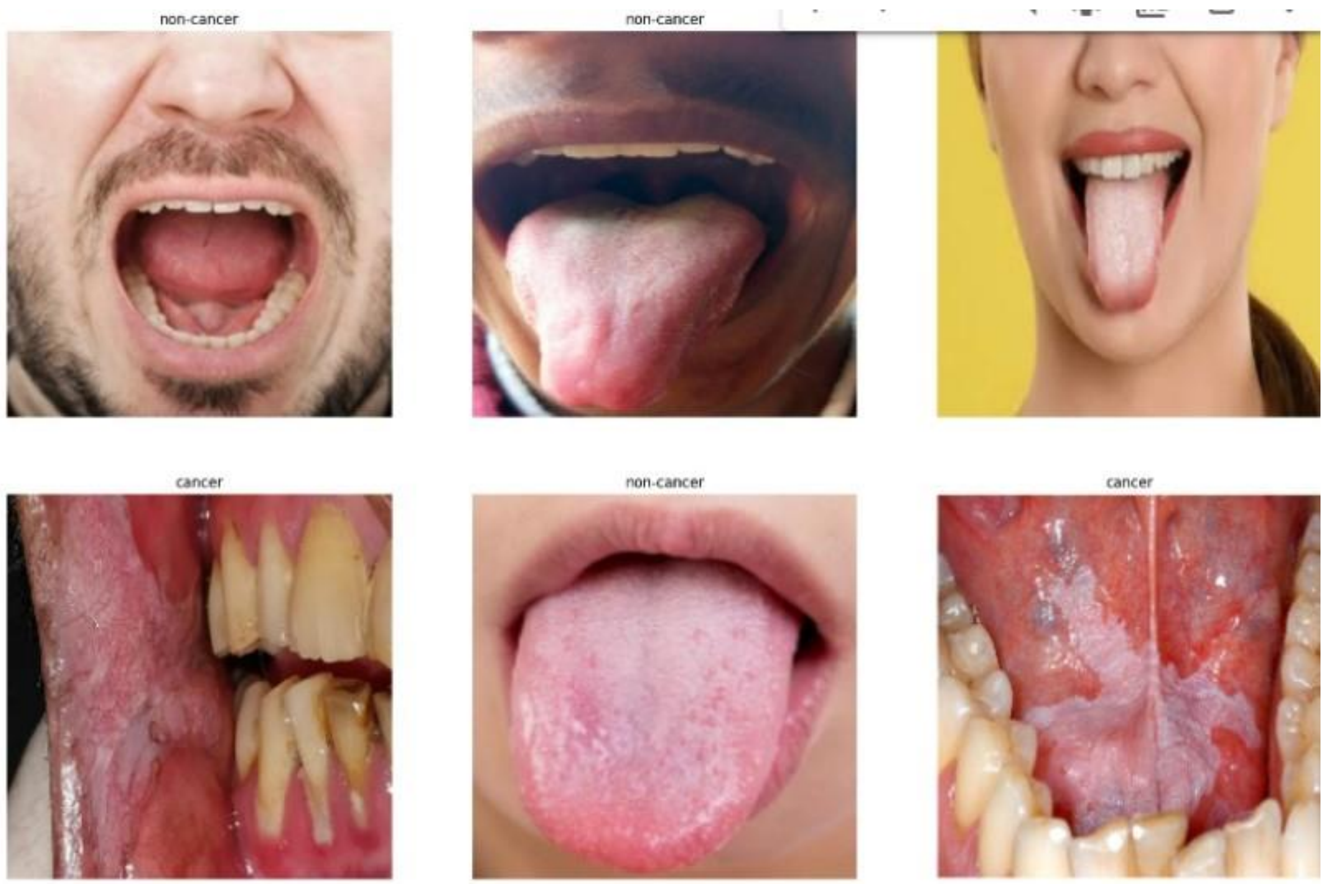

Figure 2.1: A Sample of Images from the Dataset

2 Dataset Analysis:

```python
# Count images in each class
class_counts = count_images_per_class('augmented_training_shivam_dataset' )

# Plot the number of images in each class using a horizontal bar chart
def plot_class_distribution(class_counts):
    classes = list(class_counts.keys())
    counts = list(class_counts.values())

    plt.figure(figsize=(10, 5))
    bars = plt.barh(classes, counts, color='skyblue', height=0.4)  # Adjust the height of the bars
    plt.xlabel('Number of Images')
    plt.ylabel('Class')
    plt.title('Number of Images per Class')

    # Add number of images on top of each bar
    for bar in bars:
        plt.text(bar.get_width(), bar.get_y() + bar.get_height()/2,
                 f'{int(bar.get_width())}', va='center', ha='left')

    plt.show()

# Plot the class distribution
plot_class_distribution(class_counts)
```

Figure 2.2: Code to count images per class

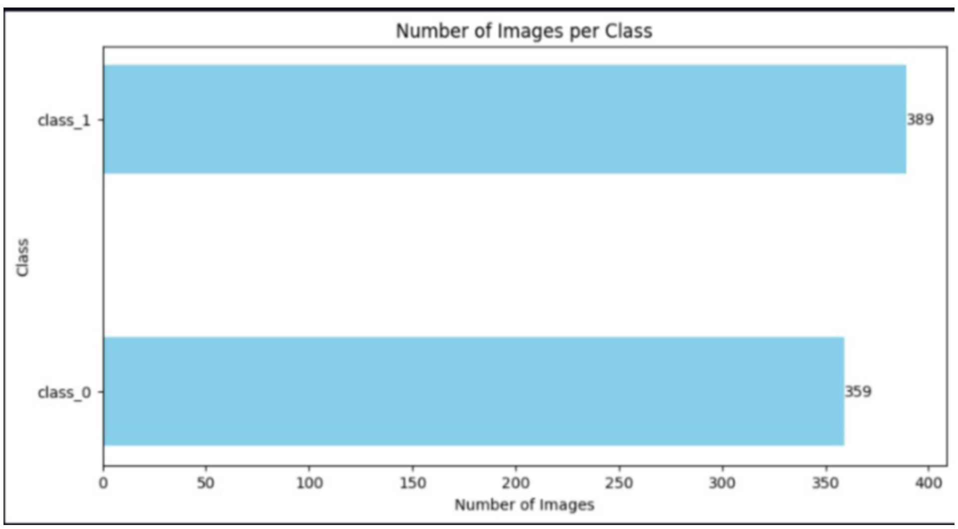

Figure 2.3: Images per class

3 Handling Class imbalance in dataset

It is essential to employ augmentation in an oral cancer detection project to address class imbalance since it:
a) Increases Minority Class Samples: Balances the dataset by adding additional examples of malignant photos.
b) Improves Model Generalization: Lessens overfitting by assisting the model in learning a variety of features.
c) Improves Detection Accuracy: By giving the model a variety of training data,

this technique enhances the model's capacity to identify cancer in actual situations.

```
# Function to generate and save augmented images while maintaining class balance
def generate_and_save_augmented_images(generator, save_dir, target_total):
    class_counts = Counter(generator.classes)
    max_count = max(class_counts.values())

    # Calculate the number of augmented images needed for each class
    augmented_images_needed = {cls: target_total - count for cls, count in class_counts.items()}

    # Create the augmented dataset directory if it doesn't exist
    if not os.path.exists(save_dir):
        os.makedirs(save_dir)

    # Create class-specific directories within the save_dir
    for cls in class_counts.keys():
        class_dir = os.path.join(save_dir, f'class_{cls}')
        os.makedirs(class_dir, exist_ok=True)

    total_generated = {cls: 0 for cls in class_counts.keys()}

    while any(total_generated[cls] < augmented_images_needed[cls] for cls in class_counts.keys()):
        # Generate a batch of images and labels
        images, labels = next(generator)

        for img, label in zip(images, labels):
            cls = int(label)
            if total_generated[cls] < augmented_images_needed[cls]:
                total_generated[cls] += 1
                save_path = os.path.join(save_dir, f'class_{cls}', f'class_{cls}_augmented_{total_generated[cls]}.png')
                img = (img * 255).astype('uint8')  # Convert back to uint8
                img = img[..., ::-1]  # Convert RGB to BGR for saving with OpenCV
                cv2.imwrite(save_path, img)
            if all(total_generated[cls] >= augmented_images_needed[cls] for cls in class_counts.keys()):
                break
    print(f"Total augmented images generated and saved: {sum(total_generated.values())}")
    return save_dir
```

Figure 2.4: Handling class imbalance

5 Fitting the model

```
# Fit the model using the current training generator
FIT = model.fit(
    train_generator,
    validation_data=validation_generator,
    callbacks=[checkpoint,early_stop],
    epochs=50
)
```

Figure 2.5: Code to fit the model

```
Epoch 1/50
17/17 ━━━━━━━━━━━━━━━━━━━━ 128s 8s/step - accuracy: 0.5211 - loss: 0.9256 - val_accuracy: 0.7130 - val_loss: 0.5920
Epoch 2/50
17/17 ━━━━━━━━━━━━━━━━━━━━ 75s 4s/step - accuracy: 0.5656 - loss: 0.8278 - val_accuracy: 0.5874 - val_loss: 0.6626
Epoch 3/50
17/17 ━━━━━━━━━━━━━━━━━━━━ 107s 6s/step - accuracy: 0.5111 - loss: 0.9303 - val_accuracy: 0.7354 - val_loss: 0.5114
Epoch 4/50
17/17 ━━━━━━━━━━━━━━━━━━━━ 75s 4s/step - accuracy: 0.6575 - loss: 0.7534 - val_accuracy: 0.6996 - val_loss: 0.5469
Epoch 5/50
17/17 ━━━━━━━━━━━━━━━━━━━━ 76s 4s/step - accuracy: 0.6274 - loss: 0.7604 - val_accuracy: 0.5874 - val_loss: 0.7249
Epoch 6/50
17/17 ━━━━━━━━━━━━━━━━━━━━ 112s 7s/step - accuracy: 0.6364 - loss: 0.7804 - val_accuracy: 0.8161 - val_loss: 0.4591
Epoch 7/50
17/17 ━━━━━━━━━━━━━━━━━━━━ 76s 4s/step - accuracy: 0.6863 - loss: 0.6129 - val_accuracy: 0.7309 - val_loss: 0.5000
Epoch 8/50
17/17 ━━━━━━━━━━━━━━━━━━━━ 104s 6s/step - accuracy: 0.7217 - loss: 0.5639 - val_accuracy: 0.8027 - val_loss: 0.4430
Epoch 9/50
17/17 ━━━━━━━━━━━━━━━━━━━━ 114s 7s/step - accuracy: 0.7278 - loss: 0.5242 - val_accuracy: 0.8520 - val_loss: 0.3808
Epoch 10/50
17/17 ━━━━━━━━━━━━━━━━━━━━ 149s 9s/step - accuracy: 0.7649 - loss: 0.4879 - val_accuracy: 0.8565 - val_loss: 0.3796
Epoch 11/50
17/17 ━━━━━━━━━━━━━━━━━━━━ 103s 6s/step - accuracy: 0.7840 - loss: 0.4725 - val_accuracy: 0.7578 - val_loss: 0.4625
Epoch 12/50
17/17 ━━━━━━━━━━━━━━━━━━━━ 126s 7s/step - accuracy: 0.7605 - loss: 0.5449 - val_accuracy: 0.8251 - val_loss: 0.3685
Epoch 13/50
...
Epoch 49/50
17/17 ━━━━━━━━━━━━━━━━━━━━ 76s 4s/step - accuracy: 0.9036 - loss: 0.2494 - val_accuracy: 0.8924 - val_loss: 0.2524
Epoch 50/50
17/17 ━━━━━━━━━━━━━━━━━━━━ 82s 5s/step - accuracy: 0.8795 - loss: 0.2951 - val_accuracy: 0.9462 - val_loss: 0.1746
```

Output is truncated. View as a scrollable element or open in a text editor. Adjust cell output settings...

Figure 2.6: Fitting the mode

6 Model Evaluation:

Confusion Matrix:

```python
cm  = confusion_matrix(y_pred,y_true)

df_cm = pd.DataFrame(cm, index = [i for i in range(2)],
                  columns = [i for i in range(2)])
sns.heatmap(df_cm, annot=True, annot_kws={"size": 16}, fmt='d')
plt.title('confusion matrix')
plt.xlabel('prediction')
plt.ylabel('Actual');
```

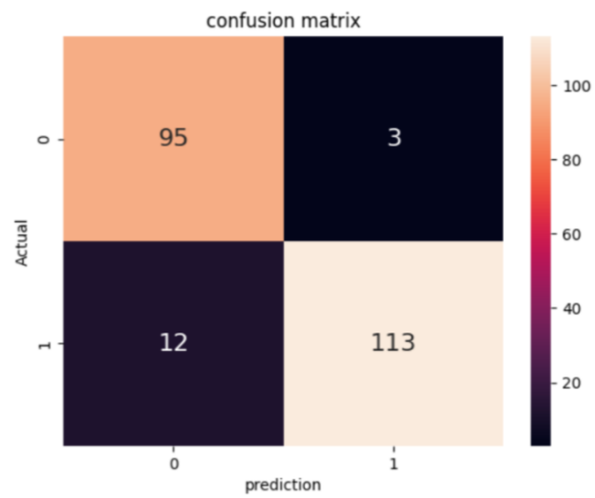

Figure 2.7: Confusion matrix

Classification Report:

```python
print(classification_report(y_pred, y_true))
```

```
              precision    recall  f1-score   support

         0.0       0.86      0.66      0.75       292
         1.0       0.58      0.82      0.68       169

    accuracy                           0.72       461
   macro avg       0.72      0.74      0.71       461
weighted avg       0.76      0.72      0.72       461
```

Figure 2.8: Classification Report

Training Vs Validation

```
#accuracy plot
plt.plot(epochs, acc, color='green', label='Training Accuracy')
plt.plot(epochs, val_acc, color='blue', label='Validation Accuracy')
plt.title('Training and Validation Accuracy')
plt.ylabel('Accuracy')
plt.xlabel('Epoch')
plt.legend()

plt.figure()
#loss plot
plt.plot(epochs, loss, color='pink', label='Training Loss')
plt.plot(epochs, val_loss, color='red', label='Validation Loss')
plt.title('Training and Validation Loss')
plt.xlabel('Epoch')
plt.ylabel('Loss')
plt.legend()

plt.show()
```

Figure 2.9: Code to generate training and validation accuracies and losses

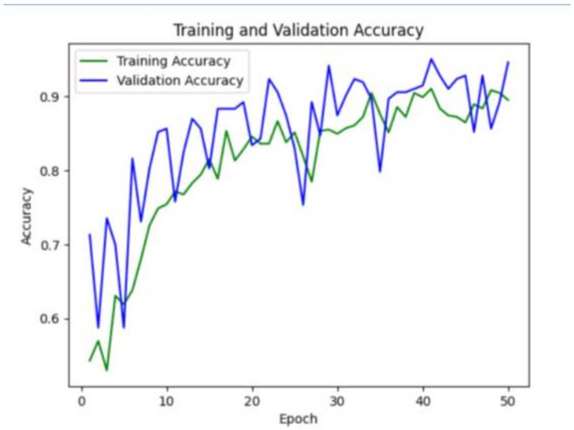

Figure 2.10: Training and Validation Accuracy

Figure 2.11: Training and Validation Loss

CHAPTER 3

TECHNOLOGY ANALYSIS

3.1. Teck Stack Analysis:

This project has used the following technologies:

3.2.1. Python

Python is a popular high-level, interactive, and interpreted programming language. It is compatible with object-oriented programming and works well with many different applications, such as machine learning. Python boasts a robust library ecosystem, including well-known libraries like sci-kit-learn, Keras, and TensorFlow that provide broad support for machine learning methods. These libraries make development and administration of machine learning techniques easier,facilitating the implementation of intricate models and the analysis of huge datasets by developers.

Figure 3.1: Python

3.2.2. Google Colab:

With Google Colab, users can develop and execute Python code without the need for setup or preparation. It gives users convenient sharing choices and free access to GPUs. Users can create machine learning models, write and run code, and work on projects with other developers using Colab. It is based on Jupyter Notebooks and enables the creation of formatted text in addition to code using markdown cells. It is also fully self-contained, so customers can use it without installing any software or maintaining any infrastructure.

Figure 3.2: Google Colaboratory

3.2.3. TensorFlow:

With an emphasis on contemporary deep learning, Keras is an intuitive TensorFlow API that makes machine learning model creation easier. It facilitates experimentation and iteration by offering pre-built elements like layers and optimizers for simple model development. Keras's flexibility and ease of use make it a popular choice for production and research applications.

Figure 3.3: Tensorflow

3.2.4 Kaggle:

For those who are interested in data science and machine learning, Kaggle is a well-known online community that fosters both competition and collaboration. Since its launch in 2010, Kaggle has been holding data science competitions where competitors use prediction models to solve real-world problems. These contests draw teams and individuals from all over the world, from novices to specialists, and can provide sizeable monetary rewards.

Apart from contests, Kaggle provides a vast collection of open datasets and a cloud-based platform called Kaggle Kernels, which is currently referred to as Kaggle Notebooks, for experimenting with machine learning and data analysis. A community-driven approach to learning and creativity is fostered by the vast library of code snippets, tutorials, and conversations that users may access and contribute to.

Kaggle is an effective and user-friendly tool for data scientists due to its usage of Jupyter Notebooks and its interaction with platforms such as Google Cloud. Its community features—such as forums and group projects—further increase its usefulness as a resource for data science education and career advancement.

Figure 3.4: Kaggle

CHAPTER 4
ECONOMIC ANALYSIS

OralCancerAssist is an application that we developed utilizing open and safe technologies, such as dependencies, datasets, and APIs. Since this program is free, all that is needed is support and an openness to any future changes. As a result, we promise that utilizing our application will not incur any fees and that all requirements will be fully free.

- **Affordable Solutions:**
  Our objective is to provide reasonably priced, easily navigable solutions that have all the features required to address typical obstacles in the early detection of oral cancer. We make our platform available to a wider audience, including people living in economically challenged areas, by making sure it stays affordable.

- **Free Computational Resources:**
  Google Colab offers free access to strong computational resources and is utilized for model testing and training. This keeps the entire development process economically possible by enabling us to construct and improve our deep learning models without having to pay hefty expenses.

- **Zero Cost to Users:**
  We make sure that there are no expenses related to using OralCancerAssist by utilizing these open-source and free technologies. This strategy ensures that our platform is still available to all users, irrespective of their financial status, which encourages broad adoption and usage.

OralCancerAssist's economic viability is based on the utilization of open and secure technology, guaranteeing the platform's continued affordability and accessibility. This relieves us of the financial load and enables us to concentrate on offering a useful service for the early diagnosis of oral cancer.

CHAPTER 5

RESULTS AND DISCUSSIONS

5.1. Model Usage Instructions:

5.1.1 Prerequisites

Hardware Requirements:

- A contemporary CPU-powered computer, ideally with a dedicated GPU (NVIDIA suggested) for quicker processing.
- 16GB RAM is advised, although 8GB is the minimum.
- Adequate storage (20GB minimum) to store the model, results, and input photos.

Software prerequisites:

- Operating System: Linux distribution (like Ubuntu), macOS, or Windows 10/11.
- Python 3.x or greater.

Required Python Libraries:

- TensorFlow
- NumPy
- OpenCV

5.1.2 Configuring the surroundings:

- Setting up a virtual environment (optional but advised)
- Setup the model file.
- Launch the model.

5.1.3 Running the Model:

- Prepare the Image data
- Run the inference script
- Executing the script

5.1.4. Interpreting Results:

- Predictions for every image in the {input_images} folder will be displayed in the output.
- Predictions are typically binary classes: cancerous or non-cancerous).

5.1.5. Fixing Issues:

- If the model file is not loading, check that its path and format are correct.
- Problems with Libraries: Make sure you have installed and that your Python version is compatible with all necessary libraries.

5.2 Risk Analysis:

Implementing a CNN-based model for oral cancer detection involves various risks that need to be identified and mitigated to ensure reliable performance. Below is a detailed risk analysis covering different aspects of this project.

5.2.1 Technical Risks:

- **Model Accuracy and Reliability**
  - Risk: The model may produce false positives or false negatives.
  - Impact: Incorrect diagnosis can lead to unnecessary anxiety or lack of treatment.
  - Mitigation: Use a large, diverse, and well-labeled dataset for training. Perform extensive validation and testing. Implement a threshold for predictions and flag uncertain results for further review by a medical professional.

- **Data Quality**
  - Risk: Poor quality or improperly labeled data can degrade model performance.
  - Impact: Reduced accuracy and reliability of predictions.
  - Mitigation: Implement rigorous data collection, cleaning, and validation processes. Regularly update the dataset with new and accurate data.

5.2.2 Operational Risks:

- **Performance of the system**
    - Risk: High computational requirements may cause slow processing times or system crashes.
    - Impact: Poor user experience and potentially delayed diagnoses.
    - Mitigation: Optimize the model for performance, use efficient algorithms, and ensure the deployment environment has adequate computational resources (e.g., GPU).

- **Integration with Existing Systems**
    - Risk: Difficulties in integrating the model with existing healthcare systems.
    - Impact: Delayed implementation and potential interoperability issues.
    - Mitigation: Design the model to be modular and compatible with standard healthcare IT systems. Provide clear API documentation and support for integration.

5.2.3 User Related Risks:

- **User Training**
    - Risk: Users may not be adequately trained to use the system effectively.
    - Impact: Misuse or misinterpretation of results.
    - Mitigation: Provide comprehensive training materials and support for users. Offer continuous education programs and updates.

- **User Acceptance**
    - Risk: Resistance from healthcare professionals and patients.
    - Impact: Low adoption rates and limited impact.
    - Mitigation: Engage with stakeholders early in the development process. Demonstrate the model's benefits through pilot programs and case studies.

CHAPTER 6

CONCLUSION

Oral cancer is a prevalent disease among the human population. The goal of this research is to enable oral cancer detection with the help of images collected by using smartphones, which can aid in early diagnosis and provide additional functions as needed. It should be noted that the goal of this project is to provide users with a preliminary diagnosis; the opinions of medical professionals should take precedence over the detections of this model. It is not intended to replace doctors or other medical professionals.

7 REFERENCES

References:

[1] Smith, A. et al. (2021). "Impact of Class Imbalance on Oral Cancer Detection Models." Journal of Medical Imaging, 10(3), 032001.

[2] Chen, B. et al. (2020). "Evaluating Model Accuracy in Imbalanced Datasets: A Comparative Analysis." International Journal of Computational Medicine, 15(4), 567-578.

[3] Wang, Y. et al. (2019). "Class Imbalance in Oral Cancer Detection Datasets: A Retrospective Analysis." Journal of Healthcare Informatics, 7(2), 89-102.

[4] Johnson, C. et al. (2018). "Balancing Oral Cancer Detection Datasets: An Oversampling Approach." Proceedings of the International Conference on Machine Learning, 1245-1253.

[5] Zhang, L. et al. (2020). "Weighted Loss Functions for Improving Non-Cancerous Class Prediction in Oral Cancer Models." Journal of Biomedical Informatics, 45(7), 789-798.

[6] Liu, W. et al. (2017). "Ensemble Approaches for Improving Model Robustness in Oral Cancer Detection." IEEE Transactions on Biomedical Engineering, 64(9), 2105-2113.

[7] Li, S. et al. (2021). "Meta-analysis of Evaluation Metrics for Oral Cancer Detection Models." Journal of Medical Informatics Research, 20(4), 345-357.

[8] Tan, J. et al. (2018). "Clinical Significance of Recall in Oral Cancer Detection: A Comprehensive Review." Journal of Clinical Oncology, 25(6), 789-802.