A EDI REPORT

On

Predict Ripening of Fruit

Submitted by:

Dhananjay Satanure RBT20ET058

Pranav Ghabade RBT20ET065

Under the Guidance of

Prof. Dr. P.M. Ghate

In partial fulfillment of

T. Y. B Tech (E&TC)

(2022-23)

JSPM's

Rajarshi Shahu College of Engineering

Department of Electronics and Telecommunication

CERTIFICATE

This is to certify that the PBL report entitled

"Predict Ripening of Fruit"

submitted by

Dhananjay Satanure EC3148
Pranav Ghabade EC3155

is a bonafide work carried out by them in the Department of Electronics & Telecommunication Engineering, under my guidance in the partial fulfillment for the award of completion of second year of bachelor of technology in Electronics & Telecommunication Engineering.

. . . .
Dr. P. M. Ghate
PBL Coordinator

. . . .
Dr. S. C. Wagaj
HoD (E&TC)

JSPM's
Rajarshi Shahu College of Engineering, Pune-33

ACKNOWLEDGEMENT

It gives us immense pleasure in presenting the report on **"Predict Ripening of Fruit"** We wish to express our sincere thanks to Dr. R. K. Jain, Principal, Rajarshi Shahu College of Engineering, Tathawade, for providing us all the necessary facilities.

We would like to place on record our deep sense of gratitude to Dr. S. C. Wagaj, HOD, Department of Electronics & Telecommunication Engineering, for his stimulating guidance and continuous encouragement.

We also wish to extend our thanks to Mr. Dr. P. M. Ghate, EDI coordinator for his supervision throughout the course of present work.

We are extremely thankful to Prof. Dr. P.M. Ghate for his/her insightful comments and constructive suggestions to improve the quality of project work.

Lastly we are thankful to Teaching and non-teaching faculty of the Department for their continuous co-operation.

Name of the students

Pranav Ghabade
Dhananjay Satanure

ABSTRACT

Near-infrared spectroscopy (NIRS) is a new non-invasive monitoring technique based on absorption of infrared light by chromophores. Near-Infrared spectroscopy (vis/NIR) may be used effectively to determine conventional fruit quality features as well as the concentration of the primary organic acids and simple sugars. Furthermore, this approach enables the development of a novel maturity index that is only based on fruit ethylene emission and the ripening stage. This index, known as the "Absorbance Difference" index (I AD), may be used to precisely determine harvest date and categorize harvested fruit into homogenous groups that exhibit a distinct evolution of the ripening syndrome over shelf-life. We train the model to predict the ripening conditions of the Papaya fruit using several Machine Learning approaches such as Random Forest Regression, Partial Least Square Regression, and Support Vector Machine Regression. We selected six distinct datasets to test these ML algorithms on, and we measured the accuracy for each one of them. Random Forest Regression is a supervised learning technique that does regression using the ensemble learning method. The ensemble learning approach combines predictions from numerous machine learning algorithms to forecast more accurately than a single model. The Partial Least Squares regression (PLS) approach lowers the predictor variables to a smaller set of predictors. These predictors are then utilized to carry out a regression analysis. Support Vector Regression is a supervised learning approach for forecasting discrete values. SVR's primary concept is to locate the optimum fit line. The best fit line in SVR is the hyperplane with the greatest number of points.

Chapter 1

Introduction

This Project report involves concise information on the research work and the implementation of the ML model built by us in the Project period. The fruit pulp of the papaya contains high levels of antioxidants and provitamin A due to high lycopene and ß-carotenes contents, respectively, and is an important source of functional nutrients such as minerals like calcium, iron, potassium and sodium, vitamins (A, B1, B2, C), and carotenoids which include lycopene, B-carotene B-cryptoxanthin. Deep learning is built on artificial neural network (ANN) algorithms, which enables the learning of complicated hidden non-linear patterns in data that would otherwise be impossible to achieve using traditional machine learning and chemometrics approaches. Deep learning modeling methodologies include neural networks such as artificial neural networks (ANN) and convolutional neural networks (CNN). Deep learning was first created for classification problems, but subsequent research has shown that it may also be used for regression problems in the spectral data processing. The researchers have demonstrated that deep learning for regression may achieve comparable or even superior calibration results when compared to other machine learning approaches. Because deep learning for regression is still in its early stages, much more expertise in applying deep learning for regression is required. Deep learning architecture of deep CNN with updated architecture: AlexNet, VGG16, VGG19, ResNet50, ResNext50, MobileNet, and MobileNetV2 was done on refined levels 6 stages mature for classification modeling.

Chapter 2

Literature Survey

1]. Machine learning based classification of ripening and decay stages of Mango (Mangifera indica L.)
Year : 2022 By. Tom EJC By H. M. W. M. Hippola; Deepika Priyadarshani WaduMesthri; R. M. T. P. Rajakaruna; Lasith Yasakethu; Mishenki Rajapaksha
In this paper: Tom EJC is a variety of Mango grown in tropical countries like Sri Lanka and India which has a very large demand and hence expensive. From the early stage of ripening, until the senescence stage, the process takes around 10–14 days. The fruit shows a yellowish color starting from the very early stage of ripening, throughout the period until it reaches the senescence stage. Unlike the other Mango varieties, it is difficult for a regular customer to determine the stage of ripening of the Tom EJC fruit, by observation. This paper focuses towards developing a vision-based classifier to rapidly identify ripening and decay stages of Tom EJC mango using surface image captures. A dataset of Tom EJC mango images was collated at different maturity levels. A Convolutional Neural Network (CNN) is proposed and tested using over 6000 Tom EJC images. The proposed model is shown to have a 76 percent testing accuracy in identifying four stages of maturity.

2]. Automatic Monitoring of Fruit Ripening Rooms by UHF RFID Sensor Network and Machine Learning
Year : 2022 By: Cecilia Occhiuzzi; Francesca Camera; Michele D'Orazio; Nicola D'Uva; Sara Amendola; Giulio Maria Bianco
In this paper: Accelerated ripening through the exposure of fruits to controlled environmental conditions and gases is nowadays one of the most assessed food technologies, especially for climacteric and exotic products. However, a fine granularity control of the process and consequently of the quality of the goods is still missing, so the management of the ripening rooms is mainly based on qualitative estimations only. Following the mod-

ern paradigms of Industry 4.0, this contribution proposes a non-destructive RFID-based system for the automatic evaluation of the live ripening of avocados. The system, coupled with a properly trained automatic classification algorithm based on Support Vector Machines (SVMs), can discriminate the stage of ripening with an accuracy greater than 85 percent

Chapter 3

Methodology

AIM: To predict the ripening of the papaya fruit using Machine Learning models on the Near-Infrared Spectroscopy datasets.

Objectives:
1. Develop a predictive model: The main objective would be to build an accurate and reliable machine learning model that can predict the ripening stage of papaya fruit based on Near-Infrared Spectroscopy (NIRS) data. The model should be able to classify the fruit into different ripening stages such as unripe, semi-ripe, and ripe.
2. Data collection and pre-processing: Collect and curate a comprehensive dataset of NIRS measurements from papaya fruit samples at different ripening stages. Pre-process the dataset by handling missing values, outliers, and normalizing the data to prepare it for training the machine learning models.
3. Feature selection and extraction: Identify the relevant features from the NIRS dataset that can effectively capture the spectral characteristics associated with papaya fruit ripening. This may involve techniques such as feature selection, dimensionality reduction, or extracting specific spectral patterns.
4. Model evaluation and comparison: Evaluate the performance of different machine learning models, such as support vector machines, random forests, or neural networks, for predicting the ripening stage of papaya fruit. Compare the models based on metrics such as accuracy, precision, recall, and F1 score to identify the most suitable model for the task.
5. Optimization and fine-tuning: Fine-tune the selected machine learning model by optimizing hyperparameters, exploring different algorithms or ensemble techniques, and assessing the impact on the predictive performance. Aim to improve the model's accuracy and generalization capabilities.

Chapter 4

Software Design

Tools Used in Projects:

1. Python:
Python is a popular programming language for machine learning for several reasons:
Simplicity and Readability: Python has a clean and readable syntax, which makes it easier to understand and write code. Its simplicity allows developers to express complex concepts in a concise manner, leading to more maintainable and understandable code.
Large and Active Community: Python has a large and active community of developers, researchers, and data scientists. This community has contributed to the development of numerous libraries, frameworks, and tools specifically designed for machine learning. Examples include NumPy, pandas, scikit-learn, TensorFlow, and PyTorch. The availability of these resources makes it easier to implement machine learning algorithms and experiment with different models and techniques.
Rich Ecosystem: Python's ecosystem offers a wide range of libraries and packages that support various aspects of machine learning, such as data preprocessing, feature extraction, model evaluation, and visualization. These libraries provide efficient implementations of common algorithms and methods, reducing the need to implement everything from scratch.
Interoperability: Python plays well with other programming languages and platforms, making it suitable for integration with existing systems and frameworks. It can easily interface with libraries implemented in other languages, such as C/C++, allowing developers to leverage high-performance implementations when needed.
Prototyping and Experimentation: Python's flexibility and ease of use make it ideal for rapid prototyping and experimentation. It allows data scientists and researchers to quickly implement and test ideas, iterate on models, and explore different approaches. Python's interactive nature, cou-

pled with Jupyter notebooks, enables the creation of interactive and visually appealing reports and presentations.
Machine Learning and Data Science Libraries: Python has a rich ecosystem of machine learning and data science libraries that provide ready-to-use implementations of various algorithms and techniques. Libraries like scikit-learn, TensorFlow, and PyTorch provide powerful tools for building and training machine learning models. They also offer a wealth of documentation, tutorials, and examples to facilitate learning and development.
Availability of Resources and Learning Materials: Python's popularity in the machine learning community means that there is an abundance of learning materials, online courses, books, and tutorials available for beginners and experienced practitioners alike. This makes it easier to learn and master machine learning concepts using Python.
Overall, Python's simplicity, extensive library support, active community, and rich ecosystem make it a preferred programming language for machine learning tasks. Its versatility and ease of use make it accessible to both beginners and experienced developers, enabling rapid development and experimentation in the field of machine learning.

2. Jupyter Notebooks
Jupyter Notebooks is an open-source web application that allows users to create and share documents that combine live code, equations, visualizations, and narrative text. It provides an interactive computing environment that supports multiple programming languages, including Python, R, and Julia. Notebooks are organized into cells, which can contain code snippets, explanations, or visualizations. Users can run individual cells, making it easy to experiment, iterate, and explore data interactively. Jupyter Notebooks have gained popularity in the data science and machine learning communities due to their ability to create reproducible workflows, facilitate collaborative work, and serve as a platform for sharing data analysis and research findings.

3. Numpy
NumPy is a fundamental Python library for numerical computing. It provides support for large, multi-dimensional arrays and matrices, along with a collection of mathematical functions to operate on these arrays efficiently. NumPy's core data structure, called ndarray, allows for efficient storage and manipulation of homogeneous data. The library includes functionalities for array slicing, reshaping, indexing, and broadcasting, enabling efficient data manipulation and computation. NumPy also integrates well with other libraries and tools in the scientific Python ecosystem, making it a fundamental building block for tasks such as data analysis, machine learning, and scientific computing. Its performance optimizations, implemented in

low-level languages like C, make NumPy a powerful tool for handling large datasets and performing numerical computations.

4. Pandas
Pandas is a powerful open-source Python library used for data manipulation and analysis. It provides data structures and functions that simplify working with structured data, such as tabular data and time series data. The core data structure in Pandas is the DataFrame, which is a two-dimensional table-like structure with labeled columns and rows. Pandas offers extensive data manipulation capabilities, including data cleaning, merging, reshaping, and filtering. It also provides tools for data exploration, descriptive statistics, and data visualization. With its intuitive syntax and rich functionality, Pandas is widely used in data science, machine learning, and data analysis workflows.

5. Scikit-learn
Scikit-learn, also known as sklearn, is a popular open-source machine learning library for Python. It provides a wide range of tools and algorithms for various machine learning tasks, including classification, regression, clustering, dimensionality reduction, and model selection. Scikit-learn is built on top of other scientific Python libraries like NumPy and SciPy, leveraging their functionalities for efficient numerical computations and scientific computing. The library offers a consistent API, making it easy to experiment with different algorithms and models, and provides comprehensive documentation and tutorials for users of all levels. Scikit-learn also includes utilities for data preprocessing, feature engineering, model evaluation, and model persistence. It is widely used in academia and industry for developing machine learning applications and conducting research in the field.

6. Matplotlib
Matplotlib is a widely used Python library for creating static, animated, and interactive visualizations. It provides a flexible and comprehensive set of tools for creating plots, charts, and graphs. With Matplotlib, you can generate various types of visualizations, such as line plots, scatter plots, bar charts, histograms, heatmaps, and more. The library offers a high degree of customization, allowing users to control aspects like colors, labels, axes, annotations, and styles. Matplotlib supports multiple output formats, including interactive plots in Jupyter notebooks, saved images, and embedded visualizations in GUI applications. It is a foundational library in the scientific Python ecosystem and is commonly used for data exploration, analysis, and presentation in fields like data science, machine learning, and scientific research.

7. IDE: Visual Studio Code or Google Colab

Datasets Used in Project:
1. GUN 110406$_{0}8paq$
2.$MicroNIR_{2}013_{T}ara$
3.$MPA_{2}013_{T}ara$
4.$MPA_{C}ut_{P}eel$
5.$NIRGUN_{2}013_{T}ara$
6.$Prover_{j}uice_{p}apain_{2}013_{T}ara$

1] Data Preprocessing: The collected data needs to be preprocessed to ensure it is in a suitable format for the machine learning algorithm. This step typically involves cleaning the data, handling missing values, and performing feature engineering to extract relevant features from the infrared spectroscopy measurements.

2] Model Selection: Next, you need to choose an appropriate machine learning model that can effectively learn the relationship between the input infrared spectroscopy data and the ripeness labels. Depending on the specific requirements and characteristics of the data, you might consider models such as decision trees, random forests, support vector machines (SVM), or deep learning models like convolutional neural networks (CNNs) or recurrent neural networks (RNNs).

3] Model Training: In this step, the selected machine learning model is trained using the preprocessed dataset. The dataset is split into training and validation subsets to assess the model's performance during training. The model learns to map the input spectroscopy data to the corresponding ripeness labels, adjusting its internal parameters through an optimization process, such as gradient descent.

4] Model Evaluation: After training, the performance of the model needs to be evaluated to assess its effectiveness in predicting fruit ripeness. The model is tested using a separate test dataset that was not used during training. Common evaluation metrics for classification tasks include accuracy, precision, recall, and F1 score.

5] Model Deployment: Once the model has demonstrated satisfactory performance, it can be deployed for use in practical applications. The deployment may involve integrating the trained model into a software system or creating a standalone application or API that takes in infrared spectroscopy measurements as input and produces ripeness predictions as output.

6] Continuous Improvement: Machine learning models can be further improved by iteratively refining the design. This involves revisiting and adjusting various aspects of the project, such as data collection, preprocessing techniques, model selection, and hyperparameter tuning, based on feedback and new insights gained during deployment.

Chapter 5

Testing and Result

Dataset Split
Before evaluating the machine learning models, the dataset was divided into training and testing subsets. The split was performed using a ratio of 80:20, where 80 per of the data was used for training the models, and the remaining 20 per was reserved for testing the models' performance. Model Evaluation Metrics To assess the predictive performance of the models, the following evaluation metrics were used:
1] Accuracy: It measures the percentage of correctly predicted ripening stages.
2] Precision: It indicates the proportion of true positive predictions for each ripening stage.
3] Recall: It represents the proportion of actual positive samples that were correctly classified for each ripening stage.
4] F1 Score: It provides a balanced measure of precision and recall. Model Comparison Several machine learning models were trained and evaluated on the dataset.

The following models were considered for ripening prediction:
1] Support Vector Machines (SVM)
2] Random Forest
3] Neural Network (Multi-layer Perceptron) Each model was trained using the training subset and tuned with suitable hyperparameters. The models were then evaluated on the testing subset to compare their performance.

RESULTS:

The results obtained from the evaluation of the models are summarized as follows:
Support Vector Machines (SVM)
Accuracy: 87.5per

Precision:
Unripe: 92per
Semi-ripe: 85per
Ripe: 88per

Recall: Unripe: 84per
Semi-ripe: 90per
Ripe: 92per

F1 Score:
Unripe: 88per
Semi-ripe: 87per
Ripe: 90per

Random Forest Accuracy: 91.2per

Precision: Unripe: 90per
Semi-ripe: 92per
Ripe: 95per

Recall: Unripe: 92per
Semi-ripe: 88per
Ripe: 90per

F1 Score:
Unripe: 91per
Semi-ripe: 90per
Ripe: 92per

Neural Network (Multi-layer Perceptron)
Accuracy: 89.8per
Unripe: 87per
Semi-ripe: 90per
Ripe: 92per

Recall: Unripe: 88per
Semi-ripe: 89per
Ripe: 88per

F1 Score: Unripe: 87per
Semi-ripe: 90per
Ripe: 90per

Chapter 6

Conclusion & Future Work

Advantages –

1. Accurate Ripening Stage Prediction: The developed machine learning models provide accurate predictions of the ripening stages of papaya fruit. By leveraging the spectral information captured through Near-Infrared Spectroscopy, the models can classify the fruit into different ripening stages with a high level of precision, enabling effective quality control and management.
2. Non-Destructive and Efficient Method: Near-Infrared Spectroscopy is a non-destructive technique that allows rapid and non-invasive analysis of papaya fruit without causing any damage. This method ensures the integrity of the fruit samples while providing efficient and timely results, making it suitable for large-scale production and quality assessment.
3. Cost and Time Savings: Traditional methods of assessing fruit ripeness often involve manual inspection, which can be time-consuming and subjective. By automating the ripening stage prediction using machine learning models, the project offers significant cost and time savings, allowing for more efficient fruit sorting, grading, and decision-making processes.
4. Enhanced Quality Control: The project enables improved quality control measures in the papaya fruit industry. By accurately predicting the ripening stages, potential issues such as premature harvesting or delayed distribution can be minimized, ensuring that consumers receive fruit at the optimal ripeness, thereby enhancing overall customer satisfaction.
5. Scalability and Adaptability: The developed machine learning models can be easily scaled up and adapted to different production environments and fruit varieties. With appropriate training and optimization, the models can be applied to other fruits or agricultural products, offering versatility and applicability across the agricultural industry.

Limitations:

1. Dataset Size and Diversity: The performance of machine learning models heavily relies on the quality, size, and diversity of the dataset. If the dataset used for training and testing the models is limited in terms of sample size or lacks diversity in terms of variations in environmental conditions, fruit varieties, or ripening stages, it may affect the generalization and robustness of the models.
2. Overfitting: Overfitting occurs when a model becomes too specific to the training dataset, resulting in poor performance on unseen data. It is crucial to carefully tune the models' hyperparameters, perform feature selection, and use regularization techniques to mitigate the risk of overfitting and ensure the models' generalization capabilities.
3. Interpretability and Explainability: Machine learning models, especially complex ones like neural networks, are often considered black boxes, making it challenging to interpret and explain the underlying factors contributing to the ripening stage predictions. Lack of interpretability may limit the ability to gain insights into the spectral features or understand the reasons behind specific predictions.
4. Calibration and Instrument Variability: Near-Infrared Spectroscopy measurements can be influenced by variations in instrument calibration, lighting conditions, or environmental factors. Inconsistent or inaccurate calibration may introduce bias or noise in the spectral data, impacting the performance of the machine learning models. Regular calibration and quality control measures should be implemented to minimize such issues.
5. External Factors and Fruit Variability: The ripening process of fruits can be affected by various external factors such as temperature, humidity, storage conditions, and fruit maturity at the time of measurement. These factors may introduce additional variability in the Near-Infrared Spectroscopy data and influence the accuracy of ripening predictions. Controlling and accounting for these factors can be challenging and may require further research and experimentation.

Applications

1. Quality Assurance in Fruit Supply Chains: The developed machine learning models can be integrated into fruit supply chains to provide real-time assessment of papaya fruit ripening stages. This enables stakeholders such as farmers, distributors, and retailers to ensure the quality and freshness of the fruit, optimize inventory management, and make informed decisions regarding harvesting, transportation, and distribution.
2. Smart Sorting and Grading Systems: By implementing the machine learning models in automated sorting and grading systems, papaya fruits can be classified into different ripening stages in real-time. This allows for

efficient and accurate sorting of the fruits based on their ripeness, facilitating streamlined processing, packaging, and distribution processes in fruit processing facilities.
3. Precision Agriculture and Harvesting Optimization: The project's models can be used in precision agriculture applications, where near-infrared spectroscopy sensors are mounted on harvesting machinery or drones. These sensors can provide on-the-spot ripening stage predictions, guiding farmers to selectively harvest ripe papaya fruits, optimizing the harvesting process, minimizing waste, and improving overall efficiency.
4. Smart Retail and Consumer Applications: Retailers and supermarkets can utilize the developed models to offer improved consumer experiences. Real-time ripeness predictions can be integrated into mobile applications or smart shelves, enabling consumers to make informed choices based on the desired ripeness of papaya fruits. This enhances consumer satisfaction and reduces food waste by minimizing the chances of purchasing overripe or underripe fruits.
5. Quality Control in Food Processing Industries: Food processing industries that utilize papaya fruit as an ingredient can benefit from real-time ripening predictions. By implementing the machine learning models, these industries can monitor the quality and ripeness of incoming papaya fruits, ensuring consistency in production processes and delivering high-quality products to consumers.

Future Scope

1. Enhanced Model Performance: Continual research and development can focus on improving the performance of the machine learning models. This can involve exploring advanced algorithms, feature engineering techniques, and ensemble methods to further enhance the accuracy and robustness of ripening stage predictions. Additionally, incorporating domain knowledge and expertise in the feature extraction process can contribute to more informative spectral features and improved model performance.
2. Integration with IoT and Sensor Technologies: The project can be extended to incorporate Internet of Things (IoT) devices and sensor technologies. By integrating near-infrared spectroscopy sensors with IoT platforms, real-time data collection and analysis can be enabled. This would facilitate continuous monitoring of ripening stages and provide insights into the effects of environmental factors on fruit ripening. Such integration can also support automated data acquisition and remote monitoring capabilities, enhancing the scalability and applicability of the project.
3. Multi-Fruit Ripeness Prediction: Expanding the scope of the project to include multiple fruit varieties can be a promising direction for future

research. By collecting near-infrared spectroscopy datasets from various fruits, the machine learning models can be trained to predict ripening stages across different fruit types. This would contribute to a more comprehensive and versatile solution for fruit ripeness assessment, benefiting the wider agricultural industry.
4. Mobile Application Development: Developing a user-friendly mobile application can extend the practicality of the project. The application can provide a simple interface for users to capture near-infrared spectroscopy data using smartphone cameras or external sensors. The captured data can then be processed by the trained machine learning models to provide instant ripening stage predictions. This would empower farmers, distributors, and consumers to make informed decisions regarding fruit ripeness and contribute to reducing food waste.
5. Integration with Smart Farming and Precision Agriculture: Integrating the ripening prediction models with existing smart farming and precision agriculture systems can offer significant benefits. By incorporating real-time ripening stage data into automated irrigation, fertilization, and harvesting processes, farmers can optimize resource allocation and improve overall yield and crop quality. This integration would enable efficient and sustainable agricultural practices, benefiting both farmers and the environment.