המכללה האקדמית להנדסה בכרמיאל

בראודה

Capstone Project Phase A

RedGreen

Project No 21-1-D-2

Supervisors: Dr. Benny Mounits and Dr. Dan Lemberg

Students	ID	Email
Lee Tal	209517879	Lee.bergman@e.braude.ac.il
Dimitry Simon	321375388	Dmitri.simon@e.braude.ac.il

Contents Table

Abstract

Selecting a ripe, sweet watermelon is a challenge frequently encountered by consumers, often relying on anecdotal techniques, such as tapping the rind and visually inspecting color or shapes that can yield inconsistent results. This project presents a novel, machine learning- based approach to evaluating watermelon quality by integrating both visual and acoustic analyses.

Two specialized models are developed: one processes image-based features (e.g., shape, color distribution, rind patterns), while the other interprets tapping sounds recorded through a mobile device. Trained on extensive labeled datasets of watermelon images and recorded tapping sounds, the combined system delivers a reliable assessment of sweetness and overall ripeness.

By incorporating these insights into an intuitive mobile application, end users are empowered to make more confident selections, potentially reducing dissatisfaction and food waste. This work highlights how emerging technologies can transform everyday consumer decisions, reinforcing the growing trend of integrating machine learning into common shopping experiences.

1. Introduction

Watermelons are a popular summertime treat, yet identifying an ideal specimen in terms of ripeness and sweetness can be surprisingly difficult. Many consumers rely on informal methods, such as gently tapping the rind to evaluate its resonance or judging by exterior coloration, often with inconsistent results. While these traditional strategies sometimes succeed, they do not guarantee a perfectly sweet and juicy watermelon every time.

Recent advances in artificial intelligence offer a unique opportunity to address this everyday problem by combining cutting-edge image recognition and audio analysis techniques. By capturing both photographs and tapping sounds via a mobile device, specialized machine learning models can extract critical indicators of quality- such as shape, coloration patterns, and acoustic signatures, thereby providing more consistent assessments.
Through a dedicated mobile application, shoppers gain instant access to an objective evaluation of a watermelon's sweetness and ripeness. This approach not only saves time and money but also reduces waste by helping consumers avoid purchasing watermelons that are unlikely to meet their expectations.

Ultimately, this project serves as a compelling example of how AI can seamlessly integrate into daily routines, enhancing decision-making processes in the modern marketplace.

2. Related Work and Existing Solutions

In this secti-on, we will review some existing applications of finding watermelons and other fruits. By analyzing these applications, we can identify successful strategies and learn valuable lessons to enhance the development and effectiveness of our own solution.

Existing Applications for Watermelons

- **Melony**
  - Functionality: Uses algorithms to analyze sound frequency when tapping a watermelon to determine ripeness.
  - Platform: Android and iOS
  - Rating: 2.4 out of 5 stars based on 9 ratings
  - Downloads: Over 10,000 downloads.

- **Watermelon Prober**
  - Functionality: Analyzes sound from tapping the fruit to assess ripeness.
  - Platform: Android
  - Rating: 2.8 out of 5 stars based on 402 reviews
  - Downloads: Over 100,000 downloads.

- **Melon Aid**
  - Functionality: Uses AI to analyze vibration responses and sound frequency to detect ripeness.
  - Platform: iOS
  - Rating: 2.2 out of 5 stars based on 5 ratings
  - Downloads: numbers are not provided on the App Store.

Existing Application for Determining General Fruit Sweetness

- **Clarifruit**
  - Functionality: Utilizes AI-powered vision to assess fruit quality, including ripeness and sweetness, from photos.
  - Platform: Android & iOS
  - Rating: 3.0 out of 5 stars based on 3 ratings on the App Store
  - Downloads: Over 5,000 downloads on Google Play

Conclusions and Insights for Building Our App

Building an application to reliably detect the ripeness and sweetness of watermelons requires a meticulous approach that combines technical expertise with user-focused design. Below are the key conclusions and insights drawn from our research and analysis:

- **Comprehensive Analysis with Sound and Visual Features**
  Integrating both visual and acoustic data provides a robust evaluation of watermelon ripeness. While competitors like Melony and Watermelon Prober rely solely on sound analysis, their low ratings reveal room for improvement. By using advanced models such as YOLOv8 for visual detection and ECAPA-TDNN for sound analysis and combining their predictions through a fusion layer, our app can deliver a more accurate and holistic ripeness score. This dual-modal approach enhances reliability and user satisfaction.

  Recent research supports the effectiveness of this combined methodology. The study "Fruit Ripeness Identification Using YOLOv8 Model" by Xiao et al. demonstrated the high accuracy of YOLOv8 in detecting fruit ripeness based on visual characteristics, achieving an impressive 99.5% accuracy.

  By leveraging these two models, our system ensures a multi-modal ripeness evaluation, integrating both visual and acoustic data to provide a comprehensive and reliable assessment of watermelon quality. These findings reinforce our decision to integrate both image and sound-based models to enhance prediction accuracy and usability.

- **Importance of High-Quality Data**
  The effectiveness of our machine learning models depends heavily on diverse and well-annotated datasets. Collecting images and sound recordings under various conditions ensures robust training and minimizes inaccuracies caused by environmental variability. In addition, leveraging techniques such as Mel spectrogram transformation can improve feature extraction from audio signals, as demonstrated in Liu et al.'s research.

- **Leveraging Cloud-Based Solutions**
  To overcome the limitations of mobile devices, we plan to offload computationally intensive tasks to *Google Cloud*. This approach allows for real-time processing of large datasets while maintaining a responsive user experience. The research conducted by *Xiao et al***.** highlighted the importance of efficient model deployment for real-world applications.

- **Optimizing User Experience**
  A seamless, intuitive user interface is critical for adoption. The app should provide clear instructions, rapid feedback, and an engaging design to encourage repeated use. Furthermore, addressing potential challenges such as background noise and lighting conditions will enhance user satisfaction. The ECAPA-TDNN model presented by *Liu et al***.** demonstrated how preprocessing techniques such as noise reduction and feature extraction can improve model robustness in real-world environments.

- **Enhancing Accuracy Through Model Fine-Tuning**
  Fine-tuning pre-trained models such as YOLOv8 and ECAPA-TDNN with domain-specific data is essential for achieving high accuracy. If necessary, custom models may be developed to address specific challenges like environmental noise and complex visual patterns. The success of the ECAPA-TDNN approach suggests that fine-tuning deep learning architectures can significantly enhance classification performance.

- **Collaboration Opportunities**
  Partnering with retailers or agricultural organizations could facilitate data collection and promote the app. These collaborations may also provide marketing opportunities and help establish the app's credibility in the marketplace.

- **Continuous Improvement Through Feedback**
  User feedback will be invaluable for refining both the app's functionality and user interface. Iterative updates, informed by user insights, will ensure the app evolves to meet consumer needs effectively.

By addressing these insights, our application aims to revolutionize the way consumers select watermelons, offering a reliable, convenient, and enjoyable solution that bridges technology and everyday life.

2.1. Detection Approaches

Existing Detection Modals for Sound and Pictures

The application integrates both visual and auditory data to assess watermelon sweetness. This dual-modal approach ensures high accuracy by leveraging complementary information from images and sounds. Below, we detail the models and techniques employed for pictures and sound.

Object Detection methods

- **YOLO**

  You Only Look Once is a widely used deep-learning model for real-time object detection. It processes an entire image in a single pass, predicting bounding boxes and class probabilities simultaneously. This approach makes it highly efficient for detecting visual ripeness indicators such as color patterns, surface textures, and shape variations. The study "Fruit Ripeness Identification Using YOLOv8 Model" by Xiao et al. demonstrated that YOLOv8 achieves a 99.5% classification accuracy in fruit ripeness detection, making it an optimal choice for our image analysis model.

- **RCNN**

  Region-Based Convolutional Neural Networks first picks out "regions" in the image that might have objects (using selective search) and then classifies them one by one. It’s accurate but slow because it processes each region separately. Think of it as taking a magnifying glass to every part of the image to make sure you don’t miss anything. It works well for detailed detection but isn’t the best for real-time needs.

- **Faster RCNN**
  Improvement upon traditional RCNN by utilizing a Region Proposal Network (RPN) to reduce computational overhead. While it provides high accuracy, it is computationally expensive compared to YOLOv8. Due to the need for real-time processing in mobile applications, YOLOv8 was chosen as the primary model for image analysis.

- **SSD**
  Single Shot MultiBox Detector also does detection in one go, like YOLO, but uses multiple layers to detect objects at different scales. This makes it better at handling objects of various sizes compared to YOLO. It balances speed and accuracy pretty well, making it suitable for many real-world applications. However, for tiny objects, it might need a little extra help.

Sound Detection methods

- **ECAPA-TDNN**
  Emphasized Channel Attention, Propagation, and Aggregation Time Delay Neural Network is an advanced deep-learning architecture optimized for audio classification. Unlike traditional sound models, it utilizes Mel Spectrogram feature extraction, which enhances classification performance by retaining critical frequency domain information while reducing complexity. The research *"Non-destructive Ripeness Judgement of Watermelon Based on Mel Spectrogram and ECAPA-TDNN"* by *Liu et al*. demonstrated that this method achieved 89.5% accuracy in classifying watermelon ripeness based on tapping sounds. This model is particularly effective in real-world settings, where background noise and environmental variability pose challenges for accurate classification.

- **YAMNet**
  A lightweight, pre-trained audio classification model from Google, built on the MobileNetV1 architecture. It can recognize over 500 audio event classes and is efficient for mobile applications. By fine-tuning it with a custom dataset, it can be adapted for analyzing watermelon tapping sounds. YAMNet is available on TensorFlow Hub, making it easy to implement.

- **VGGish**
  A pre-trained audio model based on the VGG architecture, designed to extract strong embeddings from sound data. It serves as a robust baseline for audio classification and can be fine-tuned for tasks like analyzing watermelon tapping sounds. Available in TensorFlow and PyTorch, it is ideal for feature extraction and classifier training.
- **OpenL3**
  A deep audio feature extractor that creates embeddings from raw audio or mel-spectrograms. It supports both audio and visual inputs, leveraging large-scale pre-trained datasets for quick adaptation. This model is suitable for generating embeddings of tapping sounds for watermelon ripeness classification and is available via the OpenL3 library in Python.

Our choices

For image analysis, we selected YOLOv8 due to its exceptional ability to detect and classify visual ripeness indicators with high accuracy and real-time performance. The anchor-free architecture and C2f module of YOLOv8 make it highly efficient in identifying subtle surface patterns and texture variations, as supported by research from *Xiao et al*.

For sound analysis, we opted for ECAPA-TDNN, as it outperforms traditional machine learning models in non-destructive ripeness detection. The combination of Mel Spectrogram feature extraction and deep learning attention mechanisms significantly improves classification accuracy, as shown in the research by *Liu et al*.

By leveraging these two models, our system ensures a multi-modal ripeness evaluation, integrating both visual and acoustic data to provide a comprehensive and reliable assessment of watermelon quality.

3. Expected Achievements

- **Empowering Users to Make Informed Decisions**
  Equip users with a reliable tool to assess watermelon quality, enabling better purchasing decisions. By providing accurate results, we aim to build user trust and reduce uncertainty in selecting high-quality fruit.

- **Reducing Waste and Promoting Sustainability**
  By helping consumers identify sweet watermelons, the app can minimize food waste caused by unripe or spoiled fruit. Additionally, we aim to promote sustainable practices by encouraging mindful consumption through this technology.

- **High Accuracy in Sweetness Detection**
  Achieve precise predictions for watermelon sweetness through advanced models trained on custom datasets. By combining sound-based analysis with visual cues, we strive for a classification accuracy of at least 80%, ensuring reliability for users.

- **Educational Value**
  Gain valuable knowledge from the development process of sound and visual analysis technologies. This experience will not only educate our team but also empower us to create similar innovative applications for other fruits or industries in the future.

4. Research Process

The research process involves a systematic approach to integrating visual and auditory data for assessing watermelon ripeness. This comprehensive methodology ensures the development of a robust and accurate application.

Gathering Information and Reviewing Research

- **Objective:** Study current models and technologies for backend integration using cloud-based AI solutions.
- **Focus:**
  - Analyze YOLOv8 for image-based object detection.
  - Explore ECAPA-TDNN for sound-based classification.
  - Research Mel Spectrogram feature extraction as a preprocessing technique for acoustic analysis.
- **Outcome:** Define application requirements and backend architecture using insights from state-of-the-art methodologies, specifically those demonstrated in *"Fruit Ripeness Identification Using YOLOv8 Model"* by *Xiao et al***.** and *"Non-destructive Ripeness Judgement of Watermelon Based on Mel Spectrogram and ECAPA-TDNN"* by *Liu et al***.**

Designing System Functionality

- **Core Features:** Implement a cloud-based architecture to process image and sound data.
- **User Flow:** Ensure seamless integration of image capture, sound recording, and real-time feedback.
- **Functionality:** Establish an efficient REST API for communication between the mobile app and the cloud-hosted AI models.

Planning Development Process

- **Milestones:**
  - Data collection and preprocessing.
  - Model selection and cloud deployment.
  - Cloud deployment and optimization.
  - UI design and app development.
  - Backend integration and testing.
- **Timeline:** Establish clear deadlines to synchronize model development with app implementation.

Data Collection and Preprocessing

- **Image Data:** Capture a diverse dataset of watermelon images to train **YOLOv8**, ensuring accurate identification of visual ripeness indicators.
- **Audio Data:** Collect tapping sounds from watermelons at different ripeness levels.
- **Preprocessing:**
  - **For images:** Label and standardize images for YOLOv8 training.
  - **For sound:** Extract Mel Spectrogram features from tapping sounds to train ECAPA-TDNN, following the methodology described in *Liu et al.*

Models For Detection and Cloud Deployment

- **YOLOv8**
  - Train on labeled watermelon images to detect visual ripeness indicators such as surface texture, color patterns, and shape.
  - Deploy the trained model to *Google Cloud*, ensuring low latency and scalability.
- **ECAPA-TDNN**
  - Train on Mel Spectrogram-transformed watermelon tapping sounds to classify ripeness levels.
  - Deploy the optimized model to the cloud to handle real-time acoustic analysis.

Developing the Android Interface

- **UI Design**
    - Provide an intuitive interface for capturing images and recording audio.
    - Ensure smooth navigation and clear feedback on ripeness assessment.
- **Tooling:** Use Android Studio with Kotlin for app development.
- **Backend Communication:** Implement API calls to interact with the cloud models and retrieve predictions.

Integrating Object Detection and Sound Analysis Models

- **YOLOv8 Integration**
    - Enable real-time detection of surface features using cloud-hosted inference.
    - Validate and refine API responses for accuracy.
- **ECAPA-TDNN Integration**
    - Classify tapping sounds using cloud-based acoustic analysis.
    - Implement noise reduction techniques to enhance real-world usability, as demonstrated in *Liu et al.*

Combining Results from Both Models

- **Decision Fusion**
    - Develop a backend algorithm to merge YOLOv8 and ECAPA-TDNN predictions into a single ripeness score.
    - Assign appropriate weights to visual and acoustic indicators based on performance metrics.

Testing and Optimization

- **Device Compatibility:** Conduct tests on various Android devices to ensure smooth operation.
- **Performance Metrics:** Evaluate cloud processing speed, model accuracy, and response time.
- **Refinements:** Optimize API efficiency and model inference speed for real-time usage.

Deployment

- **Preparation:** Finalize app functionality and backend deployment.

- **User Feedback:** Conduct trials with users to refine app performance and improve usability.

Tools and Frameworks

- Android Studio and Kotlin for mobile app development.
- *Google Cloud* Platform for scalable AI model hosting.
- REST API for communication between app and cloud models.
- YOLOv8 and ECAPA-TDNN for ripeness classification based on images and sound.

4.1. Related Papers

Fruit Ripeness Identification

As part of our research to develop a robust system for analyzing watermelon quality, we carefully studied an insightful article titled "*Fruit Ripeness Identification Using YOLOv8 Model*" by *Bingjie Xiao*, *Minh Nguyen*, and *Wei Qi Yan*, published in Multimedia Tools and Applications in August 2023:

- This paper provides a comprehensive exploration of how deep learning-based visual object detection, specifically the YOLOv8 model, can classify fruits as ripe or overripe by analyzing their visual features. The authors detail the advanced capabilities of YOLOv8, including its anchor-free architecture and the integration of the C2f module, which enables the extraction of fine-grained surface patterns and characteristics. Their model achieved an impressive accuracy of 99.5%, demonstrating its potential for high-precision classification tasks in agricultural and retail settings.
- This research directly aligns with our project's objectives, particularly in evaluating visual characteristics such as spots, lines, and surface patterns to determine watermelon quality. It reinforced our decision to utilize YOLOv8 as a leading candidate for image analysis, emphasizing its ability to detect subtle and complex details. By leveraging insights from this study, we aim to fine-tune YOLOv8 for our specific dataset, optimizing it to recognize watermelon-specific features and validating its performance in a mobile application context.

Non-destructive Ripeness Judgement of Watermelon Based on Mel Spectrogram and ECAPA-DTNN

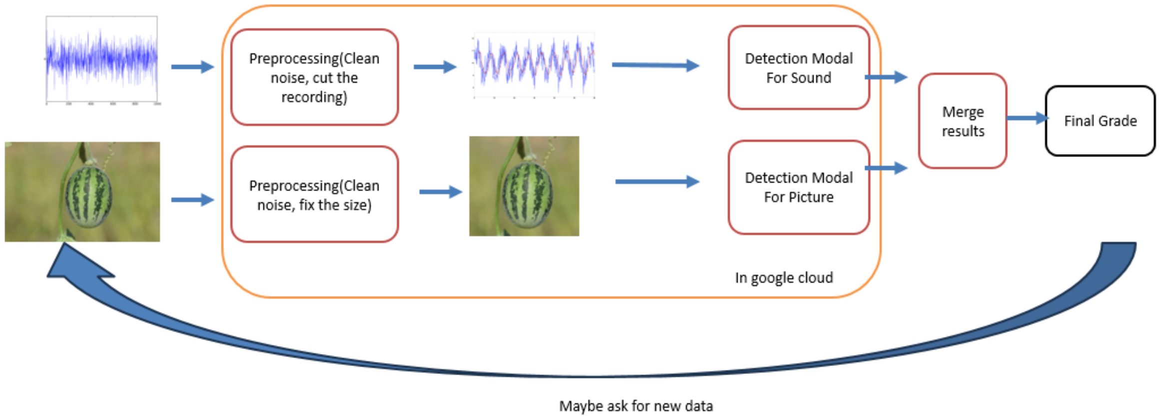
This research presents a way for people to check if a watermelon is ripe using their smartphones. The authors recorded the sound made when tapping watermelons and converted the sound waves into pictures called Mel spectrograms. They trained a special deep learning model (ECAPA-TDNN) to recognize patterns in these sounds. The model achieved 89.5% accuracy, showing it can reliably judge ripeness. This solution is easy to use and could be turned into a mobile app. Future work includes testing more watermelon types and combining audio with photos.
This research is highly relevant to our app idea because it provides a complete approach to predicting watermelon ripeness from tapping sounds using smartphone recordings. It outlines:

1. Data Collection: Using smartphones for recording, which fits our app concept.
2. Feature Extraction: Converting tapping sounds into Mel spectrograms, which are effective for analyzing sound patterns.
3. Deep Learning Model: Using ECAPA-TDNN, a powerful model for audio classification.
4. High Accuracy: Achieved 89.5%, showing real-world potential.

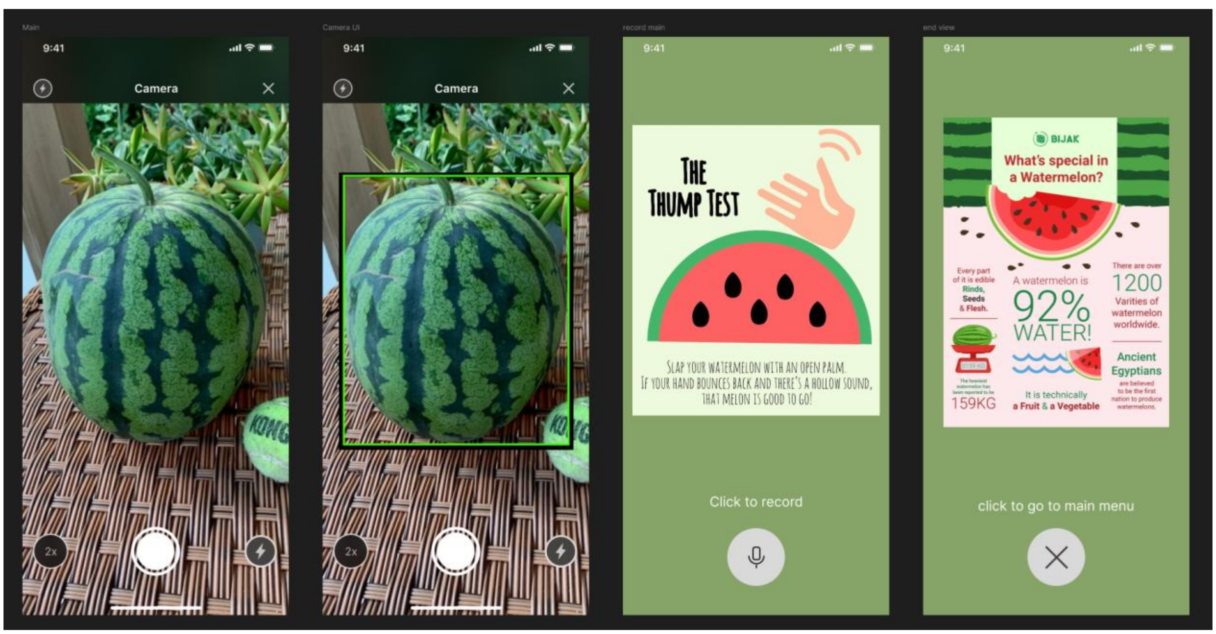
We can adapt their methods to predict sweetness by adding more audio features or integrating additional models.

5. App Architecture concepts

back-end:



front-end:



6. Challenges:

Limited Smartphone Capabilities

Mobile devices often have limited processing power, which may result in slower processing speeds or reduced analysis quality.

- Mitigation Strategies:
  - Offload processing to *Google Cloud* for advanced model inference.
  - Evaluate and fine-tune various versions of YOLO to identify the optimal model for visual pattern detection.
  - Use preprocessed and annotated datasets to train models, reducing misclassification risks.

Complexity of Detection:

Watermelons possess intricate visual patterns, such as lines and stripes, which can confuse the detection model by:

  - Misinterpreting lines as separate objects.
  - Overlooking key features essential for ripeness evaluation.

- Mitigation Strategies:
  - Implement advanced image processing techniques to enhance feature extraction.
  - Train models on diverse datasets to improve robustness against pattern variability.

Noise in Recordings

Background noise can distort the clarity of watermelon tapping sounds, affecting the reliability of the sound analysis.

- Mitigation Strategies:
  - Integrate noise-reduction techniques, such as spectral noise gating, during preprocessing.
  - Train the ECAPA-TDNN model using diverse datasets that simulate real-world environments with various noise levels.
  - Use filtering algorithms in *Google Cloud* to preprocess audio before analysis.

Training Data Quality

The accuracy of both visual and auditory models depends significantly on the diversity and comprehensiveness of training datasets.

- Mitigation Strategies:
  - Collect a wide variety of watermelon images under different conditions to ensure model robustness.
  - Record tapping sounds in various environments, capturing variability in acoustics and watermelon characteristics.
  - Regularly update datasets with new samples to improve model performance.

Environmental Lighting

Users will operate the app in varied lighting conditions, such as indoors, outdoors, or under bright sunlight, which can affect image analysis.

- Mitigation Strategies:
  - Use data augmentation techniques to train models with images captured in different lighting conditions (e.g., bright sunlight, dim light).
  - Incorporate dynamic brightness and contrast adjustment in the app to preprocess images before sending them to the cloud.
  - Leverage YOLO's robustness to lighting variations by fine-tuning the model on diverse datasets.

7. Verification

System Requirements

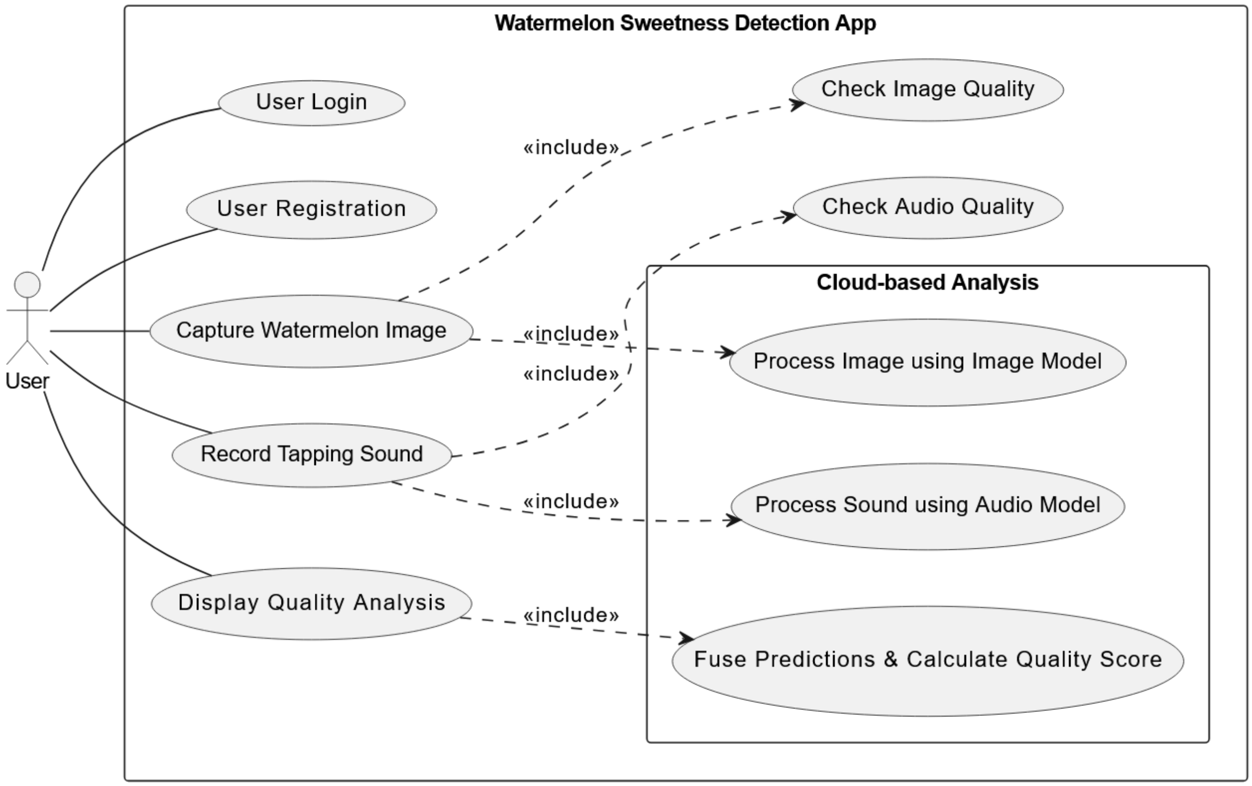
ID	Requirement
1	The app should be user-friendly
2	The app should correctly classify watermelon quality at least 90% of the time
3	The app should work fast without significant delays
4	The app should ask from the user to tap again if it was recorded in loud environment
5	The app should provide clear and easy-to-understand feedback

To verify the aforementioned requirements, we propose the following verification methods:

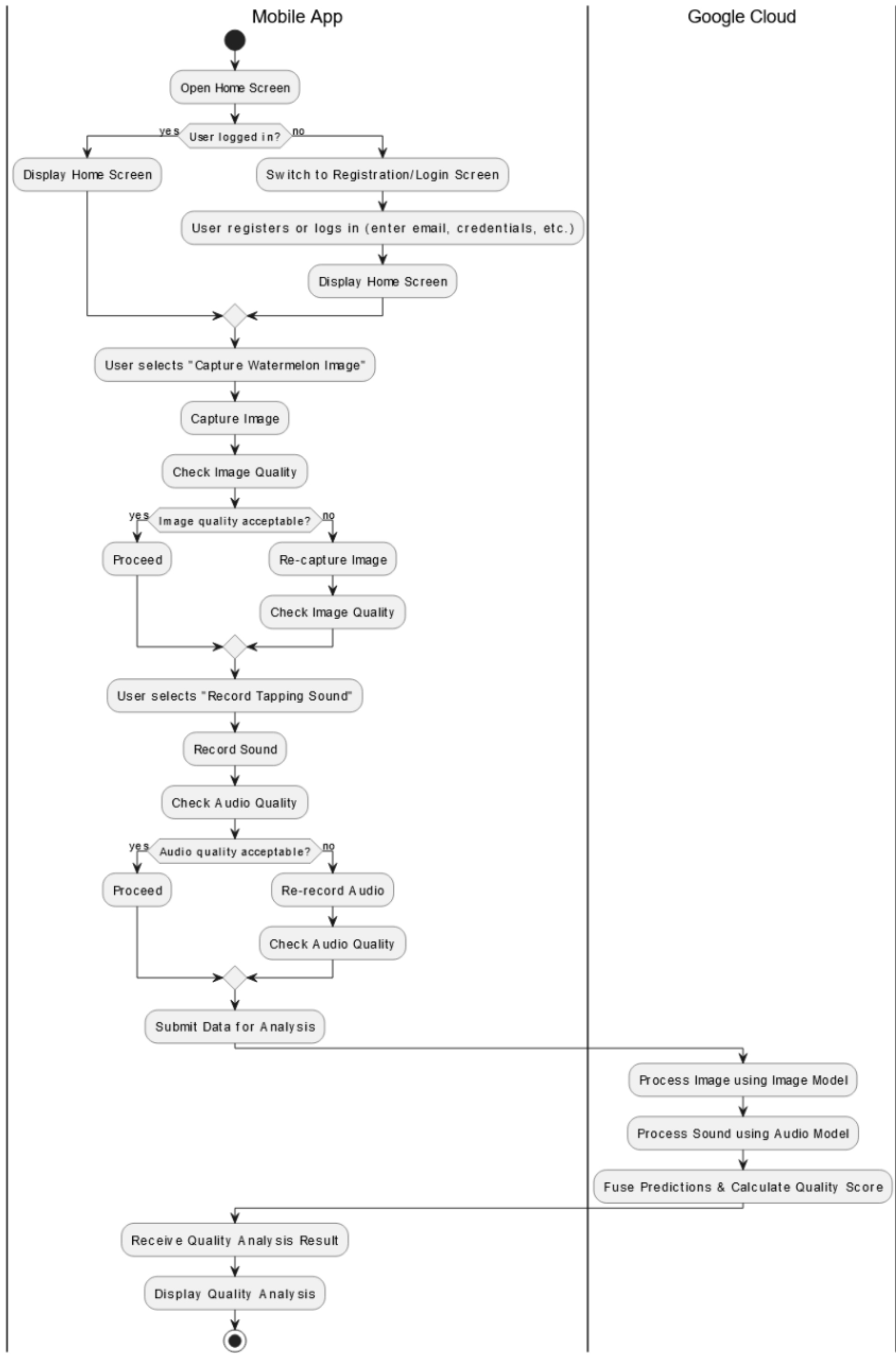
1. **User Experience Evaluation**: Gather feedback from a representative sample of users regarding their experience with the app. This will focus on aspects such as ease of use, navigation, and clarity of the user interface.
2. **Classification Accuracy Testing**: During the testing phase, conduct classification tests on multiple watermelons and calculate the average accuracy rate. This will enable us to assess the app's performance and identify areas for improvement to achieve the desired accuracy level.
3. **Performance Stress Testing**: Simulate high user demand or network congestion scenarios to evaluate whether the app can maintain optimal performance and responsiveness under load.
4. **Environmental Noise Testing**: Evaluate the app’s ability to detect noisy environments by testing it under varying sound conditions (e.g., quiet, moderately loud, and very loud). This will ensure that the app prompts users to retake recordings when necessary due to excessive background noise.
5. **Feedback Clarity Evaluation**: Conduct user interviews or surveys to assess whether the feedback provided by the app is clear, understandable, and meaningful to users. Additionally, evaluate whether the app effectively explains the classification results in a manner that users can comprehend.

8. Diagrams

Use Case



Activity Diagram



9. Summary

This project outlines a comprehensive plan to develop a mobile application designed to evaluate watermelon quality by analyzing both visual and acoustic features. The app will classify watermelons into categories such as "Not Sweet," "Sweet," and "Very Sweet" based on machine learning-based assessments.

The development process involved extensive research into fruit ripeness detection, leading to the selection of YOLOv8 for image analysis and ECAPA-TDNN for sound analysis. YOLOv8 is used to detect visual ripeness indicators like surface texture and color distribution, while ECAPA-TDNN analyzes tapping sounds recorded through a smartphone to assess ripeness and sweetness. The ECAPA-TDNN model has been chosen for its high accuracy in distinguishing sound patterns and its robustness against environmental noise.

The application will send the collected data to cloud-hosted models on Google Cloud Platform via a REST API, ensuring fast and efficient processing while minimizing the computational burden on mobile devices. Android Studio and Kotlin will be used for app development, with Android Jetpack Components ensuring a structured and efficient architecture.

The verification and evaluation phase will assess usability, performance, and accuracy. Testing scenarios will include both controlled and real-world conditions to ensure the app delivers reliable results. This project leverages cutting-edge AI technologies while maintaining a user-friendly approach, aiming to provide an accessible, practical, and accurate tool for selecting high-quality watermelons.

10. References

- Xiao, B., Nguyen, M., & Yan, W. Q. (2024). Fruit ripeness identification using YOLOv8 model. *Multimedia Tools and Applications*, 83, 28039–28056. https://doi.org/10.1007/s11042-023-16570-9
- Liu, J., Shi, H., Xia, Y., & Li, J. (2024, July). Non-destructive Ripeness Judgement of Watermelon Based on Mel Spectrogram and ECAPA-DTNN. In *2024 20th International Conference on Natural Computation, Fuzzy Systems and Knowledge Discovery (ICNC-FSKD)* (pp. 1-8). IEEE. https://doi.org/10.1109/icnc-fskd64080.2024.10702215
- *Google Cloud* Platform. (n.d.). *Cloud-based AI solutions for mobile applications*. Retrieved from https://cloud.google.com.
- TensorFlow Hub. (n.d.). *YAMNet: A pre-trained model for sound classification*. Retrieved from https://www.tensorflow.org/hub.
- *RedGreen Project Repository*. (n.d.). GitHub. Retrieved from https://github.com/PyExtr/RedGreen
- *Zhang, L., Chen, R., Hao, H., He, E., Ning, M., Tang, J., & Fan, A.* (2024). *Watermelon appearance and knock correlate data sets with sugar content*. IEEE DataPort. https://ieee-dataport.org/documents/watermelon-appearance-and-knock-correlate-data-sets-sugar-content