

Computer Science Project

Graph-Based Design

Bug Busters Group

Problem Statement : Supermarket Product Manager: Optimizing Inventory with Graph-Based Visualization

Food waste is a major problems in supermarkets ,where perishable goods often expire without being sold. Managing stock efficiently while ensuring that customers receive fresh products is a hard task. Traditional inventory systems rely on simple expiration tracking , but they often fail at reducing the waste as well as rotating the stock regularly.



Solution

Develop a graph based image analysis system that can detect expired goods and relay how goods need to be placed in the supermarket.



Project Justification

It simplifies the large amount of products that need to be checked to only those that need to be changed , like perishable goods. It also reduces waste in supermarkets because graphs model relationships and dependencies between food items. It also has an extra feature for identifying waste products in the store

Technical Algorithms to use

- KNN Graph
- Graph Clustering

Overview

This is a system using JavaFX application for managing inventory using a graph-based approach. This system combines data structures, image processing, and visualization for practical retail use while also using hidden calculations as a way to interact with the products to form groups or clusters.

This system is designed to collect data of the products in the supermarket and use this data to perform tasks which would be a bit cumbersome for people to do. The aim is to reduce waste product in South African supermarkets and this system allows for an easier view of all products in the store ,thus highlighting the necessary actions needed to be taken to reach our goal.



This system is the answer as it is easy to use, simple and efficient in its execution to minimize product waste in supermarkets using a graph based system. The system has a lot of functions such as to :

- Add products
- Delete expired products once they are removed
- View how fresh products are
- Visualize how products are related(using a graph)

Data Structure

A custom data Structure is used in this system to oversee all operations related to the products.

A custom `TreeGraph.java` data structure ,which operates like an Adjecency List data structure is used, and the strategic use of HashMap and ArrayList demonstrate a strong understanding of data structure design. It is well-suited for the project's needs, supporting product similarities, dynamic updates ,visualization and similarity detection.

A `HashMap` is used for efficient node positioning, clustering, and shelf grouping.

An `ArrayList` is also used for product collections and edge tracking, appropriate for dynamic lists.

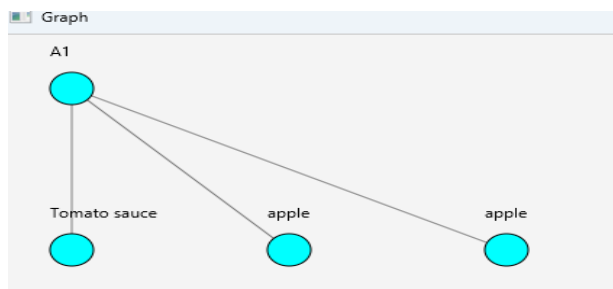
Structures are correctly implemented, with robust handling of lookups and iterations

A custom `Point` class is also used to keep track of coordinates for graph visualisation.

Graph Visualization

A custom class `GraphVisualizer.java` is used to handle the visualisation of our graph-based system. The store shelves are used as dummy nodes at the top, products below, with similar products stacked vertically. Edges show similarities between products.

This visualization makes it easy for retailers to see which products are similar, like grouping milk brands together. It makes it easier to see how products should be placed in the store based on similarity with each hierarchy representing each shelf thus not missing a single product given.



Core Functionalities

These are the main features of the project in helping manage and if possible, organize the stock on the shelves of South African supermarkets.

Add Product

The main function to add products the products inside the store

Delete Expired Products

Responsible for removing expired products which are removed from the store.

Find Similar Products

Finds all products (using KNN) similar to product that is added.

Freshness Tracking

Notifies manager about which products need attention in store.

Save/Load Data

Allows for the simple capturing of data instead of add all the products again.

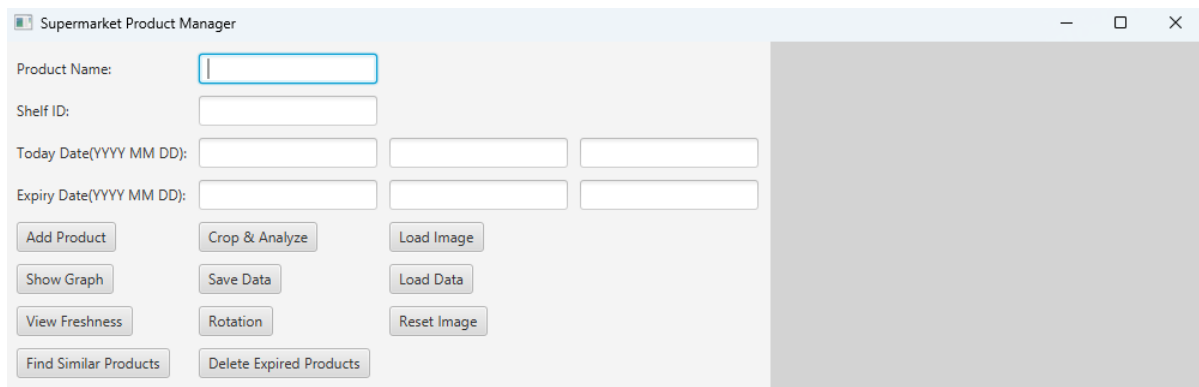
Technical Implementation

The app uses a graph to connect similar products, like a map of relationships. Project is divided into 4 packages each handling different aspects needed to achieve the system's goal.

Ensuring that the system captures details fully these are the `Product.java` requirements :

```
public class Product {  
    private String name; //Name of product being stored  
    private Features feats; //feature of each image of product ,e.g shape etc..  
    private Shelf shelf; //shelf ,product is found in  
    private Image img; //image of product  
    private Date date; //how many days before a product expires  
    private FreshnessLvl fresh; //how fresh products are  
    private String imagePath;  
}
```

The system also uses JavaFX to create its UI :

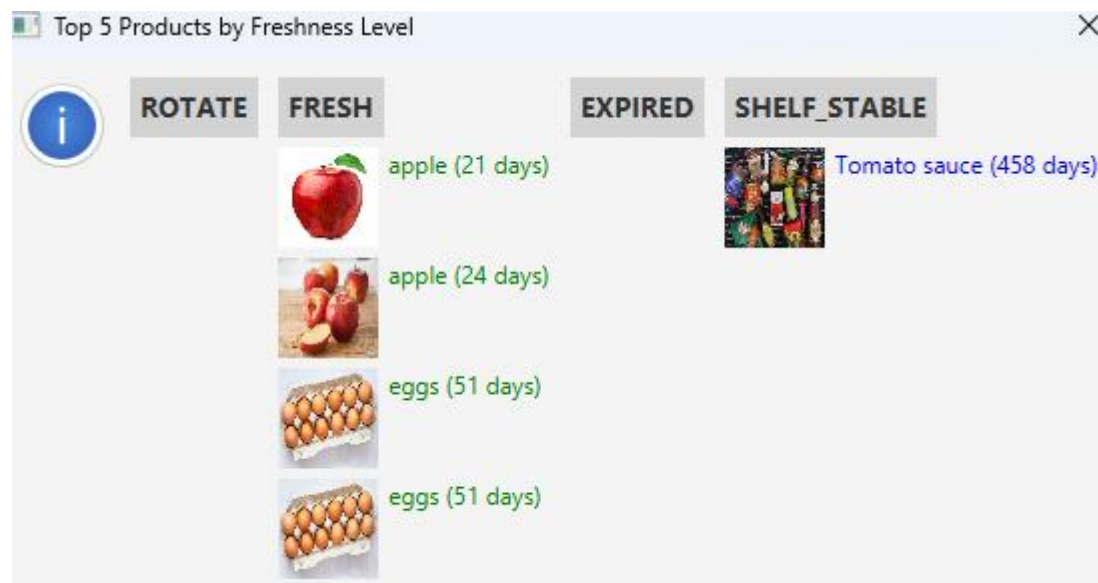


Thus ensuring an easy to use UI system for users. All of the buttons are checked for any errors or user errors making it more effective and robust.

It also simplifies the capturing of an image as shown below , to make it easier to view the part of the image which is relevant. This is to make it easier for the system to take in any shelf images given and to only focus on the product highlighted in red.



There are many more actions the user can use to interact with the system to improve help improve store needs.



Member Tasks

Member 1 (AO Cushe) 223220776

Suggests similar food items based on freshness and expiration date using K-NN graph.

Member 2 (P Mabeso)223039521

Classifies food items based on freshness levels using clustering techniques to group them.

Member 3 (MY Ntombela)222222816

Building user-friendly GUI for supermarket employees and managers to visualize food stock levels and make data informed decisions.

Member 4 (TD Selane)222045938

Tracks and updates food item freshness and expiry using its corresponding graph representation and date metadata.

How the Project should work

Steps	How the project should work
1	User uploads shelf image/s or they load a previous graph
2	GUI lets user crop/select regions to be shown
3	For each region: system turns the cropped region into a graph
4	System compares to existing product graphs
5	Suggests likely match (classification)
6	User is prompted to input product expiry and today date , as well as name and shelf Id for product.
7	The selected product image is converted into a graph (Nodes are the products ,edges are the relationships of products).
8	The system calculates how many days are left before the product expires and based on this, it assigns a freshness label(fresh,expired,expires soon(in 2 days) ,etc)
9	The system compares the current product graph with all previously stored product graphs using KNN to find visually similar products.
10	System groups products into clusters based on similarity and freshness score.
11	Display in GUI : cropped product image,graph of the image(these 2 can be left out to uncomplicate the task), Expiry status, Table of all products in inventory,as well as recommendations for which products to rotate
12	Repeat for each product on shelf/s

Conclusion

This is a graph-based system used to solve a crisis of Product or Food wastage in South Africa using images to recognize and determine which products need immediate attention from the store. It simplifies everyday activity inside a supermarket and makes it easier for the people to focus on the right products while also visualising similarities between products for other uses they might find it useful for (such as organization).

It has been a pleasure for us Bug Buster to finally present our work and hope it achieves its objective.