Hands-Free Food Ordering: A Gesture-Based Interface for Restaurants

Hoang Long Tran
Deakin University
221 Burwood Hwy, Burwood VIC 3125, Australia
Email: s223128143@deakin.edu.au
ORCID: 0009-0000-9372-6246

***Abstract*—The motivation behind this paper is to propose a gesture-based food ordering system to enhance the dinning experience for those with hearing impairments. The system uses computer vision to recognize gestures for users to navigate and interact with the menu. The machine learning model can be integrated into user friendly front-end design to facilitate food ordering. The main application of this system is for drive-though and restaurant settings, but it could be extended to other ordering system if needed. This solution does not only empower individual with hearing disabilities, but also offer a hygienic approach which is one of the major concerns in post COVID-19.**

***Index Terms*—Computer Vision, Food Ordering, Touchless Interfaces, Accessibility, Machine Learning**

I. Introduction

Traditional food ordering services rely heavily on verbal communication. This makes ordering difficult for those who cannot hear or speak, which often can lead to misunderstandings and delays in service. In recent years, we have seen a glimpse on the potential of how technology can influence our daily life. By leveraging those new technologies in Computer Vision, this paper introduces a simple design of using machine learning model to predict hand recognitions to address the communication issue. Besides addressing the communication problem, a touchless food ordering system offers hygienic practice in public places, which is critical after the COVID pandemic.

The solution proposed is a simple integration of machine learning model and user interface. The model has 3 labels where the user can cycle through menu option with the "5" gesture. The "yes" and "no" gestures are for confirming and canceling the selections. The system can be implemented in traditional restaurant environments and drive-though settings.

In this paper, we will first do a literature review on similar papers. Then discuss the system architecture underlying the gesture recognition model and front-end design. Finally, we will present our findings, discuss the implications for and further improvements in our systems.

II. Literature Review

A. Hand Gesture Based Food Ordering System

[1] This article proposes a design on hand gesture-based ordering system for businesses like restaurants. The design is a machine learning model that can solve problems like preventing the spread of germs, tackling language barriers and communication difficulties with just a simple wave, swipe, and other gestures to order food. They divided the machine learning problem into data collection, data processing, and model training.

In the data collection, the data was collected from many open repositories. The data cleaning process involves using MediaPipe and OpenCV python library to crop the images. For the model building, they suggested using a hybrid model via CNN and Decision Tree. CNN is implemented first to learn spatial hierarchies of features from the images. This CNN is different from regular ones because instead of moving the output layer, that layer will be used as input for the Decision Tree.

After finished building the model, the research proposed some suggestions on integrating it into the system.

- **Interface Layout**: A real-time video feed window, an interactive button to confirm the order, and a display pane to show the outcome of the processed gesture.
- **System Implementation**: Gesture capture via a video capture module, processing each frame, and feeding them to the hybrid model for prediction.
- **Feedback Display**: Displays the food prediction feedback in real-time and allows users to correct any mistakes in the order. Users can also rate the predictions. [1]

B. Restaurant Menu with Gesture Recognition

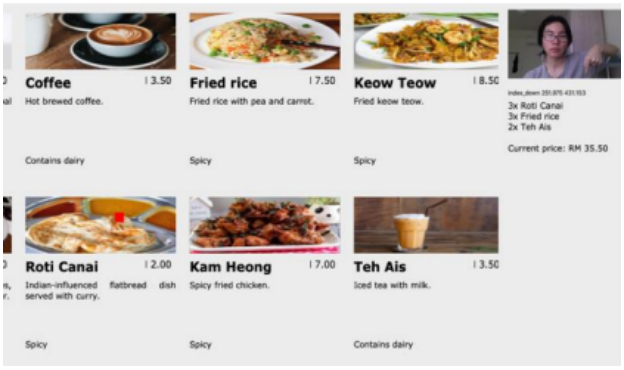
[2] The paper proposes a digital restaurant menu system using hand recognition. They divided the research into a few sections. The first step is data collecting, which uses "pose elimination" from MediaPipe to crop hands from collected images.

The next step is developing a system development life cycle (SDLC). A process that design, build and deliver to users for them to understand the business's needs. There are 4 main phases in the SDLC, which are planning (plan to design the system), analysis (examines the current system), designing (design the hardware, software, network infrastructure, user interfaces and databases), and finally is the application where the system is built.

The following step is the system design, where they have designed a diagram for 2 distinct operations.

The "Server" is for employees to see and act once an order is received. The "Gesture Recognition" contains classes for the back-end to output the predicted gesture.

The final part is the system implementation where they demonstrate how the system might look. The following picture shows the front-end design of the app.



Most of the screen is taken by the menu except the top right. There is a camera to view as feedback, below it is the detected gestures and a list of items currently in the cart. They also propose specific back-end design and evaluation methods that is suitable for their system [2].

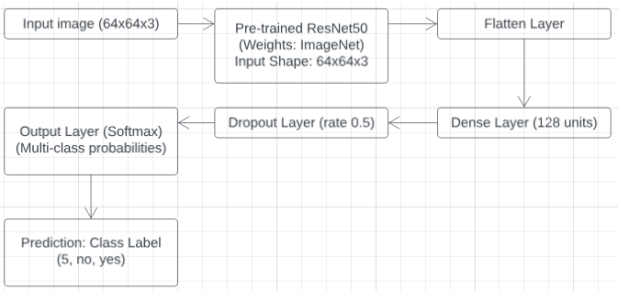
III. System Architecture/Methodology

A. Data collection

The data was taken from a variety of open repositories. After having enough data, "MediaPipe" and "OpenCV" was used to crop out hand images.

B. Model Architecture

The classification model is the core of this system. We need to build a model that can generalize well for real-time data. This paper proposes using the pre-trained "ResNet50" model. The picture below shows the model architecture diagram.



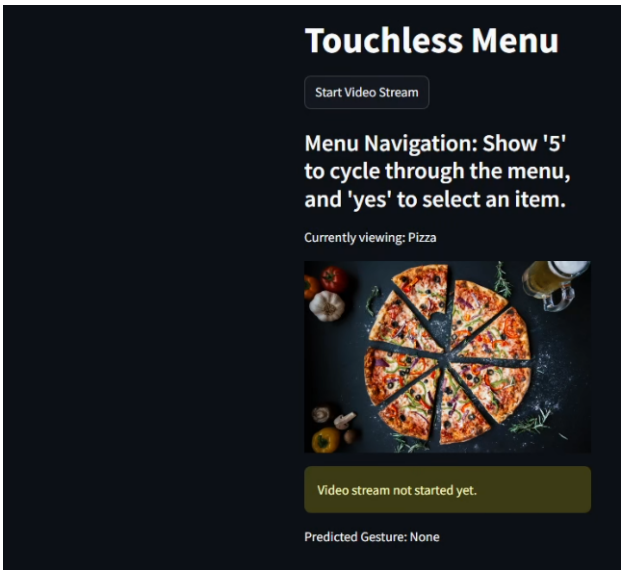
The input data are images of shape 64x64 pixels with 3 channels (RGB). Then, the pre-trained ResNet50 will do its magic to extract features. Then, we will flatten the feature maps from the ResNet50 and input them into a dense layer of 128 units with "ReLU" activation function. To prevent overfitting, a Dropout layer has been added. The last layer is the output layer using "SoftMax" activation function to output the probability distribution of our 3 classes. The final output is the predicted class that has the highest probability.

The performance of the model on the test set is 98.4% accuracy. This shows that our model can generalize well and is ready to be used for predicting real-time data.

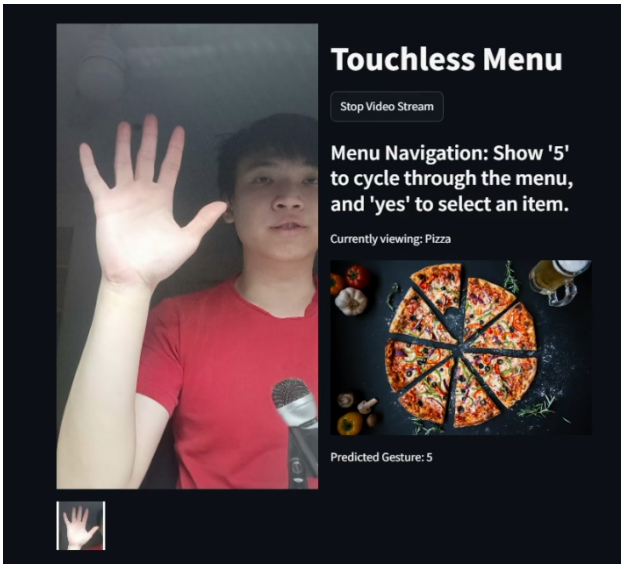
C. System Demonstration

To understand the system architecture, a video demonstration is needed. The following explanation and pictures are taken from the video [3].

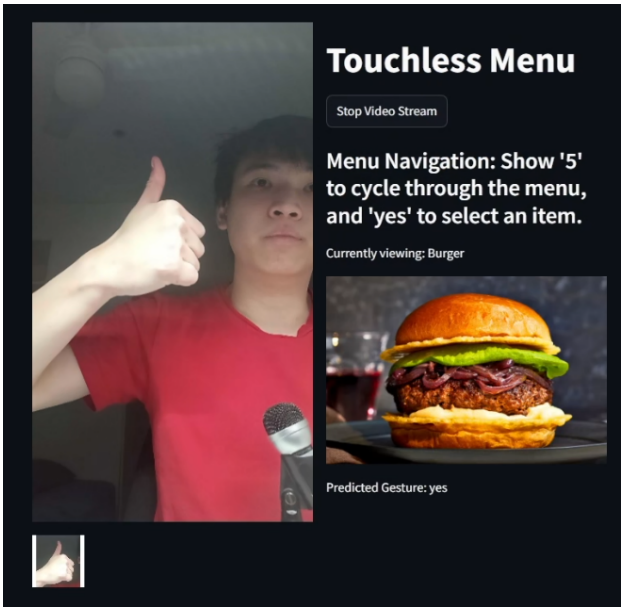
When starting the dashboard, this layout will be shown.



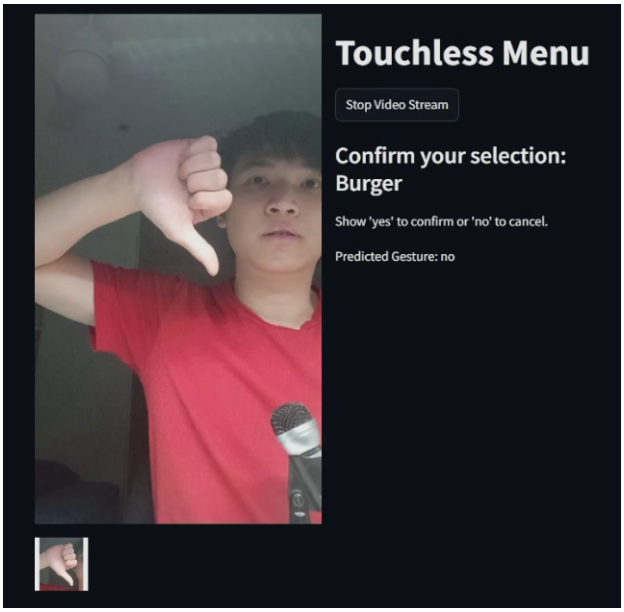
The customer needs to press "Start Video Stream" to start the ordering process. Once the process has been started, a camera viewing the customers will be shown. If the user wants to stop the ordering, they can click on the "Stop Video Stream". The starting and stopping ordering option is currently set to manual.



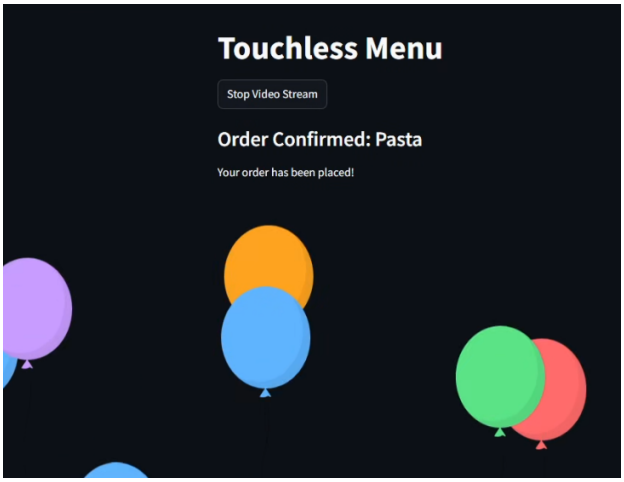
The small image below the camera is the cropped image of the customer's hand, and that image will be fed into the machine learning model. There are 3 items on the menu, and the user can cycle through the menu by raising the "5" gesture.



To confirm the selected item, the user needs to raise a thumbs up or "yes" gesture. This will then open a confirmation window for the user to re-confirm their order.

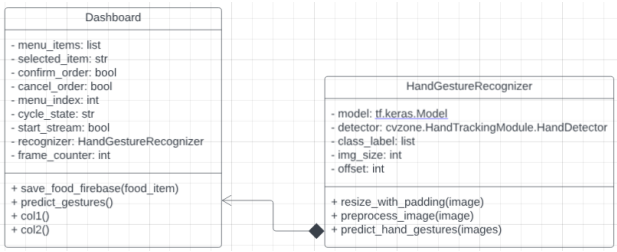


In the confirmation window, the user can either confirm or cancel the order. The cancellation gesture is a thumbs down. If cancel, the state of the system will be in the menu selection. If the user raises a thumbs up, then the food has been placed and the ordered menu item will be recorded on Firebase Database.



D. System Architecture

The picture below shows the UML design of Streamlit Dashboard and class "HandGestureRecognizer".



The Dashboard contains variables to control the state of the app. The function "predict_gesture" connects the Dashboard to the class "HandGestureRecognizer". The class relationship from the Dashboard is composition, and from "HandGestureRecognizer" it is association. The "predict_gesture" function will send an image to the class "HandGestureRecognizer" to process the image and make predictions. The "HandGestureRecognizer" then returns the predicted gesture label and a cropped hand image for the Dashboard to display. In the Dashboard, there is a mechanism that only predicts every 3 images and captures predictions for 10 seconds, then returning the most frequent label. This mechanism is added due to the delay of the live camera.

IV. Results and Discussion

A. Model results

The table below shows some common metrics to measure the performance of a classification model.

Gesture	Precision	Recall	F1-Score	Support
5 (Gesture 0)	0.99	1.00	0.99	1453
Yes (Gesture 1)	0.99	0.97	0.98	1457
No (Gesture 2)	0.97	0.99	0.98	1418
Accuracy	98.38% (4258/4328)			
Macro Avg	0.98	0.98	0.98	4328
Weighted Avg	0.98	0.98	0.98	4328

The overall test accuracy of the model is 98.38%, the precision, recall, f1 scores are all close to 1, which shows our model does an extremely good job of in classifying the 3 labels. There seems to be a slight decrease for "Yes" gesture in recall. This slight dip might be due to a slight variation of the hand captured, or the data is not clean to its fullest. For the most part, the model generalizes really well and there is no error in the live demonstration.

B. System Responsiveness

As seen in the live demonstration, there is some latency between camera and the video displayed. The system has introduced a mechanism that captures predictions every 3 frames and aggregates them over a 10 second window to determine the most frequent gesture. However, the system should aim to decrease this latency as low as possible. One of the reasons that can possibly be explained why there is some delay is because of some code overheads in the back-end design.

V. Conclusion

In this paper, we presented a touchless food ordering system that aims to enhance the ordering and dining experience of individuals with hearing impairments or trouble speaking. By leveraging the advancement in the field of Machine Learning, customers can now order food from drive-through and in restaurant settings with just some simple hand gestures. The system also integrates Firebase Database to track the preferences of customers in order to understand the consumers better.

The results show that our system did a fantastic job in recognizing gestures such as "5", "yes", and "no". Having said that, gesture recognition is limited to only 3 labels, and it means that there is many rooms for improvement. Some potential improvements are to expand the model for more gestures and enhance the processing speed. These two changes can allow our system to accommodate more complex interactions and reduce the delay caused by live video capture.

References

[1] M. A. Shaik, M. Azam, T. Sindhu, K. Abhilash, A. Mallala, and A. Ganesh, "Hand gesture based food ordering system," in *2023 International Conference on Self Sustainable Artificial Intelligence Systems (ICSSAS)*, 2023, pp. 867–872.

[2] I. C. Susanto, K. Subaramaniam, and A. S. bin Shibghatullah, "Restaurant menu with gesture recognition," *Journal of Advances in Artificial Life Robotics*, vol. 3, no. 2, pp. 102–112, 2022.

[3] Tomadonna, "Sit225 - 9.2hd: Hands-free food ordering: A gesture-based interface for restaurants," September 25 2024, youTube video. [Online]. Available: https://www.youtube.com/watch?v=yZaxhg8c7cg