Visual and Thermal Imaging Camera-based System for a Smart Cooking Assistant

Gustavo Ruiz
Electrical and Computer Engineering Department
California State University, Fullerton
Fullerton, California 92831, USA
gustavoruiz064@csu.fullerton.edu

Sai Chaitanya Kilambi
Computer Science Department
California State University, Fullerton
Fullerton, California 92831, USA
skilambi@fullerton.edu

Pratishtha Soni
Computer Science Department
California State University, Fullerton
Fullerton, California 92831, USA
psoni@fullerton.edu

Kiran George
Electrical and Computer Engineering Department
California State University, Fullerton
Fullerton, California 92831, USA
kgeorge@fullerton.edu

Anand Panangadan
Computer Science Department
California State University, Fullerton
Fullerton, California 92831, USA
apanagadan@fullerton.edu

***Abstract*—Homelessness is a condition that not only deprives individuals of a place to live but also exposes them to various risks, including malnutrition. Personalized cooking recipe recommendation systems have the potential to guide homeless individuals or those recently out of homelessness towards making informed dietary choices. However, a significant gap in current systems is their failure to cater to individuals with minimal cooking experience, offering little to no guidance on the actual cooking process. Addressing this critical gap, the present work introduces an innovative system designed to function as a smart cooking assistant. This system is not merely a passive repository of recipes but an active participant in the cooking process. It is built on the premise of observing users as they cook, utilizing a combination of hardware and advanced machine learning software to guide them through each step of the recipe meticulously. The system's hardware infrastructure is centered around the Raspberry Pi mini-computer, a compact yet powerful device capable of integrating various sensors, including a standard camera, an infrared thermal camera, and a humidity and temperature sensor. These components are strategically mounted above the cooking area, specifically focusing on the stovetop where the cooking vessel is placed. This setup enables the system to continuously monitor the cooking process. The core of the system's software is a deep learning image classification model that is generated using Google Vertex AI's Transfer learning functionality and trained on a dataset of 330 images collected from cooking pasta, captured through a smartphone camera and custom-designed hardware. By applying this method, the algorithm efficiently interprets the series of images to precisely identify the current stage of cooking and offer timely and automated suggestions. By offering a personalized, interactive, and educational cooking experience, the system not only aims to improve the nutritional intake of homeless individuals but also empowers them with the skills and confidence needed to cook healthy meals.**

Keywords—IoT, machine learning, Smart Home, supportive housing

I. INTRODUCTION

People who have only recently escaped homelessness and are beginning to live in supportive housing can benefit from personalized knowledge of nutrition and cooking due to factors including low income, limited cooking skills, and access to affordable food [1, 2]. Recently, personalized cooking recipe recommendation systems have been developed that attempt to provide guidance on nutrition [3, 4]. Such systems can also be based on emerging Artificial Intelligence (AI) technologies [5].

One important shortcoming of current recipe recommendation systems is that they cannot provide feedback on the actual process of cooking. We describe a hardware-software system that acts as a smart "cooking assistant". The proposed system observes a resident attempt to follow a particular recipe, using a camera, infrared thermal camera, and temperature sensor. The sensor data is classified to identify the specific step of the recipe that has been completed so that the resident can be prompted to continue to the next step (or complete the current step). The sensors are integrated into a Raspberry Pi 4 mini-computer. The system is mounted over a cooking range to continuously monitor the cooking area. The software component consists of image classification algorithms that translate the images from the cameras to a specific cooking step.

In prior work [6], we described the overall system with a particular focus on the hardware design. In this paper, we provide more details of the image classification approach used to identify the cooking stages. The contributions of this work are: (1) the design of both hardware and software components of a cooking assistant system, (2) steps to facilitate training of the image classifier using annotations, and (3) use of the Vertex AI algorithm suite to perform machine learning and its demonstration for detecting the sequence of stages for a cooking a simple pasta dish.

II. RELATED WORK

AI techniques, notably machine vision and image processing, have been applied in multiple aspects of food processing. These techniques are primarily used to identify the type and quality of food, grade food products, and detect defective spots or foreign objects [7]. The dataset including Chinese recipes is presented in the paper with several photos representing different stages of cooking. Various models are trained independently for distinct images in distinct categories, such as initial, intermediate, and advanced stages[8].

However, these tasks do not align with the primary steps of home cooking for personal use. There has been relatively little research on assistive cooking systems. The Cognitive Orthosis for coOKing (COOK) is a smart tablet application connected to a stove, designed to assist individuals with cognitive impairments during meal preparation[9,10]. Monitoring and tracking objects during cooking is done using real-time detection and tracking techniques such as YOLO (You Only Look Once) and KCF (Kernelized Correlation Filter). A variety of challenges are discussed in the paper, including object disappearance and appearance, occlusion, and motion blur. An evaluation of the system's performance shows that the ability to trace and identify kitchen utensils is greatly enhanced by combining detection and tracking data [11].

The study by Jelodar et al. [12] created a dataset of cooking-related images containing 11 states that represent the most frequently used cooking objects. In order to identify objects, they used a deep model based on Resnet. To detect cooking-specific items, such systems require retraining their object detection models.

Another popular object detection model is YOLO, which is also similar to Resnet. An individual network is used to detect objects in the YOLO model instead of conventional object identification methods. When compared with traditional methods, the YOLO framework simplifies detection and classification tasks [13]. MobileNets [14] is used as the model for image detection in our work from the perspective of embedding it in an embedded system.

III. System Design

A. Hardware Design

a) Sensors

The MLX90614, an infrared thermometer, operates in a -70°C to 382.2°C range, suitable for cooking applications, with tested accuracy and ease of use. Replacing the DHT11, the system introduces the AHT21. Utilizing a I2C communication protocol, the sensor measures humidity, crucial for distinguishing cooking stages like boiling, and complements other sensor data for enhanced system functionality.

b) Cameras

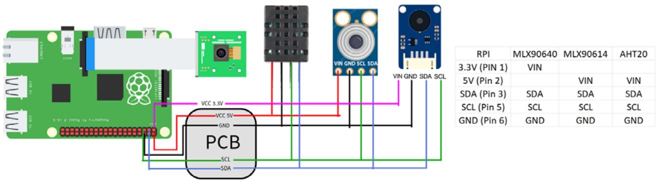
The OV5647 Mini Camera Module, designed for Raspberry Pi, captures images (2592x1944) and 720P videos at 60FPS, essential for a cooking recognition ML model. The MLX90640 Infrared Camera develops a matrix that tracks temperature changes from -40°C to 300°C with ±2°C accuracy, ideal for monitoring changes in the ingredients.

c) Processing Unit

Initially utilizing an ESP32, the system required more computational power for image recognition and use of multiple sensors, leading to the adoption of the Raspberry Pi 4. This platform efficiently handles multiple sensor control scripts and runs ML image detection models.

d) Display

A 5-inch LCD display, coupled with a PCB, offers portability and facilitates testing of camera and sensor data. The PCB ensures a tidy arrangement and effective use of GPIO pins, compatible with the LCD's pin requirements. The setup is detailed in Fig. 1.



RPI	MLX90640	MLX90614	AHT20
3.3V (PIN 1)	VIN		
5V (Pin 2)		VIN	VIN
SDA (Pin 3)	SDA	SDA	SDA
SCL (Pin 5)	SCL	SCL	SCL
GND (Pin 6)	GND	GND	GND

Fig. 1. Hardware Layout

e) Physical Design

The physical design of the hardware was constructed using SOLIDWORKS and printed on ABS material for sturdiness and protection against heat and cooking elements. It uses the base of the Raspberry Pi with a bracket for sensors. For testing purposes, multiple camera mounts was used for easy adjusting and sturdiness.A photograph of the prototype setup is shown in Fig. 3.

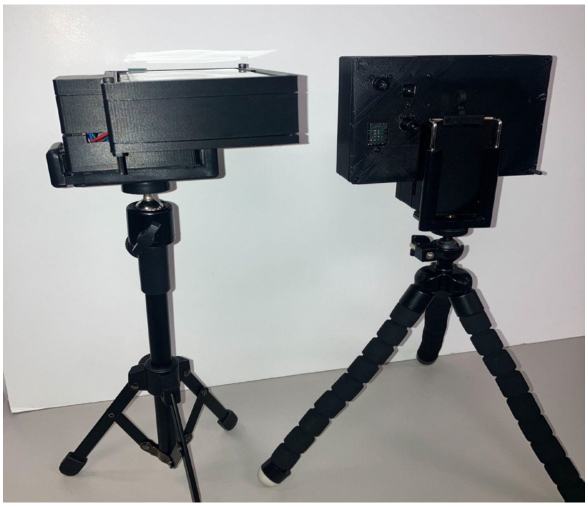


Fig. 2. Prototype hardware

B. Image Processing

The process flow illustrated in Fig 3 for the cooking assistance system emphasizes data collection and processing. The system integrates temperature data from the MLX90614 infrared sensor, humidity data from the AHT21 sensor, and visual data from the OV5647 Mini Camera Module and MLX90640 Infrared Camera. This collected data is subsequently processed by a Raspberry Pi, which employs an AutoML model for image classification. The processed information is then displayed on an LCD screen, offering real-time cooking guidance. The integration of multiple sensors and processing units constitutes the core of the cooking assistance application.

In this study, we utilize "AutoML," a pre-trained model from Google Vertex AI, to tackle the challenge of cooking stage identification, a subset of image classification problem. Google Vertex AI is a machine learning platform designed for deploying and generating machine learning models and monitoring their performance. As part of the Google Cloud ecosystem, AutoML (Automated Machine Learning) within Google Vertex AI automates many of the complex and repetitive tasks involved in building machine learning models. It offers pre-trained models, supports transfer learning, and allows us to leverage existing models and adapt them to our real-time food image dataset. The model's efficacy is tested on a simplified culinary task—cooking pasta. We categorize the cooking process into six distinct stages, each characterized by specific items observable on the stovetop, ranging from an "Empty burner" to a "Pot with cooked pasta."

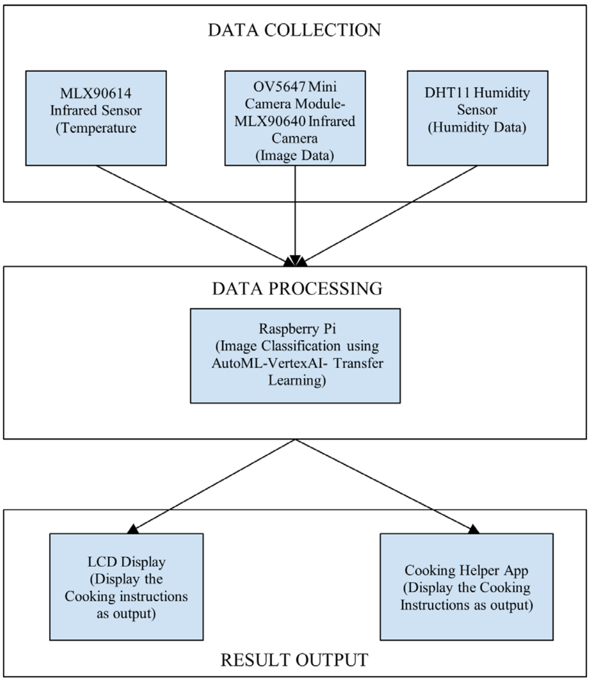

DATA COLLECTION
MLX90614 Infrared Sensor (Temperature
OV5647 Mini Camera Module-MLX90640 Infrared Camera (Image Data)
DHT11 Humidity Sensor (Humidity Data)
DATA PROCESSING
Raspberry Pi (Image Classification using AutoML-VertexAI- Transfer Learning)
LCD Display (Display the Cooking instructions as output)
Cooking Helper App (Display the Cooking Instructions as output)
RESULT OUTPUT

Fig. 3. Process Flow For the cooking assistance system

a) Image Collection

The collection and labeling of images for the dataset is a non-trivial task, requiring consideration of the model's scalability for future use. In this study, we aimed to detect various stages of cooking pasta and created labels for the collected images accordingly. As our research focuses on developing an image classification for supportive housing, the images were collected from a video recording demonstrating how to cook pasta. Parameters such as camera orientation and the type of pans used were kept identical to those in supportive housing to better emulate the desired conditions. Frames were extracted using a Python script. The dataset comprises 330 images, categorized based on the stages of cooking pasta and the corresponding camera view at each instance, specifically: “Empty burner,” “Empty pot,” “Pot with water,” “Pot with boiling water,” “Pot with pasta,” and “Pot with cooked pasta.” Sample images from these classes are presented in Fig 4.

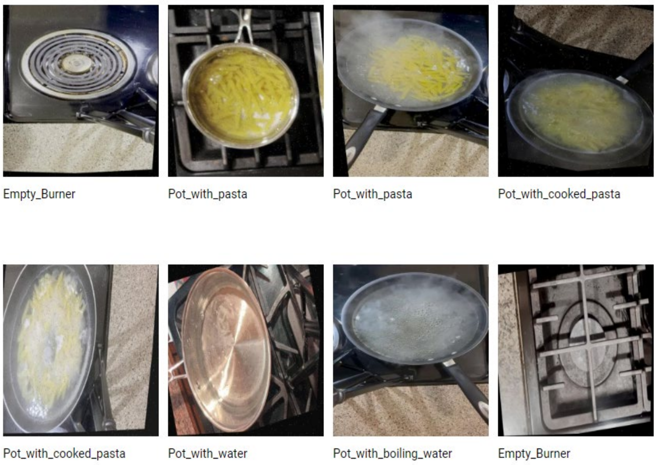


Fig. 4. Sample collected images for Empty burner, Pot with boiling water, Pot with Cooked pasta, Empty pot, Pot with pasta, Pot with water.

b) Image Augmentation

For the model to generalize effectively, it is crucial for our dataset to exhibit a wide variation of images. To create a larger and more diverse dataset, image augmentation was employed. RoboFlow was utilized to streamline the image preprocessing and augmentation phases of our project, as illustrated in Fig 5. This ensured that our model was trained on high-quality, consistently formatted, and diversified images. The uniformity and augmentation facilitated by RoboFlow prepared our dataset for effective model training, enhancing its ability to generalize across new, unseen images in practical applications. The augmentations applied included transformations such as rotating, flipping, and adjusting image color and brightness. Specifically, the augmentations included horizontal and vertical flipping, 90° rotation (clockwise, counter-clockwise, and upside down), rotation between -15° and +15°, shear ±10° (horizontal and vertical), hue adjustment between -15° and +15°, saturation adjustment between -29% and +29%, brightness adjustment between -15% and +15%, exposure adjustment between -13% and +13%, blur up to 2.5 pixels, and noise up to 0.58% of pixels. Below are the augmented images of an empty burner, a pot with cooked pasta, a pot with pasta, and a pot with boiling water.

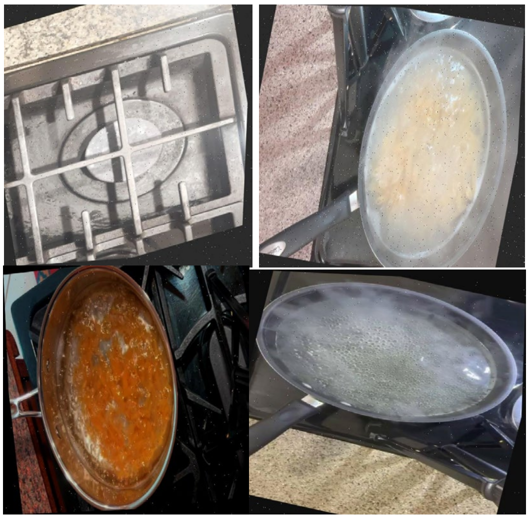
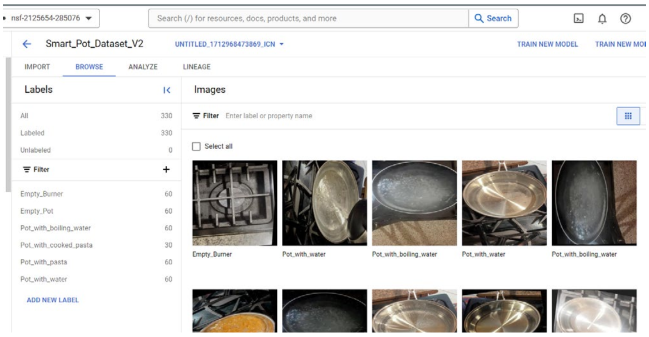


Fig. 5. Augmented images used for training the image classification models.

c) Labeling Images

The augmented images are subsequently uploaded into a storage bucket and classified according to their respective categories on the Vertex AI portal. These classified images are then used to train the model. One significant advantage of using Vertex AI is its ability to streamline tasks such as labeling, thereby enhancing the overall efficiency of the model training process.



All	330
Labeled	330
Unlabeled	0
Empty_Burner	60
Empty_Pot	60
Pot_with_boiling_water	60
Pot_with_cooked_pasta	30
Pot_with_pasta	60
Pot_with_water	60

Fig. 6. Image depicting the labeling interface on VertexAI

d) Transfer Learning

Transfer learning leverages pre-trained models to address new tasks with limited datasets, thereby accelerating the training process and enhancing performance. AutoML facilitates the utilization of this feature and additionally employs Neural Architecture Search (NAS), which automates the design of neural networks. NAS optimizes network architecture by evaluating multiple configurations, leading to improved performance.

In our study, we generated two image classification models, leveraging these features, and trained them using different approaches to determine which method yields better results. For the first model, Smart_Pot_Dataset_V2,we trained it exclusively on the newer dataset. For the second model, Smart_Pot_Dataset_V2_Incremental, we performed incremental training by pretraining it on a previously collected dataset of cooking pasta (Fig 7), which had poorer image quality, and then fine-tuning it on a newer dataset with better image variation.

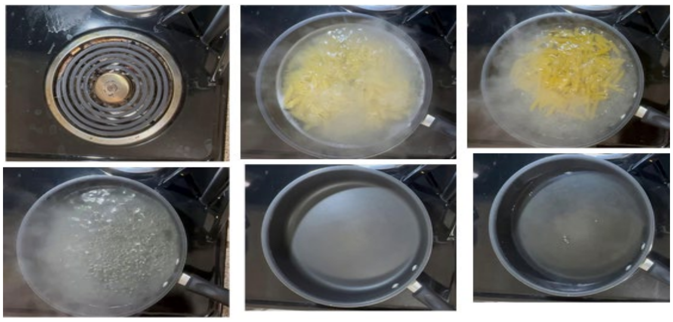


Fig. 7. Sample collected images for Empty burner, Pot with boiling water, Pot with Cooked pasta, Empty pot, Pot with pasta, Pot with water from the previous dataset.

IV. Results and Discussions

The evaluation of two models using the Vertex AI platform yielded notable results. The first model, Smart_Pot_Dataset_V2, achieved an average precision of 0.978, with a precision of 96.8% and a recall of 90.9%. The second model, Smart_Pot_Dataset_V2_Incremental, achieved an average precision of 0.999, with a precision of 89.2% and a perfect recall of 100%. Both models used the same dataset distribution: 330 images, with 264 for training, 33 for validation, and 33 for testing.

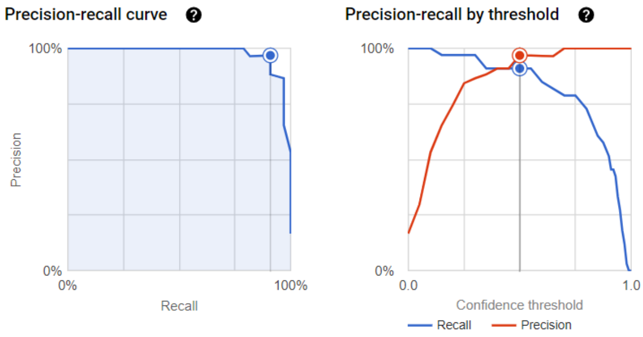


Fig. 8. Precision and Recall curves for Smart_Pot_Dataset_V2

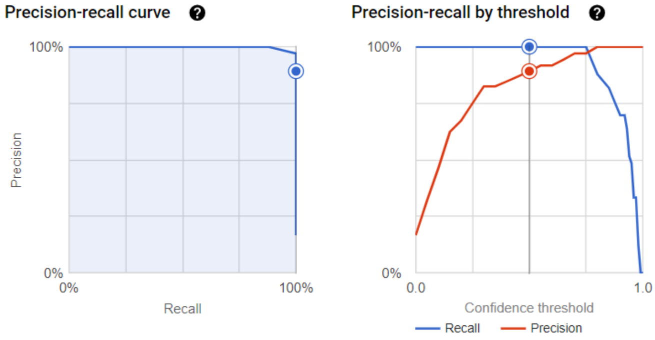
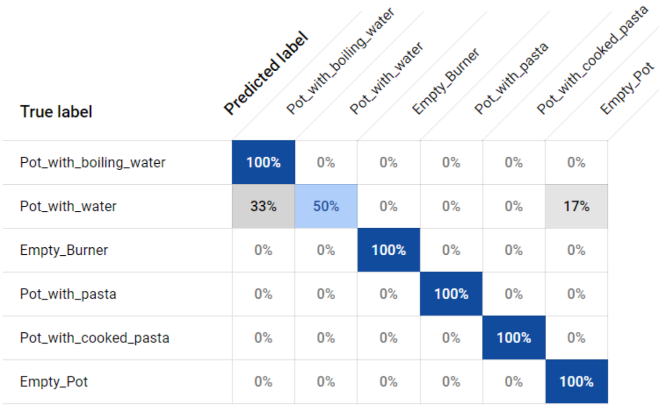


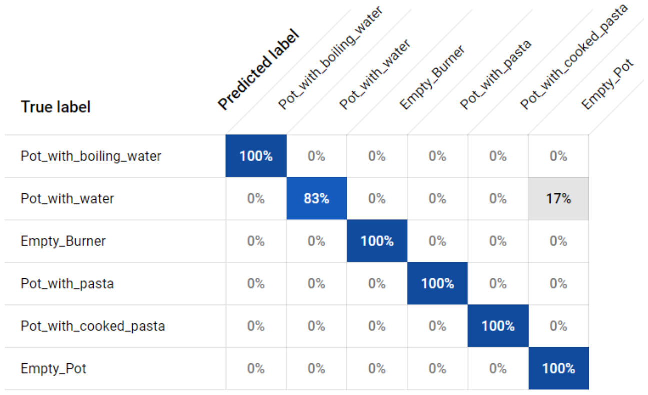
Fig. 9. Precision and Recall curves for Smart_Pot_Dataset_V2_Incremental

The first model demonstrated higher precision, while the incremental model showed superior recall, indicating a trade-off between precision and recall. The incremental training approach allowed the model to leverage prior knowledge from a less diverse dataset and fine-tune it with high-quality data, resulting in nearly perfect recall. Both models exhibited strong generalization capabilities, suggesting robustness and the ability to handle data variations.



True label	Predicted label	Pot_with_boiling_water	Pot_with_water	Empty_Burner	Pot_with_pasta	Pot_with_cooked_pasta	Empty_Pot
Pot_with_boiling_water	100%	0%	0%	0%	0%	0%	
Pot_with_water	33%	50%	0%	0%	0%	17%	
Empty_Burner	0%	0%	100%	0%	0%	0%	
Pot_with_pasta	0%	0%	0%	100%	0%	0%	
Pot_with_cooked_pasta	0%	0%	0%	0%	100%	0%	
Empty_Pot	0%	0%	0%	0%	0%	100%	

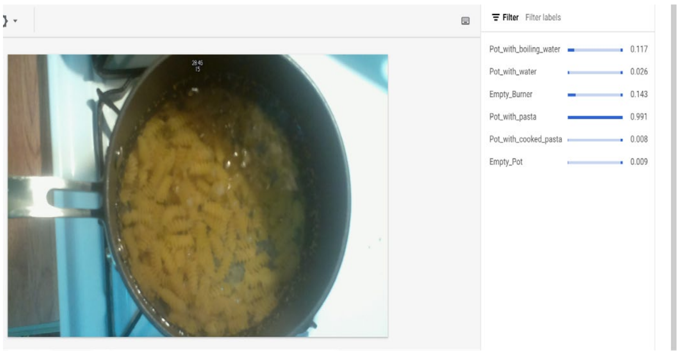
Fig. 10. Confusion Matrix for Smart_Pot_Dataset_V2



True label	Predicted label	Pot_with_boiling_water	Pot_with_water	Empty_Burner	Pot_with_pasta	Pot_with_cooked_pasta	Empty_Pot
Pot_with_boiling_water	100%	0%	0%	0%	0%	0%	
Pot_with_water	0%	83%	0%	0%	0%	17%	
Empty_Burner	0%	0%	100%	0%	0%	0%	
Pot_with_pasta	0%	0%	0%	100%	0%	0%	
Pot_with_cooked_pasta	0%	0%	0%	0%	100%	0%	
Empty_Pot	0%	0%	0%	0%	0%	100%	

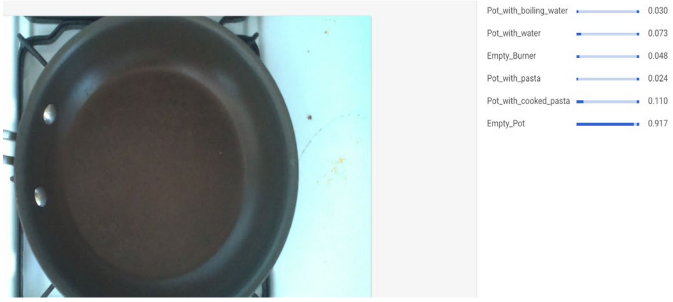
Fig. 11. Confusion Matrix for Smart_Pot_Dataset_V2_Incremental

The evaluation metrics for both models were validated when deployed and tested on actual images collected from the designed apparatus. The models demonstrated proper generalization across a wide variety of images. Fig. 12-15 illustrate some of the predictions made by the models.



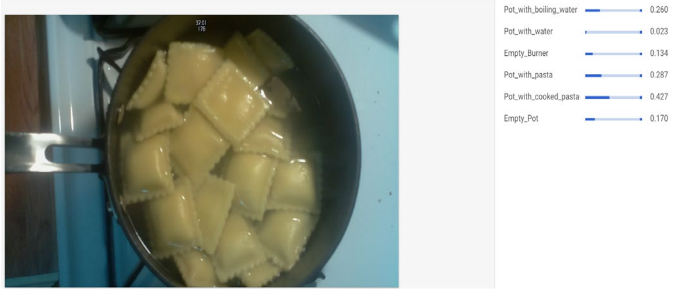
Pot_with_boiling_water	0.117
Pot_with_water	0.026
Empty_Burner	0.143
Pot_with_pasta	0.991
Pot_with_cooked_pasta	0.008
Empty_Pot	0.009

Fig. 12. Smart_Pot_Dataset_V2 prediction on actual image of 'Pot with Pasta' from the apparatus with 99% accuracy.



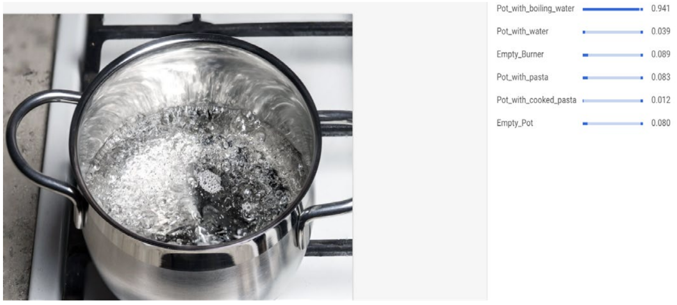
Pot_with_boiling_water	0.030
Pot_with_water	0.073
Empty_Burner	0.048
Pot_with_pasta	0.024
Pot_with_cooked_pasta	0.110
Empty_Pot	0.917

Fig. 13. Smart_Pot_Dataset_V2 prediction on actual image of 'Empty pot' from the apparatus with 92% accuracy.



Pot_with_boiling_water	0.260
Pot_with_water	0.023
Empty_Burner	0.134
Pot_with_pasta	0.287
Pot_with_cooked_pasta	0.427
Empty_Pot	0.170

Fig. 14. Smart_Pot_Dataset_V2_Incremental prediction on image of 'Pot with pasta' which was out of the trained variety from the apparatus with 43% accuracy.



Pot_with_boiling_water	0.941
Pot_with_water	0.039
Empty_Burner	0.089
Pot_with_pasta	0.083
Pot_with_cooked_pasta	0.012
Empty_Pot	0.080

Fig. 15. Smart_Pot_Dataset_V2 prediction of 'Pot with boiling water' image from the internet with 94% accuracy.

V. Conclusion and Future Work

The hardware prototype is created for swift development and testing. Its physical design and classification model will be revamped to endure more challenging environments. Specifically, the following issues will be investigated.

One issue to address is the device's power supply. During testing, a wall outlet near the stove was used, as a battery pack did not last long enough to complete a cooking recipe.

We utilized the image classification method in Google Cloud Vertex AI AutoML to develop our model. This approach has significantly enhanced the model's accuracy and precision in making predictions. It effectively distinguishes between various classes, such as "boiling water." However, despite these advancements, the model still requires additional data to accurately differentiate between closely related classes, specifically "empty pot" and "pot with water."

Integrating temperature metadata with temperature images can greatly improve the model's ability to distinguish between an empty burner and a pan with water. By combining these two data sources, the model can more accurately identify the presence of water based on the heat patterns and temperature readings, leading to more precise classifications.

It is observed that when the camera is placed directly above the pot the steam generated by the boiling water obstructs the vision and may fail in properly detecting the stage. This problem can be solved by also including the temperature and the humidity data.

Acknowledgement

This material is based upon work supported by the National Science Foundation under Grant No. 2125654.

References

[1] Sprake, Eernest F., John M. Russell, and Maria E. Barker. "Food choice and nutrient intake amongst homeless people." Journal of Human Nutrition and Dietetics 27.3 (2014): 242-250.

[2] Wiecha, Jean L., Johanna T. Dwyer, and Martha Dunn-Strohecker. "Nutrition and health services needs among the homeless." Public Health Reports 106.4 (1991): 364.

[3] Pecune, Florian, Lucile Callebert, and Stacy Marsella. "A Recommender System for Healthy and Personalized Recipes Recommendations." HealthRecSys@ RecSys. 2020.

[4] Ge, Mouzhi, Francesco Ricci, and David Massimo. "Health-aware food recommender system." Proceedings of the 9th ACM Conference on Recommender Systems. 2015.

[5] Yera, Raciel, Ahmad A. Alzahrani, and Luis Martínez. "Exploring post-hoc agnostic models for explainable cooking recipe recommendations." Knowledge-Based Systems 251 (2022): 109216.

[6] G. Ruiz, S. C. Kilambi, P. Soni, K. George, and A. Panangadan. "Design of a Multisensor System for a Smart Cooking Assistant." In IEEE International Conference on Artificial Intelligence x Medicine, Health, and Care (AIMHC). IEEE, 2024.

[7] Zhu, Lili, et al. "Deep learning and machine vision for food processing: A survey." Current Research in Food Science 4 (2021): 233-249.

[8] ZHANG, Yixin, Yoko YAMAKATA, and Keishi TAJIMA. "Stage-Aware Recognition Method for Foodstuffs Changing in Appearance in Different Cooking Stages on Chinese Recipe."

[9] Giroux, Sylvain, et al. "Cognitive assistance to meal preparation: design, implementation, and assessment in a living lab." 2015 AAAI Spring Symposium Series. 2015.

[10] Yaddaden, Amel, et al. "Using a cognitive orthosis to support older adults during meal preparation: Clinicians' perspective on COOK technology." Journal of Rehabilitation and Assistive Technologies Engineering 7 (2020): 2055668320909074.

[11] Ngankam, Hubert, et al. "Real-Time Multiple Object Tracking for Safe Cooking Activities." International Conference on Smart Homes and Health Telematics. Cham: Springer Nature Switzerland, 2023.

[12] Jelodar, Ahmad Babaeian, Md Sirajus Salekin, and Yu Sun. "Identifying object states in cooking-related images." arXiv preprint arXiv:1805.06956 (2018).

[13] Redmon, Joseph, et al. "You only look once: Unified, real-time object detection." Proceedings of the IEEE conference on computer vision and pattern recognition. 2016.

[14] Howard, Andrew G., et al. "Mobilenets: Efficient convolutional neural networks for mobile vision applications." arXiv preprint arXiv:1704.04861 (2017).