

Smart Microwave Oven with Image Classification and Temperature Recommendation Algorithm

Tareq Khan

School of Engineering Technology, Eastern Michigan University, USA

Article Info

Article history:

Received May 25, 2018

Revised Jul 5, 2018

Accepted Aug 8, 2018

Keyword:

Closed-loop

Embedded system

Histogram

Image classification

Temperature sensing

ABSTRACT

When food is warmed in a microwave oven, the user guesses the estimated time for the heating. This cognitive process of guessing can be incorrect - resulting the final food temperature to be too hot or still cold. In this research, a novel closed-loop microwave oven is designed which automatically suggests the target temperature of a food by learning from previous experiences and the heating stops automatically when the food temperature reaches the target temperature. The proposed microwave captures and classifies the food image, and recommends the target temperature, thus the user does not need to remember the target food temperature each time the same food is warmed. The algorithm gradually learns the type of foods that are used in that household and becomes smarter in the recommendation. The proposed algorithm can recommend target temperature with an accuracy of 86.31% for solid food and 100% for liquid food. A prototype of the proposed microwave is developed using the embedded system and tested.

Copyright © 2018 Institute of Advanced Engineering and Science.

All rights reserved.

Corresponding Author:

Tareq Khan,
School of Engineering Technology,
Eastern Michigan University,
118 Sill Hall, Ypsilanti, MI 48197, USA.
Email: tareq.khan@emich.edu

1. INTRODUCTION

Digital technology, sensors, and learning algorithms are changing the way we interact with devices and appliances. Recent advances in miniature sensors, embedded processors, and wireless technologies have caused a rapid growth in smart applications such as controlling and monitoring home appliances [1]-[4], monitoring of smart parking lots [5], [6], automatic meter reading (AMR) using smart meters [7] etc. A microwave oven (commonly referred to as a microwave) is a kitchen appliance that heats and cooks food by exposing it to electromagnetic radiation in the microwave frequency range. It has become a common appliance in the modern kitchen since its release to the public for residential and commercial use in 1967 [8]. Since then, some sort of timing mechanism, whether it be through dials or through push-button programming, is used to control how long the microwave heats the food.

In this process, the user needs to estimate the exact time mentally by considering several variables such as the current temperature of the food (for instance, the food could be taken out from refrigerator, freezer, or could be at room temperature), quantity and thermal properties of the food, power level of the microwave oven, and finally the desired target temperature of the food. This cognitive process of calculating the exact required time is complex and may be estimated incorrectly. If a longer time is estimated than the required time, then the food becomes too hot to eat. It may cause burn in the mouth if the food is consumed with a spoon without checking the temperature by touching it first. When the food is too hot, the user needs to wait until the food cools down. If a shorter time is estimated than the required time, then the user needs to

repeat the process of checking the current temperature of the food by touch, estimating time, and then heating again until the desired target temperature is reached. This process is tiresome and error-prone.

In this paper, a solution of the above problems is proposed. The overall operation of the system is shown in Figure 1. The main contributions of the paper are mentioned below.

1. In this project, a closed-loop microwave oven is designed which continuously measures the food temperature without physical contact while the food is being heated, and stops the microwave automatically when the food temperature reaches the target temperature. In this process, the user does not need to guess the extract required time mentally. This is more convenient to use and ensures the exact target temperature of the food.
2. In this research, an automatic target temperature recommendation algorithm is also proposed. When a food is inserted in the proposed microwave oven and the door is closed, it takes an image of the food and stores the color pattern (such as histogram) of the food in a database. Once a temperature is set for that food using push button switches, that temperature is assigned to the stored histogram in the database. Later, when the user heats the same kind of food again (even if the food is on a different plate or cup), the proposed algorithm automatically recommends the temperature which was assigned for that food previously. The user just presses the 'start' button and do not need to remember or re-enter the target temperature. The algorithm calculates the color pattern similarity between the current food and the histograms stored in the database. The closest match histogram's assigned temperature is recommended as the target temperature. If there is no significant match between the current food color pattern and the stored histograms, then the new histogram for that food and its desired temperature is added to the database. In this way, the microwave oven gradually learns the type of foods that are used in that household and becomes smarter in recommending or predicting the target temperature.
3. For 2018, total unit shipments of microwave ovens are projected to reach 12.69 million units [9] and 96% of U.S. homes use a microwave oven [10]. The survey in [11] shows that Americans are using microwaves to warm and heat more, rather prepare dishes from scratch. The market report in [12] published in December 2017 states that microwaves lag behind other major appliances in terms of "smart" features. The proposed novel machine learning based autonomous microwave can fill this market and research gap.

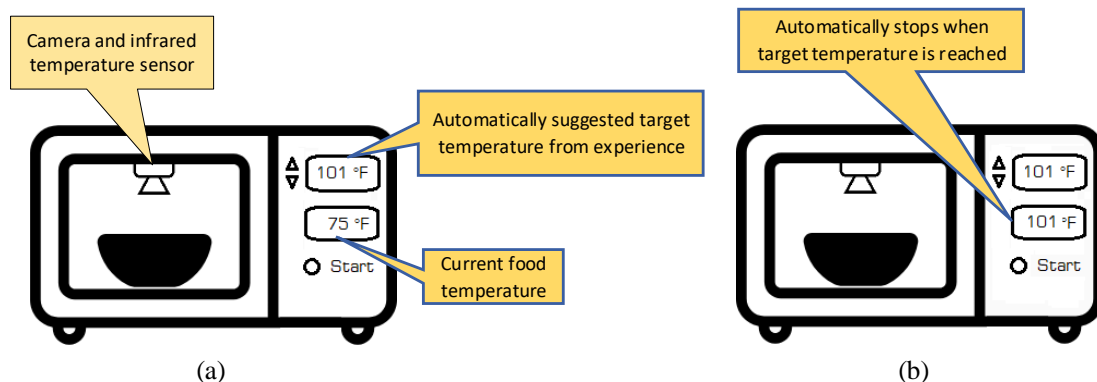


Figure 1. The overall operation of the proposed system. (a) when a food is inserted and the door is closed, the proposed microwave suggests the target food temperature using image classification algorithm; (b) After the "Start" button is pressed, the microwave heats the food until food temperature reaches the target temperature

Several works are found in the literature about measuring the temperature of the sample or food while they are heated by microwave in real time. The use of conventional electric temperature sensors inside a microwave oven is problematic because of the strong electromagnetic field environment. Fiber-optic sensors [13] are insensitive to electromagnetic fields. However, they require direct contact with the food and requires cleaning each time the food is warmed. This is inconvenient. In [14], the use of an infrared (IR) fiberoptic radiometer is proposed for non-contact temperature measurements inside a microwave oven. In this method, the tip of the IR fiber is situated directly above the sample and transmits the thermal radiation emitted by the heated sample to a radiometer. In [15], thermal imaging technique with forward-looking infrared (FLIR) camera is used, where temperature and other data is transferred to a PC for monitoring and

controlling the power level of the microwave. In [16], FLIR camera is used to capture the thermal image of the food surface. However, FLIR cameras are expensive [17] and can significantly increase the price of the microwave oven. In [18], temperature monitoring inside a microwave oven is proposed by using conventional color charge-coupled device (CCD) camera. This method could be low cost, but may not be accurate. An inexpensive temperature controller [19] and optimum power control strategies for microwave oven are discussed in [20], [21]. However, they do not discuss the automatic shutdown of microwave power when the food temperature reaches the desired temperature.

A recent heating element based oven (not microwave-based heating) in [22], uses a high definition (HD) camera to identify from a common predefined set of foods that are put inside and recommend cooking times and temperature. A resistance temperature detectors (RTD) temperature sensor probe is inserted manually into the food and user can get a notification on smart devices when the food is cooked. However, this method of temperature sensing requires physical contact of the temperature sensor with the food. Moreover, the food recognition algorithm is fixed to a set of predefined foods, thus it does not learn any new food item and unable to recommend their temperatures. In this proposed research, food temperature is measured using IR temperature sensor. It is insensitive to microwaves, much cheaper than FLIR camera, and does not need physical contact with the food. In this proposed work, the automatic shutdown of microwave power when the food temperature reaches the desired temperature is also discussed using a microcontroller-based embedded system. The proposed food classification algorithm gradually learns the food items by image processing and recommends their target temperatures; rather than a predefined set of foods.

2. RESEARCH METHOD

2.1. Food Classification using Image Processing and Temperature Recommendation Algorithm

The proposed microwave oven captures an image of the food when it's door is closed. The image is then processed and the algorithm tries to classify the image by comparing its color pattern with the stored color patterns in the database. If the food is classified, then it recommends the temperature that is assigned to that food. If the food is unclassified, then it's color pattern and desired temperature is added to the database. A block diagram of the proposed classification algorithm is shown in Figure 2. A brief description of the algorithm is discussed below.

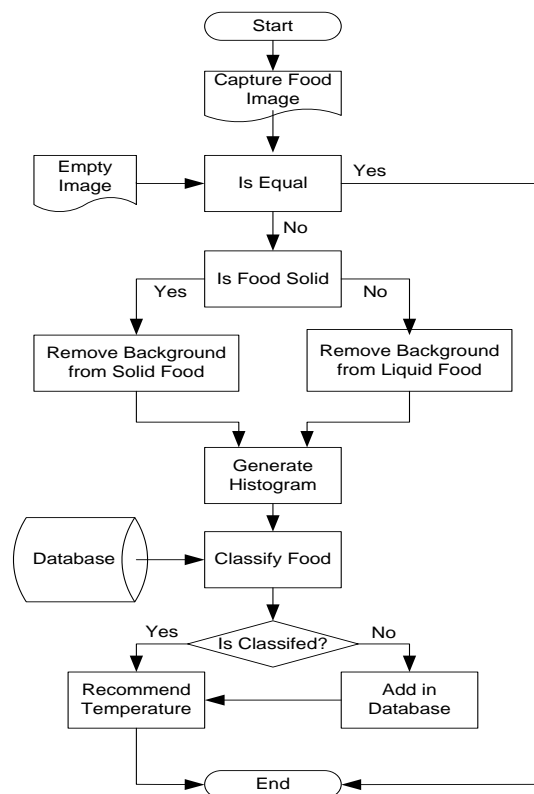


Figure 2. Block diagram of the food classification and temperature recommendation algorithm

2.1.1. Existence of Food Detection

As soon as the door of the microwave oven is closed, an image, I_{captured} , is captured using an HD camera mounted on the roof of the oven. The image is an RGB color image of size 320×240 . The first step of the algorithm is to determine whether there is food in the oven or it is empty. To do that, a preexisting image of the microwave oven without any food, I_{empty} , is compared with I_{captured} . If there is a significant match, then the microwave is empty and the algorithm terminates without temperature recommendation. If there is a significant mismatch, then there is food in the microwave and it goes to the next block of the algorithm. As the microwave turntable tray can rotate, it could be in a different position than the preexisting I_{empty} image. So, a hard pixel comparison between I_{captured} and I_{empty} will not work.

In Figure 3, an example of a preexisting empty image (a), a captured image without food (c), and a captured image with food (e) along with their grayscale histograms (b), (d) and (f) are shown namely. Here we see that, even though (c) is rotated compared with (a), their histograms (d) and (b) are quite similar because histograms do not contain the position information of the pixels [12]. On the other hand, the histogram of the image with food, (f), is quite different from (b) and (d). To compare the images, histogram difference between the two images are calculated. If the difference of the histograms is less than a threshold, $\text{HIST_DIFF_THRESHOLD}$, then the images are considered equal, else they are considered different. Experiments have been conducted with 10 empty images, I_{empty} , having different rotations of the turntable tray and a threshold value is assigned as 100. The steps of determining whether the images are equal are shown in the pseudocode in Figure 4.

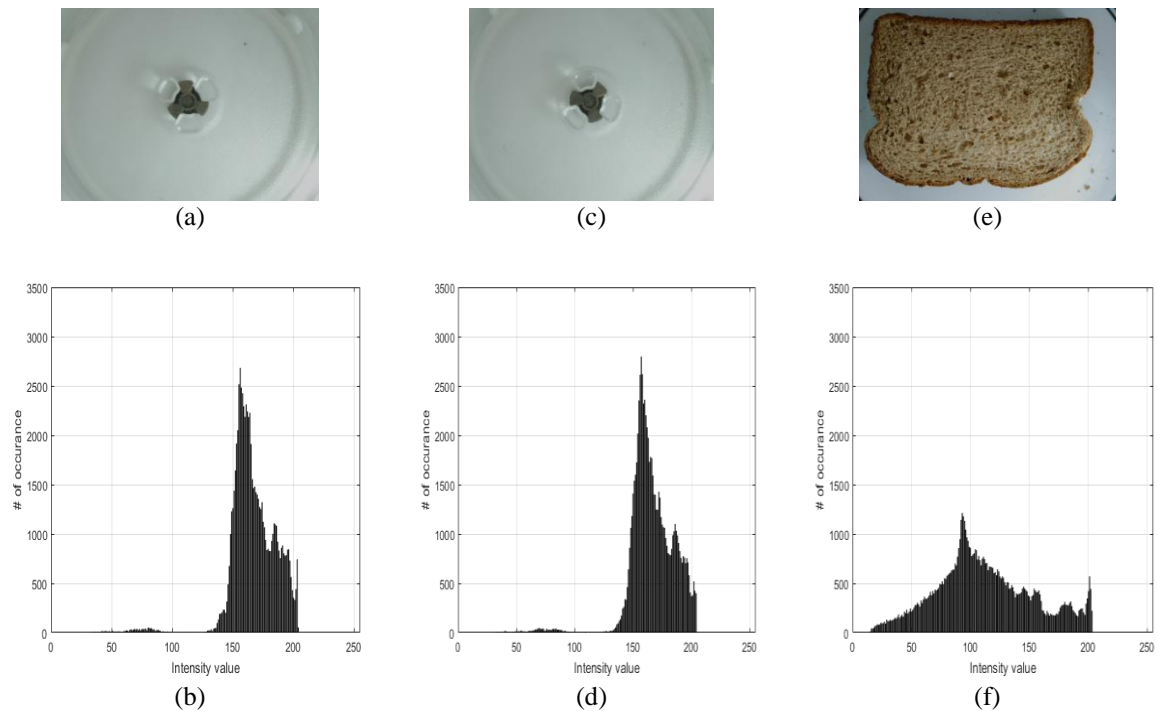


Figure 3. (a) Preexisting image when no food in oven; (b) Histogram of (a); (c) Captured image when no food in oven, the tray could be rotated compared with (a); (d) Histogram of (c); (e) Captured image when food is present; (f) Histogram of (e)

```

Icaptured_gray := grayscale of Icaptured
Iempty_gray := grayscale of Iempty
h1 := histogram of Icaptured_gray
h2 := histogram of Iempty_gray
Calculate the average of histogram differences as d

$$d := \frac{\sum |h1 - h2|}{256}$$

IsEqual := (d < HIST_DIFF_THRESHOLD)

```

Figure 4. Pseudocode for comparing images

2.1.2. Solid and Liquid Food Classification

After the existence of the food is determined, the next block of the algorithm determines whether the food is *solid* or *liquid*. Liquids (commonly referred as *drinks*) are generally served in cups or glasses, whereas solids are served on plates or bowls. Classifying whether the food is solid or liquid is required to detect and remove the food background in the next block of the algorithm. The premise for classifying solid and liquid food is that, in most cases, the texture of solid food is coarse, whereas the texture of liquid food is smooth and homogenous. So, the variance (σ^2) of solid food image will be much higher than the variance of liquid food.

In Figure 5, the color image of a solid food is shown in (a), its grayscale image is shown in (b) with a 32×32 pixel block drawn in magenta color at middle, and the 3D plot of the block where height is proportional to the pixel value is shown in (c). Similarly, (d), (e), and (f) shows the images for a liquid food namely. Here, we see that the pixel values in the block change more in solid food as shown in (c), whereas the pixel values are almost the same in liquid food as shown in (f). For this example, the variance of (c) is 746.9 and the variance of (f) is 1.7. The variance is calculated using (2) and (3) where, N is the total pixel in the block, x_i is the i th pixel value, and m is the mean of all pixel values in the block. Simulations have been conducted with 263 solid food and 53 liquid food images for a block size of 32×32 in the middle. It has been found that the maximum variance for liquid food is 4.6. A threshold, $\text{MIN_SOLID_FOOD_VAR}$, is assigned as 10 in the algorithm. If the block variance is less than or equal to the threshold, then the food is classified as liquid, else it is classified as solid.

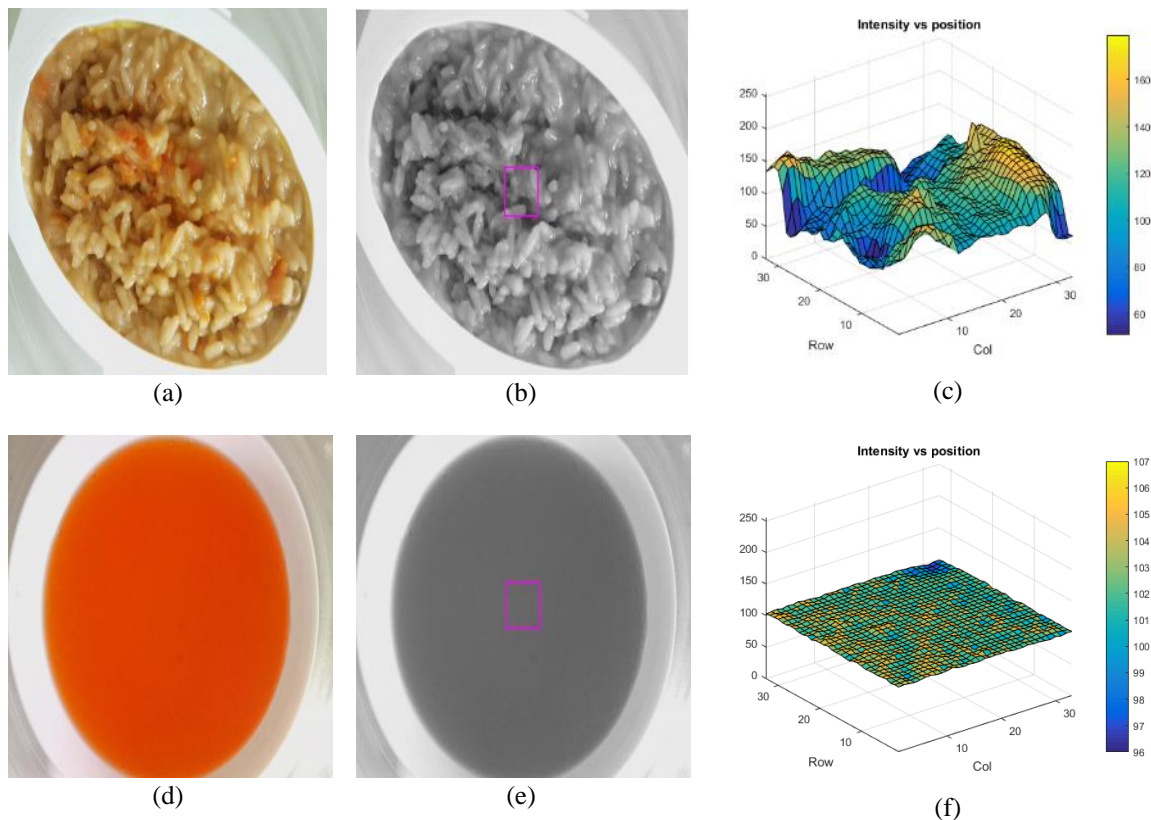


Figure 5. (a) Color image of solid food; (b) Grayscale image of (a) with a block drawn in magenta color at middle; (c) 3D plot of the block in (b) where height is proportional to the pixel value; (d) Color image of liquid food; (e) Grayscale image of (d) with a block drawn in magenta color at middle; (f) 3D plot of the block in (e) where height is proportional to the pixel value

2.1.3. Background Removal

In this context, only the food portion of the captured image is defined as foreground, as shown inside the green region in Figure 6(a), and the rest of the image is defined as background as shown in Figure 6(a) and Figure 6(b). The color information of the food is contained in the foreground, and not in the background. The background may vary if the same food is served in a different type of plate, bowl, or cup.

So, the algorithm removes the background from the image so that the generated histogram (discussed in the next section) do not contain background color information, and thus the food can be classified accurately even if it is put on a different container.

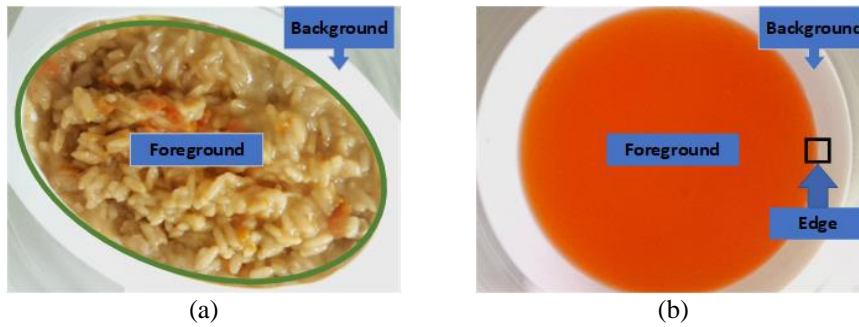


Figure 6. (a) Foreground (inside the green region) and background (outside the green region) of solid food; (b) foreground (the orange part) and background (outside the orange part) of liquid food

Solid food is generally put on a plate or pot. When food is placed on a plate, it is generally surrounded by smooth texture having a monotonous color of the plate as shown in Figure 6(a). For solid food, the foreground has high variance, as shown in Figure 5(c), and its background has low variances in the pixel values. This heuristic is used to detect the background of the solid food. Liquid food or drink is generally served in a mug or glass. For liquid food, the foreground has low variance, as shown in Figure 5(f), and it is surrounded by the edge of the container as shown in Figure 6(b). The block covering an edge, as shown in Figure 6(b) in black color, will have higher variance in pixel values. These high variance edge blocks are used to detect the background of the liquid food.

To automate the background removal process, a copy of the image is first converted to grayscale and then divided into four quadrants – top-right, top-left, bottom-left, and bottom-right namely as shown in Figure 7. Then starting from the center of the grayscale image, the variance is calculated for each 10×10 blocks according to the horizontal sequence (i.e. row after row in raster scan fashion) as shown Figure 7(a) at the top-right quadrant. For solid food, if the variance of CONSEC_BG_BLK number of consecutive blocks is less than a threshold value, BG_VAR_THRESHOLD, then the background is reached and the remaining blocks of that row are made black on the actual color image. For liquid food, if the variance of a block is greater than a threshold value, EDGE_VAR_THRESHOLD, then the background is reached and the remaining blocks of that row are made black on the actual color image. Then this process is repeated for the remaining top-left, bottom-left, and bottom-right quadrants according to the sequence shown in Figure 7 (a). After that, the entire process is repeated according to the vertical sequence as shown in Figure 7 (b). Another similar horizontal and vertical sequence are repeated where it searches for fully black block and once found, it fills the rest of the row or the column with black namely. The background of the images shown in Figure 5(a) and Figure 5(d) is removed using the proposed algorithm and the results are shown in Figure 8 (a) and Figure 8 (b) namely.

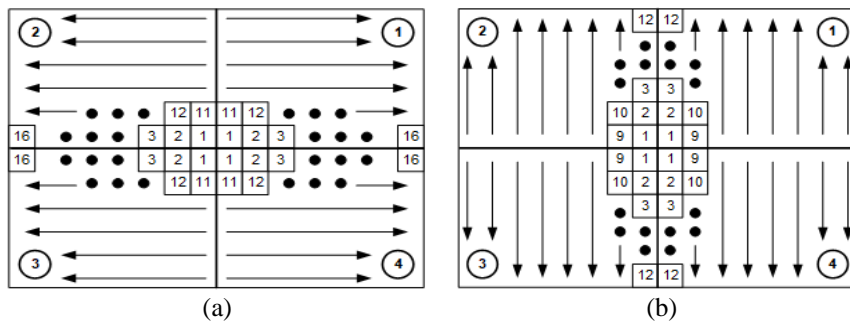


Figure 7. Block variance checking sequences in the four quadrants of the image: (a) horizontal pass; (b) vertical pass



Figure 8. (a) Background removed from Figure 4 (a); (b) background removed from Figure 4 (d)

2.1.4. Histogram Generation

There are several feature extraction techniques available in the literature targeting different types of images [23],[24]. For instance, a face recognition algorithm will extract features such as between-eye distance, width-length ratio etc. [25]. However, the classification of food image is quite different as the food does not have any fixed shape or texture. A food will have different shape each time when it is placed on a plate or when it is stirred. The overall color pattern (i.e. histogram) of the food is the only significant feature for classifying food. In the proposed algorithm, the histogram of the image, h_{test} , is generated for red (R), green (G), and blue (B) channels after removing the background. The black pixel count in the histogram (i.e. R=0, G=0, B=0) is set to zero as black is used to cover the background and it has no effect on the food color pattern. The generated histograms of Figure 8(a) and Figure 8(b) are shown in Figure 9(a) and Figure 9(b) namely.

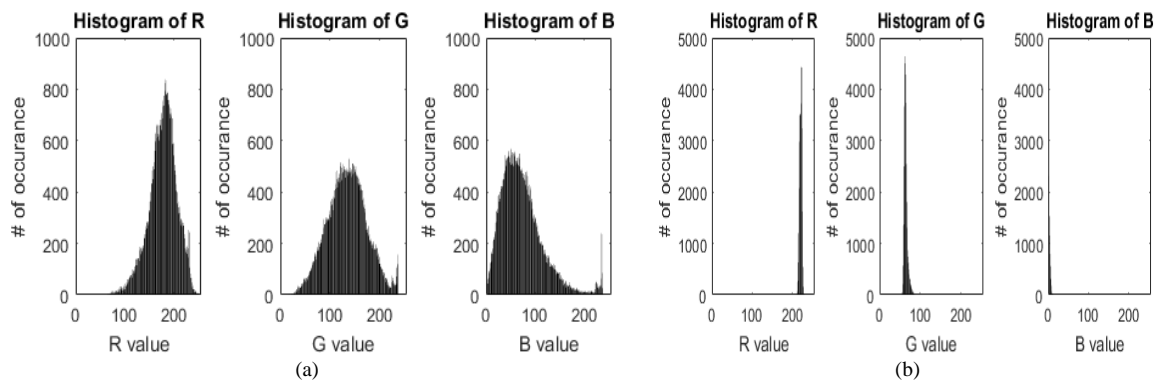


Figure 9. (a) Histogram of Figure 7(a); (b) Histogram of Figure 7(b)

2.1.5. Matching Score Calculation

In this step, the generated test histogram, h_{test} , is compared with the previously generated histograms stored in the database and matching scores are calculated. Let's consider that, a food was heated previously and its histogram, h_{comp} , is already stored in the database. Now, when the same food is heated again, the food can be in a different place on the plate, in a rotated position, scaled (i.e. different food quantity), or stirred compared with its previously stored histogram. The algorithm should consider all these transformations and should indicate they are the same food. The translation (i.e. different position in the plate) and the rotation of the food will not have much effect on the histogram because position information of pixels is not stored in histograms. The different stirring condition can cause some minor differences, but the overall color pattern will be similar. Figure 10(a) shows the same food as shown in Figure 5(a), however, the food is rotated and in a different stirring condition. Figure 10(b) shows the image after background removal and Figure 10(c) shows the histogram of Figure 10(b). If we compare the shapes of the histograms in Figure 9(a) with the histograms shown in Figure 10(c), we see that they are quite similar. This shows that the histograms the same food having a different translation, rotation and stirring condition will be similar. If there is scaling of the food, then one of the histograms is scaled before comparing. The smaller area histogram is scaled to the larger area histogram using the pseudocode as shown in Figure 11. This is done for

each color channel.

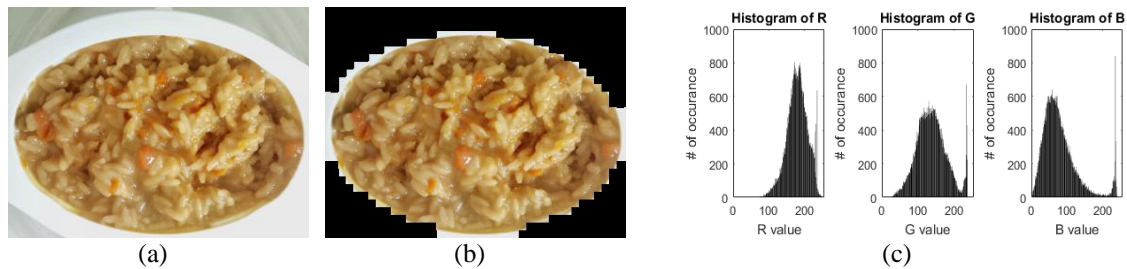


Figure 10. (a) The same food as shown in Figure 5(a), however, in a different rotation and stirring condition; (b) Background removed from Figure 10(a); (c) Histogram of Figure 10(b), which is quite similar to Figure 9(a)

```

1. Area_h_test := sum (h_test)
2. Area_h_comp := sum (h_comp)

3. If Area_h_test >= Area_h_comp Then
4.   h_comp := h_comp * (Area_h_test / Area_h_comp)
5. Else
6.   h_test := h_test * (Area_h_comp / Area_h_test)
7. End

```

Figure 11. Pseudocode for scaling histograms

After scaling, Euclidian distance is calculated between the histograms for each color channel using (1). Then the total matching score is calculated by adding the distances, d , for R, G, and B channel histograms. The smaller the distance (or matching score), the better the match.

$$d = \sqrt{\sum (h_{test} - h_{comp})^2} \quad (1)$$

2.1.6. Temperature Recommendation

Matching scores are calculated between the generated test histogram, h_{test} , and with the food item histograms stored in the database. The assigned temperature is looked up for the food item which has the minimum matching score. This temperature is recommended by the algorithm to the user. The user can then increase or decrease the recommended temperature if wants by push button switches or can accept the recommended temperature for the food. If there is no significant match between the generated test histogram, h_{test} , and with the histograms stored in the database, (i.e. the minimum matching score is greater than a threshold value, MAX_MS), then the food is considered as a *new* food item. In this case, a default temperature is recommended by the algorithm to the user. The user may change this temperature by push button switches. After the user press the start button on the microwave, the h_{test} and its assigned temperature is added to the database as a new food item. In this way, the algorithm gradually learns what temperature it should recommend for different foods used in that household. The algorithm also stores the date when a new item is created, how many times an item is used since its creation and the average time of the day when it is used.

2.2. Temperature Feedback Closed-Loop System

After the target food temperature is set, the proposed microwave oven continues to heat the food item until it reaches the target temperature. Once the target temperature is reached, the heating is turned off automatically. Every physical object radiates infrared (IR) waves proportional to its temperature. The food temperature is measured using a non-contact IR temperature sensor [26]. The sensor is mounted on the outer side of the microwave oven cavity roof through a 4-mm hole. The microwave has a wavelength of around 120 mm [27] and infrared has a wavelength in the range of 1 mm - 750 nm [28]. As the microwaves are long waves, they do not pass the 4-mm hole and do not damage the sensor. The IR waves can pass through the hole and thus the food temperature can be sensed.

3. PROTOTYPE IMPLEMENTATION

An embedded system is implemented in a commercial microwave oven [29] after disconnecting its mechanical timing mechanism. The hardware and the firmware part of the prototype are briefly described below.

3.1. Hardware Design

The block diagram of the hardware is shown in Figure 12. The single board computer, Raspberry Pi (RPi) v3 [30], is used as the processor. As image processing needs high memory and processing speed [31], the RPi is chosen over other microcontrollers such as Advanced Virtual RISC (AVR) and Peripheral Interface Controller (PIC) which has lesser memory capacity and speed. An RPi HD camera module [32] is mounted at the center of the roof of the microwave oven prototype through a 4-mm hole. To get a sharper image, the focus length is adjusted to the height of the oven cavity which is 15 cm. A non-contact IR temperature sensor [33] is mounted slightly away from the center of the roof of the oven through a 4-mm hole.

The sensor has a field of view (FOV) of 5 degrees and distance from the sensor to the turntable tray is 15 cm. Thus, the sensor gets the average temperature of a circular area having a radius of $15 \times \tan(5) = 1.3$ cm. When the turntable tray rotates, the sensor gets the temperature of different portions of the food as it is placed slightly away from the center of the oven cavity. The sensor is interfaced with the RPi using two wire I2C protocol. A 16x2 character liquid crystal display (LCD) [34] is interfaced with the RPi using universal asynchronous receiver-transmitter (UART) protocol. To interact with the user, the design contains four push-button keys labeled *Up*, *Down*, *Start*, and *Stop*. A buzzer is included in the design to generate beep sounds. One of the door switches of the oven is connected with an interrupt pin of RPi, and the other door switch is connected to the path of the AC circuit so that current can only flow when the door is closed. A daylight white LED bulb [35] is connected through the contacts of a single pole single through normally open (SPST-NO) relay [36]. A solid-state relay (SSR) [37] is used to turn on/off the magnetron circuit for generating microwave and the motor connected to the turntable tray. The power supply for the RPi board and the camera is supplied using a 110V AC to 5.1V DC adapter [38]. The RPi board generates 3.3 V DC and it is used to power the LCD, IR temperature sensor, and buzzer.

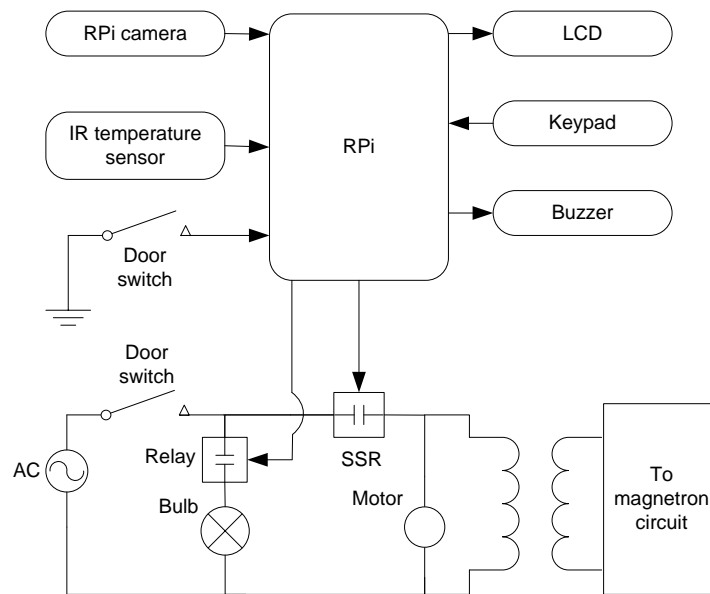


Figure 12. Block diagram of the hardware

3.2. Firmware Design

A Debian-based Linux operating system, Raspbian [30], is installed on a 16 GB SD card of the RPi board. The firmware for the proposed microwave is developed in Python language. The firmware is built on two layers-the driver layer and the application layer. The driver layer consists of low-level firmware for accessing different hardware peripherals. The application layer access the hardware by calling the functions of the driver layer. A pseudocode of the application layer is shown in Figure 13.

```

On DoorClose Event:
  BulbOn()
  I := CaptureImage()
  BulbOff

  C := ClassifyImage(I)
  If (C = NOT EMPTY)
    Tr := GetRequiredTemp(C)
    BulbOn()
    HeatOn()
    isFinish := False
    while (NOT isFinish)
      FIFO.add (ReadIRTemp())
      Tc := FIFO.avg ()
      DisplayLCD ()
      ButtonInput ()
      wait (SAMPLE_RATE_DELAY)
      is_food_ready := (Tc >= Tr)

      isFinish := is_food_ready OR is_stop_pressed OR is_door_opened
    BulbOff()
    HeatOff()

    if is_food_ready
      PlayBuzzer()

```

Figure 13. Pseudocode for application layer

4. RESULTS AND DISCUSSION

4.1. Simulation Results

The proposed image classification and temperature recommendation algorithm, as discussed in Sec. 2.1, has been implemented in MATLAB for simulation on a personal computer (PC). In the simulation, 263 images of solid food, 53 images of liquid food, and 10 empty images (i.e. when no food in the microwave) were used. All images were captured by the camera mounted on the top of the microwave. Different scenarios were created with different foods placed in different containers. Thirteen different types of solid foods and 3 plates having a color of white, blue, and red were used. Each solid food was placed on each type of the plates in turn. Then the orientation of the plate was changed and the food was stirred before putting it into the microwave so that the same food is in a different situation each time. Nine different types of drinks (i.e. liquid food) were also used in the test. Figure 14 shows some sample images of the foods used in the tests.

During the simulation, the target temperature for each type of food was assigned by generating normally distributed random numbers having a mean (μ) of 100 and standard deviation (σ) of 15. The accuracy of the image classification and temperature recommendation algorithm is shown in Table 1. The proposed algorithm can distinguish between an empty and a microwave oven with food with 100% accuracy. It can also classify whether the food is solid or liquid having an accuracy of 100%. Then the algorithm removes the background from the image, generate histograms, classify the image, and recommend a target temperature. It can recommend the exact target temperature with an accuracy of 86.31% for solid and 100% for liquid food. As the algorithm is learning, it needs to detect new food items. When the algorithm classifies a new food image as one of the food items that are already in the database, it is considered as incorrect detection of new food. Conversely, when the algorithm classifies a food image that is already in the database as a new food item, it is considered as incorrect detection of new food. The proposed algorithm can detect new food items having an accuracy of 89.35% for solid and 92.45% for liquid food.

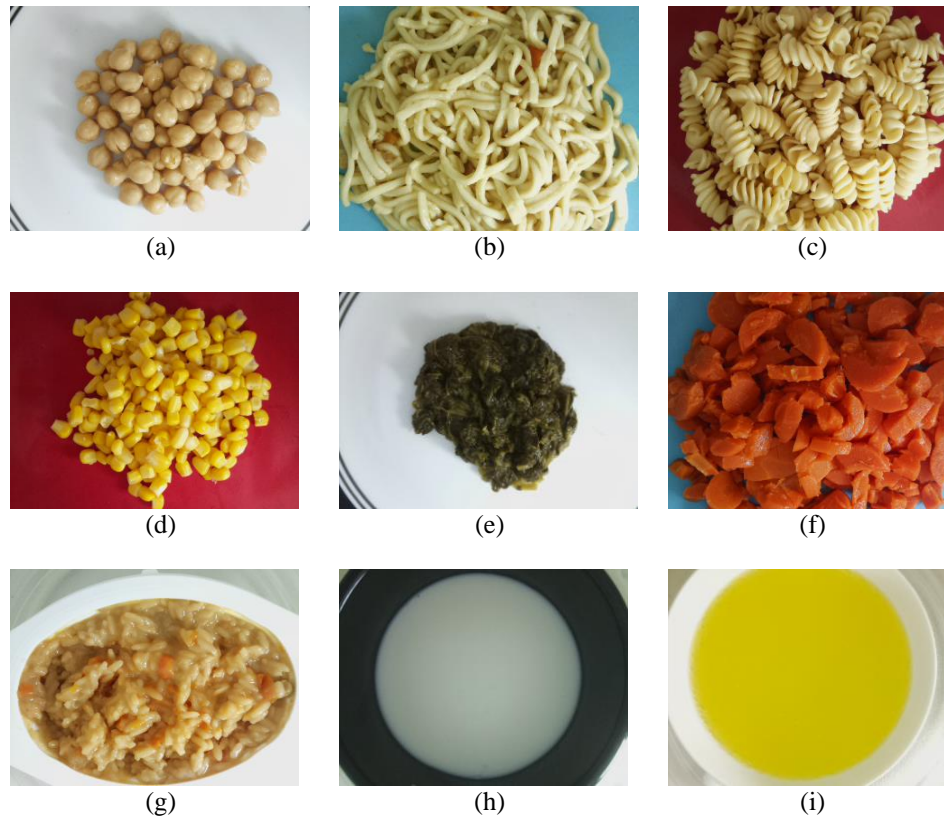


Figure 14. Sample food images in different containers. (a) Chick peas; (b) Noodles; (c) Pasta; (d) Corn; (e) Spinaches; (f) Carrot slices; (g) Rice with chicken; (h) Milk; (i) Pineapple drink

Table 1. Accuracy of Classification and Temperature Recommendation Algorithm

Task		Accuracy
Empty detection		100.00 %
Classifying food as solid or liquid		100.00 %
Temperature recommendation	Solid	86.31 %
	Liquid	100.00%
Detection of new food	Solid	89.35 %
	Liquid	92.45%

4.2. Prototype Results

The proposed microwave oven is prototyped in the laboratory and its photographs are shown in Figure 15. The classification algorithms as discussed in Sec 2 and the firmware as discussed in Sec 3.2 are written in Python and programmed to the Raspberry Pi board. The prototype can classify and recommend temperature having an accuracy similar to the simulation result in Table 1. After the target temperature for the food is recommended by the algorithm or edited by the user, the proposed microwave oven heats the food until the food's current temperature reaches the target temperature. It then stops the microwave oven automatically and plays buzzer to notify the user.

The microwave oven has an inherent property of non-uniform heating [14]. As the turntable tray rotates, the IR sensor samples temperature data from different areas of the food. The sampled data is put in a FIFO. The average value of the contents of the FIFO is used as the current temperature of the food. Thus, the average value can represent the overall food temperature even though the food is non-uniformly heated. Experiments with solid and liquid food have been conducted to verify the accuracy of the final food temperature. After the food is heated, the temperature of different portions of the food was measured. The average standard deviation of the temperature of different areas of the food was found to be 4.22 °F for solid food and 1.34 °F for liquid food. The accuracy of the final food temperature for solid food is 99.32% and liquid food is 99.46%.

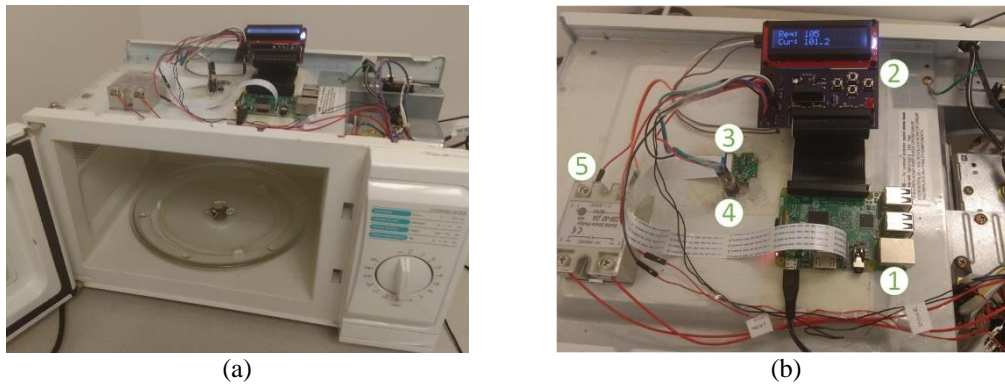


Figure 15. (a) Photograph of the prototype microwave oven; (b) Photograph of the designed hardware mounted on top of the microwave: 1. Raspberry Pi board, 2. Custom designed printed circuit board (PCB) containing LCD, keys, relay, buzzer, connectors etc., 3. HD camera, 4. IR temperature sensor, 5. SSR

4.3. Comparison with Other Works

A comparison with other related works is shown in Table 2. The works in [15] and [16] uses FLIR camera for contactless temperature sensing. However, (FLIR) camera is expensive [11] and it can significantly increase the price of the microwave. CCD camera-based temperature sensing is used in [18] to reduce the price. The work in [22] uses temperature measurement using RTD sensor requiring physical contact with the food. The sensor needs to be cleaned often and this is inconvenient. The proposed method uses non-contact IR temperature sensor. This sensor is much cheaper than FLIR and CCD camera. In [15], temperature and other information are transferred to a personal computer and power level of the microwave oven is controlled.

The proposed work does not need any external PC, rather the controlling is done using low cost embedded system [30]. The proposed work classify a food item by image processing and recommends the target temperature of the food based on its previous experience. The proposed algorithm also has the ability to learn new food items and store its target temperature in the database. Thus the microwave gradually becomes smarter in recommending the target temperature of food items that are used in that household. The work in [22] uses a feature called “Food ID” that automatically identifies a set of predefined foods using image processing. However, it cannot learn new food items and their target temperatures. It also uses a smartphone notification system whenever the food is ready. It should be noted that the work in [22] is an oven for cooking food, and it is not a microwave oven for warming up food. Cooking food in the oven may take hours and smartphone notification is well justified. However, warming up food using microwave generally do not take more than a few minutes, thus the user can stay near the microwave until the food is ready. So, a smartphone notification feature might be an overdesign for a microwave.

Table 2. Comparison with other works

Work	Contactless temperature sensing	Auto turn-off heating	Auto target temperature recommendation	Able to learn new food's target temperature	Smartphone notification
E. Belotserkovsky <i>et. al</i> [14]	Yes, IR fiberoptic radiometer	No	No	No	No
G. Cuccurullo <i>et. al</i> [15]	Yes, FLIR camera	Controls power level using a PC	No	No, only for heating apple slices	No
J. Bows <i>et. al</i> [16]	Yes, FLIR camera	No	No	No	No
C. Liyan <i>et. al</i> [18]	Yes, CCD camera	No	No	No	No
June Intelligent Oven [22]	No, RTD temperature sensor	Yes	Yes, for set of preprogrammed food	No	Yes
Proposed	Yes, IR temperature sensor	Yes, control using embedded system	Yes	Yes	No

4.4. Constraints

According to the background removal algorithm for liquid foods as discussed in Sec. 2.1.3, the food needs to be opaque, and should not be transparent. For instance, if transparent water is put on a cup, then the

proposed algorithm will not be able to distinguish between foreground (i.e. the water) and background (i.e. the cup). As water is transparent, the algorithm will generate a histogram based on the color of the cup. If water is warmed in that household on a few number of cups repeatedly, the proposed learning algorithm will learn the cup colors and can recommend the correct temperatures eventually. Another constraint for the background removal algorithm is that the food color needs to be different from its container color. For instance, if white milk is put on a white cup, then the proposed algorithm will not be able to differentiate between foreground (i.e. the food) and background (i.e. the container). In this case, the background will not be removed and the cup image will be considered as part of the milk image. This will increase the area or the apparent quantity of the food. However, this will have no effect on the accuracy of the algorithm as histograms are scaled as shown in Figure 10.

5. CONCLUSION

In this paper, a closed-loop microwave oven using non-contact IR temperature sensor is discussed. A novel food image classification and a target temperature recommendation algorithm are also presented. The proposed algorithm has the ability to learn, and it gradually becomes smarter in the recommendation. A hardware prototype of the proposed microwave oven has been developed and its accuracy has been tested.

REFERENCES

- [1] H. Sayuti, *et al.*, "Smart Home and Ambient Assisted Living Based on the Internet of Things," *International Journal of Electrical and Computer Engineering (IJECE) of Institute of Advanced Engineering and Science (IAES)*, vol/issue: 7(3), 2017.
- [2] C. Ryu and C. W. Hur, "A Monitoring System for Integrated Management of IoT-based Home Network," *International Journal of Electrical and Computer Engineering (IJECE) of Institute of Advanced Engineering and Science (IAES)*, vol/issue: 6(1), 2016.
- [3] I. Mukhlash, *et al.*, "Mining Fuzzy Time Interval Periodic Patterns in Smart Home Data," *International Journal of Electrical and Computer Engineering (IJECE) of Institute of Advanced Engineering and Science (IAES)*, vol/issue: 8(5), Part II, 2018.
- [4] N. Navya and G. Dumala, "Web Based Automated Smart House Management," *TELKOMNIKA Indonesian Journal of Electrical Engineering of Institute of Advanced Engineering and Science (IAES)*, vol/issue: 13(3), pp. 493-496, 2015.
- [5] H. Shahinzadeh and A. H. Khosroshahi, "Implementation of Smart Metering Systems: Challenges and Solutions," *TELKOMNIKA Indonesian Journal of Electrical Engineering of Institute of Advanced Engineering and Science (IAES)*, vol/issue: 12(7), pp. 5104-5109, 2014.
- [6] F. Karray, *et al.*, "WiRoTip: an IoT-based Wireless Sensor Network for Water Pipeline Monitoring," *International Journal of Electrical and Computer Engineering (IJECE) of Institute of Advanced Engineering and Science (IAES)*, vol/issue: 8(5), Part I, 2018.
- [7] H. P. Singh, *et al.*, "An Approach to Implement Cost Efficient Space Detection Technology with Lower Complexity for Smart Parking System," *TELKOMNIKA Indonesian Journal of Electrical Engineering of Institute of Advanced Engineering and Science (IAES)*, vol/issue: 15(3), 2015.
- [8] J. M. Osepchuk, "A History of Microwave Heating Applications," *IEEE Transactions on Microwave Theory and Techniques*, vol/issue: 32(9), pp. 1200-1224, 1984.
- [9] Total unit shipments of microwave ovens in the U.S. from 2005 to 2018, 2018. <https://www.statista.com/statistics/220122/unit-shipments-of-microwave-ovens/>
- [10] A. Williams, "Surveys of Microwave Ovens in U.S. Homes," *Lawrence Berkeley National Laboratory*, 2014. <https://escholarship.org/uc/item/3s29h7wd>
- [11] Americans are eating at home more; microwave usage increases but not cooking, *NPD*, 2018. https://www.npd.com/wps/portal/npd/us/news/press-releases/pr_091112/
- [12] Microwaves in the US, *Euromonitor*, 2017. <http://www.euromonitor.com/microwaves-in-the-us/report>.
- [13] M. H. Sun, *et al.*, "Fiber optic temperature sensors in the medical setting," *Proc. SPIE*, vol. 1067, pp. 15-21, 1989.
- [14] E. Belotserkovsky, *et al.*, "Infrared fiberoptic temperature control of the heating process in a microwave oven," *IEEE Transactions on Microwave Theory and Techniques*, vol/issue: 42(5), pp. 901-903, 1994.
- [15] G. Cuccurullo, *et al.*, "Infrared thermography assisted control for apples microwave drying," *Journal of Food Engineering*, vol/issue: 112(2012), pp. 319-325, 2012.
- [16] J. Bows and K. Joshi, "Infrared imaging feels the heat in microwave oven," *Physics World*, vol/issue: 5(8), pp. 21-22, 1992.
- [17] FLiR Dev Kit, 2018. <https://www.sparkfun.com/products/13233>.
- [18] C. Liyan, *et al.*, "Temperature monitoring based on image processing for intelligent microwave heating," *The 27th Chinese Control and Decision Conference (2015 CCDC)*, Qingdao, China, pp. 1397-1401, 2015.
- [19] I. Sanchez, *et al.*, "Temperature control in microwave combination ovens," *Journal of Food Engineering*, vol/issue: 46(2000), pp. 21-29, 2000.
- [20] J. M. Cabrera, *et al.*, "Feedback control procedure for energy efficiency optimization of microwave-heating ovens," *Measurement*, vol/issue: 42(2009), pp. 1257-1262, 2009.

- [21] Z. Li, *et al.*, "Optimal power control strategies in microwave drying," *Journal of Food Engineering*, vol/issue: 99(2010), pp. 263–268, 2010.
- [22] June Intelligent Oven, 2018. <https://juneoven.com>.
- [23] W. A. Albukhanajer, *et al.*, "Evolutionary Multiobjective Image Feature Extraction in the Presence of Noise," *IEEE Transactions on Cybernetics*, vol/issue: 45(9), pp. 1757-1768, 2015.
- [24] M. Mackiewicz, *et al.*, "Wireless Capsule Endoscopy Color Video Segmentation," *IEEE Transactions on Medical Imaging*, vol/issue: 27(12), pp. 1769-1781, 2008.
- [25] I. Marques and M. Grana, "Face Recognition Algorithms," *Universidad del Pais Vasco - Euskal Herriko Unibersitatea*, 2010.
- [26] MLX90614 infrared thermometer, 2018. <https://www.melexis.com/en/product/MLX90614/Digital-Plug-Play-Infrared-Thermometer-TO-Can>.
- [27] Microwave Oven, 2018. <http://hyperphysics.phy-astr.gsu.edu/hbase/Waves/mwoven.html>.
- [28] Infrared, 2018. <http://hyperphysics.phy-astr.gsu.edu/hbase/ems3.html>.
- [29] Westinghouse Microwave oven, Model WMW-17668, 2018. <http://westinghouse.com>.
- [30] Raspberry Pi, 2018. <https://www.raspberrypi.org>.
- [31] T. H. Khan and K. Wahid, "Universal bridge interface for DVP-compatible image sensors," *Microprocessors and Microsystems, Elsevier*, vol/issue: 35(6), pp. 547-556, 2011.
- [32] Raspberry Pi camera module, 2017. <https://www.raspberrypi.org/products/camera-module-v2>.
- [33] MLX90614 infrared thermometer, 2018. <https://www.melexis.com/en/product/MLX90614/Digital-Plug-Play-Infrared-Thermometer-TO-Can>.
- [34] Serial Enabled 16x2 LCD, 2018. <https://www.sparkfun.com/products/9067>.
- [35] Daylight White E17 LED Bulb, 2018. <https://www.amazon.com/dp/B01KX765TE>.
- [36] SPST-NO Relay, 2018. <https://www.digikey.com/product-detail/en/panasonic-electric-works/APAN3103/255-5322-5-ND/6009005>.
- [37] Solid State Relay, 2018. <https://www.sparkfun.com/products/13015>.
- [38] DC Power Supply, 2018. <https://www.sparkfun.com/products/13831>.

BIOGRAPHIES OF AUTHORS



Dr. Tareq Khan received his Ph.D. degree from the department of Electrical and Computer Engineering of University of Saskatchewan, Canada. Dr. Khan is now an Assistant Professor in the School of Engineering Technology of Eastern Michigan University, USA. To date, Dr. Khan has authored (and co-authored) 2 books, 1 book chapter, 18 peer-reviewed journals and 26 international conference papers. He currently has 2 US patents granted. In addition to his academic research, he has also industrial experiences on embedded system projects such as designing pre-paid electricity and gas metering system, automatic meter reading (AMR), data accusation and monitoring systems etc. His research interests include smart home, embedded systems targeting healthcare applications, Internet of Things (IoT), machine learning, capsule endoscopy, and image processing. He is a member of the Institute of Electrical and Electronics Engineers (IEEE).