IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received 3 December 2022, accepted 12 December 2022, date of publication 15 December 2022,
date of current version 21 December 2022.

Digital Object Identifier 10.1109/ACCESS.2022.3229895

RESEARCH ARTICLE

Real-Time Occupancy Detection System Using Low-Resolution Thermopile Array Sensor for Indoor Environment

B. SHUBHA[1,2] AND V. VEENA DEVI SHASTRIMATH[1,2], (Member, IEEE)
[1]Department of Electronics and Communication Engineering, NMAM Institute of Technology, Nitte, Karkala, Karnataka 574110, India
[2]Visvesvaraya Technological University, Belagavi, Karnataka 590018, India

Corresponding author: B. Shubha (shubhab@nitte.edu.in)

ABSTRACT Low-Resolution Thermopile Array Sensors are widely used in several indoor applications such as security, intelligent surveillance, robotics, military, and health monitoring systems. It is compact, cost-effective, and offers a low-resolution thermal image of the environment, attracting its use in privacy-focused applications. Many industries migrating towards Industry 4.0 are facing challenges in using sensors and automating the systems. One of the areas in which automation could be implemented is by using sensors to operate the systems smartly based on occupancy. The major challenge in such applications is maintaining privacy; conventional imaging mechanisms using optical camera systems fail to achieve it. The same could be achieved by using thermopile sensors which provide thermal data of the desired region. This generates the possibility to identify the number of people in a specified area without revealing their identity. This paper proposes various approaches to detect human occupancy using a low-resolution infrared thermopile array sensor to keep their identity safe and avoid privacy issues. The proposed system detects IR-emitting objects using a low-resolution thermopile array Grid-EYE sensor (AMG8833). The sensor acquires 8×8 pixels of thermal distribution. These thermal distribution data are subjected to interpolation, filtering, adaptive thresholding, and background suppression to attain the set goal of human detection.

INDEX TERMS Thermopile array sensor, human target detection, bicubic interpolation, Gaussian filter, adaptive threshold, Raspberry Pi.

I. INTRODUCTION

The occupancy detection system is vital in a wide variety of applications in any indoor environment. For example, the temperature and lighting load of a room could be estimated and controlled based on the occupancy data to save energy [1], [2], [3]. It can also be used in the home, libraries, museums, and offices to monitor the movement activities of people. This helps in minimizing the use of space facilities and building resources. Human detection could be managed with many portable devices like wearable sensors [4], optical cameras [5], Radio Frequency Identification Devices (RFIDs) [6], and WLAN devices [7]. However, wearable sensors are not convenient for users. Moreover, cameras are not only affected by lighting conditions but also pose identity privacy issues [8]. In RFID and WLAN-based localization, users need to carry the receiving devices with them. This paper addresses these challenges by proposing thermal sensors for human detection to keep people's personal identities private. In these sensors, thermal values of a region are converted into electrical signals [9], [10]. Human detection could be done by processing and analyzing the received signals. These cost-effective thermal sensors are also gaining popularity in surveillance and smart housing due to their accuracy and performance [11].

The human target detection is one of the most essential and desirable information in real world applications. This could be achieved by using multiple Pyroelectric Infrared (PIR) sensors [12], [13] and sensors need to be placed in different places at different angles. This conventional method is easy to implement, however poses the limitation like – detection is possible only when the target is in motion.

The associate editor coordinating the review of this manuscript and approving it for publication was Zhongyi Guo.

A thermopile thermal array sensor could be an alternative to address the limitation as it can detect both static and moving objects. The detectivity and responsivity of the thermopile array sensors are high and have a better signal-to-noise ratio, which makes them more suitable for temperature measurements in different applications compared to the pyroelectric sensors [14], [15].

The advancement in the human detection field used thermopile sensors to detect a person [16] for applications like counting at doors and pedestrian detection using robots. These efforts could not address the limitations like object detection with background removal. To enhance the detection capability, a greater number of thermopile sensors are used [17] to identify and detect the object locations. To remove the image background fuzzy techniques are tried and results are suffered due to poor thermal image.

The probabilistic method using Grid-EYE sensor [18] provided promising results in the form of human detection but posed limitation due to the object size. The Grid-EYE sensor-based detection work is being extended to tracking [19] and also energy-related activities are recognized in the workplace [20]. The application of Grid-EYE sensor is observed in the applications like – detection of finger movements [21], detect and track people within a FOV of 2.5m $\times$ 2.5m [22] and passenger detection inside the car [23]. All the work carried out using Grid-EYE sensor involves minimal post processing of the data and very less real time implementations.

Various applications like - tracking of fall detection of elderly people using k-Nearest Neighbor [24] using 16 $\times$ 4 thermopile array, detection of human posture like walking, lying, sitting [25] using 16 $\times$ 16 thermopile array sensors and tracking of person walking on two polygonal paths with a high-resolution 24 $\times$ 32 [26] thermopile sensor. But there is a distortion in the edges of the detection area due to the large angle of view of the sensor. People have worked to improve the detection area by employing more sensors [27] by sacrificing on the system cost.

In this paper, the proposed method uses a single thermopile array sensor (Grid-EYE) to detect multiple objects in real time. The sensor data is collected and processed in real time to detect the occupancy of the object. In this paper, effective human detection using a thermal sensor is achieved by interfacing the thermal sensor to a Raspberry Pi and implementing signal processing algorithms in real time. An initial simulation study is carried out by using a single thermopile array sensor to detect multiple targets in the detection area using MATLAB. Later, using the Raspberry Pi and Python programming, the same is implemented in real-time. A comparison of different interpolation and filtering techniques is carried out to select the most suitable method for the detection of the target. The computation time taken for detection of the target using simulation in MATLAB and real time implementation is compared. The main focus of this paper is to achieve real time implementation of human detection using thermopile sensors. The effort is towards achieving computation time of signal processing algorithm computed in Raspberry Pi to be less than the frame rate of sensor.

The rest of this paper is ordered as follows. Section II presents the specifications of the thermopile array sensor (Grid-EYE sensor), the details of the Grid-EYE sensor evaluation kit used, and the specifications of Raspberry Pi. Section III introduces the method used to detect stationary/multiple humans. Section IV presents the experimental results of human target detection in real time. Section V is the conclusion part.

II. THERMOPILE ARRAY SENSOR

The infrared radiation of the target is captivated by the thermal infrared sensors to detect the temperature variation and translate this temperature variation into an electrical signal. Many thermocouples are connected in series in the thermopile thermal detectors, and they work on the principle of the Seebeck Effect [13]. The thermopile array sensors receive infrared radiation emitted by humans and detect both moving and stationary objects by giving the thermal values of the target. The thermal values are received from the thermopile array sensor (Grid-EYE-AMG8834) by using the Grid-EYE sensor evaluation kit as shown in Figure 1 [28].

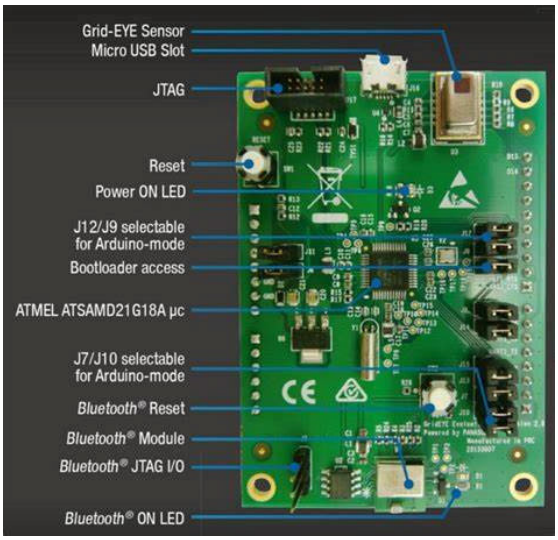

FIGURE 1. Grid-EYE infrared array sensor evaluation kit [28].

A component AMG8834 Grid-EYE sensor is implanted in the evaluation kit. The Grid-EYE sensor is interfaced with the microcontroller (ATSAMD21G18A) through the I^2C protocol. The kit is powered through a micro USB port. The processed temperature values in the microcontroller are fed to the Personnel Computer (PC) through the USB protocol. The connection between Grid-EYE, the microcontroller, and the PC is shown in Figure 2. Specifications of the Grid-EYE sensor are as in Table 1.

The Grid-EYE sensor gives a thermal image of size 8 $\times$ 8. Figure 3 shows the thermal sensor output, where the human is present in the detection area. In this image, red and blue colors signify high and low temperature respectively.

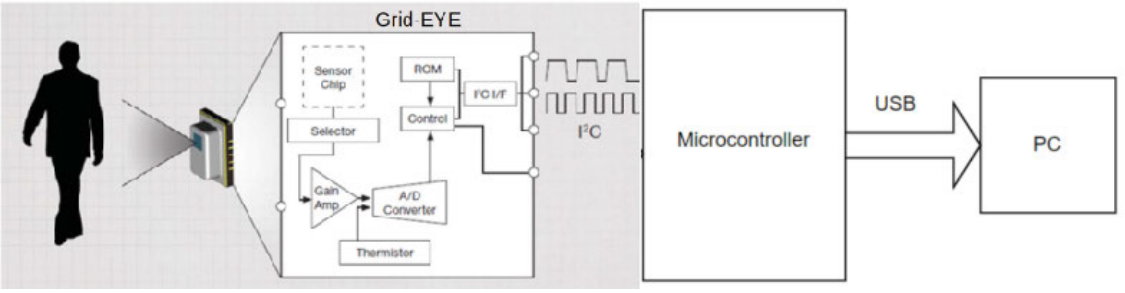

FIGURE 2. Connection block diagram.

TABLE 1. Specifications of grid eye sensor [29].

Parameter	Details
Number of Pixels	64(8 rows x 8 columns)
Temperature range of targets	-20^0C~100^0C
Absolute temperature accuracy	$\pm 3^0C$
Range of Detection	7m
Acquisition Frequency	1 frame/s or 10 frames/s
Field of View (FoV)	60^0C

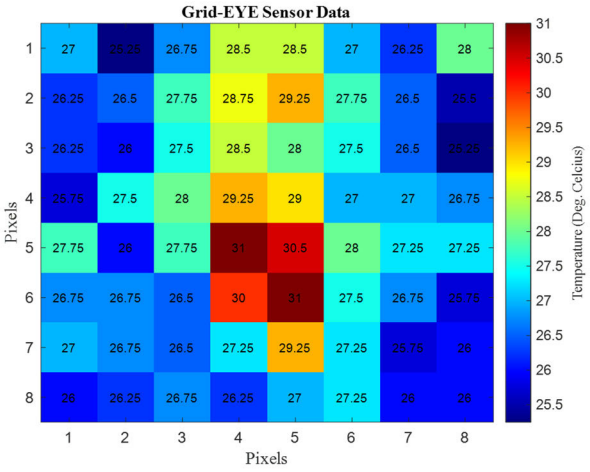

FIGURE 3. Grid-EYE sensor data.

The histogram of the thermal image with the human is shown in Figure 4. It is observed from the figure that, the number of occurrences of a particular temperature in a room [30].

A. AREA OF TARGET IN THERMAL IMAGE

The 64 (8×8) pixels received from the Grid-EYE sensor are used for further data processing. If the range between the Grid-EYE sensor and the target increases, the sensor covers more detection area. A target covered in the detection area is shown in the output image of the Grid-EYE sensor with some high-temperature pixel values as shown in Figure 5.

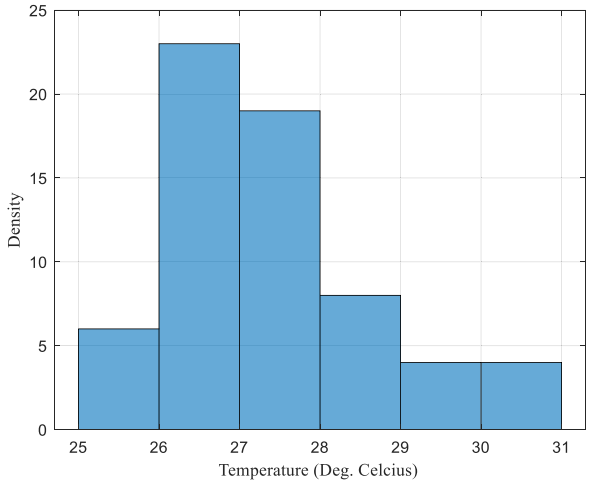

FIGURE 4. Histogram of thermal image.

The thermopile array consists of 64 sensing elements and each element has a FoV of 7.5°.

Figure 5 also shows the area enclosed by each pixel for the different distances between the object and the sensor. The heat target occupies more pixels in the thermal image when the target is very close to the sensor. Similarly, if the sensor is kept at a far distance (5m), it can cover a human within a pixel. The human presence in the output image diminishes as the human moves away from the sensor.

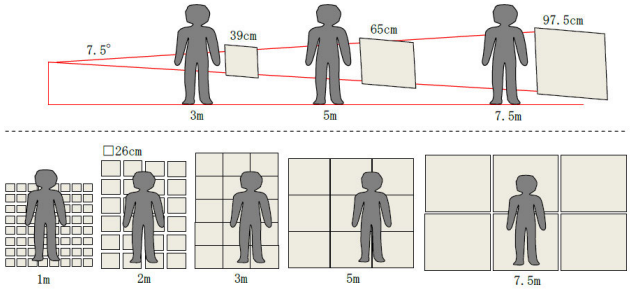

FIGURE 5. Distance versus object size [29].

Figure 6 shows a number of heat pixels indicating the presence of the target in an interpolated thermal image (100×100). The number of heat pixels in the thermal image

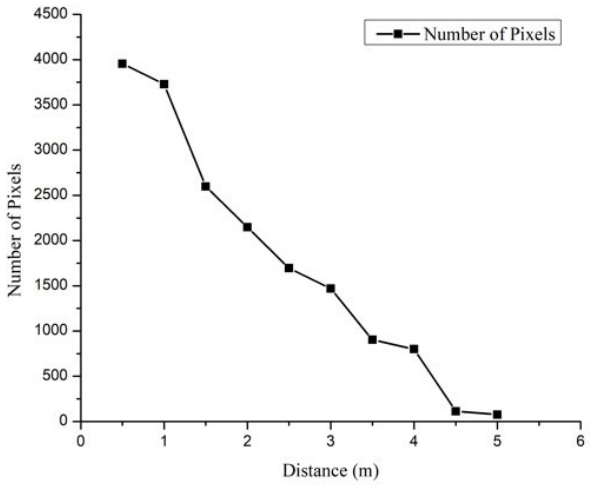

FIGURE 6. Detected heat pixels versus distance.

decreases as the target moves away from the sensor. This feature is then used to estimate the location of the human in the FoV [31]. From this analysis, one can keep the sensor at suitable distance depend on their application.

B. TEMPERATURE OF TARGET

The different temperature values measured from the thermopile sensor by varying the range between the human and the sensor is shown in Figure 7. The distance between the sensor and the human varied from 0.5 to 5m at the step of 0.5m. It is observed that as the range between the human and the sensor increases, the measured thermal values of the target decrease. From this, it could be concluded that there is a limit on the range of the sensor for detection [32].

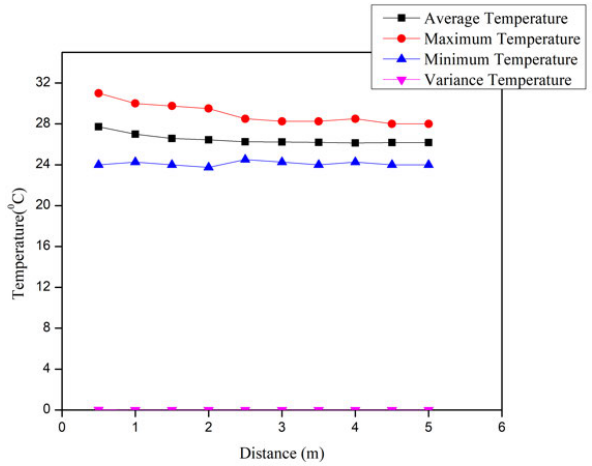

FIGURE 7. Target temperature versus distance.

C. FIELD OF VIEW OF GRID-EYE SENSOR

The detection area and FoV of the Grid-EYE sensor are shown in Figure 8. The sensor was placed horizontally and covers the square detection area below it.

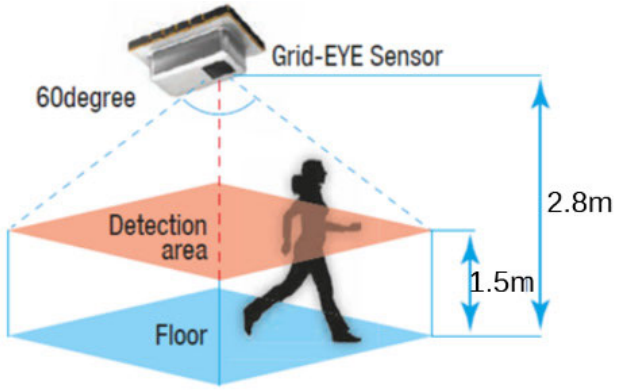

FIGURE 8. Field of view and detection area of Grid- EYE [29].

The larger detection area can be covered, if the range between the sensor and the floor is increased. The detection area range, R is calculated based on the geometry as:

$$R = 2 * X * \tan\left(\frac{\theta}{2}\right) \tag{1}$$

where, X is the distance between the sensor and the object and θ is the FoV. In this study, the sensor was kept at the height of 2.8m on the ceiling, and the detection area on the floor is $9m^2$.

D. RASPBERRY PI

In this paper, for real-time detection of human the Grid-EYE sensor is interfaced to Raspberry Pi. Raspberry Pi will act as a mini cheap PC. It is mainly used for real time image/video processing. Signal processing algorithms are implemented in Raspberry Pi to process the received data frames from the sensor. The different components of the Raspberry Pi 4 are shown in Figure 9 [33].

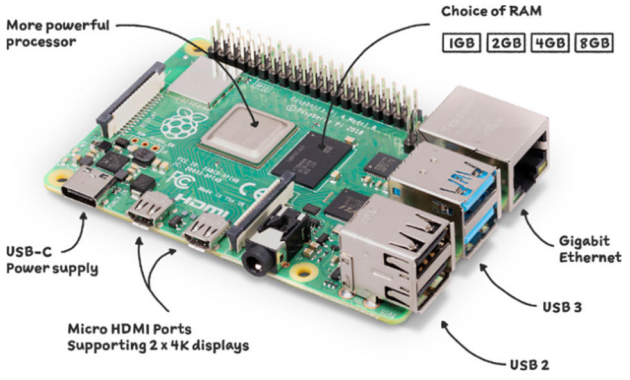

FIGURE 9. Raspberry Pi 4 [33].

The specifications of Raspberry Pi 4 are listed in Table 2.

III. STATIONARY/MOVING MULTIPLE HUMAN DETECTION SYSTEM

The steps involved to detect the stationary multiple humans in the FoV of the sensor are shown in Figure 10. The thermal values are collected from the infrared thermopile array sensor

TABLE 2. Specifications of raspberry Pi [33].

Parameter	Details
CPU	64 BIT Quad-core ARM Cortex-A72 (1.5GHz)
RAM	4GB
USB Ports	2 USB 3.0 2 USB 2.0
Networking	Gigabit Ethernet

(8 × 8 frame) and these temperature values are interpolated to get 100 × 100 frame. The interpolated frame is subtracted from the interpolated background temperature. The result is further smoothed by using a Gaussian filter. The smoothed frame is used for the target detection using adaptive thresholding.

A. DATA FROM GRID-EYE SENSOR

The thermal values collected from the thermopile sensor are displayed in the form of an array of 8 × 8 pixels when human targets move in the detection area of the Grid-EYE sensor. The Grid-EYE sensor thermal values are collected each time at the rate of 10 frames/sec through the serial port to the PC. These 12-bit thermal values are converted to degree Celsius.

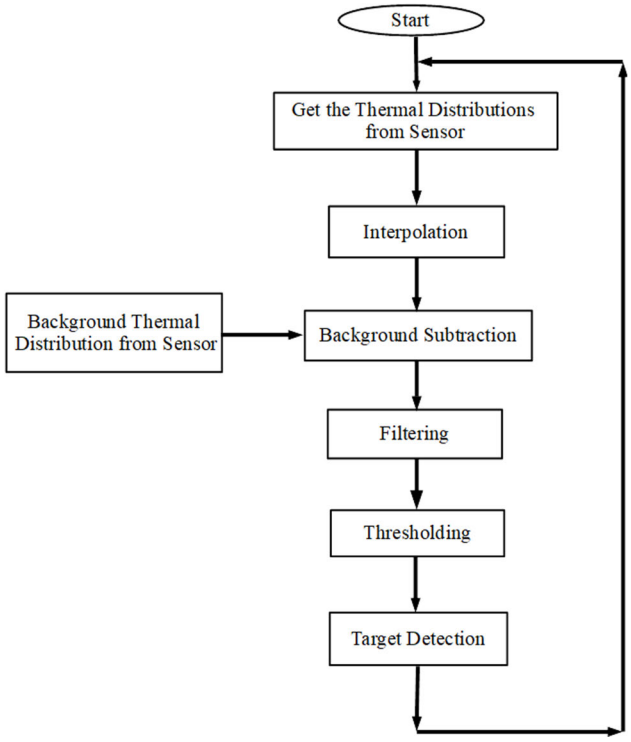

FIGURE 10. Flow chart of the detection of human targets.

B. INTERPOLATION

The Grid-EYE sensor output has an 8 × 8 resolution and it consists of 64 pixels, which is very low resolution for human detection. It is required to increase the resolution of the original thermal image. Interpolation of an image increases the resolution of the image without losing the quality of the image. Initially, the Grid-EYE sensor output thermal image is interpolated, and the thermal image of resolution 100 × 100 is obtained. Different interpolation techniques like Nearest Neighbor, Bilinear, and Bicubic Interpolation are compared and the suitable method of interpolation is selected for the human detection.

1) NEAREST NEIGHBOR INTERPOLATION

In this interpolation technique, the new interpolated pixel value is selected from the very nearby surrounding coordinates of it. The nearest neighbor interpolation just copies the existing values. The nearest-neighbor interpolation kernel is [34]:

$$f(x) = \begin{cases} 0x < 0 \\ 1x > 0 \end{cases} \tag{2}$$

where, x is the distance between interpolated pixel and nearby pixel points. For a two-dimensional image, let interpolated pixel is $A = f(x, y)$. The neighboring pixels be at (i, j), $(i, j+1)$, $(i+1, j)$, $(i+1, j+1)$.

The distance between (x, y) and (i, j), $(i, j+1)$, $(i+1, j)$ and $(i+1, j+1)$ are calculated, then the values of (x, y) is set as the value of the point which is nearest to (x, y) [32]. The distance D, between the two pixels $f(i, j)$ and $f(x, y)$ are calculated:

$$D = \sqrt{(x-i)^2 - (y-j)^2} \tag{3}$$

The actual image of size of 2 × 2 is shown in Figure 11a. Each pixel in the actual image is interpolated to 4 pixels and the final interpolated image has sixteen pixels as in Figure 11b. Both the images should have similar characteristics.

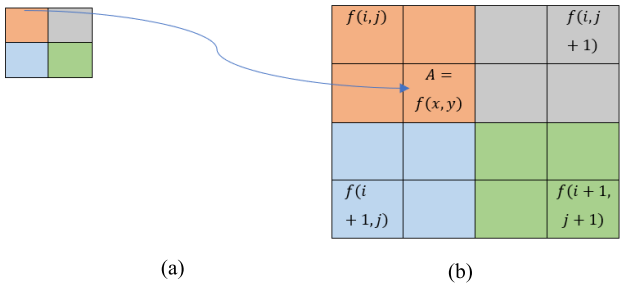

FIGURE 11. a. Actual image b. Interpolated image.

2) BILINEAR INTERPOLATION

In this, the new interpolated pixel value is calculated from the weighted mean value of the 4 nearest pixels. It uses three linear interpolations, two in the horizontal direction and one in the vertical direction. The bilinear interpolation

kernel is [34]:

$$f(x) = \begin{cases} 0 & |x| > 1 \\ 1 - |x|, & |x| < 1 \end{cases} \tag{4}$$

where, x is the range between interpolated pixel and other pixel grid point.

In a two-dimensional image, let P be the pixel to be interpolated and Q_1, Q_2, Q_3, Q_4 are the four points closer to P, as illustrated in Figure 12.

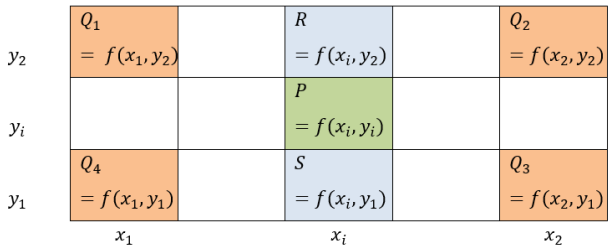

FIGURE 12. Bilinear interpolation.

Initially the point R is calculated by doing the linear interpolation in the horizontal direction taking reference of Q_1 and Q_2 points [35]. This gives:

$$R = f(x_i, y_2) = \frac{(x_2 - x_i)}{(x_2 - x_1)} Q_1 + \frac{(x_1 - x_i)}{(x_2 - x_1)} Q_2 \tag{5}$$

The point S is calculated by doing the linear interpolation in the horizontal direction taking reference of Q_3 and Q_4 points. This gives:

$$S = f(x_i, y_1) = \frac{(x_i - x_1)}{(x_2 - x_1)} Q_3 + \frac{(x_2 - x_i)}{(x_2 - x_1)} Q_4 \tag{6}$$

Finally, the value P is calculated by doing the linear interpolation in the vertical direction taking reference of P and S points. This gives:

$$P = B_l(x_i, y_i) = \frac{(y_2 - y_i)}{(y_2 - y_1)} S + \frac{(y_i - y_1)}{(y_2 - y_1)} R \tag{7}$$

$$\begin{aligned} P = B_l(x_i, y_i) = {} & \frac{1}{(y_2 - y_1)(x_2 - x_1)} \\ & \times \{(x_2 - x_i)(y_i - y_1) Q_1 + (x_i - x_1)(y_i - y_1) Q_2 \\ & + (x_i - x_1)(y_2 - y_i) Q_3 \\ & + (x_2 - x_i)(y_2 - y_i) Q_4\} \end{aligned} \tag{8}$$

Here $Q_1 = f(x_1, y_2)$, $Q_2 = f(x_2, y_2)$, $Q_3 = f(x_2, y_1)$ and $Q_4 = f(x_1, y_1)$.

3) BICUBIC INTERPOLATION

In Bicubic interpolation, the interpolated value is calculated by using the weighted average of 4×4 (16) neighbor's pixels. The distance between these calculated and unknown pixels in an image are different. The bicubic interpolation kernel is given as [36]:

$$f(x) = \begin{cases} \frac{3}{2}|x|^3 - \frac{5}{2}|x|^2 + 1 & 0 \le |x| < 1 \\ -\frac{1}{2}|x|^3 + \frac{5}{2}|x|^2 - 4|x| + 2 & 1 \le |x| < 2 \\ 0 & 2 < |x| \end{cases} \tag{9}$$

where, x is the distance between grid point and interpolated point. For two-dimensional image, it uses the third order polynomial to evaluate the value of image F on the subpixel (r_f, c_f) [37]. A Bicubic interpolation is defined by:

$$B_c(x, y) = \sum_{m=0}^{3} \sum_{n=0}^{3} \alpha_{mn} x^m y^n \tag{10}$$

To calculate the value of pixel at location (r_f, c_f), the 16 neighbor coefficients α_{mn} are measured from the values of the image and 4-pixel location $[F(r, c), F(r+1, c), F(r, c+1), F(r+1, c+1)]$ derivatives at the surrounding of (r_f, c_f) shown in Figure13.

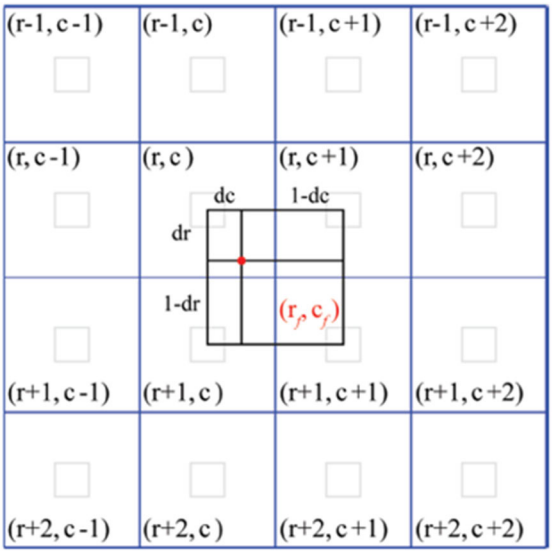

FIGURE 13. Bicubic interpolation [37].

The calculation steps for Bicubic interpolation as follows:

1) The original image data is taken and is interpolated to generate smooth image. The interpolation is done along row, column and diagonals.
2) Let F be the image with size $R \times C$, a new image S is created from image F with size $R' \times C'$ with pixel positions $x \times y$, where $x = 1, 2, \ldots .R'$ and $y = 1, 2, \ldots .C'$.
3) The pixel values of the output image $S(r', c')$ is determined by the neighboring pixels (4 values), that is by the $F(r, c)$, $F(r+1, c)$, $F(r, c+1)$ and $F(r+1, c+1)$.
4) The partial derivatives of each these 4 neighboring pixels along the row and along the column is $\frac{\partial I}{\partial r} = F_r$ and $\frac{\partial I}{\partial c} = F_c$

5) The cross derivatives of each these 4 neighboring pixels along the diagonal is $\frac{\partial^2 I}{\partial r \partial c} = F_{rc}$. Thus, the calculation involves 16 values in the case of the Bicubic Interpolation that is for each of the neighboring 4-pixel values, their partial derivatives along row, column and finally its cross derivatives (along diagonals).
6) The cross derivatives of each these 4 neighboring pixels along the diagonal is $\frac{\partial^2 I}{\partial r \partial c} = F_{rc}$. Thus, the calculation involves 16 values in the case of the Bicubic Interpolation that is for each of the neighboring 4-pixel values, their partial derivatives along row, column and finally its cross derivatives (along diagonals).
For example, consider pixel $F(r, c)$:

$$B_c(r, c) = F(r, c) \tag{11}$$
$$\frac{\partial B_c(r, c)}{\partial r} = \frac{\partial F(r, c)}{\partial r} \tag{12}$$
$$\frac{\partial B_c(r, c)}{\partial c} = \frac{\partial F(r, c)}{\partial c} \tag{13}$$
$$\frac{\partial^2 B_c(r, c)}{\partial r \partial c} = \frac{\partial^2 F(r, c)}{\partial r \partial c} \tag{14}$$

where,

$$\frac{\partial B_c(r, c)}{\partial r} = \sum_{m=0}^{3}\sum_{n=0}^{3} \alpha_{mn} m r^{m-1} c^n \tag{15}$$
$$\frac{\partial B_c(r, c)}{\partial c} = \sum_{m=0}^{3}\sum_{n=0}^{3} \alpha_{mn} n r^{m} c^{n-1} \tag{16}$$
$$\frac{\partial^2 B_c(r, c)}{\partial r \partial c} = \sum_{m=0}^{3}\sum_{n=0}^{3} \alpha_{mn} m n r^{m-1} c^{n-1} \tag{17}$$
$$F_r = \frac{\partial F(r, c)}{\partial r} = \frac{F(r+1, c) - F(r-1, c)}{2} \tag{18}$$
$$F_c = \frac{\partial F(r, c)}{\partial c} = \frac{F(r, c+1) - F(r, c-1)}{2} \tag{19}$$
$$F_{rc} = \frac{\partial^2 F(r, c)}{\partial r \partial c} = \frac{F(r+1, c+1) - F(r+1, c-1)}{2} + \frac{F(r-1)(c-1) - F(r-1)(c+1)}{2} \tag{20}$$

7) Similarly, the function values and its derivatives at other 3-pixel locations $[F(r, c+1), F(r+1, c), F(r+1, c+1)]$ are calculated.
8) The $[r, c]$ takes the continues values over the interval $[0 \times 1]$ and the values of the source image $F(r, c), F(r+1, c), F(r, c+1), F(r+1, c+1)$ are redefined as $F(0, 0), F(1, 0), F(0, 1), F(1, 1)$.
9) The above equations are reformulated into a matrix for the linear equation $AX = B$. Where,

$$X = [\alpha_{00}\ \alpha_{10}\ \alpha_{20}\ \alpha_{30}\ \alpha_{01}\ \alpha_{11}\ \alpha_{21}\ \alpha_{31}\ \alpha_{02}\ \alpha_{12}\ \alpha_{22}\ \alpha_{32}\ \alpha_{03}\ \alpha_{13}\ \alpha_{23}\ \alpha_{33}]^T \tag{21}$$

$$B = [F(0, 0), F(1, 0), F(0, 1), F(1, 1), F_r(0, 0) \ldots\ldots F_c(0, 0) \ldots\ldots F_{rc}(0, 0)]^T \tag{22}$$

10) The 16 coefficients are given by:

$$X = A^{-1}B \tag{23}$$

11) Using these 16 coefficients the new interpolated image is obtained by using the formula:

$$S(x, y) = \sum_{m=0}^{3}\sum_{n=0}^{3} \alpha_{mn} x^m y^n \tag{24}$$

The experimental observations and comparison of the different interpolation techniques are discussed in the Section IV. It is found that the Bicubic Interpolation technique produces better results in the enhancement of thermal image resolution.

C. BACKGROUND SUBTRACTION

Background frames are collected and mean temperature value of the background frames are calculated. The mean temperature value of the background frames is subtracted from the foreground frames as,

$$B(i, j) = F(i, j) - \frac{1}{N}\sum_{k=1}^{N} B^k(i, j) \tag{25}$$

$B(i, j)$, is the background subtracted image. $F(i, j)$, is the thermal image with human. $B^k(i, j)$, is the background image without human beings. N, is the number of frames of the image. Background subtraction helps in removing all the static noises. The human presence in the detection area will have the peak temperature value in the foreground image [38], [39], [40].

D. FILTERING

The Grid-EYE sensor is susceptible to noises from the external environment. The image is processed with filtering techniques to enhance the image and to suppress noise. In image processing, the spatial domain refers to the image plane itself, where the operations are performed directly on the pixels of the input image [41]. In the filtering process, any pixel value in the output image is calculated by using a predefined relationship to the neighboring pixel values of the input pixel. The process uses a filter mask which is moved on the image from point to point. Spatial domain operation is being expressed as:

$$P(p, q) = T[f(p, q)] \tag{26}$$

where the input image is $f(p, q)$, the output image is $P(p, q)$ and T is the operator on the input image f defined by its neighboring points. By using smoothing spatial filters, the images are blurred and noises are removed. The blurring process is applied to the image before the object removal. In the blurring process, small details from an image are removed and it connects the small gaps between the lines or curves. The different types of Spatial smoothing filters are Mean

Filter, Gaussian Filter, and Median Filter. In this paper, different filtering techniques are compared and Gaussian filtering method is selected as most suitable for this application. The experimental comparison of the different filtering techniques is done in Section IV.

The Mean and Gaussian Filters is a Linear Spatial Filter. Linear spatial filtering of an input image *MxN* with filter mask *w* of size *mxn* is:

$$P(p,q) = \sum_{s=-a}^{a} \sum_{t=-b}^{b} w(s,t) f(p+s, q+t) \tag{27}$$

The p and q are varied so that every pixel in w overlaps each pixel in f. But some pixels do not overlap. So, the padding of zeros on either side f makes each pixel in w to overlap with each pixel in f. If the filter mask is of size m, $m-1$ zeros are padded on either side of f.

1) GAUSSIAN FILTER

By using 2D Gaussian function, the Gaussian filtering is performed on the images. The 2D Gaussian function is given as:

$$w(x,y) = \frac{1}{2\pi\sigma^2} e^{-(\frac{x^2+y^2}{2\sigma^2})} \tag{28}$$

where, σ is the standard deviation of the Gaussian distribution with mean of 0 [43]. The 3 × 3 filter mask for Gaussian is given as:

$$w = \frac{1}{16}\begin{bmatrix} 1 & 2 & 1 \\ 2 & 4 & 2 \\ 1 & 2 & 1 \end{bmatrix} \tag{29}$$

The 2D Gaussian distribution with mean (0,0) and $\sigma = 1$ is shown in Figure 14.

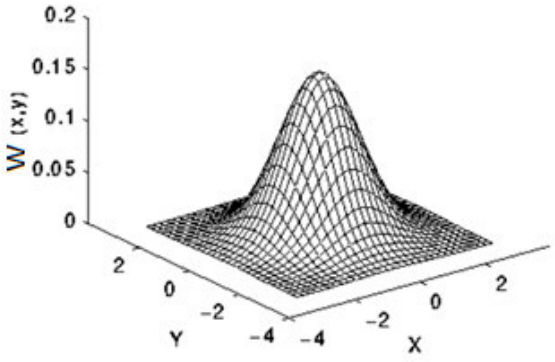

FIGURE 14. 2D Gaussian distribution [42].

2) MEAN FILTER

The filtered value at every pixel of $f(x, y)$ is calculated by taking average of the pixel values of the neighborhood defined by the filter mask.

$$G(x,y) = \frac{1}{mn} \sum_{(x,y)} f(x,y) \tag{30}$$

where, *mxn* is the mask size. For example, taking a 3 × 3 neighborhood about (x, y) gives:

$$w = \frac{1}{9}\begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{bmatrix} \tag{31}$$

In the mean filter, all the elements have the same weights [41].

3) MEDIAN FILTER

It is a nonlinear filter. Each pixel value in an image is replaced with the median of the values in the local neighborhood. For example, take a 3×3 window and in each window the median of the pixels is calculated as:

i) Keep all the pixel values of the window in ascending order.

ii) Select the middle pixel value as the median value for the pixel (x, y) [41].

E. ADAPTIVE THRESHOLDING

The temperature can vary frequently in the indoor environment. When the Grid-EYE sensor is placed in a different environment, the atmospheric temperatures and thermal values of human targets may vary. It is assumed that the background temperature of a room has a constant mean value and it varies according to the Gaussian distribution. There are multiple temperature peaks in the sensor output image other than the target place, which gives an error in the measurement. It is necessary to remove these peaks otherwise it could produce false detection. For the purpose, threshold value of the temperature is calculated by using adaptive thresholding method. In this method, the threshold (T) is calculated by [26]:

$$T = Max * 0.6 \tag{32}$$

where, *Max* is the maximum value of heat distribution of the background temperature. The threshold is selected as 60% of the maximum value. Since the temperature of the human target is knowingly greater than the threshold value, the human targets can be detected easily.

F. BINARY THERMAL IMAGE AND TARGET DETECTION

The foreground region is obtained by using the above calculated threshold value and converted into a binary thermal image. The Gaussian filtered thermal distribution value is compared with the threshold value. If the filtered value is greater than the threshold value, then the output thermal image value is set to one, otherwise to zero. From this, the segmented binary image is obtained. In a binary thermal image, all the foreground object temperature values are binary ones and the background are binary zero [43], [44]. The foreground image shows the occupancy of humans.

IV. EXPERIMENTAL RESULTS

Initially, the sensor captures temperature values of an empty room for two seconds. All the frames obtained from this unoccupied room are interpolated and the mean of all frames gives the background temperature. The thermal frames with

FIGURE 15. Data acquisition setup.

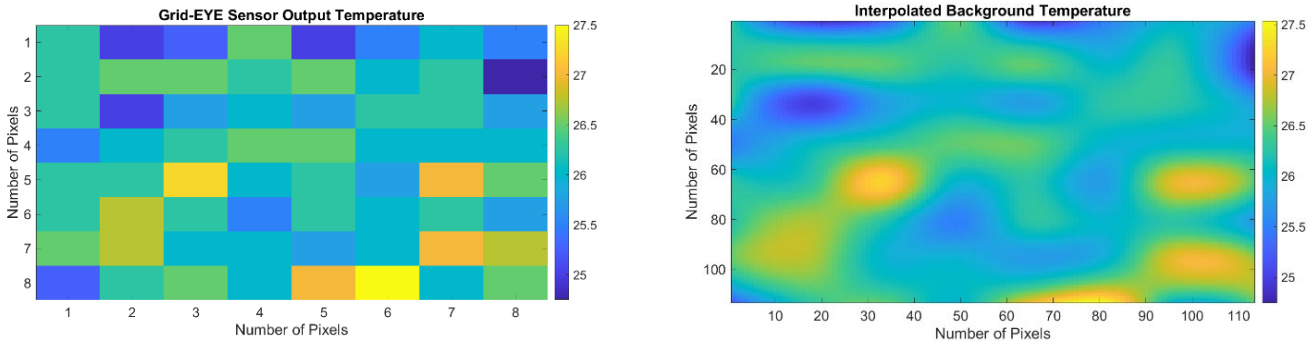

FIGURE 16. Sensor and interpolated image of empty room.

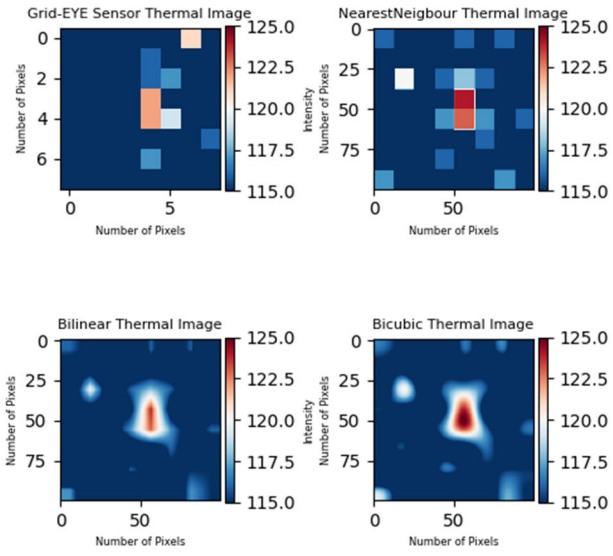

FIGURE 17. Comparison of different interpolation techniques.

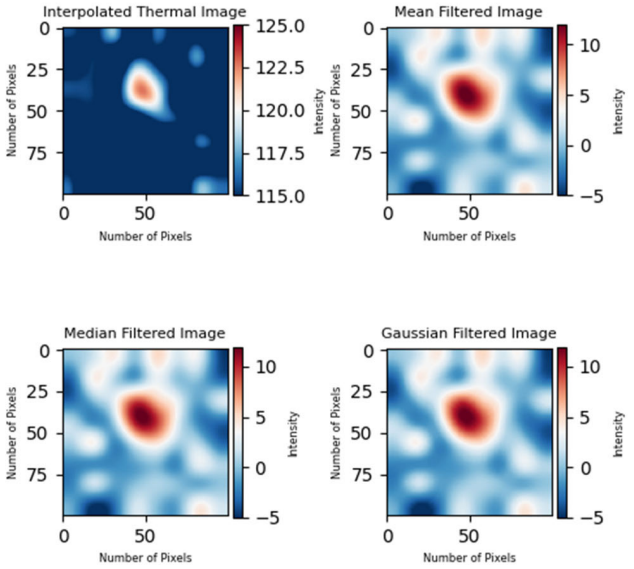

FIGURE 18. Comparison of different filtering methods.

humans are captured and these frames are interpolated. After the interpolation, thermal frames with humans are subtracted by the background temperature. Any random noises that appear in the image are filtered and smoothened using Gaussian filter. In the Gaussian thermal image, the accurate region is extracted by converting it into a binary thermal image. The thermal image is converted into a binary thermal image by selecting a suitable threshold value. The threshold value is used to segment an image into 2 regions, all the temperature

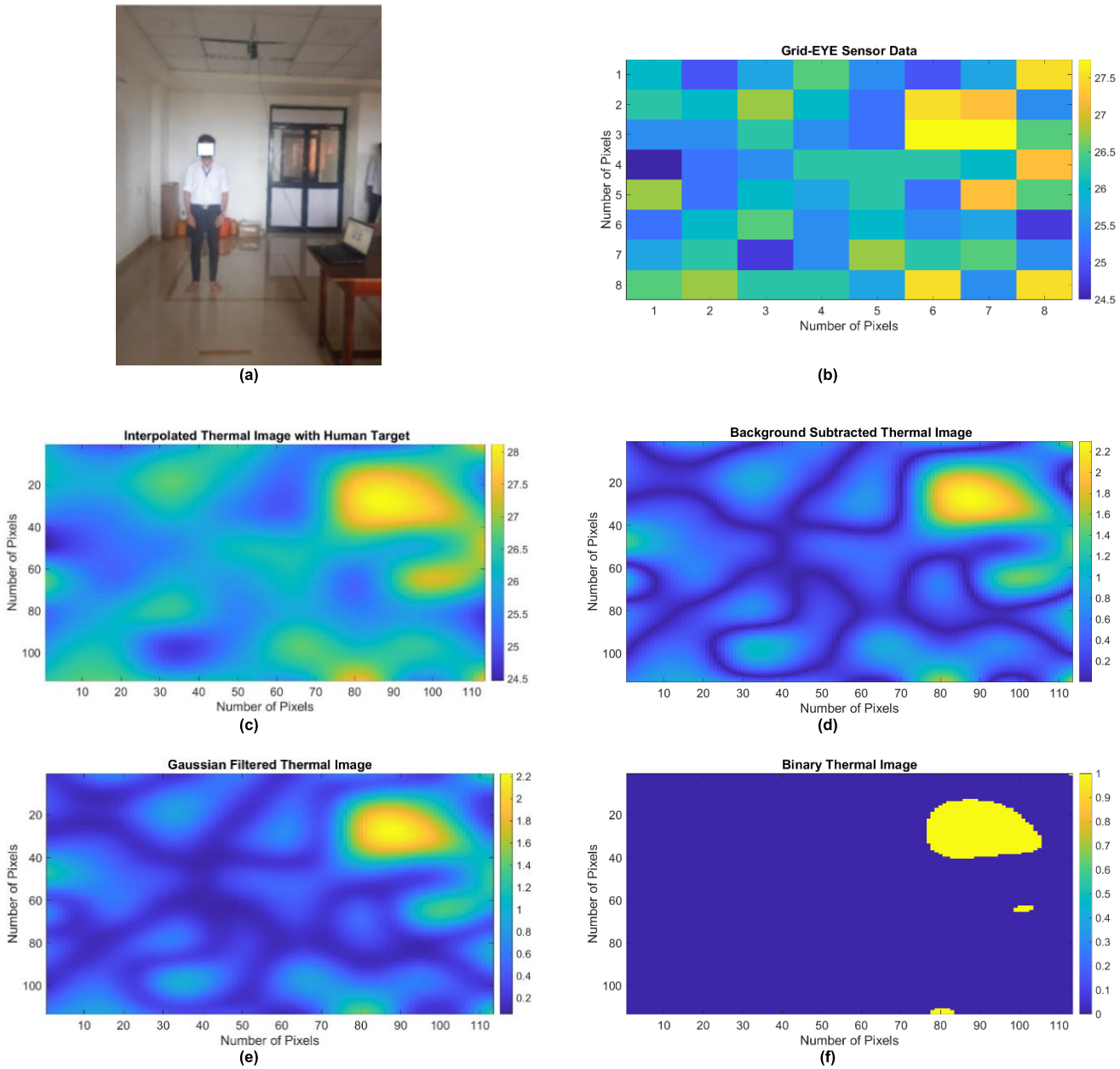

FIGURE 19. (a) Camera image (b) Sensor data (c) Interpolated image (d) Background subtracted image (e) Gaussian filter (f) Binary thermal image of a single person.

values above a threshold are selected as one and below a threshold are made zero.

From this, the required temperature values in the foreground region are extracted. After obtaining the threshold temperature, the desired foreground region is detected and is represented using the boundary box. Initially, the algorithm was implemented using MATLAB and later the same algorithm is tested in real-time by interfacing the Grid-EYE sensor to Raspberry Pi.

A. DATA ACQUISITION SETUP

The experiment was carried out indoors. The data acquisition setup is shown in Figure 15. The Grid-EYE sensor has a FoV of 60°. Here, the sensor is placed horizontally on the ceiling at a height of 2.8m and covers the detection area of approximately $9m^2$. The Grid-EYE sensor detection area is as shown in Figure 15. The outer square box indicates the detection area of $9m^2$ and the inner square box indicates the area of $2m^2$. The thermal image of the empty room and its interpolated output is shown in Figure 16.

B. COMPARISON OF DIFFERENT INTERPOLATION ALGORITHMS

The experiment was conducted by considering the 8×8 pixels from the thermal sensor. This resolution of the thermal image is resized into 100×100 pixels by using the Nearest Neighbor,

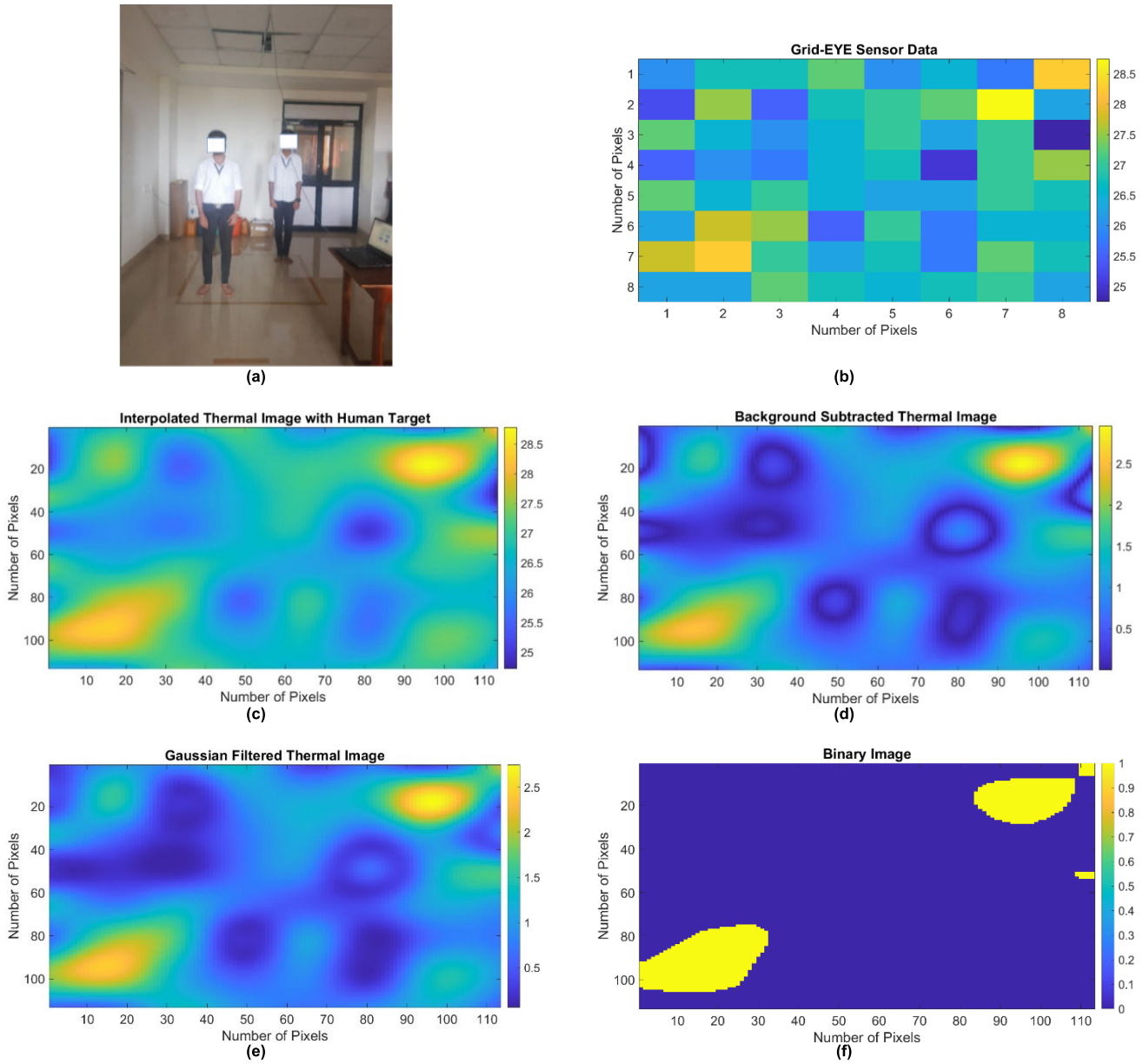

FIGURE 20. (a) Camera image (b) Sensor data (c) Interpolated image (d) Background subtracted image (e) Gaussian filter (f) Binary thermal image of a two person.

Bilinear, and Bicubic interpolation. The following performance metrics are used to compare the different interpolation methods: Mean Absolute Error (MAE)

1) Mean Square Error (MSE)
2) Root Mean Square Error (RMSE)
3) Signal to Noise Ratio (SNR)
4) Peak Signal to Noise Ratio (PSNR)
5) Similarity Index (SSIM)
6) Execution Time

It is observed from Table 3 that, the performance metric values for different interpolations can reflect the quality of the images. The higher the SNR, the quality of image is higher. It is observed that for thermal images, the SNR value of the bicubic interpolation is more compared to other interpolation methods. From the SSIM value, the nearest neighbor interpolated image has a 39% similarity with the original image compared to other interpolations.

The PSNR value of the Bicubic interpolation is 28dB, which is more compared to other interpolation methods. The MAE and MSE value of Bicubic Interpolation is less compared to other interpolations. By observing these performance metrics for different interpolations, the Bicubic interpolation performance is better compared to other methods.

The nearest neighbor interpolation is simple, requires less time, and easy to implement compared to other methods. In this interpolation, some of the information at the edges is lost. Bilinear interpolation gives better performance than

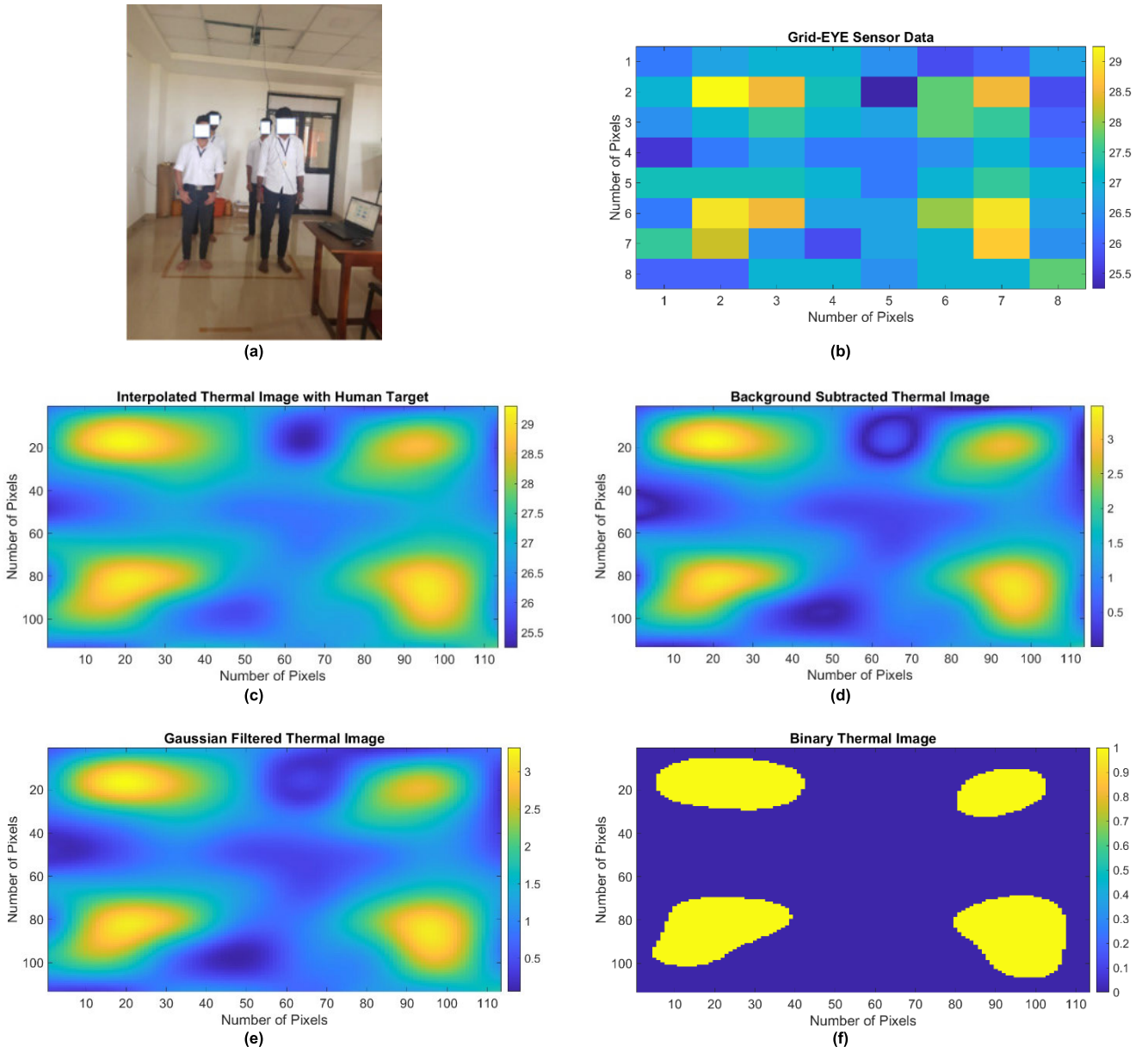

FIGURE 21. (a) Camera image (b) Sensor data (c) Interpolated image (d) Background subtracted image (e) Gaussian filter (f) Binary thermal image of four persons.

nearest neighbor and takes less computation time compared to bicubic interpolation.

The execution time for the bicubic interpolation is more compared to other methods. Even though the execution time is longer, the image of the Bicubic interpolation is more precise and sharper compared to other interpolation methods shown in Figure 17. From these observations, the Bicubic interpolation method's performance is better and it is used in detection algorithm.

C. COMPARISON OF DIFFERENT FILTERING ALGORITHMS

After the interpolation, spatial filtering is used to reduce noise and to smoothen the image. The different filtering methods used are Mean, Median, and Gaussian filters. The same above-mentioned performance metrics are used to compare the different filtering methods. By applying the Gaussian filtering to the image, the MSE and RMSE values of the thermal images are less compared to the other filtering methods as in Table 4. The PSNR value of the Gaussian filtering is 28dB, which is more compared to other filtering methods. The execution time required for the Gaussian filtering is less compared to the Median filtering.

By observing these performance metrics, the Gaussian filtering gives better performance compared to other filtering methods and is selected for the detection algorithm. The images of the different filtering methods are shown in Figure 18.

TABLE 3. Performance metrics for different interpolation techniques.

Performance Metrics/ Interpolation Methods	Nearest Neighbor	Bilinear	Bicubic
MAE	4.8745	4.9347	4.5823
MSE	30.48390	28.9773	25.9361
RMSE	5.32849	5.3232	4.9986
SNR (dB)	34.0404	35.3925	34.7337
PSNR (dB)	27.3034	27.5236	28.0051
SSIM	0.3970	0.1197	0.0912
Execution Time (s)	0.000160	0.00018	0.00020

TABLE 4. Performance metrics for different filters.

Performance Metrics/ Filtering Methods	Mean	Median	Gaussian
MAE	4.5367	3.3417	4.3569
MSE	24.9739	23.7182	21.9198
SNR (dB)	36.5180	36.4138	36.0045
PSNR (dB)	28.1693	28.3933	28.5444
SSIM	0.0668	0.0644	0.05353
Execution Time (s)	0.00027	0.0049	0.00029

D. ROOM TEMPERATURE VERSUS OBJECT SIZE

The experiment is conducted in two room temperatures to check the effect of the temperature on the thermopile sensor. The experiment was repeated by detecting a person in different locations of the detection area. In each location, 10 frames are measured and the average number of active pixels are calculated. The Table 5 shows an active number of pixels in an interpolated thermal image at a different location of a detection area.

It is observed that, as the room temperature decreased the number of active pixels in a thermal image is increased. If the person enters into room which contain less temperature compare to outside, then the detected active pixels in the thermal image will be more.

E. OCCUPANCY DETECTION ALGORITHM

Camera image and different steps to detect the thermal image of a single person in the detection area is shown in Figure 19. In this image, the person is standing still. The different images in the Figure 19 shows the different steps used in the occupancy detection algorithm. The yellow-colored region in the Binary thermal image indicates the detected object.

The camera image and different steps to detect thermal image of two persons in the detection area is shown in Figure 20. It is observed from the binary thermal image that the two yellow regions indicate two humans present in the detection area and marked with the rectangular boundary

TABLE 5. Room temperature versus object size.

Room Temperature	Number of Pixels				
	Left	Right	Middle	Top	Bottom
26.5^0C	490	561	613	599	505
24^0C	575	452	946	628	596

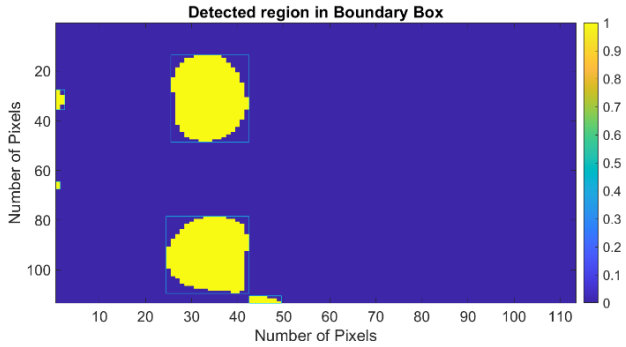

FIGURE 22. Detected two people in boundary box.

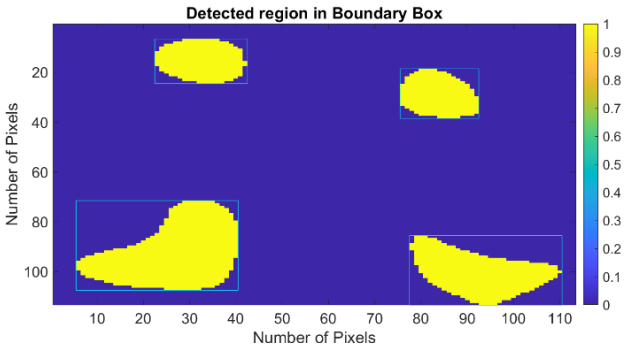

FIGURE 23. Detected four people in boundary box.

as shown in Figure 22. The camera image and different steps to detect the thermal image of four persons in the detection area is shown in Figure 21. The four yellow regions in the binary thermal image indicate that four people are detected and marked with the rectangular boundary as shown in Figure 23.

F. REAL TIME IMPLEMENTATION OF DETECTION ALGORITHM

The real time implementation is carried out by using the Raspberry pi processor by connecting Grid-EYE sensor to it. The real time detection of a single human in the FoV of the sensor is shown in Figure 24. The interpolation technique used in the algorithm is Bicubic Interpolation. The detected human is represented inside the square box. If the person is moving in the detection area, it can also be detected immediately within a millisecond and represented inside the rectangular box. The multiple object detection (3 people) in real-time is shown in Figure 25. The computation time taken by the detection algorithm in real time implementation using Raspberry Pi is 90ms. The computation time taken by the detection algorithm using the MATLAB simulation

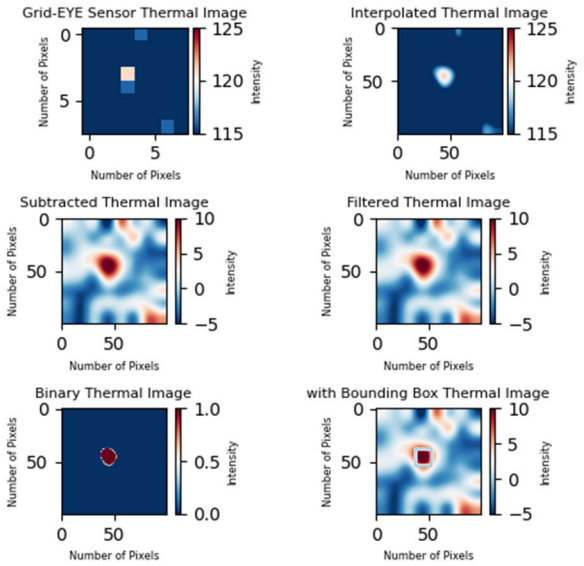

FIGURE 24. Single person detection.

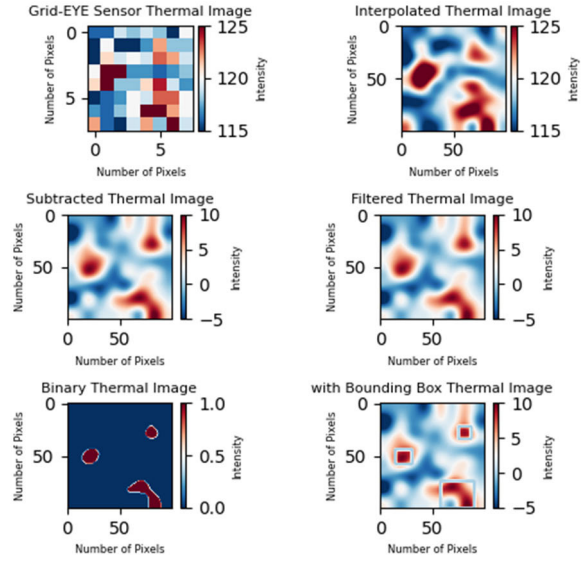

FIGURE 25. Multiple person detection.

tool is 10 seconds in PC. This indicates the response of the algorithm in real time implementation is faster than the MATLAB simulation. As the computation time of real time implementation (90ms) is less than the frame rate (100ms) of the sensor it is possible to carry out human detection without loss information.

V. CONCLUSION AND FUTURE WORK

Initially, the paper proposes the study of detection of multiple humans using Grid-EYE sensor (AMG3388) by calculating the thermal distribution radiated by the body in an indoor environment using a MATLAB simulation tool. Later, the paper discusses real-time multiple human detection using the same sensor. With this low-resolution sensor, human targets are detected easily. One of the main advantages of the thermopile array sensor compared to the optical camera-based systems is that it is not expensive and is not affected by the environment's lighting variations. So, instead of expensive optical cameras, thermopile sensors can be used for human detection.

Real time implementation of Bicubic interpolation and Gaussian filter method are carried out in Raspberry Pi and found that the computational time is less than the frame rate of the sensor. These results encourage the use of this sensor in real time applications. The room temperature highly affects the thermopile sensor. The research work could be further improved by optimal implementation of algorithms to reduce the computational time and could be extended to high resolution and frame rate sensors. In addition, the obtained results open the door for extending the technique to multiple sensor installation scenario.

ACKNOWLEDGMENT

The authors would like to thank the Research Innovation Center, NMAMIT, Nitte, for providing the space to set up the thermal sensor.

CONFLICTS OF INTEREST

The authors report no conflicts of interest. The authors alone are responsible for the content and writing of this article. The work submitted is being carried out in NMAMIT, Nitte, for research purpose only and has no relevance to any industries.

REFERENCES

[1] P. Anand, C. Deb, K. Yan, J. Yang, D. Cheong, and C. Sekhar, "Occupancy-based energy consumption modelling using machine learning algorithms for institutional buildings," *Energy Buildings*, vol. 252, Dec. 2021, Art. no. 111478, doi: 10.1016/j.enbuild.2021.111478.

[2] P. Anand, D. Cheong, C. Sekhar, M. Santamouris, and S. Kondepudi, "Energy saving estimation for plug and lighting load using occupancy analysis," *Renew. Energy*, vol. 143, pp. 1143–1161, Dec. 2019, doi: 10.1016/j.renene.2019.05.089.

[3] P. Anand, D. Cheong, and C. Sekhar, "Computation of zone-level ventilation requirement based on actual occupancy, plug and lighting load information," *Indoor Built Environ.*, vol. 29, no. 4, pp. 558–574, Apr. 2020, doi: 10.1177/1420326X19875802.

[4] Z. Wang, Z. Yang, and T. Dong, "A review of wearable technologies for elderly care that can accurately track indoor position, recognize physical activities and monitor vital signs in real time," *Sensors*, vol. 17, no. 2, p. 341, Feb. 2017, doi: 10.3390/s17020341.

[5] X. Luo, Q. Guan, H. Tan, L. Gao, Z. Wang, and X. Y. Luo, "Simultaneous indoor tracking and activity recognition using pyroelectric infrared sensors," *Sensors*, vol. 17, no. 8, p. 1738, Apr. 2017, doi: 10.3390/s17081738.

[6] T. Miyazaki and Y. Kasama, "Multiple human tracking using binary infrared sensors," *Sensors*, vol. 15, no. 6, pp. 13459–13476, Jun. 2015, doi: 10.3390/s150613459.

[7] W. Liang, Y. Wang, Z. Wu, B. Mao, and J. Cao, "Indoor region localization with asynchronous sensing data: A Bayesian probabilistic model," *IEEE Sensors J.*, vol. 18, no. 24, pp. 10174–10182, Dec. 2018, doi: 10.1109/JSEN.2018.2872825.

[8] S. Kanazawa, K. Taniguchi, A. Kazunari, K. Kuramoto, S. Kobashi, and Y. Hata, "A fuzzy automated object classification by infrared laser camera," in *Proc. SPIE*, vol. 8058, pp. 342–350, Jun. 2011.

[9] M. C. Foote, M. Kenyon, T. R. Krueger, T. A. McCann, R. Chacon, E. W. Jones, M. R. Dickie, J. T. Schofield, and D. J. McCleese, "Thermopile detector arrays for space science applications," in *Proc. SPIE*, vol. 4999, pp. 443–447, Mar. 2003.

[10] A. W. Van Herwaarden and P. M. Sarro, "Thermal sensors based on the Seebeck effect," *Sens. Actuators A, Phys.*, vol. 10, nos. 3–4, pp. 321–346, Nov. 1986, doi: 10.1016/0250-6874(86)80053-1.

[11] C. Chitu, G. Stamatescu, I. Stamatescu, and V. Sgarciu, "Wireless system for occupancy modelling and prediction in smart buildings," in *Proc. 25th Medit. Conf. Control Autom. (MED)*, Jul. 2017, pp. 1094–1099.

[12] J. Lu, T. Zhang, F. Hu, and Q. Hao, "Preprocessing design in pyroelectric infrared sensor-based human-tracking system: On sensor selection and calibration," *IEEE Trans. Syst., Man, Cybern., Syst.*, vol. 47, no. 2, pp. 263–275, Feb. 2017, doi: 10.1109/TSMC.2016.2523914.

[13] X. Luo, T. Liu, B. Shen, L. Gao, and X. Luo, "Human indoor localization based on ceiling mounted PIR sensor nodes," in *Proc. 13th IEEE Annu. Consum. Commun. Netw. Conf. (CCNC)*, Jan. 2016, pp. 868–874.

[14] D. Xu, Y. Wang, B. Xiong, and T. Li, "MEMS-based thermoelectric infrared sensors: A review," *Frontiers Mech. Eng.*, vol. 12, no. 4, pp. 557–566, Dec. 2017, doi: 10.1007/s11465-017-0441-2.

[15] B. Shubha, V. V. D. Shastrimath, and L. V. Prabhu, "Parameter analysis of pyroelectric and thermopile detectors," in *Proc. Int. Conf. Emerg. Technol. (INCET)*, Jun. 2020, pp. 1–7.

[16] J. L. Honorato, I. Spiniak, and M. Torres-Torriti, "Human detection using thermopiles," in *Proc. IEEE Latin Amer. Robotic Symp.*, Oct. 2008, pp. 151–157.

[17] M. Kuki, H. Nakajima, N. Tsuchiya, K. Kuramoto, S. Kobashi, and Y. Hata, "Mining multi human locations using thermopile array sensors," in *Proc. IEEE 43rd Int. Symp. Multiple-Valued Log.*, May 2013, pp. 59–64.

[18] Y. Jeong, K. Yoon, and K. Joung, "Probabilistic method to determine human subjects for low-resolution thermal imaging sensor," in *Proc. IEEE Sensors Appl. Symp. (SAS)*, Feb. 2014, pp. 97–102, doi: 10.1109/SAS.2014.6798925.

[19] D. C. Kallur, "Human localization and activity recognition using distributed motion sensors," M.S. thesis, Dept. Electron. Commun. Eng., Oklahoma State Univ., VTU, Belguam, Karnataka, 2014.

[20] L. I. L. Gonzalez, M. Troost, and O. Amft, "Using a thermopile matrix sensor to recognize energy-related activities in offices," *Proc. Comput. Sci.*, vol. 19, pp. 678–685, Jan. 2013.

[21] A. D. Shetty, B. Disha, and K. Suryanarayana, "Detection and tracking of a human using the infrared thermopile array sensor—'Grid-EYE,'" in *Proc. Int. Conf. Intell. Comput., Instrum. Control Technol. (ICICICT)*, Jul. 2017, pp. 1490–1495.

[22] C. Basu and A. Rowe, "Tracking motion and proxemics using thermal-sensor array," 2015, *arXiv:1511.08166*.

[23] A. Geczy, R. D. J. Melgar, A. Bonyar, and G. Harsanyi, "Passenger detection in cars with small form-factor IR sensors (grid-eye)," in *Proc. IEEE 8th Electron. Syst.-Integr. Technol. Conf. (ESTC)*, Sep. 2020, pp. 1–6.

[24] W.-H. Chen and H.-P. Ma, "A fall detection system based on infrared array sensors with tracking capability for the elderly at home," in *Proc. 17th Int. Conf. E-Health Netw., Appl. Services (HealthCom)*, Oct. 2015, pp. 428–434.

[25] Y. Taniguchi, H. Nakajima, N. Tsuchiya, J. Tanaka, F. Aita, and Y. Hata, "Estimation of human posture by multi thermal array sensors," in *Proc. IEEE Int. Conf. Syst., Man, Cybern. (SMC)*, Oct. 2014, pp. 3930–3935, doi: 10.1109/SMC.2014.6974545.

[26] N. Gu, B. Yang, and T. Li, "High-resolution thermopile array sensor-based system for human detection and tracking in indoor environment," in *Proc. 15th IEEE Conf. Ind. Electron. Appl. (ICIEA)*, Nov. 2020, pp. 1926–1931.

[27] H. M. Ng, "Poster abstract: Human localization and activity detection using thermopile sensors," in *Proc. 12th Int. Conf. Inf. Process. Sensor Netw. (IPSN)*, 2013, pp. 337–338.

[28] Panasonic User Mannual. *Grid-EYE Evaluation Kit*. Accessed: Jun. 10, 2022. [Online]. Available: https://api.pim.na.industrial.panasonic.com/file_stream/main/fileversion/4712

[29] Panasonic Product Catalog. (2022). *Infrared Array Sensor Grid-EYE*. Accessed: Jun. 10, 2022. [Online]. Available: https://api.pim.na.industrial.panasonic.com/file_stream/main/fileversion/247996

[30] V. Chidurala and X. Li, "Occupancy estimation using thermal imaging sensors and machine learning algorithms," *IEEE Sensors J.*, vol. 21, no. 6, pp. 8627–8638, Mar. 2021, doi: 10.1109/JSEN.2021.3049311.

[31] A. Naser, A. Lotfi, and J. Zhong, "Towards human distance estimation using a thermal sensor array," *Neural Comput. Appl.*, pp. 1–11, Jun. 2021, doi: 10.1007/s00521-021-06193-2.

[32] M. Maaspuro, "Infrared occupancy detection technologies in building automation—A review," *ARPN J. Eng. Appl. Sci.*, vol. 13, no. 19, pp. 8055–8068, 2018.

[33] *Raspberry Pi 4*. Accessed: Jun. 10, 2022. [Online]. Available: https://www.raspberrypi.com/products/raspberry-pi-4-model-b/

[34] L. Ngeljaratan and M. A. Moustafa, "Implementation and evaluation of vision-based sensor image compression for close-range photogrammetry and structural health monitoring," *Sensors*, vol. 20, no. 23, p. 6844, Nov. 2020, doi: 10.3390/s20236844.

[35] L. Liu, Y. Chen, and B. Yan, "An efficient classification based image interpolation algorithm," in *Proc. 4th Int. Congr. Image Signal Process.*, Shanghai, China, Oct. 2011, pp. 1063–1066.

[36] E. V. Diana and M. Sumathi, "A comprehensive study of 1D and 2D image interpolation techniques," in *ICCCE* (Lecture Notes in Electrical Engineering), vol. 500. Singapore: Springer, 2018.

[37] Richard Alan Peters II. (Oct. 29, 2015). *Image Processing Lecture Notes: Resizing Images*. [Online]. Available: https://ia902707.us.archive.org/23/items/Lectures_on_Image_Processing/EECE_4353_15_Resampling.pdf

[38] L. Tao, T. Volonakis, B. Tan, Y. Jing, K. Chetty, and M. Smith, "Home activity monitoring using low resolution infrared sensor," in *Proc. Comput. Vis. Pattern Recognit.*, 2018, pp. 1–8.

[39] A. A. Trofimova, A. Masciadri, F. Veronese, and F. Salice, "Indoor human detection based on thermal array sensor data and adaptive background estimation," *J. Comput. Commun.*, vol. 05, no. 4, pp. 16–28, 2017, doi: 10.4236/jcc.2017.54002.

[40] M. Maaspuro, "A low-resolution ir-array as a doorway occupancy counter in a smart building," *Int. J. Online Eng.*, vol. 16, pp. 4–19, Jun. 2020, doi: 10.3991/ijoe.v16i06.13915.

[41] C. R. Gonzalez and E. R. Woods, *Digital Image Processing*. 3rd ed. Upper Saddle River, NJ, USA: Prentice-Hall, 2008.

[42] *Gaussian Filtering*. Accessed: Jun. 10, 2022. [Online]. Available: https://homepages.inf.ed.ac.uk/rbf/HIPR2/gsmooth.htm

[43] H. Mohammadmoradi, S. Munir, O. Gnawali, and C. Shelton, "Measuring people-flow through doorways using easy-to-install IR array sensors," in *Proc. 13th Int. Conf. Distrib. Comput. Sensor Syst. (DCOSS)*, Jun. 2017, pp. 35–43.

[44] X. Zhang, D. Li, W. Yang, J. Wang, and S. Liu, "An efficient iterative thresholding algorithms for color images of cotton foreign fibers," in *Proc. 4th Conf. Comput. Comput. Technol. Agricult. (CCTA)*, Nanchang, China, Oct. 2010, pp. 710–719.

B. SHUBHA received the B.E. degree in electronics and communication engineering and the M.Tech. degree in digital electronics and communication systems from Visvesvaraya Technological University, Belagavi, in 2004 and 2010, respectively. She is currently pursuing the part-time Ph.D. degree with the Department of Electronics and Communication Engineering, NMAM Institute of Technology, Nitte, Karkala, India. She is also serving as an Assistant Professor with the Department of Electronics and Communication Engineering, NMAM Institute of Technology. Her research interests include image processing, signal processing, embedded systems, and control systems. She is a Life Member of the Indian Society for Technical Education (ISTE) and a member of the Indian Society of Systems for Science and Engineering (ISSE).

V. VEENA DEVI SHASTRIMATH (Member, IEEE) received the B.E. degree (Hons.) in electronics and communication engineering from the Sri Jayachamarajendra College of Engineering (SJCE), University of Mysore, Karnataka, India, in 1991, the M.Tech. degree (Hons.) in digital electronics and advanced communication from the Manipal Academy of Higher Education (MAHE), Manipal Institute of Technology (MIT), Manipal, Karnataka, in 2002, and the Ph.D. degree in digital image processing and remote sensing from Mangalore University, in 2015. She has teaching experience in various colleges, including SJCE Mysore, BITS Pilani, India, Mangalore University, P. A. College of Engineering, Mangalore, SJEC Mangalore, and the NMAM Institute of Technology (NMAMIT), Nitte. She is currently working as a Professor with the Department of Electronics and Communication Engineering, NMAMIT. She has more than 25 years of teaching experience and two years of industrial experience. She is a Life Member of the Indian Society for Technical Education (ISTE).