International Journal of Engineering & Extended Technologies Research (IJEETR)

|ISSN: 2322-0163| www.ijeetr.com | A Bimonthly, Peer Reviewed, Scholarly Indexed Journal |

| Volume 8, Issue 2, March - April 2026 |

DOI:10.15662/IJEETR.2026.0802115

Development of Raspberry Pi–Based Image Segmentation System for Automated Wound Area Measurement and Temperature Monitoring

Harsha Bala, Petchiyammal, Mary James

Department of Biomedical Engineering, Sethu Institute of Technology, 626115 Virudhunagar, India

Publication History: Received: 25.02.2026; Revised: 20.03.2026; Accepted: 25.03. 2026; Published: 28.03.2026.

ABSTRACT: This project proposes a wound healing monitoring system. It's efficient in checking how a wound is healing. It also finds infections that may happen during healing. The system uses an ID for each patient. This helps store the patients data on wound healing. The data is stored as CSV files for easy analysis. The system takes high-resolution images of the wound. It then accurately identifies the wound area. The system also calculates the wound area. It uses images from the start and current stages of healing. This helps calculate the healing percentage. So it gives a measure of wound healing. The system also has a temperature sensor. This sensor finds infections. So it helps evaluate wound healing.

KEYWORDS: Wound healing, Image segmentation, Temperature sensor, Patient monitoring, CSV data storage, Infection detection, Healing percentage.

I. INTRODUCTION

TWounds, whether chronic or acute are a problem worldwide.Millions of patients suffer from wounds every year. Wounds like foot ulcers, pressure ulcers and surgical site wounds need constant monitoring. This helps avoid complications like infections and amputations. According to health reports diabetic foot ulcers are increasing rapidly. This is especially true for the elderly and in developing countries. Good wound management relies on measuring and evaluating the wounds size, depth and temperature changes.. Current wound assessment methods are mostly subjective and manual.

Conventionally wound size is measured using a ruler, paper or planimeter methods. The wound outline is. The area is estimated. These methods are widely used in hospitals and clinics. They have limitations. Firstly they have variability between observers. Secondly they may increase the risk of infection. This is especially true if methods are repeated frequently. Thirdly they have limitations in recording wound healing progress. This is especially true if documentation is not electronic..

In the recent past, digital imaging, as well as computer vision, has come up as an effective tool for the automated assessment of wounds. Image-based analysis of wounds has the potential for the objective quantification of wound size, color distribution, as well as healing patterns, without the need for any kind of contact with the wound. Sophisticated computer vision methods, such as deep learning, especially convolutional neural networks (CNNs), have also come up as highly effective for the analysis of wounds, as they can classify different types of tissues, such as granulation tissue, necrotic tissue, as well as epithelialized tissue, etc. Although deep learning methods come up as highly effective for the analysis of wounds, they also require high computational capabilities, memory, as well as GPU capabilities, thus making them not suitable for embedded systems.

In many developing countries, the facilities for healthcare may not be equipped with the necessary computational capabilities, internet access, or affordability. Hence, there is an urgent need for developing embedded systems that can be used for wound monitoring without the need for cloud computing facilities or advanced computational capabilities. For the purpose of developing embedded systems for medical monitoring, single-board computers may be considered as an alternative, as they can be integrated with cameras, sensors, and display modules while consuming less power.

This research presents an embedded system for wound monitoring using a Raspberry Pi platform. The system combines image capturing, wound segmentation, temperature sensing and patient information recording. It uses a color-based segmentation technique. This technique uses the Hue Saturation Value (HSV) color model. The HSV model helps differentiate the wound area from tissues. The system also includes a temperature sensor. This sensor measures the wound surface temperature during assessment.

The use of the color-based segmentation method is also motivated by the fact that the proposed method is computationally efficient and can be used to implement the proposed solution in real-time embedded systems. The use of HSV thresholding in the proposed method can reduce the processing complexity compared to the use of neural network inference models. Therefore, the proposed solution can operate smoothly even in low-power hardware platforms without the use of GPUs. In addition, the use of image resizing and pixel-level analysis can also improve the performance of the proposed solution.

Another important parameter to be evaluated in the proposed wound assessment solution is the temperature. An increase in temperature can indicate infection, inflammation, and poor wound healing. Therefore, the proposed solution includes the use of a digital temperature sensor to measure the temperature of the wound surface during the wound assessment session. In order to monitor patients longitudinally, a data storage mechanism was incorporated into the system using comma-separated value files. Each patient is given a unique identification number that can be selected from buttons on the system. Once an image is captured, the system will automatically log the timestamp, percentage of area of the wound, and temperature reading into a file specific to each patient. This will enable healthcare professionals to track the healing process of each patient by comparing the initial percentage of area of the wound with that of future images captured from the same patient.

User interaction is facilitated through the use of simple hardware interfaces such as push buttons and an OLED display. The push buttons are used to select the ID of the patient, to take an image, and to delete the data. Therefore, the device can operate independently without the use of a keyboard, mouse, and monitor. The use of the OLED display provides the user with real-time information such as the selected ID of the patient, the percentage wound area, temperature values, and the healing progress. The proposed system has several advantages over the available solutions. To start with, the system does not require any dependency on cloud connectivity as well as high-performance computing. Moreover, the system provides data privacy as the data is stored locally. Furthermore, the system does not require any additional hardware cost as it does not require the use of GPUs as well as imaging devices. Finally, the system provides a multi-parameter monitoring system that uses visual as well as thermal images.

With the focus on the simplicity and affordability of the computation, the proposed system closes the gap between the highly accurate wound analysis models implemented in the laboratory and the application in the rural health care sector. The proposed system shows the possibility of effective wound monitoring without the use of complex deep learning techniques.

II. MATERIALS AND METHODS

A. SYSTEM OVERVIEW

The proposed wound monitoring system is designed for integration into a embedded form factor. It's capable of performing image acquisition, image segmentation-based area estimation, temperature detection and patient-specific data recording in time. The system includes hardware and software elements. These elements work together to provide an automated -contact wound assessment tool for resource-scarce clinical environments.
The proposed wound monitoring system architecture includes the following elements:

- Image acquisition module
- Temperature sensing module
- Embedded processing module
- Display module/user interface module
- Data storage module

The system uses a three-button user interface. This interface is used for identification, image capture and data deletion. The healing rate is determined by comparing wound area measurements with previously stored data in CSV file format.

|ISSN: 2322-0163| www.ijeetr.com | A Bimonthly, Peer Reviewed, Scholarly Indexed Journal |

| Volume 8, Issue 2, March - April 2026 |

DOI:10.15662/IJEETR.2026.0802115

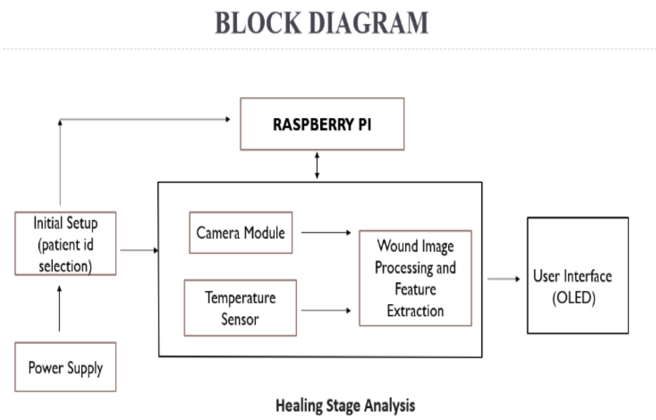

FIGURE 1 Wound Healing Monitor Overview

B. HARDWARE ARCHITECTURE

1) EMBEDDED PROCESSING UNIT

The processing unit of the system is the Raspberry Pi 4 Model B, which has been used due to its sufficient processing capabilities, GPIO pins, and the availability of camera and sensor modules. The Raspberry Pi has the following functionalities:

- Image acquisition
- Image preprocessing and segmentation
- Temperature data acquisition
- CSV data logging
- OLED control
- Button interrupt

The device uses a standard 5-voltage supply, and the operating system used by the device is Linux.

2) IMAGE ACQUISITION MODULE

A Raspberry Pi camera module, with a camera sensor is used to take images of the wounds. It supports high-resolution image capture and integrates well with the rpicamera software. The camera captures images. The resolution is customizable 640 x 480 pixels. This ensures processing. The color is converted from RGB to grayscale during preprocessing. The image capture method is triggered by a push button.

images are shown when the user wants to see them. This helps the system process information efficiently.

3) TEMPERATURE SENSING MODULE

A DS18B20 digital temperature sensor is used to measure the temperature of the wound surface. The DS18B20 digital temperature sensor works on the 1-Wire protocol and offers digital output with a precision of ±0.5°C in the clinical temperature range.

Key features:

- Operating range: -55°C to +125°C
- Resolution: 9 to 12-bit programmable
- Communication: Single data line (GPIO-based)

Temperature values are recorded during the course of each image capture cycle.

4) DISPLAY MODULE

A 0.96-inch I2C OLED display with a screen resolution of 128×64 pixels is used. The OLED screen acts as the output interface. It displays all the wound monitoring parameters to the user in a compact form. During the operation of the system the selected patient ID is displayed on the screen.

The wound area value obtained by the system is displayed on the screen. The system also computes the healing percentage.

International Journal of Engineering & Extended Technologies Research (IJEETR)

|ISSN: 2322-0163| www.ijeetr.com | A Bimonthly, Peer Reviewed, Scholarly Indexed Journal |

| Volume 8, Issue 2, March - April 2026 |

DOI:10.15662/IJEETR.2026.0802115

It indicates the wound area's progression of recovery over time.
The temperature reading obtained from the wound area is displayed to the user. It helps in the detection of any conditions that may be developing in the wound area. If the temperature reading goes beyond a set threshold an infection alert message appears on the screen.

5) USER INTERFACE BUTTONS

The system also comprises three push buttons that connect to the Raspberry Pi GPIO to allow user interaction and control of the system. The push buttons have been configured using internal pull-up resistors to provide reliable digital inputs and prevent interference from external sources. The push buttons have the following functions: the selection of the patient ID, which allows users to scroll through the IDs sequentially to enable the tracking of each patient's wounds individually; the initiation of the image processing and analysis procedure, which activates the camera module to calculate the parameters of the wounds; and the deletion of the stored data based on the selected patient ID, which offers the system the flexibility to manage the data recorded. This is done by the effective implementation of software-based debouncing methods, which prevent the false multiple triggers that may result due to the contact bounce phenomenon. The debouncing methods make the input signals more stable, thus improving the responsiveness of the system when it is subjected to multiple uses. The fact that the interface is controlled by the button eliminates the need for the inclusion of keyboards or touchscreens, thus simplifying the hardware part of the system. The simple user interface makes the system easy to use, especially when the system is required to function in rural health facilities. The inclusion of the button makes the system more durable, thus simplifying the complexity of the system as a whole.

C. SOFTWARE ARCHITECTURE

1) PROGRAMMING AND DEVELOPMENT FRAMEWORK

For our wound monitoring system we use Python 3. We choose Python 3 because it is flexible and has libraries. It is also easy to integrate with embedded systems. Python is a tool. It can combine image processing, temperature monitoring, data storage and user interface control into one software tool. The Python program is light. So it can run on Raspberry Pi without slowing down. We use programming. This lets us process operations individually. Modularity makes the system easy to maintain and improve. We make sure the system computes efficiently and ensures real-time computation. It does not use memory. We do not use learning frameworks. This keeps the software efficient for embedded systems. The system handles errors. It deals with communication errors with the sensor and GPIO interruptions. The system works offline. It does not need internet and makes it reliable for health facilities. Python provides a platform and good for developing software for embedded systems like our wound monitoring system. The Python programming language is suitable, for our wound monitoring system. Python 3 helps us build a wound monitoring system.

2) IMAGE PROCESSING AND NUMERICAL COMPUTATION LIBRARIES

OpenCV is the library used for image acquisition, processing, segmentation, contour detection and calculating wound area. These techniques help isolate the wound region accurately and efficiently on Raspberry Pi. The library works with NumPy for tasks like counting pixels and doing math to find out how much of the wound has healed. This combination makes image processing efficient without needing a GPU. Image data is treated as data to speed up computations and reduce processing time and delays in the segmentation algorithm. The goal is to keep image processing techniques accurate. OpenCV and NumPy together make this possible. They help in getting results quickly. The image segmentation algorithm is key, to maintaining accuracy. It works well on Raspberry Pi hardware. The processing time is reduced significantly. The wound area computation is done efficiently.

3) SENSOR INTERFACING AND HARDWARE CONTROL

To get the temperature we use a library called w1thermsensor. This library helps the DS18B20 temperature sensor talk to the 1-Wire communication protocol. It makes getting temperature data easy by handling the communication details. We get temperature readings by doing each measurement cycle. We use this temperature data along with wound area data to check for infections. The RPi.GPIO library is used to set up and control the push buttons on the Raspberry Pis GPIO pins. The pull-up resistor configuration helps get digital signals and reduces electrical noise. We also do debouncing of switches. This is done by using software techniques to prevent switch activations due to switch bounce effects. The GPIO interface lets us interact with the system easily. We can select patients take pictures and delete data using this interface. The temperature measurement and GPIO interactions are useful, for our application. The w1thermsensor. Rpi.GPIO library make it all work together.

International Journal of Engineering & Extended Technologies Research (IJEETR)

|ISSN: 2322-0163| www.ijeetr.com | A Bimonthly, Peer Reviewed, Scholarly Indexed Journal |

| Volume 8, Issue 2, March - April 2026 |

DOI:10.15662/IJEETR.2026.0802115

4) DISPLAY CONTROL AND DATA LOGGING

The OLED screen is managed using the Adafruit SSD1306 library, which allows effective communication using the I2C protocol. The library helps in displaying various parameters such as the patient's identification, wound area, percentage of healing, and temperature readings using text and numerical data types. The screen will be updated in real time after each measurement cycle. The data logging has been achieved using the CSV library, which comes as a standard library with Python. The library allows effective storage of wound parameters of various patients in a structured manner. Each patient's identification has a separate file using the CSV library, which allows effective storage of wound area, temperature value, time, and the calculated percentage of healing.

5) SOFTWARE WORKFLOW AND EXECUTION SEQUENCE

The software works in a step by step way. It starts with setting up the system, where all the hardware parts and libraries are put together. At this point the GPIO pins and other parts like the camera, temperature sensor and OLED display are all set up. Ready to go. The system then waits for the user to do something.

To pick the ID we use the first push button. This button helps us deal with patients. Once we have the ID we press the button to take pictures using the camera. We then do some work on the picture we took like turning it into black and white and getting rid of any noise. We take the picture. Find the part that is the wound. We then find the edge of the wound. Figure out how big it is by counting the pixels. At the time the software uses the temperature sensor to see how hot the wound is. We figure out how much the wound has healed by looking at how big it's now and how big it was, at the start. We write down all the things we have figured out in a file called a CSV file. We also update the OLED display to show us the size of the wound how much it has healed, the temperature. If it is infected. The software is always using the wound and patient ID to keep track of everything. The wound is the thing we are looking at and the patient ID helps us know who the wound belongs to.

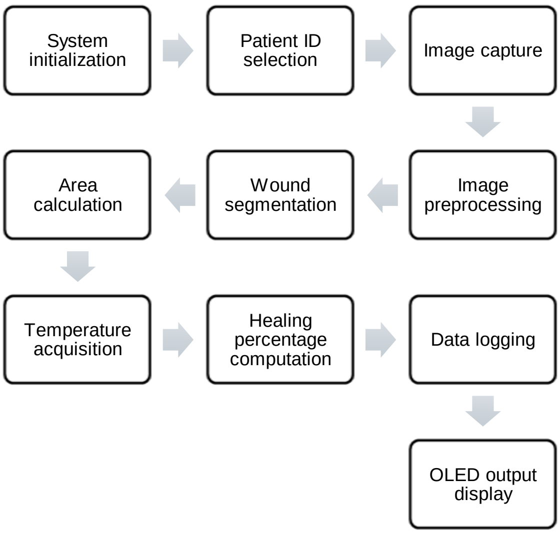

FIGURE 2 Software Workflow

D. IMAGE PROCESSING AND SEGMENTATION METHODOLOGY

Classical image processing techniques are used instead of deep learning techniques to ensure computational efficiency on embedded hardware.

1) IMAGE PREPROCESSING

The wound image processing begins with taking a picture of the wound using the Raspberry Pi camera module. This camera takes a clear picture of the wound. The picture that is taken is then changed from RGB format to grayscale format. This is done to make it easier to work with the wound image and to make the processing faster. The wound image is changed to grayscale format to simplify things. The conversion of the wound image from RGB format, to grayscale format can be explained with math.

$$I(x,y) = 0.299R+0.587G+0.114B$$

|ISSN: 2322-0163| www.ijeetr.com | A Bimonthly, Peer Reviewed, Scholarly Indexed Journal |

| Volume 8, Issue 2, March - April 2026 |

DOI:10.15662/IJEETR.2026.0802115

I(x,y) - the intensity of the grayscale image at the point (x,y)
R, G, B - the intensity of the red, green, and blue colors of the RGB format of the image, respectively.
Gaussian filtering is then applied to the grayscale images to remove the minor texture variations on the surface of the skin that may affect the reliability of the segmentation of the wound from the skin. The expression of the smoothing function of the images using the Gaussian filter is given by:

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

Where σ controls the amount of smoothing applied to the images.

2) THRESHOLD-BASED WOUND SEGMENTATION

Finally, the image is segmented by performing an intensity-based thresholding technique that distinguishes the wound region from the adjacent skin tissues. An appropriate threshold value, *T*
T, is determined based on the intensity distribution of the pixels. The binary image is obtained by performing the following thresholding operation:

$$B(x,y) = \begin{cases} 1, & \text{if } I(x,y) < T \\ 0, & \text{otherwise} \end{cases}$$

B(x,y) is an image that shows where a wound might be. It uses "1" for the area that could be the wound and "0" for the background or healthy skin. When pictures are taken in lighting we use a special way to find the right threshold for each part of the image. This helps the method work with pictures taken in various lighting conditions. The binary image might have some stuff like noise in it. This noise can be part of what looks like the wound area. The technique is more robust because it can handle images taken in lighting conditions. It works by finding a threshold that's just right for each part of the image. B(x,y) has "1" for the wound area and "0" for the background. The method uses thresholding to handle different lighting. This makes B(x,y), for different images. It finds the wound area and background clearly.

3) MORPHOLOGICAL REFINEMENT AND CONTOUR EXTRACTION

To make the image better we use a technique called image processing. We use erosion to get rid of noise pixels and shrink tiny artifacts. This helps clean up the image. Then we use dilation to restore the edges of the wound area. These operations are defined using math as follows:

$$B \ominus S = \{z \mid S_z \subseteq B\}$$
$$B \oplus S = \{z \mid (\hat{S})_z \cap B \neq \emptyset\}$$

The erosion and dilation operations are denoted by $\ominus$ and $\oplus$, respectively. Once the image is morphologically processed, the image contour is detected using connected component analysis to identify the continuous regions in the image. It is assumed that the largest detected image contour corresponds to the wound region to avoid misclassification of noise regions in the image.

4) WOUND AREA ESTIMATION

Once the wound contour is obtained, the area of the wound is calculated based on the total white pixels within the segmented region. The wound area A in pixel units is calculated as follows:

$$A = \sum_{x=1}^{M} \sum_{y=1}^{N} B(x,y)$$

where *M* and *N* are the dimensions of the image. Although absolute calibration such as conversion from pixels to mm² can be incorporated within the system, currently it is designed primarily for relative analysis between multiple monitoring sessions. By comparing the initial wound area A(initial) and the current wound area A(current), the progression of healing can be calculated as:

$$\text{Healing Percentage} = \frac{A_{\text{initial}} - A_{\text{current}}}{A_{\text{initial}}} \times 100$$

5) COMPUTATIONAL EFFICIENCY CONSIDERATIONS

The Raspberry Pi is what this segmentation algorithm is made for. It works well on the Raspberry Pi without needing help from a GPU or complicated learning tools. When it looks at pictures it uses OpenCV to do things. This way it can look at a picture. Give an answer in just a few seconds. It is almost like it is happening in time. To make sure it does not use much memory it makes the pictures smaller before it looks at them. This algorithm uses methods to look at pictures, which is much easier on the computer than using complicated neural networks. This makes it a good choice for tools that need to work in places where computers are not very powerful. The Raspberry Pi segmentation algorithm is still able to give results even in these situations. The segmentation algorithm on the Raspberry Pi is very useful, for these kinds of jobs.

E. HEALING PROGRESS EVALUATION

1) BASELINE AREA DETERMINATION

The healing assessment in the suggested system is done by comparing the measurements of the wound area over time. When we first look at the wound area of the chosen patient ID we save the wound area calculation as the baseline reference value, which we call A_initial. This is what the wound looks like at the beginning before it starts to heal. We need to set this reference value so we can see how the wound is healing over time. We save this value in the patients CSV file and it stays the same unless we manually change it.

This baseline area is what we use to compare all the measurements of the wound area so we can see how the wound is healing. We do this by calculating the area, in pixels, which's simple. We do not need to do a calculation that takes into account the environment where the picture of the wound is taken. The baseline value is what we use to compare all the measurements so we can see how the wound healing process is going. The wound healing process is what we are trying to assess. We use the wound area measurements to do this. We compare the wound area measurements to the baseline value, which's the initial wound area to see how the wound is healing over time.

2) TREND ANALYSIS AND CLINICAL INTERPRETATION

The system is useful for more than looking at one session. It also helps us look at trends by using all the data from the CSV file records. We can watch the wound area change over time using the measurements of the wound area. This means we can see how the wound changes from day to day or over a week. If the percentage of the wound healing is going up slowly it usually means the wound is healing well.. If the percentage is going up and down or getting smaller it means there are problems.

The system also helps us understand the wound better by looking at the temperature readings. This gives us a complete picture of what is going on with the wound. So we can make a diagnosis and not just guess. The system is not perfect now because it only gives us basic information about the percentage of the wound that is healing.. It could be made better by adding models that can predict what will happen to the wound. This would make the wound healing part of the system useful, for watching wounds.

F. TEMPERATURE-BASED INFECTION ASSESSMENT

1) TEMPERATURE MEASUREMENT MECHANISM

Wound temperature is really important to know when someone is hurt. It can tell us if the wound is getting infected. The system we are talking about uses a thermometer called DS18B20. This thermometer is digital. It uses something called the 1-Wire protocol to measure the temperature of the wound. It is very good at measuring the temperature. It does not make many mistakes.

The thermometer takes the temperature of the wound at the time that we take a picture of the wound. This helps us to know exactly what is going on with the wound. The temperature is written down in a file along, with a picture of the wound. How much it has healed.

We need to keep track of the temperature of the wound all the time. This helps us to see if the temperature is not normal which can mean that the wound is infected. The system is designed so that the thermometer can talk to the Raspberry Pi without needing any help. This makes it easy to use. It helps us to get the information we need about the wound temperature and the wound.

|ISSN: 2322-0163| www.ijeetr.com | A Bimonthly, Peer Reviewed, Scholarly Indexed Journal |

| Volume 8, Issue 2, March - April 2026 |

DOI:10.15662/IJEETR.2026.0802115

2) INTEGRATION WITH HEALING PROGRESS

The integration of temperature monitoring and wound area evaluation creates a comprehensive assessment tool. For instance, if the wound area is decreasing and the temperature remains steady, this can be an indication of healthy wound healing. However, if the wound area is increasing and the temperature is also elevated, this could imply infection and/or poor wound healing. The wound monitoring system can record both parameters simultaneously to enable correlation during data analysis. The use of multiple parameters to evaluate wound healing minimizes the use of a single diagnostic parameter. Long-term temperature trends can also be used to measure the effectiveness of treatment and/or antibiotic use. The integration of temperature assessment greatly improves the diagnostic value of the wound monitoring system.

G. DATA LOGGING AND PATIENT MANAGEMENT

An efficient management system has been achieved in the management of the patients' data using a CSV file-based storage system. Each patient ID has a corresponding CSV file that includes all the wound monitoring parameters appended sequentially. The parameters stored in the CSV file include the timestamp, wound area in terms of pixels, temperature reading, and healing percentage calculation. Thus, the system maintains a record in an efficient way without the need to implement a complex database system. The CSV file has the advantage of compatibility with available spreadsheet and statistical analysis software. The CSV files are stored locally on the Raspberry Pi hardware, thus providing enhanced privacy. The system retrieves the historical data in each session to calculate the healing progress in an efficient way.

III. EXPERIMENTAL RESULT AND PERFORMANCE METRICS

The wound monitoring system that we came up with was tested by taking pictures of wounds times over several sessions. The system was able to figure out the size of the wound in pixels how much it had healed and the temperature for each session. When we looked at the results in a table we saw that the wound got smaller every time. On the day the wound was 18,500 pixels big but by the ninth day it was only 6,200 pixels big. This means that the wound got, about 66.49% smaller which shows that the tissue was regenerating well while we were keeping an eye on it. If you look at the wound size over time as shown in Figure 3 you can see that it kept getting smaller at a rate. This tells us that the edge of the wound was getting smaller every time we took a picture. The computer program that we used to measure the wound size was able to find the edge of the wound as long as the lighting was consistent. This means that we could trust the measurements that the program gave us.

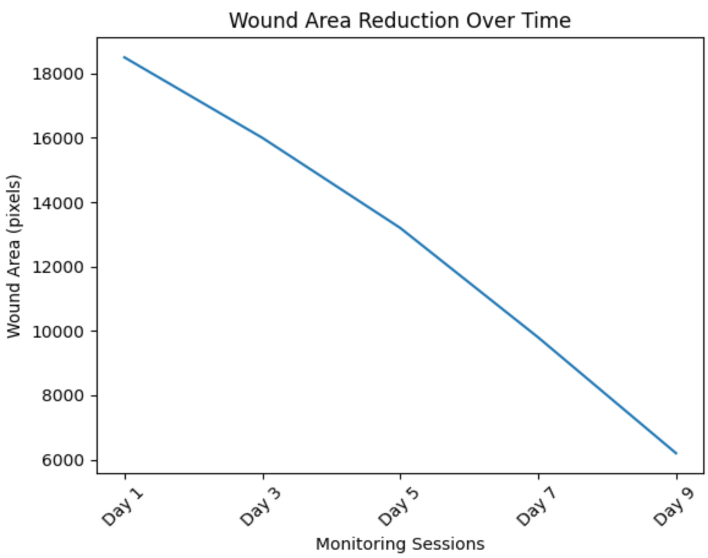

FIGURE 3 Wound Healing tracking by wound area reduction

The healing percentage progression, which you can see in Figure 4 shows that it gets better and better over time. This proves that the comparative area-based healing model is accurate. The system figures out the healing percentage by comparing it to a baseline so it is consistent even if we do not calibrate the units. The healing percentage goes up from 0% to 66.49% over nine days which means the system is good at monitoring the healing progress. The healing

International Journal of Engineering & Extended Technologies Research (IJEETR)

|ISSN: 2322-0163| www.ijeetr.com | A Bimonthly, Peer Reviewed, Scholarly Indexed Journal |

| Volume 8, Issue 2, March - April 2026 |

DOI:10.15662/IJEETR.2026.0802115

percentage of the wound increases over time. We do not see any changes in the data, which means the system is working well and it is good, at finding the edges of the wound. The healing percentage progression of the wound is stable. It increases over time.

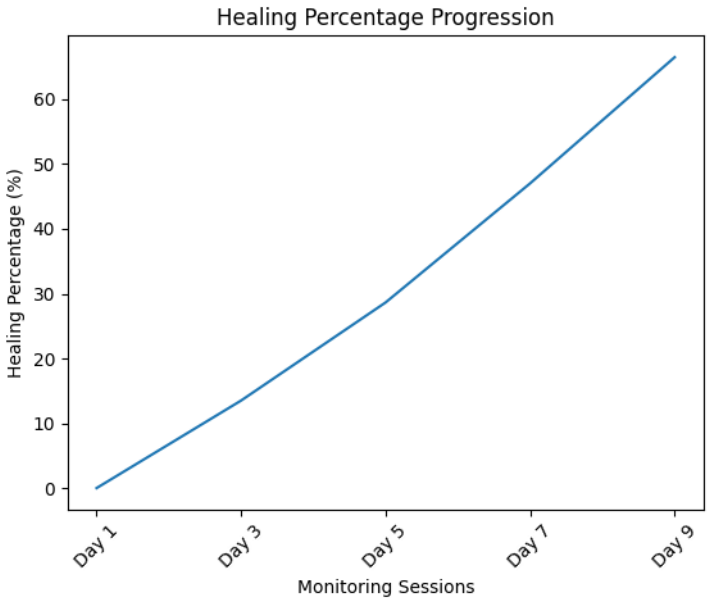

FIGURE 4 Healing Process Increases with the reduction of the Wound area

The temperature variation analysis in Figure 5 shows that the healing process temperature goes down from 38.1°C to 36.9°C. At the start the temperature was a bit high which meant there was some inflammation. As the healing process went on the temperature became normal. Using temperature checks along with looking at the shape and form of things makes it more likely that we can find infections. The system to warn about infections works well when the temperature is higher than it should be. This way of looking at two things at the time makes it easier for doctors to understand what is going on compared to only looking at one thing. The temperature variation analysis and the shape and form analysis are both important, for the healing process.

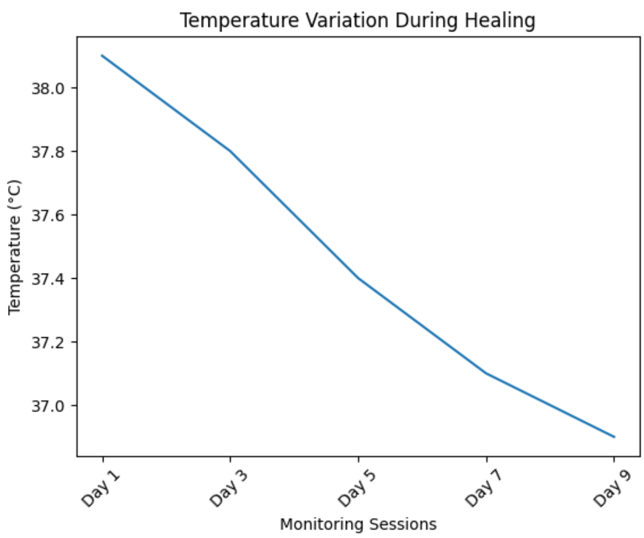

FIGURE 5 Variation in the Temperature over the wound healing.

|ISSN: 2322-0163| www.ijeetr.com | A Bimonthly, Peer Reviewed, Scholarly Indexed Journal |

| Volume 8, Issue 2, March - April 2026 |

DOI:10.15662/IJEETR.2026.0802115

IV. DISCUSSION

The proposed wound monitoring system based on Raspberry Pi proves the concept of developing a cost-effective embedded system for the objective evaluation of wounds using classical image processing techniques as well as temperature measurement. The use of grayscale-based segmentation, contour detection, and pixel-based area measurement is a computationally efficient approach for wound evaluation compared to the use of deep learning-based image processing. The experimental implementation of the proposed system proves the effectiveness of the segmentation algorithm for identifying the wound area under controlled lighting conditions. The pixel comparison method is also effective for evaluating the healing process of the wound without requiring any complex calibration steps. The addition of temperature measurement also proves effective in improving the diagnostic accuracy of the proposed system since it involves a physiological parameter related to wound infection.

One of the significant advantages of the proposed system is the simplicity and portability of the solution. Unlike the augmented reality and 3D reconstruction-based wound monitoring systems, the proposed solution is entirely offline-based and does not require the use of complex hardware and software tools. The use of CSV-based data storage ensures efficient data management and compatibility with other tools and software. The proposed solution also provides efficient management of multiple patient IDs via the push-button interface. The use of the OLED-based real-time display also ensures efficient interpretation of the wound area, healing percentage, and infection level without the use of external displays.

However, some limitations were observed during the evaluation of the system. The accuracy of the image segmentation may be affected by the variation of lighting, skin color, and color of the wound. Although the Gaussian filter has improved the robustness of the image, the extreme lighting may impact the detection of the contours. Additionally, the system may not be able to provide the exact area of the wound, as the measurements are provided in pixels, not square millimeters. However, the relative measurements provided by the system can be useful for the monitoring of the healing patterns of the wound. The avoidance of the deep learning technique has provided the required balance between the complexity of the computations and the adaptability of the system, making the system highly suitable for rural areas.

V. CONCLUSION

This paper discussed the design and implementation of a cost-effective embedded wound monitoring system using Raspberry Pi, classical image segmentation techniques, and temperature sensing to evaluate infections. The system effectively combines wound image acquisition, preprocessing, threshold-based segmentation, contour extraction, area calculation, healing percentage calculation, and temperature sensing into a single, embedded system. The proposed solution avoids the use of computationally expensive deep learning techniques and instead uses OpenCV and light-weight Python libraries to guarantee the system's performance in embedded devices. The implementation of the CSV data logging system allows the tracking of wound healing, while the use of a push-button interface and OLED screen enhances the system's portability.

The healing percentage evaluation method can be used to obtain an objective measure of the healing process over time, reducing the need to rely on subjective assessment through visual inspection. Additionally, the temperature-based infection detection feature enhances the reliability of the system in detecting infections through the combination of morphological and physiological features. While the system has demonstrated its reliability in controlled environments, further features may be incorporated in the future, such as normalization techniques to achieve consistency in the amount of illumination, as well as the ability to measure the actual physical area and the application of machine learning techniques to achieve robustness in the segmentation process.

REFERENCES

1. Yu-Hsien Lu, Meng-Hsuan Wu, Yu-Zheng Chen, Po- Liang Ou, Kuo-Shu Hung, Yi-Syuan Shin, Yuan-Yu Hsueh, Peng- Ting Chen, Chih-Lung Lin, "Multispectral Imaging for Preliminary Burn Depth Evaluation in Mice with Tissue Section Analysis", 2024 IEEE SENSORS, pp.1-4, 2024.
2. Yogesh Jadhav, K. H. Swetha, Omprakash Das, M. P. Sunil, "17513 Experimental evaluation and validation of different variants of PCA on octonionoctonion multispectral imagingmultispectral imaging", Octonion Sparse-Based Image Processing, pp.175, 2025.

|ISSN: 2322-0163| www.ijeetr.com | A Bimonthly, Peer Reviewed, Scholarly Indexed Journal |

| Volume 8, Issue 2, March - April 2026 |

DOI:10.15662/IJEETR.2026.0802115

3. Murugeshwari, B., Sudharson, K., Panimalar, S. P., Shanmugapriya, M., & Abinaya, M. (2020). SAFE–Secure Authentication in Federated Environment using CEG Key code.
4. Sugumar, R. (2025). Cyber-Secure Cloud Architecture Integrating Network and API Controls for Risk-Aware SAP Healthcare Data Platforms. *International Journal of Humanities and Information Technology, 7(4), 53-60.*
5. Sharma, K. P., Kumar, I., Singh, P. P., Anbazhagan, K., Albarakati, H. M., Bhatt, M. W., ... & Rana, A. (2024). Advancing spacecraft rendezvous and docking through safety reinforcement learning and ubiquitous learning principles. *Computers in Human Behavior, 153, 108110.*
6. Yi-Syuan Shin, Kuo-Shu Hung, Chung-Te Tsai, Meng-Hsuan Wu, Chih-Lung Lin, Yuan-Yu Hsueh, "Validation of multispectral imaging–based tissue oxygen saturation detecting system for wound healing recognition on open wounds", Journal of Biomedical Optics, vol.29, no.08, 2024.
7. C. -L. Lin et al., "Multispectral Imaging-Based System for Detecting Tissue Oxygen Saturation With Wound Segmentation for Monitoring Wound Healing," in IEEE Journal of Translational Engineering in Health and Medicine, vol. 12, pp. 468-479, 2024, doi: 10.1109/JTEHM.2024.3399232
8. R. Gupta et al., "Towards an AI-Based Objective Prognostic Model for Quantifying Wound Healing," in IEEE Journal of Biomedical and Health Informatics, vol. 28, no. 2, pp. 666-677, Feb. 2024, doi: 10.1109/JBHI.2023.3251901.
9. D. R. Seshadri, N. D. Bianco, A. N. Radwan, C. A. Zorman and K. M. Bogie, "An Absorbent, Flexible, Transparent, and Scalable Substrate for Wound Dressings," in IEEE Journal of Translational Engineering in Health and Medicine, vol. 10, pp. 1-9, 2022, Art no. 4900909.
10. Z. Ye, M. Yang, M. Farhat, M. M. . -C. Cheng and P. -Y. Chen, "Multimodal Wireless Wound Sensors via Higher-Order Parity-Time Symmetry," in IEEE Sensors Journal, vol. 24, no. 1, pp. 741-749, 1 Jan.1, 2024.
11. Y. Cao and Y. Wang, "The Impact of Artificial Intelligence and Deep Learning-Based Family-Centered Care Interventions on the Healing of Chronic Lower Limb Wounds in Children," in IEEE Access, vol. 12, pp. 125557-125570, 2024.
12. A. Mwangi, L. Navarro-Hilfiker, L. Brewka, M. Gryning, E. Fumagalli and M. Gibescu, "A Threshold-Triggered Deep Q-Network-Based Framework for Self-Healing in Autonomic Software-Defined IIoT-Edge Networks," in IEEE Transactions on Network and Service Management, vol. 23, pp. 1297-1311, 2026.
13. Agrippina Mwangi, León Navarro-Hilfiker, Lukasz Brewka, Mikkel Gryning, Elena Fumagalli, Madeleine Gibescu, "A Threshold-Triggered Deep Q-Network-Based Framework for Self-Healing in Autonomic Software-Defined IIoT-Edge Networks", IEEE Transactions on Network and Service Management, vol.23, pp.1297-1311, 2026.
14. A. Lazaro, M. Rodrigo Cujilema, R. Villarino and D. Girbau, "Smart Bandage for Wireless Pressure and Wound State Sensing Based on LC Sensor," in IEEE Sensors Journal, vol. 26, no. 1, pp. 1232-1247, 1 Jan.1, 2026.
15. D. Vital et al., "SkinAid: A Wirelessly Powered Smart Dressing Solution for Continuous Wound-Tracking Using Textile-Based Frequency Modulation," in IEEE Transactions on Biomedical Circuits and Systems, vol. 17, no. 5, pp. 985-998, Oct. 2023.
16. R. S. Fard et al., "Multimodal AI for Home Wound Patient Referral Decisions From Images With Specialist Annotations," in IEEE Journal of Translational Engineering in Health and Medicine, vol. 13, pp. 341-353, 2025.
17. C. Zeng, C. Li, M. Li, C. Chen, C. Liu and S. Han, "Wearable Organic Electrochemical Transistor System for Multiplexed Chronic-Wound Biosensing," in IEEE Sensors Journal, vol. 26, no. 3, pp. 5098-5105, 1 Feb.1, 2026.
18. C. -L. Lin et al., "Multispectral Imaging-Based System for Detecting Tissue Oxygen Saturation With Wound Segmentation for Monitoring Wound Healing," in IEEE Journal of Translational Engineering in Health and Medicine, vol. 12, pp. 468-479, 2024.
19. B. Pandey, D. Joshi, A. S. Arora, N. Upadhyay and H. S. Chhabra, "A Deep Learning Approach for Automated Detection and Segmentation of Pressure Ulcers Using Infrared-Based Thermal Imaging," in IEEE Sensors Journal, vol. 22, no. 15, pp. 14762-14768, 1 Aug.1, 2022..
20. B. K. S. Kumar, K. C. Anandakrishan, M. Sumant and S. Jayaraman, "Wound Care: Wound Management System," in IEEE Access, vol. 11, pp. 45301-45312, 2023.
21. D. M. Anisuzzaman, Y. Patel, J. A. Niezgoda, S. Gopalakrishnan, and Z. Yu, "A mobile app for wound localization using deep learning," IEEE Access, vol. 10, pp. 61398–61409, 2022.
22. S. Sarp, M. Kuzlu, M. Pipattanasomporn, and O. Guler, "Simultaneous wound border segmentation and tissue classification using a conditional generative adversarial network," J. Eng., vol. 2021, no. 3, pp. 125–134, Mar. 2021.
23. C.Nagarajan and M.Madheswaran - 'Stability Analysis of Series Parallel Resonant Converter with Fuzzy Logic Controller Using State Space Techniques'- Taylor &Francis, Electric Power Components and Systems, Vol.39 (8), pp.780-793, May 2011. DOI: 10.1080/15325008.2010.541746

24. C.Nagarajan and M.Madheswaran - 'Experimental verification and stability state space analysis of CLL-T Series Parallel Resonant Converter' - Journal of Electrical Engineering, Vol.63 (6), pp.365-372, Dec.2012. DOI: 10.2478/v10187-012-0054-2
25. C.Nagarajan and M.Madheswaran - 'Performance Analysis of LCL-T Resonant Converter with Fuzzy/PID Using State Space Analysis'- Springer, Electrical Engineering, Vol.93 (3), pp.167-178, September 2011. DOI 10.1007/s00202-011-0203-9
26. S.Tamilselvi, R.Prakash, C.Nagarajan,"Solar System Integrated Smart Grid Utilizing Hybrid Coot-Genetic Algorithm Optimized ANN Controller" Iranian Journal Of Science And Technology-Transactions Of Electrical Engineering, DOI10.1007/s40998-025-00917-z,2025
27. S.Tamilselvi, R.Prakash, C.Nagarajan," Adaptive sliding mode control of multilevel grid-connected inverters using reinforcement learning for enhanced LVRT performance" Electric Power Systems Research 253 (2026) 112428, doi.org/10.1016/j.epsr.2025.112428
28. S.Thirunavukkarasu, C. Nagarajan, 2024, "Performance Investigation on OCF and SCF study in BLDC machine using FTANN Controller," Journal of Electrical Engineering And Technology, Volume 20, pages 2675–2688, (2025), doi.org/10.1007/s42835-024-02126-w
29. C. Nagarajan, M.Madheswaran and D.Ramasubramanian- 'Development of DSP based Robust Control Method for General Resonant Converter Topologies using Transfer Function Model'- Acta Electrotechnica et Informatica Journal , Vol.13 (2), pp.18-31,April-June.2013, DOI: 10.2478/aeei-2013-0025.
30. C.Nagarajan and M.Madheswaran - 'DSP Based Fuzzy Controller for Series Parallel Resonant converter'- Springer, Frontiers of Electrical and Electronic Engineering, Vol. 7(4), pp. 438-446, Dec.12. DOI 10.1007/s11460-012-0212-0.
31. C.Nagarajan and M.Madheswaran - 'Experimental Study and steady state stability analysis of CLL-T Series Parallel Resonant Converter with Fuzzy controller using State Space Analysis'- Iranian Journal of Electrical & Electronic Engineering, Vol.8 (3), pp.259-267, September 2012.
32. C.Nagarajan and M.Madheswaran, "Analysis and Simulation of LCL Series Resonant Full Bridge Converter Using PWM Technique with Load Independent Operation" has been presented in ICTES'08, a IEEE / IET International Conference organized by M.G.R.University, Chennai.Vol.no.1, pp.190-195, Dec.2007
33. Suganthi Mullainathan, Ramesh Natarajan, "An SPSS and CNN modelling based quality assessment using ceramic materials and membrane filtration techniques", Revista Materia (Rio J.) Vol. 30, 2025, DOI: https://doi.org/10.1590/1517-7076-RMAT-2024-0721
34. M Suganthi, N Ramesh, "Treatment of water using natural zeolite as membrane filter", Journal of Environmental Protection and Ecology, Volume 23, Issue 2, pp: 520-530,2022
35. M. Goyal, N. D. Reeves, A. K. Davison, S. Rajbhandari, J. Spragg, and M. H. Yap, "DFUNet: Convolutional neural networks for diabetic foot ulcer classification," IEEE Trans. Emerg. Topics Comput. Intell., vol. 4, no. 5, pp. 728–739, Oct. 2020.