International Research Journal of Innovations in Engineering and Technology (IRJIET)

ISSN (online): 2581-3048

Volume 5, Issue 5, pp 137-142, May-2021

https://doi.org/10.47001/IRJIET/2021.505026

Array Application as a Limiting Image for Placement of Shaped Virtual Sensors Color Map Visualization

[1]Slamet Khoeron, [2]D. B. Wibowo, [3]Ojo Kurdi

[1,2,3]Department of Mechanical Engineering, Diponegoro University, Semarang 50275, Indonesia

***Abstract* - Measurements for complex areas sometimes become constraints because of the shapes and dimensions that are difficult to interpret. Another condition that affects is the state of the location. Like the human foot which has a unique shape. To measure it requires a lot of sensors, but this is not effective because it is expensive. Arranging the array as a frame is applied to help create the delimiting image. Where in this image the sensor is placed at predetermined coordinates. Laying based on the division of the area of the soleplate. The measurement results show that the sensor can represent any part of the foot area that receives interface pressure. Identify the type of foot, access the location of the coordinates around the sensor and the user can easily find out data information by forming a colormap.**

Keywords: Array delimiter, virtual sensor placement, coordinates location access, surface colormap display.

I. INTRODUCTION

Areas with complex bases measured can sometimes be a serious problem in obtaining accurate data. This is influenced bythe shape, and dimensions, or environmental conditions. Consider the type of sensor used. Expensive cost. Also, search for waypoints at a specific location. The placement of sensors as the main reference to obtain correct data information reflects the real conditions with visual conditions. One example is the measurement of the soles of the human feet. The shape that is curved in the middle, sloping, and towering must be handled with special measurements [1]. This measurement is intended to obtain data on the interface pressure of the soles of the feet and a number of sensors.

Displaying data properly from the results of sensor records in measurement or experiment is important. This becomes ineffective and inefficient when trying to represent very large numbers of data. How to differentiate the numbers in one box. This can be facilitated by the basis of numerical data being processed into color mapping [2]. With this transformation, the user will be faster in understanding the existing information.

Visualization of 2D data with color is vital in order to represent the measurement results (heat transfer, stress, etc.) to provide information for users in making it easier to read the analysis data [3]. The method applies a value matrix and an array outline as boundary generators for the graph. The measurement result value is obtained in real-time from the sensor that is turned on. The values will be assigned to the area determined according to their location. The purpose of this paper is to explain the method of making graphics with real-time data obtained from the microcontroller board and placing a number of sensors virtually within the boundary of an array frame shaped like a foot.

II. MATERIAL AND METHODS

Two Arduino Mega 2560 boards were used in this experiment. Each communication board connects directly to the computer's USB Port. This is to get a power supply on both microcontrollers [4]. On each board, there are 15 FSR 402 sensors that are stuck on pin A0 to pin A14. The total number of sensors used is 30.

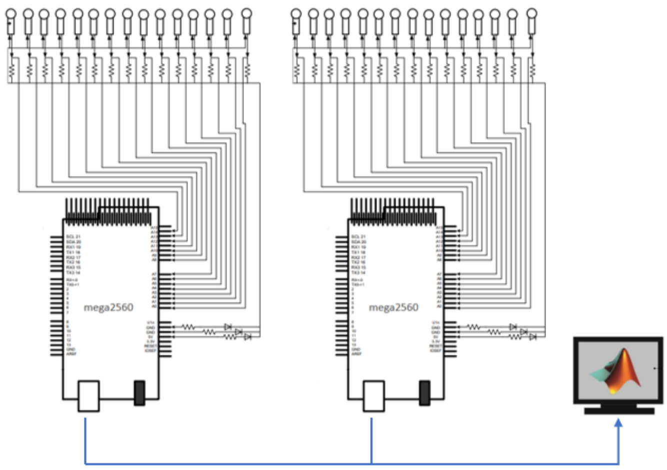

Figure 1: Hardware Design with Two Arduino Mega 2560 Boards

The first step of the microcontroller board is first to install it online by setting up Matlab and Arduino communications. This program is already available in the MATLAB library. Each board is given the initials a and b. this is intended to make it easier to indent data in syntax building. To be able to read data from sensors on each board using the same syntax, namely read Volatge for each analog pin (a, A0-A14) and (b, A0 - A14).

International Research Journal of Innovations in Engineering and Technology (IRJIET)

ISSN (online): 2581-3048

Volume 5, Issue 5, pp 137-142, May-2021

https://doi.org/10.47001/IRJIET/2021.505026

The placement of the sensor is in accordance with the predetermined division of the area. Divided into 4 areas on the soles of the feet. Heel or back center front and thumb. There are 3 sensors on the heel, 6 sensors in the middle, 5 sensors on the front, and 1 sensor on the thumb, see Figure 2 [5]. Each sensor is at different coordinates. The coordinates of the right foot are a reflection of the left foot where all values are positive. This is intended to facilitate the identification of array programs. Table 1 describes the locations of the sensors on the right and left feet in local coordinates.

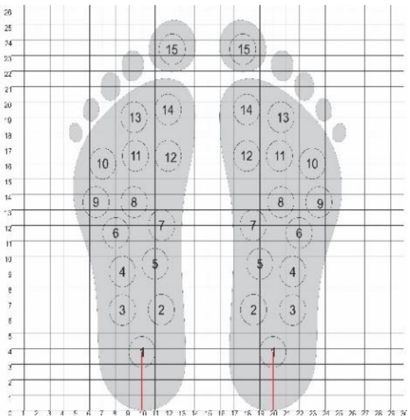

Figure 2: Sensor Numbering and Placement

Table 1: Local Coordinates of Sensor Placement

No. Sensor	Foot			
	Right		Left	
	X	Y	X	Y
1	50	38	-50	38
2	35	65	-35	65
3	65	65	-65	65
4	65	90	-65	90
5	40	95	-40	95
6	70	115	-70	115
7	35	120	-35	120
8	57	135	-57	135
9	85	135	-85	135
10	80	160	-80	160
11	55	165	-55	165
12	30	165	-30	165
13	55	190	-55	190
14	30	195	-30	195
15	27	234	-27	234

Process of Creating an Array Image and Source Code

CAD software helps to prepare the shape and dimensions of the array frame to be applied (Figure 3 to Figure 5). The shape is in the form of a curved line like the appearance of the sole of the foot. Line support that is ready to be used as a path to arrange points with a high-density level to produce a large number of coordinates. This is very important in producing a

Figure 6 shows a CAD .IGS file imported to MS. Excel to extract it to numeric form. The random number of extracted results looks bad on the MS homepage. Excel. Here, sorting is needed to determine the number that represents the X, Y, and Z coordinates. Usually, the number selected on MS. Excel shows the zero value in the Z coordinate. Figure 7 shows the local coordinates of the sensor in the extraction results where the Z value must have been given the access address for the location of each sensor. The sensor address will be directly connected by the program so that when there is a change in the input value of the sensor the number will follow it.

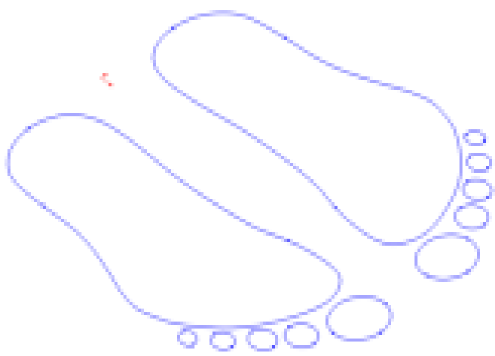

Figure 3: Sole of the Foot Display Design for Software Display

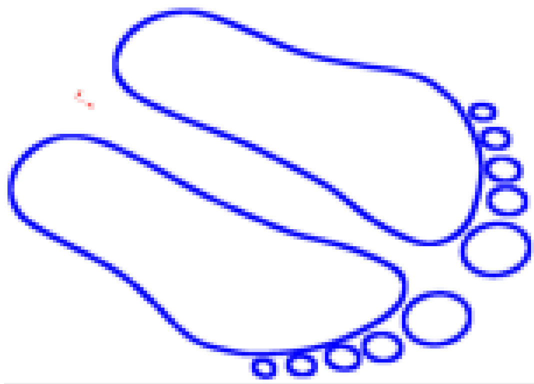

Figure 4: Points (Arrays) that Meet the Curved Lines

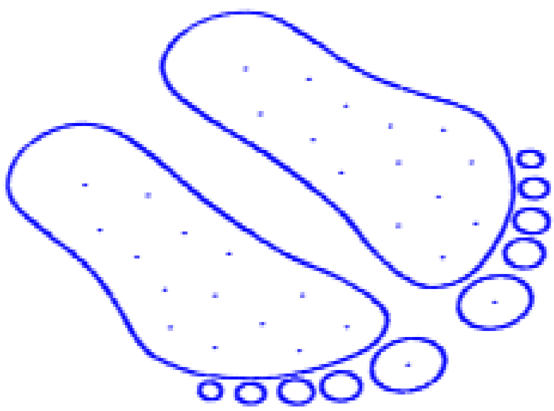

Figure 5: Sensor Point in the Boundary Array

International Research Journal of Innovations in Engineering and Technology (IRJIET)

ISSN (online): 2581-3048

Volume 5, Issue 5, pp 137-142, May-2021

https://doi.org/10.47001/IRJIET/2021.505026

-46.664261076	-0.14416382	0.
-39.294980173	0.785226214	0.
-39.835417241	0.664911284	0.
-40.363388331	0.555565048	0.
-41.048784247	0.425570234	0.
-41.9106393	0.281019607	0.
-42.497539816	0.194576656	0.
-47.64585388	-0.154483678	0.
-45.743082465	-0.110458133	0.
-49.194391177	-0.116891221	0.
-51.322673442	0.042990027	0.
-52.967288253	0.253615759	0.
-54.610083868	0.541392191	0.
-56.072098765	0.864172098	0.
-57.382772549	1.208755549	0.
-58.420222268	1.519719974	0.
52.967288253	0.253615759	0.
51.322673442	0.042990027	0.
49.194391177	-0.116891221	0.
45.743082465	-0.110458133	0.
47.64585388	-0.154483678	0.
42.497539816	0.194576656	0.
41.9106393	0.281019607	0.
41.048784247	0.425570234	0.
40.363388331	0.555565048	0.
39.835417241	0.664911284	0.
39.294980173	0.785226214	0.
46.664261076	-0.14416382	0.

Figure 6: Extraction Coordinate Number of the Array Point

X	Y	Z
-50	38	1
-35	65	2
-65	65	3
-65	90	4
-40	95	5
-70	115	6
-35	120	7
-57	135	8
-85	135	9
-80	160	10
-55	165	11
-30	165	12
-55	190	13
-30	195	14
-27	234	15
50	38	16
35	65	17
65	65	18
65	90	19
40	95	20
70	115	21
35	120	22
57	135	23
85	135	24
80	160	25
55	165	26
30	165	27
55	190	28
30	195	29
27	234	30

Figure 7: Extraction Coordinate Figure of the Sensor Point

Data from MS. Excel is then imported into MATLAB so that it becomes an embedded part of the program. The data that is entered is actually an array that is used as an object in making graphics. Variables that use arrays are reference variables. Variable declaration by specifying the identity in reading the data which is added behind the load syntax. Describing columns and rows is needed to find the sum by giving the initials M for rows and N for columns. The row and column sizes in the data matrix are displayed. Embed data from sensors in real-time and access it over again to display data changes on the sensor. The final stage is to plot all the data by running the program.

Another important addition to the graph produced by the program is how to easily distinguish the individual values on each sensor. The colors in this case are very helpful to represent the difference in value. The color that is often used is red-green-blue (RGB) [6]. To adjust RGB color in MATLAB, 'jet' colormap syntax is added. The final stage plots all the data by running the program. The following is the source code used:

```
clear all;
clc;
a = arduino();
b = arduino();
L00 = readVoltage(a,'A0');
L01 = readVoltage(a,'A1');

R13 = readVoltage(b,'A13');
R14 = readVoltage(b,'A14');
load P4.mat
M = 2883;
N = 3;
[i,j] = size(Data);
for i = 1:M
for j = 1:N
if Data (i,j) == 1
        Data (i,j) = L00;
elseif Data (i,j) == 2
        Data (i,j) = L01;

elseif Data (i,j) == 29
        Data (i,j) = R13;
    elseif Data (i,j) == 30
        Data (i,j) = R14;
end
end
end
whos Data;
Data=Data;
x=Data(:,1); y=Data(:,2); z=Data(:,3);
v_x=linspace(-150,150,500);
v_y=linspace(-50,300,500);
[xx,yy]=meshgrid(v_x,v_y); mesh(xx,yy,ones(500,500));
zz = griddata(x,y,z,xx,yy);
fig=figure; hold on;
set(fig,'renderer','zbuffer');
plot3(x,y,z,'.');
surf(xx,yy,zz);
colormap('jet');
shading flat;
yticklabels('manual')
xticklabels('manual')
box on
c = colorbar;
c.Label.String = 'Pressure (Pa)';
view(00,90); grid off
view(20,30); grid off
view(30,60); grid off
```

III. RESULTS AND DISCUSSIONS

With the GUI (graphical user interface) the user can easily give commands to the microcontroller. Send requests for sensor record data. Interacting the electronic components of the divider in representing the visual form of the data. GUI is the back of the interface programming [7]. GUI consists of several command components to carry out tasks. Figure 8 shows that the MATLAB GUI component is usually in the form of a box frame filled with command instructions. This GUI will show real-time data from a number of sensors installed on the measuring device and display the results of the sensor pressure track record. This GUI is also equipped with a probe feature, namely a cursor to show the interpolation

International Research Journal of Innovations in Engineering and Technology (IRJIET)

ISSN (online): 2581-3048

Volume 5, Issue 5, pp 137-142, May-2021

https://doi.org/10.47001/IRJIET/2021.505026

number of the load distribution results located in the visual net.

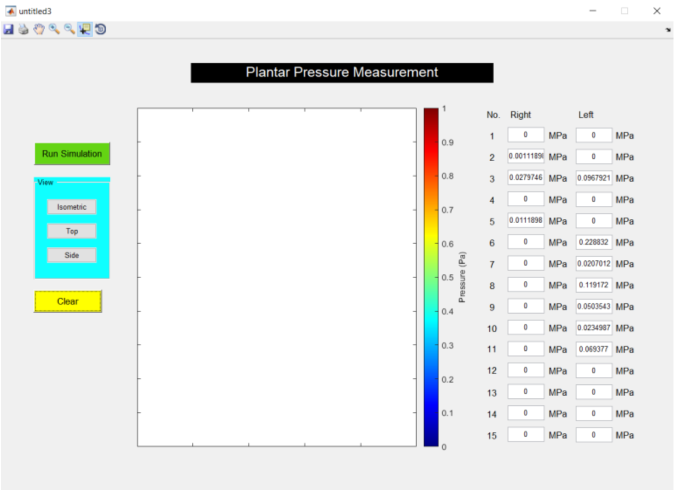

Figure 8: GUI Display to Monitor Data

The Run simulation box is a visual button for running software. By clicking Run simulation, the program will display the sensor record results and display the results of the interface pressure analysis on axes1 (the largest white background box). The right side of the software is equipped with a numeric box that will make it easier to read the interface pressure on each sensor. The figure box also shows which part of the sensor is receiving pressure peaks. Another additional feature is to view the graph results from several angles in the panel box (isometric, top, and side).

Colormap Chart Construction

The MATLAB workbox does not contain original point references like CAD software. The initial reference is assisted by making the object surface (mesh grid) as the bottom base. Figure 9. The surface object occupies the origin X and Y origin coordinates. The surface object is useful for helping in projecting the vector direction of the Z-axis. This technique is the basis for the formation of RGB colors to visualize the results of sensor data.

Figure 10 is a plot of the starting point position display of the sensor which is still at the bottom. This defines that the origin Z is still at zero all because there is no interface pressure interaction on the sensor. When the sensor starts to turn on, the visual sensor point will move upward away from the bottom bottom with a distance that represents the value of the voltage data sent by the microcontroller. This condition has not yet occurred staining. it is also difficult to explain the position of the sensor which is the farthest distance from the base (Figure 11).

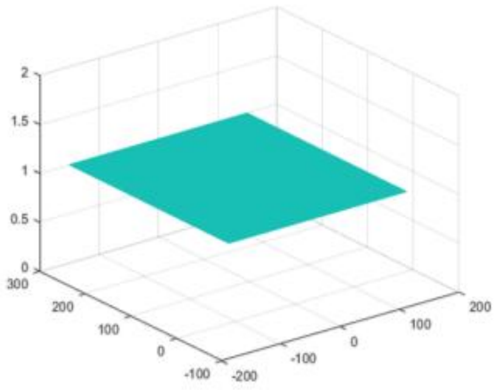

Figure 9: Base Surface (Mesh Grid)

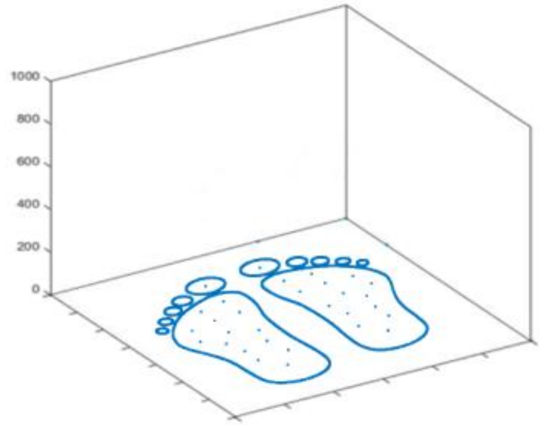

Figure 10: Initial Conditions of the Sensor Plot at Zero

When the sensor is on and sends the incoming voltage. Then one by one the points (visual sensors) in the array frame area will move away from the bottom base. The amount of distance depends on the value sent by the sensor. Next is the coloring process. There are many color maps in the MATLAB library. But the one who chose is red-green-blue (RGB). In this experiment, the data from the left foot dominates the farthest distance with a large amount of dark red filling every part of the foot area.

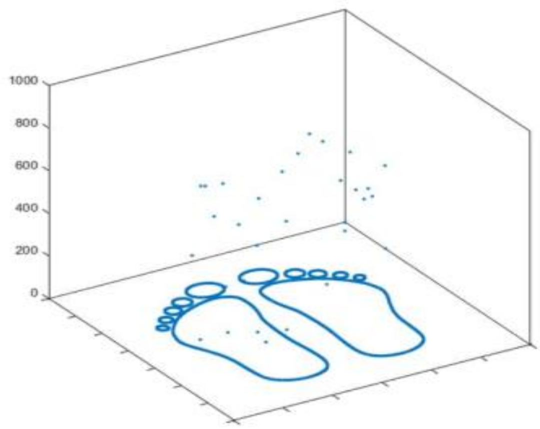

Figure 11: Displacement of the Sensor Away from the Base Surface

International Research Journal of Innovations in Engineering and Technology (IRJIET)

ISSN (online): 2581-3048

Volume 5, Issue 5, pp 137-142, May-2021

https://doi.org/10.47001/IRJIET/2021.505026

Figure 12 shows the colors starting to appear to form a surface. This can be seen by rotating the view at an isometric angle. The green color represents the distribution area. In measuring the distribution, the green color plays a role in stating the size of the area to identify scientific data in the form of values around the peak. This technique is great for generating interpolated data with only the placement of a few sensors.

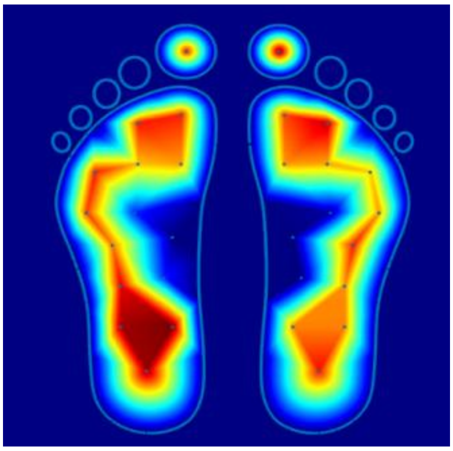

Figure 12: Formation of the Color Surface on the Sensor

The color image also gives an indication that dark red is the area close to the sensor. This makes it easier for users to detect coordinates around the sensor with the available probe features. This feature can access any location you want to find out the value at the X Y Z coordinates. It can also be used to find the signal points sent with the closest area received by the sensor.

IV. CONCLUSION

A color map visualization that represents a number of virtual sensors positioned within the array boundaries capable of producing accurate measurement data. Each sensor will be connected to form a color surface. By transforming the coordinate numbers into color it becomes easy to provide information to the user. The array formation adjusts the shape and dimensions as well as the actual conditions. This technique can be used for a variety of different types of sensors depending on measurement needs.

REFERENCES

[1] M. Bertamini, L. Palumbo, T. N. Gheorghes, M. Galatsidas, "Do observers like curvature or do they dislike angularity?," *Br J Psychol*, pp. 154–178, Feb. 2016.

[2] L. Zhou and C. D. Hansen, "A Survey of Colormaps in Visualization," vol. 2626, no. c, pp. 1–21, 2015, doi: 10.1109/TVCG.2015.2489649.

[3] J. R. Nu, C. R. Anderton, and R. S. Renslow, "Optimizing colormaps with consideration for color vision deficiency to enable accurate interpretation of scientific data," *no. Cvd*, pp. 1–14, 2018.

[4] V. M. Cvjetkovic and M. Matijevic, "Overview of architectures with arduino boards as building blocks for data acquisition and control systems," *Int. J. Online Eng.*, vol. 12, no. 7, pp. 10–17, 2016.

[5] D. B. Wibowo, A. Suprihanto, W. Caesarendra, and S. Khoeron, "A Simple Foot Plantar Pressure Measurement Platform System Using Force-Sensing Resistors," *Appl. Syst. Innov.*,pp. 1–10. Aug. 2020

[6] K. Moreland, "Diverging Color Maps for Scientific Visualization". *In: Bebis G. et al. (eds) Advances in Visual Computing. Lecture Notes in Computer Science*, vol 5876. Springer, Berlin, Heidelberg, ISVC, 2009.

[7] N. R. Shenoy, C. Kamala, and K. Vindhya, "a Matlab Gui : Designed to Perform Basic Image Processing Operations," *ijates*, vol. 4, no. 1, pp. 88–96, 2016.

AUTHOR'S BIOGRAPHIES

Slamet Khoeron, Master Student of Mechanical Engineering at Diponegoro University, Indonesia.

Dr. Dwi Basuki Wibowo, Lecturer at Diponegoro University (UNDIP) Semarang, Indonesia. Bachelor and Master Degrees in Mechanical Engineering from Bandung Institute of Technology, Indonesia. Doctoral Degree in Mechanical Engineering from Diponegoro University, Indonesia.

Ojo Kurdi PhD, Lecturer at Diponegoro University (UNDIP) Semarang, Bachelor Degrees in Mechanical Engineering from Diponegoro University, Indonesia. Master Degrees in Mechanical Engineering from Bandung Institute of Technology, Indonesia. Doctoral of Philosophy (PhD) from Universiti Teknologi Malaysia (UTM), Malaysia.

ISSN (online): 2581-3048
Volume 5, Issue 5, pp 137-142, May-2021
https://doi.org/10.47001/IRJIET/2021.505026

Citation of this Article:

Slamet Khoeron, D. B. Wibowo, Ojo Kurdi, "Array Application as a Limiting Image for Placement of Shaped Virtual Sensors Color Map Visualization" Published in *International Research Journal of Innovations in Engineering and Technology - IRJIET*, Volume 5, Issue 5, pp 137-142, May 2021. Article DOI https://doi.org/10.47001/IRJIET/2021.505026

****