Research Article

An Analytical Tool for Computing Perimeter Based on Visual and Coordinates Analysis

Mohd Agos Salim Nasir[1], Zubaidah Sadikin[2], Elaziera Panji[3], Nur Nadhirah Mohd Rohaidi[4], Nurzalina Harun[5], and Sidik Rathi[6, *]

1 Universiti Teknologi MARA Shah Alam; mohdagos066@uitm.edu.my; 0000-0002-8761-4595
2 Universiti Teknologi MARA Shah Alam; zubaidah1590@uitm.edu.my 0000-0003-3967-5243
3 Universiti Teknologi MARA Shah Alam; 2023125951@student.uitm.edu.my; 0009-0007-8096-1002
4 Universiti Teknologi MARA Shah Alam; 2023387733@student.uitm.edu.my; 0009-0002-7105-2062
5 Universiti Teknologi MARA Shah Alam; nurzalina@tmsk.uitm.edu.my; 0000-0003-0477-580X
6 Universiti Teknologi MARA Shah Alam; sidik8423@uitm.edu.my; 0000-0002-6408-3920
* correspondence: sidik8423@uitm.edu.my ; 60193693140

Abstract: *Perimeter calculation, a fundamental aspect of geometric analysis, plays a crucial role in numerous fields, ranging from architecture and engineering to urban planning and environmental assessment. Traditionally, manual methods for computing perimeters have been fraught with challenges, often requiring significant time investment and susceptible to errors, particularly when dealing with irregular shapes. The introduction of PeriGo represents a paradigm shift in perimeter calculation, offering an innovative software solution that streamlines the process through its user-friendly graphical interface. Leveraging advanced mathematical algorithms, including a straightforward formula for determining distances between points, PeriGo ensures precise results while mitigating the risk of human error. This transformative tool promises to revolutionize perimeter calculation across diverse industries, empowering users with accurate and efficient calculations in a fraction of the time. With its intuitive interface and robust functionality, PeriGo emerges as an indispensable asset for professionals seeking reliable geometric analysis. Moreover, recent research highlights the significance of algorithmic approaches in enhancing precision in geometric analysis, emphasizing the importance of user interface design considerations for geometric analysis software tools.*

Keywords: Perimeter; Coordinates; Efficient.

DOI: 10.5281/zenodo.11044196

1. INTRODUCTION

In today's dynamic landscape, accurate perimeter calculation is indispensable across diverse industries, from architecture and engineering to urban planning and environmental analysis (Saura et al., 2011). However, the manual computation of perimeters, particularly for irregular shapes, presents significant challenges that can impede efficiency and accuracy (Smith et al., 2019). These challenges include the time-intensive nature of manual calculations and the potential for errors to arise during the process. Recognizing the need for a more efficient solution, PeriGo emerged as a groundbreaking software tool designed to revolutionize perimeter calculation through its intuitive graphical user interface (GUI) and advanced mathematical algorithms.

PeriGo's development is rooted in the understanding that traditional methods of perimeter calculation often involve complex mathematical computations and meticulous measurements. For objects with irregular shapes, such calculations become even more intricate, demanding meticulous attention to detail and a substantial investment of time (Jones et al., 2020). PeriGo's primary objective is to alleviate these challenges by providing users with a streamlined and user-friendly solution that simplifies the process of perimeter calculation. By leveraging mathematical models, particularly a simple yet robust formula for determining distances between points, PeriGo aims to eliminate the guesswork and uncertainty associated with manual calculations.

Through its intuitive GUI, PeriGo enables users to input coordinates with ease, facilitating swift and accurate perimeter determination for any given object. With the ongoing advancement of control theory, analytical techniques, and computer technology, image processing technology has also progressed rapidly (Ke Xu et al., 2019). Therefore, PeriGo also allows users to upload images to obtain their parameters. By automating the computation process, PeriGo reduces the risk of human error, thereby enhancing the reliability and trustworthiness of the results it produces. As industries continue to evolve and demand increasingly sophisticated solutions, PeriGo stands poised to meet these challenges head-on, empowering users to achieve their objectives with confidence and efficiency (Brown & Johnson, 2021). In essence, PeriGo represents a pivotal advancement in perimeter calculation, offering users unparalleled accuracy and efficiency across a wide range of applications and domains.

2. METHOD & MATERIAL

Step 1: Data Collection

This study focuses on computing the arc length of the shoe insole through the utilization of the distance formula. Figure 2 illustrates a sketch of the shoe insole in figure 1, serving as the visual reference for the subsequent analysis. The shoe insole is represented by a total of 53 coordinates, obtained through a meticulous sketching method. These coordinates intricately outline the contour of the insole, accurately capturing its shape and defining its boundary for further analysis.

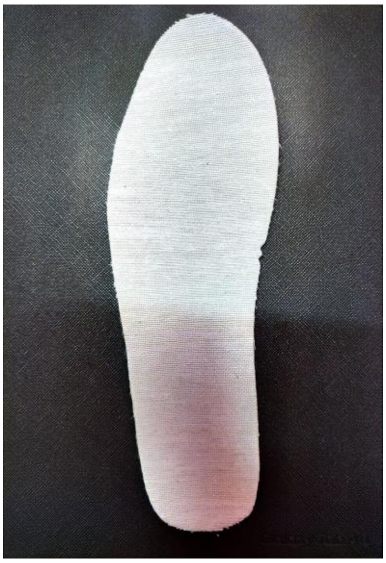

Figure 1: Original image of a shoe insole

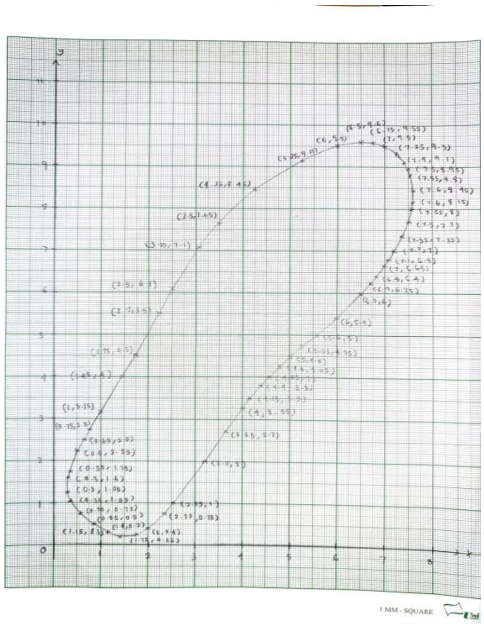

Figure 2: A plotted graph of a shoe insole

Step 2: Distance Calculation

Using the 53 coordinates, the distance between each pair of coordinates is computed utilizing the distance formula using Microsoft Excel Software. The distance formula employed in this study is

$$distance = \sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2} \tag{1}$$

which calculates the Euclidean distance between two points in a Cartesian coordinate system (Lang et al., 1988). This process ensures precise measurement of the distance along the insole's contour, facilitating accurate arc length computation.

Step 3: Perimeter Calculation

In this step, the distances obtained in Step 2 are summed up to determine the perimeter of the shoe insole. By aggregating the individual distances between consecutive coordinates, the total perimeter of the insole is derived. This comprehensive calculation accounts for all segments along the boundary of the insole, providing a comprehensive measurement of its total perimeter.

Step 4: Designing Interfaces

Utilizing the information gathered from the previous steps, interfaces are meticulously designed to automate the process without the need for manual calculations. These interfaces are tailored to seamlessly integrate with the data collection, distance calculation, and perimeter calculation steps, streamlining the entire process for user convenience. The design incorporates user-friendly features, allowing users to input or upload the necessary data effortlessly (Perry et al., 1989).

2.1 Flowchart of PeriGo System Prototype

The diagram below depicts Figure 3, which is the flowchart of the PeriGo system prototype.

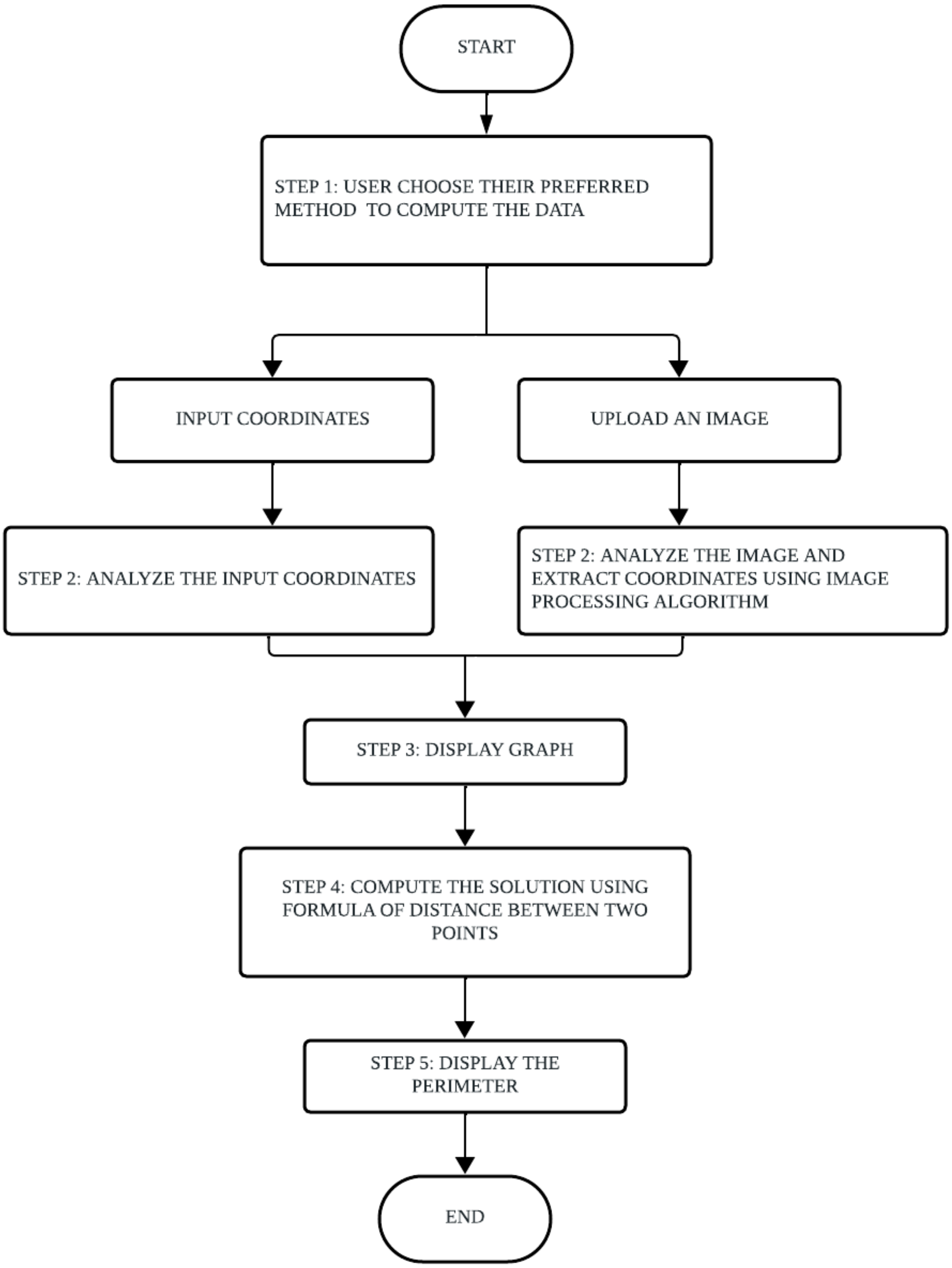

Figure 3: Flowchart of the PeriGo system prototype.

2.2 Graphical User Interface (GUI) of PeriGo

The figures below show the Graphical User Interface (GUI) of the PeriGo system prototype. PeriGo has a total of four interfaces.

Figure 4: First interface of PeriGo: Press "START" to initiate PeriGo.

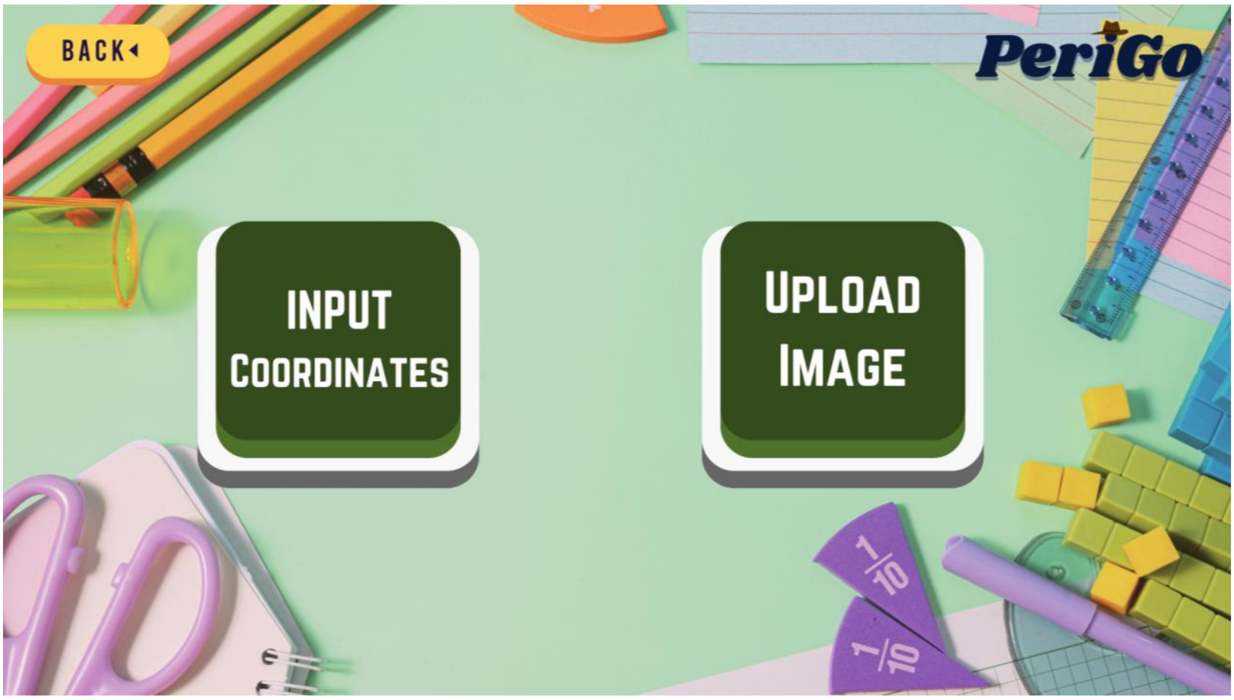

Figure 5: Second interface of PeriGo: Choose either input coordinates or upload image to find the perimeters.

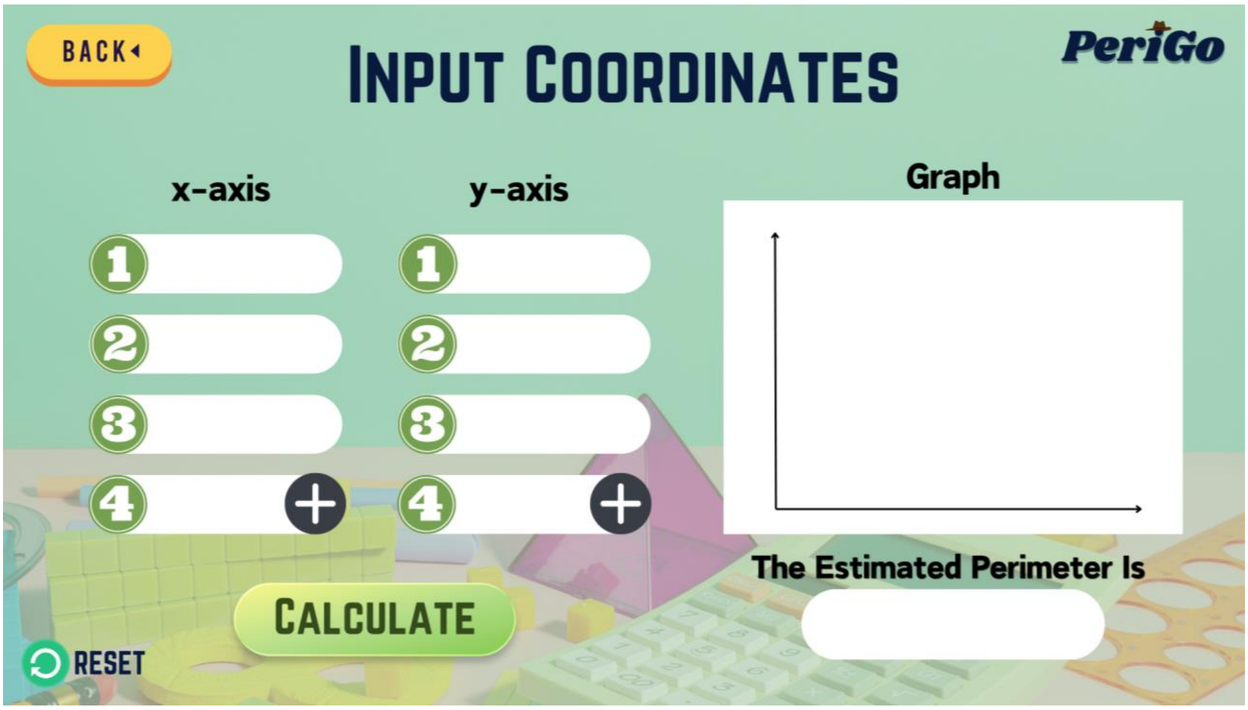

Figure 6: Third interface of PeriGo: Input Coordinates interface: Insert x-axis and y-axis coordinates. Press "CALCULATE" to obtain the graph and the perimeter.

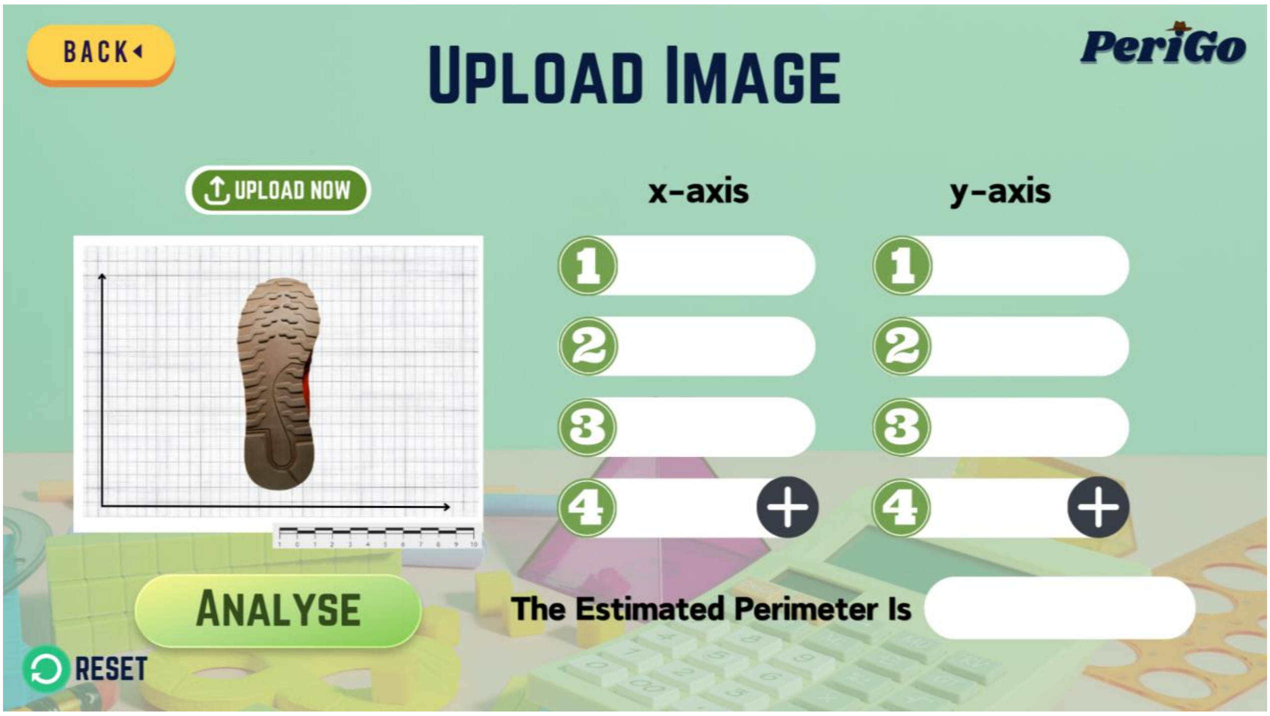

Figure 7: Fourth interface of PeriGo: Upload image interface: Upload an image. Press "ANALYSE" to obtain the graph, coordinates, and perimeter.

3. FINDINGS

3.1 The figures and the data for the prototype

Figure 8 depicts the illustration of the shoe insole along with its corresponding coordinates.

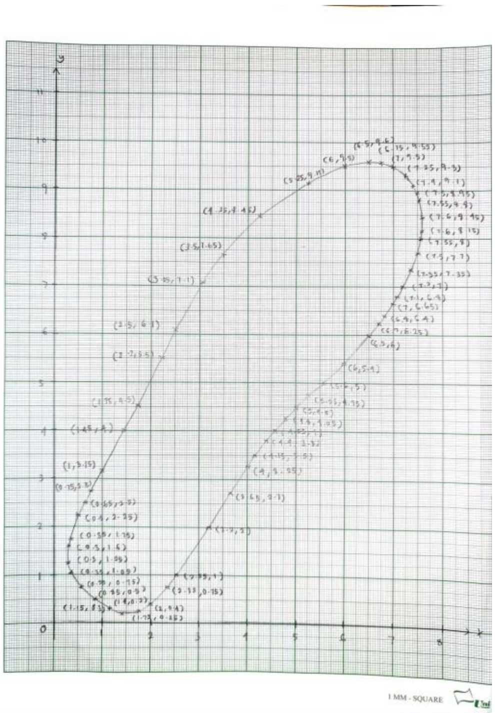

Figure 8: A plotted graph of a shoe insole.

Table 1 shows the successfully gathered data, comprising a total of 53 data points. In this context, 'x' represents the coordinate on x-axis, while 'y' represents the coordinate on y-axis.

Table 1: The coordinates of a shoe insole

x	**2.0**	**2.35**	**2.55**	**3.20**	**3.65**	**4.00**	**4.15**	**4.40**	**4.55**	**4.80**	**5.00**	**5.25**	**5.60**
y	**0.40**	**0.75**	**1.00**	**2.00**	**2.70**	**3.25**	**3.50**	**3.80**	**4.00**	**4.25**	**4.50**	**4.75**	**5.00**

x	**6.00**	**6.50**	**6.70**	**6.80**	**7.00**	**7.10**	**7.20**	**7.35**	**7.50**	**7.55**	**7.60**	**7.60**	**7.55**
y	**5.40**	**6.00**	**6.25**	**6.40**	**6.65**	**6.80**	**7.00**	**7.35**	**7.70**	**8.00**	**8.15**	**8.45**	**8.80**

x	**7.50**	**7.40**	**7.25**	**7.00**	**6.75**	**6.50**	**6.00**	**5.25**	**4.25**	**3.50**	**3.05**	**2.50**	**2.20**
y	**8.95**	**9.10**	**9.30**	**9.50**	**9.55**	**9.60**	**9.50**	**9.15**	**8.45**	**7.65**	**7.10**	**6.10**	**5.50**

x	**1.75**	**1.45**	**1.00**	**0.75**	**0.65**	**0.50**	**0.35**	**0.30**	**0.35**	**0.55**	**0.85**	**1.15**	**1.40**	**1.75**
y	**4.50**	**4.00**	**3.15**	**2.75**	**2.50**	**2.25**	**1.75**	**1.25**	**1.05**	**0.75**	**0.50**	**0.30**	**0.20**	**0.25**

From Table 1 above, we will compute the distance between two points for each set of coordinates using the distance formula. Figure 9 below shows the results of distance and perimeter computations using Microsoft Excel Software.

x2	1.75	2	2.35	2.55	3.2	3.65	4	4.15
y2	0.25	0.4	0.75	1	2	2.7	3.25	3.5
x1	2	2.35	2.55	3.2	3.65	4	4.15	4.4
y1	0.4	0.75	1	2	2.7	3.25	3.5	3.8
distance	0.2915	0.4950	0.3202	1.1927	0.8322	0.6519	0.2915	0.3905

x2	4.4	4.55	4.8	5	5.25	5.6	6	6.5
y2	3.8	4	4.25	4.5	4.75	5	5.4	6
x1	4.55	4.8	5	5.25	5.6	6	6.5	6.7
y1	4	4.25	4.5	4.75	5	5.4	6	6.25
distance	0.2500	0.3536	0.3202	0.3536	0.4301	0.5657	0.7810	0.3202

x2	6.7	6.8	7	7.1	7.2	7.35	7.5	7.55
y2	6.25	6.4	6.65	6.8	7	7.35	7.7	8
x1	6.8	7	7.1	7.2	7.35	7.5	7.55	7.6
y1	6.4	6.65	6.8	7	7.35	7.7	8	8.15
distance	0.1803	0.3202	0.1803	0.2236	0.3808	0.3808	0.3041	0.1581

x2	7.6	7.6	7.55	7.5	7.4	7.25	7	6.75
y2	8.15	8.45	8.8	8.95	9.1	9.3	9.5	9.55
x1	7.6	7.55	7.5	7.4	7.25	7	6.75	6.5
y1	8.45	8.8	8.95	9.1	9.3	9.5	9.55	9.6
distance	0.3000	0.3536	0.1581	0.1803	0.2500	0.3202	0.2550	0.2550

x2	6.5	6	5.25	4.25	3.5	3.05	2.5	2.2
y2	9.6	9.5	9.15	8.45	7.65	7.1	6.1	5.5
x1	6	5.25	4.25	3.5	3.05	2.5	2.2	1.75
y1	9.5	9.15	8.45	7.65	7.1	6.1	5.5	4.5
distance	0.5099	0.8276	1.2207	1.0966	0.7106	1.1413	0.6708	1.0966

x2	1.75	1.45	1	0.75	0.65	0.5	0.35	0.3
y2	4.5	4	3.15	2.75	2.5	2.25	1.75	1.25
x1	1.45	1	0.75	0.65	0.5	0.35	0.3	0.35
y1	4	3.15	2.75	2.5	2.25	1.75	1.25	1.05
distance	0.5831	0.9618	0.4717	0.2693	0.2915	0.5220	0.5025	0.2062

x2	0.35	0.55	0.85	1.15	1.4
y2	1.05	0.75	0.5	0.3	0.2
x1	0.55	0.85	1.15	1.4	1.75
y1	0.75	0.5	0.3	0.2	0.25
distance	0.3606	0.3905	0.3606	0.2693	0.3536

perimeter 24.8565

Figure 9: Results of distance and parameter computations in Microsoft Excel.

As shown in the Figure 9, we compute the distance by using the distance formula given in (1). After all the distances have been computed, we sum them up to obtain the perimeter of the shoe insole. The obtained perimeter is 24.8565.

4. DISCUSSION

4.1 Explanation on findings

In this section , we will elaborate on the findings from our research. Our chosen subject for data collection is a shoe insole. We represented the shoe insole on a graph and recorded its coordinates as 'x' for the x-axis and 'y' for the y-axis. We successfully gathered a total of 53 data points throughout the process. To calculate the perimeter of the shoe insole, we employed the distance formula between two coordinates. We divided the segment between two points on the shoe insole into smaller parts with shortest distance. This approach was adopted to increase accuracy, given that the formula is specifically designed for a straight line (Dong et al., 2001).

As mentioned earlier, we applied distance formula between two points to calculate the length between two coordinates. For instance, let $(x_1, y_1) = (2,0.4)$ and $(x_2, y_2) = (1.75,0.25)$. Then, we obtain

$$d = \sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2} = \sqrt{(1.75 - 2)^2 + (0.25 - 0.4)^2} = 0.2915.$$

We repeat the process for the remaining points, and then calculate the total distance to determine the perimeter of the entire shoe insole.

4.2 Explanation on PeriGo's interfaces

Next, we will delve into the interfaces and functions of PeriGo. In this project, we introduce a prototype named "PeriGo" designed to calculate the perimeter of an object based on coordinates. The prototype includes four interfaces.

In the first interface, users must click the "START" button to activate PeriGo. This action then directs users to the subsequent interface, offering them two options. The first option, "INPUT COORDINATES" leads users to a distinct interface where they input the coordinates of their desired object and click the "CALCULATE" button to determine its perimeter. A graph representing the object based on the input coordinates will be displayed, providing users with a visual representation of their object's shape. This option caters to users with sufficient information about the coordinates of their object.

The second option, "UPLOAD IMAGE" guides users to the following interface, where they need to upload an image of the object for which they want to determine the perimeter. Subsequently, they must click the "ANALYSE" button to obtain the perimeter, which will be displayed along with corresponding coordinates on the screen. The system will utilize image processing methods to extract useful information from the uploaded image (Boming Liu et al., 2018). To enable subsequent processing on the image, the system will conduct noise reduction procedures on it (Zihao Ma, 2023). Users have the flexibility to edit the generated coordinates according to their preferences to achieve the desired result. This option is suitable for users who are lacking information about their object's coordinates. However, for optimal accuracy, users must provide a clear and sharp image to achieve the best results.

5. CONCLUSION

In conclusion, the utilization of the distance formula between two points has proven to be a reliable method for determining the perimeter of an object, whether it has a regular or irregular shape. While calculating the perimeter of an irregular shape might pose challenges using this formula, these difficulties are effectively addressed by dividing the distance between two points of the object into smaller segments. Therefore, PeriGo emerges as a valuable tool for individuals seeking high accuracy in determining the perimeter of an object within a short time. Its versatility and ability to handle irregular shapes make it a practical and efficient solution for various applications requiring precise measurements. As a recommendation, researchers and developers should collaborate to prioritize the integration of user-friendly interfaces in geometric software. This collaboration should draw upon principles from graph theory and coordinate geometry, including the distance formula, with the aim of enhancing accuracy. This joint effort is designed to guarantee that geometric applications comprehensively meet the requirements of researchers, educators, and students across diverse fields of study.

Acknowledgments: Elaziera and Nur Nadhirah acknowledged the support of the lecturers which are Mohd Agos Salim Nasir, Nurzalina Harun, Zubaidah Sadikin, and Sidik Rathi from College of Computing, Informatics and Mathematics.

References

Brown, E., & Johnson, F. (2021). Advancements in perimeter calculation software. Proceedings of the International Conference on Computer Science and Technology, 78-89.

Dong, Y., & Hillman, G. R. (2001). Three-dimensional reconstruction of irregular shapes based on a fitted mesh of contours. *Image and Vision Computing*, *19*(3), 165-176.

Jones, B., Lee, C., & Wang, D. (2020). The complexities of perimeter computation for irregular shapes. Mathematical Applications in Engineering, 15(3), 102-115.

Lang, S., Murrow, G., Lang, S., & Murrow, G. (1988). The Distance Formula. *Geometry: A High School Course*, 110-122.

Liu, B., Ma, Y., Liu, J., Gong, W., Wang, W., & Zhang, M. (2018). Graphics algorithm for deriving atmospheric boundary layer heights from CALIPSO data. *Atmospheric Measurement Techniques*, *11*(9), 5075-5085.

Ma, Z. (2023). Research on calculation strategy of perimeter of irregular objects. Highlights in Science, Engineering and Technology, 49, 220-226.

Perry, T. S., & Voelcker, J. (1989). Of mice and menus: designing the user-friendly interface. *IEEE Spectrum*, *26*(9), 46-51.

Saura, S., & Martinez-Millan, J. (2001). Sensitivity of landscape pattern metrics to map spatial extent. Photogrammetric engineering and remote sensing, 67(9), 1027-1036.

Smith, A. (2019). Challenges in manual perimeter calculation. Journal of Engineering and Technology, 10(2), 45-56.

Xu, K., Li, Y., & Xiang, B. (2019). Image processing-based contour parallel tool path optimization for arbitrary pocket shape. *The International Journal of Advanced Manufacturing Technology*, *102*, 1091-1105.