IJCRT.ORG ISSN : 2320-2882

INTERNATIONAL JOURNAL OF CREATIVE RESEARCH THOUGHTS (IJCRT)

An International Open Access, Peer-reviewed, Refereed Journal

Visualization of sorting algorithms

Shubhangi S. Mahale ,[1] Pro ,Nilesh V[2]ani

[1] PG Student, id: 0009-0001-5234-111X

Godavari College of Engineering, (NAAC Accredited), Jalgaon, Maharashtra, India – 425003

Assistant P[2] rofessor, id: 0009-0003-0476-0349

Godavari College of Engineering, (NAAC Accredited), Jalgaon, Maharashtra, India – 425003

ABSTRACT

This paper focuses on reviewing sorting algorithm visualizers and their role in improving the understanding of algorithms. Visualization helps in learning complex concepts in a simple and interactive way. The proposed system demonstrates sorting algorithms step-by-step using graphical representation, making learning easier and more effective..

KEYWORDS

Sorting Algorithms, Visualization, Data Structures, Bubble Sort, Quick Sort, Learning Tool

INTRODUCTION

In computer science, sorting is a basic and important concept used to arrange data in a particular order such as ascending or descending. It is widely used in many areas like searching, data analysis, and database management.

However, many students find it difficult to understand how sorting algorithms actually work step-by-step. The internal process of comparison and swapping is not easy to visualize using traditional teaching methods.

To solve this problem, a Sorting Algorithm Visualizer is developed. It is an interactive tool that shows how different sorting algorithms work using graphical representation. Users can see each step of the algorithm in real time, which makes learning easier and more interesting.

This project mainly focuses on improving the understanding of sorting algorithms by using visualization and user interaction.

This paper focuses on reviewing sorting algorithm visualizers, their features, and their role in improving the understanding of algorithms. It also highlights the importance of visualization in learning complex concepts in a simple and interactive way.

1.1 System Architecture

The Sorting Algorithm Visualizer system is designed to help users understand sorting algorithms through graphical representation.

The system consists of three main modules:

1. Input Module
   - User enters the size of the array
   - User can generate random values or enter custom values
   - Input is displayed in the form of bars
2. Processing Module
   - User selects a sorting algorithm:
     - Bubble Sort
     - Selection Sort
     - Insertion Sort
     - Merge Sort
     - Quick Sort
   - The selected algorithm starts execution
   - System processes data step-by-step
   - Comparisons and swaps are calculated internally
3. Visualization Module
   - Data is shown using vertical bars
   - Each bar represents an element of the array
   - 1. Different colors are used:
     - Comparing elements → highlighted
     - Swapping elements → change color
     - Sorted elements → final color
   - Animation shows real-time movement of elements

The system follows these steps:

1. User selects the sorting algorithm
2. User inputs or generates array
3. Visualization starts
4. Elements are compared
5. Swapping is performed if needed
6. Steps are repeated until array is sorted
7. Final sorted output is displayed

Visualization Module

- Data is shown using vertical bars
- Each bar represents an element of the array
- Different colors are used:
  - Comparing elements → highlighted
  - Swapping elements → change color
  - Sorted elements → final color
- Animation shows real-time movement of elements

LITERATURE REVIEW

Many researchers have focused on improving the understanding of sorting algorithms using visualization te. These tools use animations and graphical representations to display the step-by-step working of algorithms. Users can observe how elements are compared and swapped during the sorting process, which helps in better understanding.

overcome this issue, various sorting algorithm visualizers have been developed…" These tools use animations and graphical representations to display the step-by-step working of algorithms. Users can observe how elements are compared and swapped during the sorting process, which helps in better understanding. Tc

In addition, modern approaches such as web-based applications and mobile applications have made these visualization tools more accessible. These tools provide a user-friendly interface and real-time feedback, which improves learning efficiency.

Overall, previous studies show that visualization plays an important role in simplifying complex algorithm concepts and enhances the learning experience for students.

METHODOLOGY

The methodology of this project focuses on designing and developing a Sort

Algorithm Visualizer that helps users understand sorting techniques throu

graphical representation.

The system works in a step-by-step process. First, the user enters the input data

the form of an array. The user can either generate random values or provide cust input values. After that, the user selects a sorting algorithm such as Bubble S Selection Sort, Insertion Sort, Merge Sort, or Quick Sort.

Once the algorithm is selected, the system starts processing the data. Dur execution, the algorithm performs comparisons and swapping of elements. Th

operations are displayed visually using bars, where each bar represents an elem

of the array.

Different colors are used to indicate different actions, such as comparis

swapping, and completion of sorting. The visualization is updated continuously

that users can observe each step clearly.

Finally, the system displays the sorted output. This step-by-step visualization he

users understand how sorting algorithms work internally and improves th

learning experience.

The working of the system is carried out in the following steps:

1. Input Data:
2. The user enters the array elements. The system also provides an option generate random values.
3. Algorithm Selection:
4. The user selects a sorting algorithm such as Bubble Sort, Selection S Insertion Sort, Merge Sort, or Quick Sort.
5. Execution Start:
6. After selecting the algorithm, the system begins processing the data.
7. Comparison of Elements:
8. The algorithm compares elements step-by-step as per its logic.
9. Swapping Process:
10. If required, elements are swapped and updated in the visualization.
11. Visualization:
12. The entire process is shown using graphical bars with different col

representing comparison, swapping, and sorted elements.

13. Final Output:
14. After completion, the system displays the sorted array.

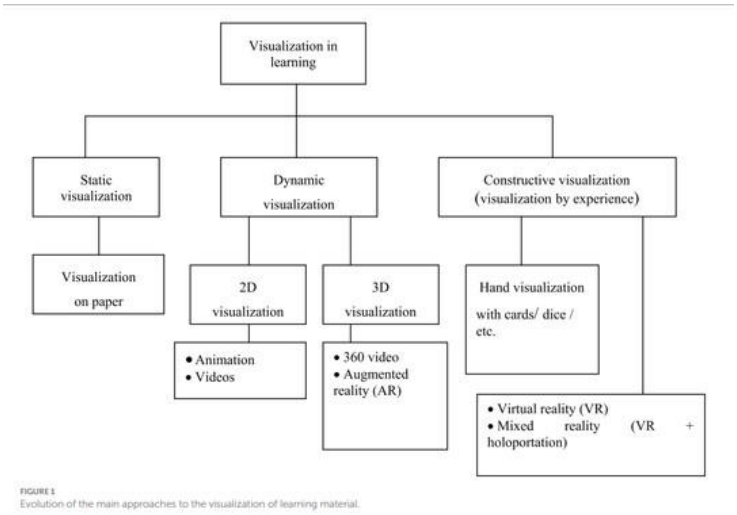

FIGURE 1
Evolution of the main approaches to the visualization of learning material.

The methodology of the proposed system focuses on the design and implementation of a Sorting Algorithm Visualizer that helps users understand sorting techniques through graphical representation.

Working Process

The working of the system is carried out in the following steps:

1. The user enters the input data (array of elements).
2. The user selects a sorting algorithm (e.g., Bubble Sort, Quick Sort).
3. The system processes the selected algorithm.
4. The visualization module displays step-by-step comparison and swapping.
5. The final sorted output is shown along with performance metrics.

Algorithm Implementation

The system implements multiple sorting algorithms:

- Bubble Sort
- Selection Sort
- Insertion Sort
- Merge Sort
- Quick Sort

Each algorithm is executed step-by-step, and intermediate steps are displayed visually using bars or graphical

RESULTS AND DISCUSSION

The implementation of the Sorting Algorithm Visualizer shows that visualization significantly improves the understanding of sorting algorithms.
Users can easily identify how different algorithms behave and compare their performance. The interactive interface increases user engagement and helps in better learning.

CONCLUSION:

The Sorting Algorithm Visualizer is an effective educational tool that simplifies the learning of sorting techniques. By providing step-by-step visualization, it helps users understand complex concepts in an easy and interactive way.
The system can be further enhanced by adding more algorithms and advanced visualization features. Overall, this project contributes to improving learning methods in computer science education.

REFERENCES

- G. Prabhakar et al., "Analysis of Algorithm Visualize to Enhance Academic Learning," 2022
- N. Yadav et al., "Path Finding Visualizer Application," 2021
- B. Nagaria et al., "Feedback Tool to Teach Algorithms," 2021
- A. B. Ghandge et al., "AlgoAssist Platform," 2021