OPEN ACCESS Freely available online

Journal of Theoretical & Computational Science

Opinion Article

Limitations and the Study of Algorithms in Computational Geometry

Vishwanathan Kumar*

Department of Medicine, University of California Irvine, California, USA

DESCRIPTION

The study of algorithms that can be expressed in terms of geometry is the focus of the computer science subfield known as computational geometry. The study of computational geometric methods can lead to some purely geometrical issues, and these issues are also regarded as a subset of computational geometry. Although the field of current computational geometry is relatively new, it is one of the oldest in computers, with roots dating back to antiquity.

In the mathematical discipline of computational geometry, effective algorithms for resolving geometric input and output issues are designed, analyzed, and put into practice. It can also refer to the solid modeling methods used to control surfaces and curves, as well as pattern recognition.

In the field of computer science known as computational geometry, algorithms that can be represented using different types of geometry are studied. Though modern computational geometry is a more recent development, historically speaking, it is regarded as one of the earliest computing topics. Computational geometry has mostly evolved as a result of advancements in computer graphics, as well as computer-aided design and production.

The advancement of computer graphics and computer-aided design and manufacturing (CAD/CAM) provided the primary impetus for the development of computational geometry as a subject, but many computational geometry issues are classical in nature and may result from mathematical visualization.

Robotics (motion planning and visibility issues), Geographic Information Systems (GIS) (geometrical location and search, route planning), Integrated Circuit design (IC geometry design and verification), Computer-Aided Engineering (CAE) (mesh generation), and computer vision are some additional significant applications of computational geometry (3D reconstruction). Parametric curves and surfaces, such as Bézier curves, spline curves, and surfaces, are the most crucial tools in this situation. The level-set method is a crucial non-parametric strategy.

This problem is typically considered a single-shot problem, or as falling within the first class, in many applications. For instance, determining which part of the screen is being clicked by a pointer is a common issue in many computer graphics programs. The polygon in question, however, is invariant in some applications, whereas the point denotes a query. Determine whether an aircraft crossed a boundary by using an input polygon that represents a country's border and a point that represents an aircraft's position. There may be realistic assumptions for the order of the inquiries in various query problem situations, which can be used for either effective data structures or more accurate estimations of computational complexity. Insights from computational graph theories used in natural geometric settings were also influenced by its development.

Problems in two-dimensional space and rarely three-dimensional space were the primary emphasis of this field in the beginning. Making theoretical discoveries more approachable for practitioners has been the focus of the majority of current research in this area. By resolving geometric degeneracies, streamlining current techniques, and building geometric libraries, this has been accomplished. They also concentrated on the characteristics of geometric problems rather than conventional continuous questions. Most of the objects in this field are flat and straight, including lines, polygons, planes, and line segments. It occasionally employs curved objects, such as circles, but stays away from solid modeling issues with intricate curves and surfaces.

Computation-based geometry may never fully satisfy practitioners' and their application areas' needs for several number of good reasons. Road networks and geographic information systems are only two examples of the many natural things that can be discretized into line segment collections. Engineers and designers that work with robotics, fluid dynamics, and solid modeling struggle to meet their objectives because computation-based geometry generally works with flat and straight objects.

Computational geometers were liberated from the unpleasantness of simulating the complexity and flaws of the physical world and dealing with the constraints of realistic computer hardware thanks to this inward research approach. They were able to concentrate on the analysis of a Platonic

Correspondence to: Dr. Vishwanathan Kumar, Department of Medicine, University of California Irvine, California, USA, E-mail: vishwak9@yahoo.com

Received: 29-Jun-2022, Manuscript No. JTCO-22-18930; **Editor assigned:** 01-Jul-2022, Pre QC No. JTCO-22-18930 (PQ); **Reviewed:** 15-Jul-2022, QC No. JTCO-22-18930; **Revised:** 22-Jul-2022, Manuscript No. JTCO-22-18930 (R);**Published:** 02-Aug-2022, DOI: 10.35248/2376-130X.22.8.154.

Citation: Kumar V (2022) Limitations and the Study of Algorithms in Computational Geometry. J Theor Comput Sci. 8:154.

universe made up of straightforward, well-behaved geometric objects that could be operated on by hypothetical computers with infinite memory and real number arithmetic.

The good news is that technological advancements have made it possible for computational geometers to model curved things using polygons or polygonal. This is the reason computation-based geometry is gaining popularity because it now calls for more technical and computer abilities rather than analytic and differential geometry knowledge. The majority of the field's attention is directed toward two-dimensional issues because they are simple to comprehend and depict, but this restricts its ability to address the most difficult three-dimensional application challenges are higher.

A vital stage in the discipline's development is being experienced computational geometry, which is a vibrant field. For the development of the field and its influence on applications, we have defined methodologies and computing paradigms that we believe are strategically important. The most important takeaway is that computational geometry should reiterate its dual aim of examining the combinatorial structure of geometric objects and offering useful methods and tools for the analysis and resolution of basic geometric issues.