International Journal of Research Publication and Reviews

Journal homepage: www.ijrpr.com ISSN 2582-7421

Autonomous Intelligent Vehicle

***Prof. Sunil Khot*[1]*, Mr. Pratik Edake*[2]*, Ms. Sayali Patil*[3]*, Mr. Rohit Shinde*[4]**

[1 2 3 4] **Department of Electronics and Telecommunication SCTR's Pune Institute of Computer Technology Pune, India**

ABSTRACT—

In a dynamic environment, autonomous intelligent vehicles (AIVs) can function independently of human drivers. Autonomous cars are the smart cars anticipated to be driver less, efficient and crash avoiding ideal urban vehicles in near future. In the modern vehicles, we can see various levels of automation where the vehicle can drive by self, some can have only some type of automation. An AIV must be able to create a precise map of the area where it is intended to operate and localize itself on that map in order to be considered completely autonomous. We have created an AIV that can create a map, locate itself on the map, and track its every move. Our AIV, in contrast to the majority of AIVs, is fully autonomous because it doesn't require any outside instruction. For the AIV's position tracking, map construction, and map updating processes, we created three brand-new algorithms. We have created the Map Building and Grid Fitting (MBGF) technique to create an outline map of the region where AIV is expected to operate using only pure geometry. We have created a Dynamic Map Building (DMB) technique to update the position when AIV moves. We have created the Runtime Position Tracking (RPT) algorithm, which uses the technologies of motor encoders and lidar sensors to localize the vehicle and track each and every movement of the vehicle. The vehicle creates a map of the region with the least amount of human help, performs localization, movement tracking, and course planning without the need for external feedback, and continuously updates the position of obstacles on the map. We have established via rigorous testing that the combination of all the algorithms makes the AIV fully autonomous and intelligent.

Keywords— Autonomous Intelligent Vehicles, AIV,Cartography, Motor encoders, Lidar, Localization, etc.

INTRODUCTION

Vehicles that can move people or products from one locationto another are becoming more and more necessary, especiallyindoors. E-commerce firms like Amazon, Flipkart, Alibaba, and others have enormous warehouses, and they now rely ontiny indoor vehicles to transfer goods throughout these vast warehouses. These earlier vehicles were referred to as autonomous guided vehicles (AGVs) because they relied on outside sensors or pathways marked on the ground to direct them from one location to another. These cars are getting smarter all the time thanks to recent developments in automation, robotics, vision, and sensor technologies. Such cars no longer require an external guidance system, which isa daily improvement. Applications for AIVs are becoming more diverse. Heavy products are transported by AIVs, whichare employed as autonomous forklifts. In the automobile industry, enormous AIVs are utilized in place of assembly lines, where the chassis is mounted on the AIV and the AIV transports the chassis from one location to another. These AIVs are also utilized in parking lots to automate the valet parking process. For many of these AIVs, the entire system, including the sensors and algorithms, can be calibrated and used. We propose a fully tested, start to finish solution for AVs using three innovative algorithms.

LITERATURE REVIEW

Any robotic system must be designed effectively, and knowledge of current technologies is crucial. Mechanical architecture, sensor technology, and navigation and tracking control strategies are essential for the development of mobilerobots. In this literature study, we examined methods for localization, path planning, map building, and vehicle tracking.

An intelligent autonomous vehicle can find its own way from a starting point to a predetermined destination. The navigation of AIVs in an indoor setting is covered in great length in this document.

This document discusses how to localise the vehicle and how lidar sensors are used in autonomous vehicles. Theauthor focused on three techniques for locating a car. the firstaccurate standstill detection that merely relied on signals from an IMU. Second, statistical filtering is applied to determine the vehicle's condition. Third, a positioning method based on LiDAR that offers data on corrected position and orientation.

Autonomous mobile robots observe their surroundings and make appropriate driving judgements. Autonomous vehicles use a variety of sensors to sense their surroundings, including cameras, radar, ultrasonic sensors, and LiDAR. These heterogeneous sensors simultaneously record a variety of environmental physical parameters. In order to map the functional area of AIV, this study demonstrates a specific method for building an autonomous mobile robot that uses aLIDAR for free space detection. It also demonstrates how toalign the output of these sensors using a geometrical model.

Currently, LiDARs are employed in a wide range ofindustries, such as robotics, agriculture, archaeology, and themeasurement of various atmospheric components. The current manuscripts discuss LiDAR's functionality, as well asits history and different uses. LiDAR data can be used to compute the separation between different objects in space andto create a 3D digital image of the scene in front of the LiDAR. The ideal lidar, the target identification algorithm, and an assessment of the lidar data accuracy and improvements using lidar-specific ground control targets in the case of diverse target types are all covered in detail in thispaper's performance analysis.

Autonomous intelligent vehicles are widely used to imitate real-world driving behaviour and test a variety of algorithms.This study uses a Raspberry Pi and a LIDAR module createdspecifically for interior navigation to show how an autonomous vehicle deployment successfully overcomes thischallenge. If we use LIDAR, we have to keep a dataset, and LIDAR is very expensive compared to IMU (magnetometer and optical encoder). The output from these sensors will be adequate for the bot to understand its location if they are fullycalibrated.

Localization is the challenge of following a movingrobot in an environment with poor sensor data and awarenessof the motion of the vehicle. This paper illustrates one approach to the bot's localization issue. It is a probabilistic- based approach to overcoming localization's difficulties. A Bayesian model of the problem is used to describe the robot'sbelief as a collection of weighted samples that are changed inaccordance with motion and sensor inputs. It is resistant to mapping errors, which invariably lower the results' accuracy.Using this function, the map is dynamically updated duringthe localization process without significantly lengthening theoperating time.

Vehicles based on point clouds recorded by LIDARsensors. This work resolves the issue when only a portion ofthe scanning data is available. For partial point clouds, the obstacle classification issue is addressed in this work using anon-shape modelling approach. The technique is examined against a well-known database and in actual environments. For AGVs, the real-time operation is provided by onboard computers of moderate complexity.

An encoded motor enables us to obtain the motor's RPM, which when processed further enables us to determinethe current location of the robot. The benefits of encoded motors for in-vehicle localization are discussed in this research. The current work describes how to perform an encoder-based real-time velocity analysis of a DC motor.

For the AIV's path planning, a powerful map generating algorithm is essential. The investigation in this work looks at autonomous motion planning for unmanned ground vehicles based on potential direction angles and localelevation mapping. Based on the configuration of a sensor system installed on an autonomous ground vehicle, an algorithm for building a local map using multiple lasers is developed, and the influence of dynamic obstacles is removed by combining it with an algorithm for tracking dynamic obstacles.

A top view outline map of the designated area of operation was created using a LIDAR sensor. The ROSoperating system, microcontrollers, rotary encoders, and a LIDAR device are all used by the wheelchair to travel. Important environmental measurements are made by the LIDAR unit, which is then used to map the area. ROS uses the rotary encoders and the LIDAR sensor to find a viable route to a user-defined destination. ROS instructs the wheelchair's microcontroller to move in the direction of the selected path. The wheelchair can be rerouted if new impediments crop up thanks to the navigation system's abilityto adjust to changes in the surroundings.

HARDWARE COMPONENTS

1. Raspberry Pi 3b+

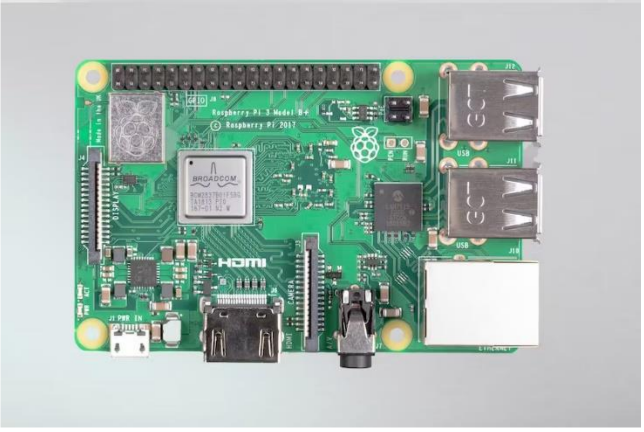


Fig-1: Raspberry Pi 3b+

Raspberry Pi 3B+ is a powerful microprocessor that runs a Raspbian operating system on it. The Raspbian operating system is a lite version of Linux. Raspberry Pi is the onboard computer. It acquires data from various sensors present on the AIV, does all the processing that is needed, and controls the motion of AIV by controlling both motors.

Proximity Sensor HC-SR04

Fig-2: Proximity sensor HC-SR04

HC-SR04 is an ultrasonic proximity sensor. This sensor is used to acquire the distance of an obstacle present in front of this sensor.

SPG30E-60K DC Geared Motor

Fig-3: SPG30E-60K DC motor

SPG30E-60K 12-volt DC geared motor equips quadrature hall effect encoder at the backside of the motor. The sensor produces 810 counts per main shaft revolution, 3 pulses per rear shaft revolution, single-channel output. This sensor is used for measuring the rpm of the motor.

RP-Lidar A2M4

Fig-4: RPLIDAR A2M4

RPLIDAR A2M4 is a 360-degree type of lidar sensor. It has 6m scanning radius and its sampling rate can go as high as6000 samples per second. This Lidar sensor's core rotates clockwise to perform 360-degree omnidirectional laser rangescanning for its surroundings, then generates 2D point clouddata of a cross-section of the environment.

2 Lipo Battery

Fig-5: Lipo Battery

An 11.1 volt, 2200mah Lipo battery is used to power both themotors of the AIV.

METHODOLOGY OF MBGF

The AIV must be familiar with the actual workspace it will use, including its perimeter measurements. We are presenting a novel method that may be applied to build the workspace's outline map and apply a grid to it. The grid is also used for AIV localization and path planning.

The four phases of this MBGF are described in further detailbelow:

1st Stage:

The user must supply the map; for this, we have provided a default polygon map as input to the raspberry pi and software; for dynamic map generation, we use the map provided by the Lidar and using that, we can give destination. For this project, we have manually provided the map as a polygon in the first stage; this polygon is known as a "map polygon," and it serves to define the workspace where AIV must operate. This first stage only needs to be completed once.

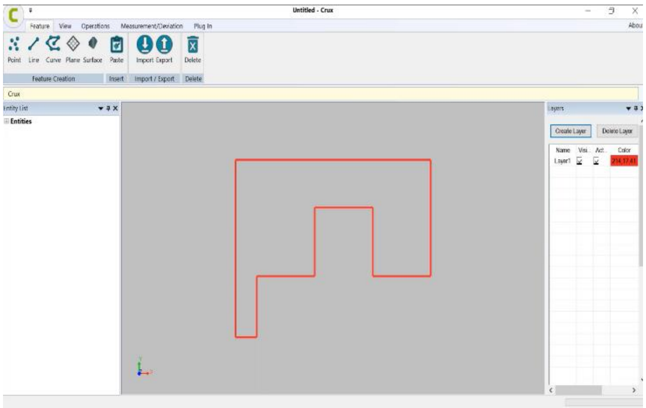


Fig-6: Map polygon

2nd Stage:

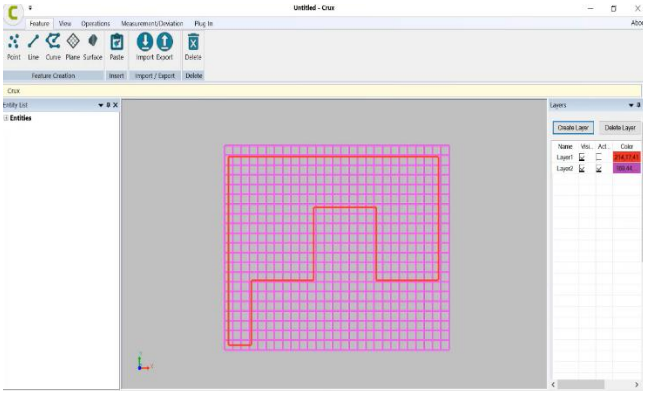


Fig-7: Full grid and map polygon

The second stage was applying the grid to the map polygon. The grid cells in this grid are all square and of the same size. When choosing the grid cell size, the AIV and map polygon sizes should be taken into account. The term "full grid" refers to this grid.

3rd Stage:

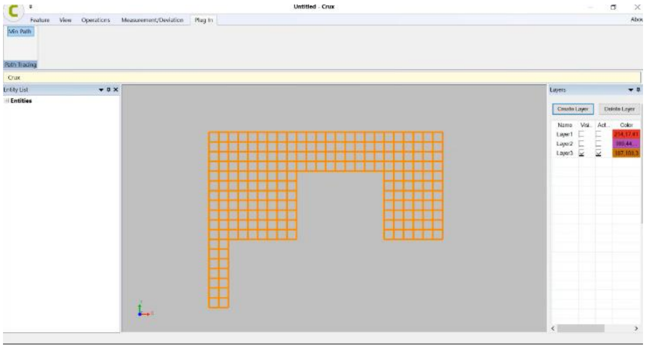
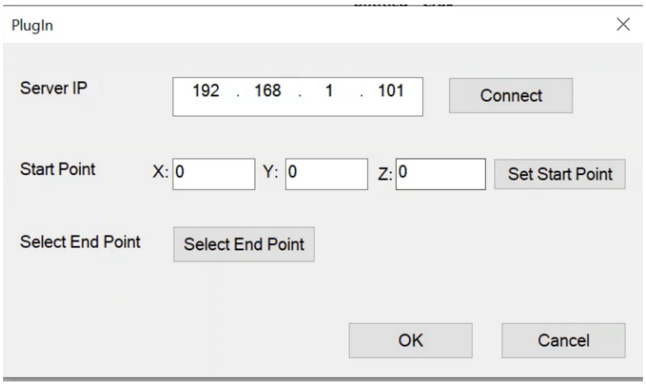


Fig-8: Valid grid, full grid, and map polygon

In the subsequent step, we only take into account the grid cells that completely encircle the polygon. The people outside the polygon are eliminated. The grid cells whose edges cross the map's polygonal boundary are also eliminated for safety reasons. As can be seen, the only legitimate cells in this example are those that are purple. A "valid grid" is what this grid is referred to as.

4th Stage:


PlugIn

Server IP: 192 . 168 . 1 . 101

Start Point X: 0 Y: 0 Z: 0

Fig-9: Selecting start point and endpoint:

The beginning point coordinates and endpoint must be entered into the software as seen in the window above. Give the IP address of the utilized Raspberry Pi first in the Server IP field.

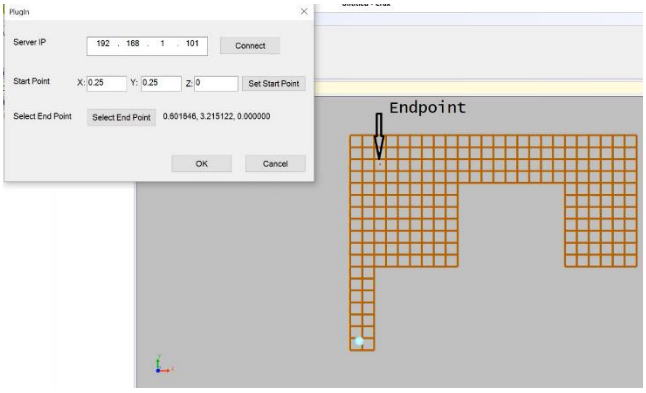
Next, include the beginning and ending points in the appropriate grid location.

PATH DESIGNING

The exact location of the AIV is always known to it. These coordinates allow us to pinpoint the grid cell containing the AIV. The "present grid cell" is the grid cell in question. When the user gives the destination location to AIV, AIV determines which grid cell the destination point is located in. The "target grid cell" is this grid cell.

Grid-based searching was chosen because it is a perfect match for the A-star algorithm. The A-star algorithm takes far less processing than other algorithms since it creates paths as needed, unlike the Dijkstra algorithm, which considers all potential paths from the beginning itself. Less calculation is better for microprocessors and microcontrollers to handle. We apply the A- star search method to discover the route from the current grid cell to the target grid cell. We can build a polyline that connects the AIV's current location to the target destination by joining the grid cell centers along the path's intended trajectory.

This polyline is used by the AIV to move from its current location to the destination. Fig. 11 displays the grid cell centers that the proposed line goes through.



Server IP 192 . 168 . 1 . 101

Start Point X: 0.25 Y: 0.25 Z: 0

Select End Point 0.601646, 3.215122, 0.000000

Fig-10: Planned path, valid grid, and map polygo

LOCALIZATION USING LIDAR

Robotics, autonomous vehicles, and augmented reality all frequently use Lidar (Light Detection and Ranging) sensors for localization. Lidar sensors produce laser pulses and time how long it takes for the light to bounce back, producing precise and high-resolution 3D point cloud data. The following methods for localization can use this data: Mapping and Simultaneous Localization and Mapping (SLAM): Environmentally complete maps can be created using lidar sensors. SLAM algorithms can map the environment while simultaneously estimating the sensor's position and orientation (localization). They do this by continuously scanning the area and comparing the readings to data from previously mapped areas. By evaluating point cloud data, algorithms may quickly locate and identify impediments. For collision avoidance systems, which enable vehicles or robots to safely navigate their environment, this information is essential.

Location in situations when GPS is denied: In some situations, such as urban canyons or enclosed places, GPS signals may be weak or unavailable. Even in areas where GPS is not available, lidar sensors can provide precise localization when combined with other sensors like cameras, wheel encoders, or inertial measurement units (IMUs). Without relying on external signals, the sensor's perception of the surrounding geometry allows for accurate pose assessment.

Lidar sensors can be used in augmented reality applications to assess a device's pose (position and orientation), such as a head-mounted display or a smartphone.

DYNAMIC MAP BUILDING ALGORITHM

The map has a grid, and the grid contains a lot of grid cells. The MBGF approach, however, does not take into account the obstacles that exist inside the workspace. To handle the barriers on the map, we developed this algorithm. This program uses proximity ultrasonic sensors to detect obstructions.

Each grid cell has a status assigned to it. Statuses can be either active or inactive. Obstacle-free grid cells are those that are active. Grid cells are rendered inactive by obstacles that are present in them. Each grid cell is awarded one of three penalty chances (penalty 0, penalty 1, penalty 2, and penalty 3). In the legal grid, every grid cell is by default active and carries a penalty of 0.

Following are the flow charts of dynamic map building andmap updating algorithm:

Fig-11: Algorithm for updating obstacles present in the area.

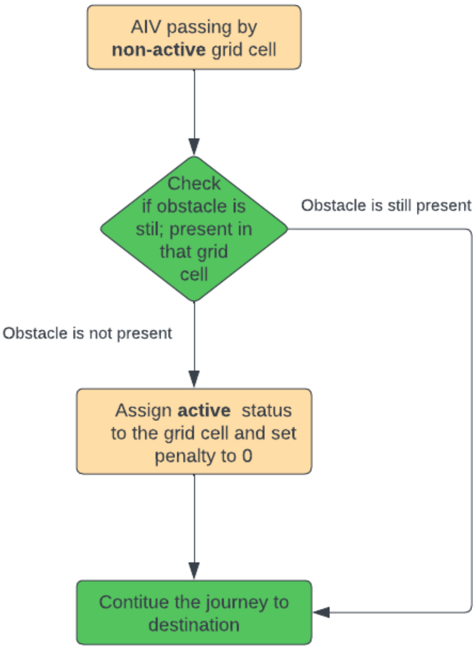


Fig-12: Algorithm for updating the obstacles that areremoved from the area.

With the help of this technique, the map will always be up to date. As the AIV moves farther, the map becomes more up to date. Any barriers erected in the region will be reflected on the map that AIV has produced. The map will show any obstacles that are eliminated. Because this method uses only proximity sensors and requires very little computing, it is perfect for use on microprocessors and microcontrollers. This algorithm is quite easy to put into practice. This method works well for continuously updating the map of a changing environment, according to extensive testing we've done.

RESULTS

On the floor:
When AIV bot is tested on plain surface like the floor, we got following results as shown.
We have given starting point as (x, y) co-ordinates as input in crux software. The endpoints are given on grid. In some cases, the bot is not able to avoid the obstacle Due to lack of response time.

Table 1: Results when bot travelled on Floor.

Sr. No.	Start Point	End Point	Time (In min)	Obstacle avoidance
1.	0.25,0.25	2,6	1	Done
2.	0.25,0.25	3,8	1.60	Not Done
3.	0.25,0.25	4,2	1.50	Done
4.	0.25,0.25	3,4	1.30	Done
5.	0.25,0.25	5,2	1.20	Done
6.	0.25,0.25	5,5	2.05	Not Done
7.	0.25,0.25	2,2	0.40	Done
8.	0.25,0.25	3,6	1.50	Not Done
9.	0.25,0.25	4,1	1	Done
10.	0.25,0.25	1,8	1.05	Done

On Rough surface:

When AIV bot is tested on little bit rough surface, we got following results as shown.
We have given starting point (x, y) co-ordinates as input in crux software. The endpoints are given on grid. In some cases, the bot is not able to avoid the obstacle Due to lack ofresponse time.

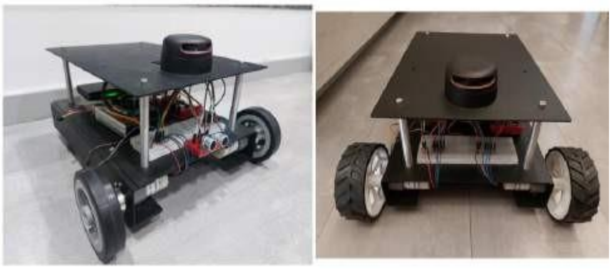


Fig-15. AIV Bot

Table 2: Results on Rough Surface

Sr. No.	StartPoint	Endpoint	Time Taken (In min)	Obstacle Avoidance
1.	0.25,0.25	3,6	3	Done
2	0.25,0.25	2,5	2.40	Done
3	0.25,0.25	1,8	1.20	Done
4	0.25,0.25	4,3	3.50	Not Done
5	0.25,0.25	3,8	3.40	Done
6	0.25,0.25	8,2	6	Not Done
7	0.25,0.25	5,3	4.50	Not Done
8	0.25,0.25	2,6	2.58	Done
9	0.25,0.25	3,7	3.30	Done
10	0.25,0.25	5,1	4.10	Not Done
11	0.25,0.25	3,9	3.55	Done
12	0.25,0.25	6,4	5.20	Not Done
13	0.25,0.25	4,9	5	Done

CONCLUSION

In this paper, we give a comprehensive explanation of an AIV system that we developed and tested. By integrating the Map Building and Grid Fitting algorithm, RPT algorithm with lidar, A-star algorithm, and Dynamic Map Building and Map Updating algorithm, an automobile can become fully autonomous and intelligent. Regardless of its intended use, every AIV must go through the mapping and tracking process. The method we developed can be modified and used with AIV of any size and shape.

REFERENCES

1. Towards a Viable Autonomous Driving Research Platform https://www.ri.cmu.edu/pub_files/2013/6/2013_IV_SRX.pd f
2. Research on Comparison of LiDAR and Camera in Autonomous Driving https://iopscience.iop.org/article/10.1088/17426596/2093/1/012032/pdf
3. Through the eyes of Autonomous Vehicles – using laser scanning technology https://www.sciencedirect.com/science/article/pii/S2352146521009650
4. Surrounding Objects Detection and Tracking for Autonomous Driving Using LiDAR and Radar Fusion https://cjme.springeropen.com/articles/10.1186/s1003-021- 00630-y
5. Lidar based self-Driving Car https://www.ijraset.com/research-paper/lidar-based-self- driving-car

6. A Survey on 3D LiDAR Localization for Autonomous Vehicles https://ieeexplore.ieee.org/document/9304812 https://www.hindawi.com/journals/js/2022/2690164/
7. Autonomous Vehicles and Intelligent Automation: Applications, Challenges and Opportunities. Volume 2022 | Article ID 7632892 https://www.hindawi.com/journals/misy/2022/7632892/
8. De Silva, V.; Roche, J.; Kondoz, A. Robust Fusion of LiDAR and Wide-Angle Camera Data for Autonomous Mobile Robots. Sensors 2018, 18(8), 2730; https://doi.org/10.3390/s18082730.
9. THE SELF-DRIVING CAR USING LIDAR https://ijcrt.org/papers/IJCRT2106279.pd
10. Autonomous Intelligent Vehicles (AIV): Research statements, open issues, challenges and road for future. January 2021, License: CC BY-NC-ND 4.0 https://www.researchgate.net/publication/354049190_Auton omous_Intelligent_Vehicles_AIV_Research_statements_open_issues_challenges_and_road_for_future
11. Lidar Sensor in Autonomous Vehicles. March 2022 https://www.researchgate.net/publication/359263639_Lidar_Sensor_in_Autonomous_Vehicles
12. Vision-Based Autonomous Vehicle Systems Based on Deep Learning: A Systematic Literature Review https://www.mdpi.com/2076-3417/12/14/6831/pdf
13. Intelligence of Autonomous Vehicles: A Concise Revisit. Volume 2022 | Article ID 2690164