iJRASET
International Journal For Research in Applied Science and Engineering Technology

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Volume: 9 **Issue:** XII **Month of publication:** December 2021

DOI: https://doi.org/10.22214/ijraset.2021.39626

www.ijraset.com

Call: 08813907089 | E-mail ID: ijraset@gmail.com

International Journal for Research in Applied Science & Engineering Technology (IJRASET)
ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538
Volume 9 Issue XII Dec 2021- Available at www.ijraset.com

Modelling Building Shadows on Rooftop Generating 3D City Model

Smit Shah[1], Francy Ruparel[2]
[1, 2]*Department of Information Technology, Faculty of Technology, Dharamsinh Desai University (DDU), College Road, Nadiad, Gujarat, India – 387001*

***Abstract:** 3D city models enable us to gain a better grasp of how various city components interact with one another. Advances in geosciences now allow for the automatic creation of high-quality, realistic 3D city models. It is not limited to visualization and navigation, however, also for shadow and solar potential analysis. Solar radiation is an example of a 3D GIS tool that is in high demand. The calculation of solar radiation that reaches 3D objects can be simple, but the shadow effect of nearby buildings is a considerably more challenging issue because some facades or roofs are only partially shadowed. The present study is analyzed into two approaches. The first approach is considered as Visualization (client-side) approach to visualize the 3D city models on the website using NodeJS and CesiumJS. The second approach is considered as Analyzation (Server-side) approach to analyze the solar potential using python for faster processing and deeming the future development aspects.*

I. INTRODUCTION

Renewable energy, such as photovoltaic systems, is becoming popular for energy production due to global warming, the greenhouse effect, and a variety of other drawbacks of existing energy sources. The outcome of a photovoltaic potentiality analysis is determined by the quality of the data and the parameters used. Shadow rapidly reduces photovoltaic system performance and is always changing due to the movement of the sun. The solar radiation incident on the earth's atmosphere is relatively constant, but the radiation incident on the earth's surface varies due to absorption, scattering, reflection, changes in spectral content, diffuse component, water vapor, clouds, and pollution, among other factors. In this project, it is being investigated how efficiently real-time shadow can be detected for both direct and diffuse radiation, taking reflection and other factors into account, in comparison to existing shadow detection methods using cutting-edge technologies, and what is the minimum data quality required for this purpose.
Numerous GIS-based approaches for analyzing solar potential have been developed. They are very effective for 2.5D surfaces such as terrain and rooftops, but if the user wants to see how sun shadows are mapped on the rooftop in the context of surrounding buildings, they must use a 3D City Model. Placing panels on building facades is more important in cities with high latitudes because the sun altitude is very low and sun rays hit the side structure of other buildings, casting shadows on lower height buildings that are placed around tall buildings. Unfortunately, even when using very high resolution DEMs, 2.5D models are insufficient to represent the shadow effect of neighboring buildings as well as other solar radiation parameters on building facades and other vertical surfaces. Three-dimensional models assist us in better understanding how various city components interact. Geoscience developments have made it possible to create high-quality, realistic-looking city models automatically. Not just for visualization and navigation, but also for more complex challenges like shadow analysis and solar potential analysis, 3D city models are necessary. The ability to create 3D city models has raised a demand for 3D GIS tools like solar radiation. The calculation of solar radiation that reaches 3D objects is rather simple, but the shadow effect of nearby buildings is a considerably more challenging issue, as certain facades or roofs can only be partially shadowed. This model can assist professionals from a variety of professions in assessing the solar energy potential of their studies, even if they lack a thorough understanding of physics or extensive programming skills. Assessing the revenues from a building installed solar system makes it accessible to all citizens, decision-makers, and local authorities as solar potential models grow into online maps, regardless of if they are unfamiliar with radiation from the sun formulations or have no background in solar energy systems.

II. LITERATURE REVIEW

As developing technologies such as GeoWeb, LBS (Location Based System), and MAR (Mobile Augmented Reality) have showed potential applications, there has been an increase in demand for 3D rooftop models in recent years. Urban planning, management, and development, environmental management, tourism, telecommunications, transportation and navigation, and public safety are just a few examples. The presence of "missing data" should be addressed while reconstructing 3D rooftop models using remotely sensed data. Consider that a rooftop is made up of a series of polygons (planar features) connected by lines (linear features). The shape of the building of interest (shape prior); how many features (polygons and lines) are necessary to describe it; and what topological rules (relations) are linked with the model are all unknown as a priori knowledge. Furthermore, the signal-to-noise ratio is always unpredictable, making fragmentary level modelling cues (features) extraction impossible to forecast.

Knowing the shape of a structure beforehand would alleviate all issues in rooftop modelling by compensating for knowledge of "missing features" and "missing relations." In fact, however, acquiring such rich priors is an uncommon occurrence: even generalizing the form prior to a semantic level (how to characterize a shape) would be a difficult task [1-4].

The parametric modelling technique poses this restriction as a model selection challenge to solve it. That is, the technique implies the presence of several shape priors (flat, gable, hip, and so on). As a result, establishing which prior (model template) would be the best fit to given observations (surface model and characteristics) and estimating associated model parameters has become the rooftop challenge. If appropriate modelling templates and sufficient observations are provided for recreating rooftop models, the procedure could be promising. However, when a building of interest is made up of N numbers of sub models, as is the case in a complex urban setting, this approach may confront the greatest difficulty [5-7].

A generic modelling technique is based on a bottom-up vision process, which restores missing data primarily using extracted characteristics from given data, as opposed to the parametric modelling approach. This could limit access to building form priors, making recovering lost features and their topological relationships more challenging. However, avoiding the usage of shape priors would be more adaptive when delineating more complicated shapes. The lack of features, on the other hand, is unavoidable. The generic modelling method should take care of recovering features that were lost during the reconstruction phase [8-11].

In a generic modelling approach, Sohn et al. (2008) presented a BSP (Binary Space Partitioning) tree for handling the missing data problem. It proved its ability to create building models while also discussing its drawbacks, such as erroneously establishing topological relationships between modelling features. This study presents a new method for correcting the topological flaws caused by the BSP algorithm of Sohn et al. (2008). The suggested method is based on a subset of cartographic regularisation algorithms, which removes erroneous vertices while keeping important building shape information. The work of Weidner and Forstner (1995) inspired our method. The generated vectors from BSP are referred as nosy model boundaries. Based on the Minimum Description Length also known as MDL, the proposed approach gradually corrects them [8-12].

III. METHODOLOGY

As discussed earlier, the study has been done into two approaches i.e., Visualization and Analysis. The methodology of both approaches is explained in this section.

A. Visualization

3D city devices are widely used as a technique for a variety of urban planning and management applications. The display of these models via desktop applications is now being developed. Due to the obvious scale of such models and the restricted mobile application devices, visualizing 3D city models in a web application is a significant technological problem. The current study presents a novel method for visualizing 3D city models in a web browser without the need to download any additional software or plugins. To perform this, the use of GeoJSON as a data model and use a web browser to see GeoJSON objects in a web page is made. CesiumJS technology is used for this, which is extremely light and renders quicker than any other contemporary GIS technology.

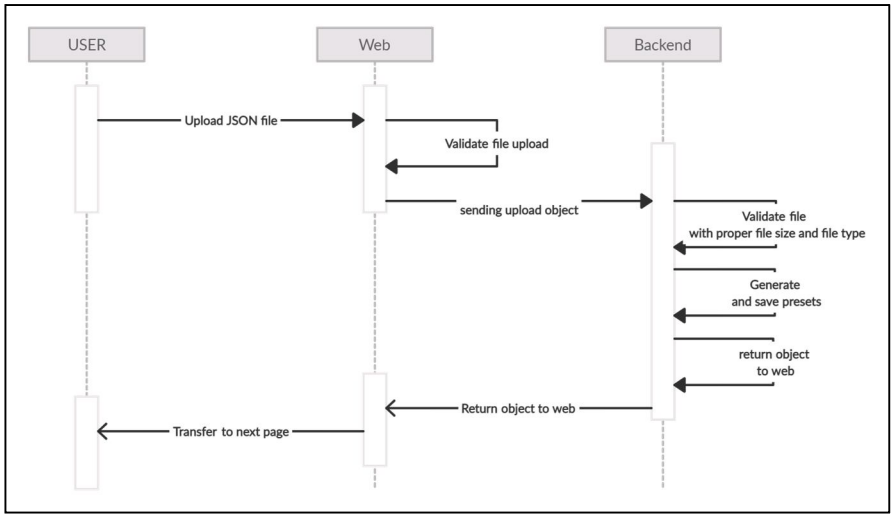


Figure 1: Process of Upload and Validation

Figure 1 represents the process of upload and validation. The user visits the uploading web page, where he or she must upload his or her JSON/GeoJSON file for visualization. The server verifies all of the marks before saving the file. The server then redirects you to the file-detail page.

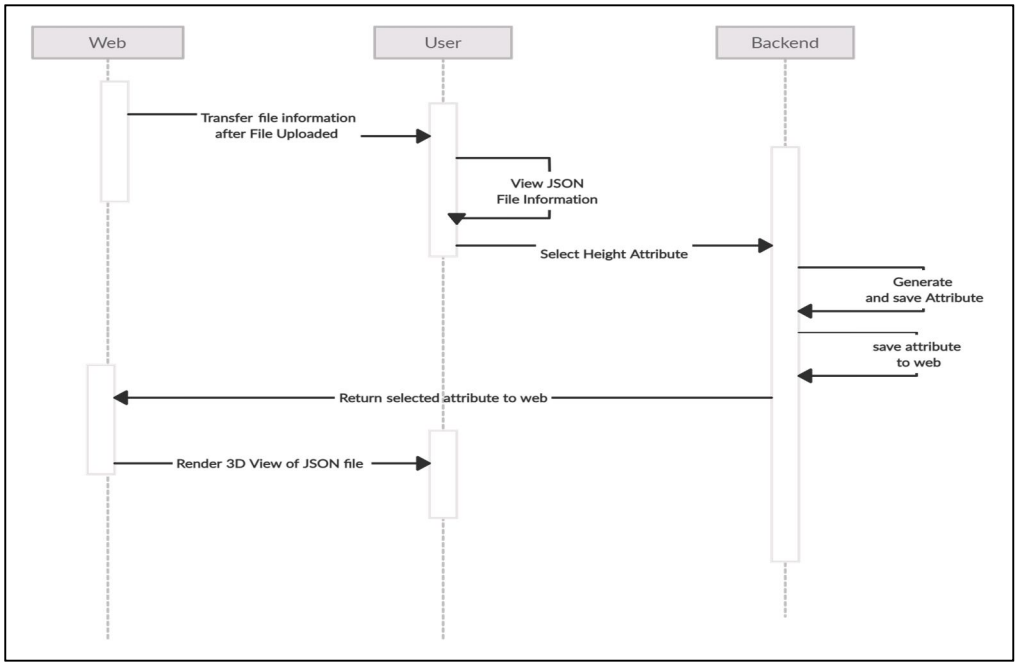


Figure 2: Process of Details Display and Selection of Height Attribute

Figure 2 depicts the process of details display and selection of height attributes. The server provides file details such as file size, file type, what sort of object it contains, CRS information if it is present in the file, and the total number of objects it contains. Aside from these details, it asks the user to select the file height attribute, which will construct a 3D object from a 2D shape and store it as a property on the file. The server will redirect to the visualization page after selecting a height property, where it will be sent as an input. In 3D mode, the user can see a city depiction. The detailed methodology is explained in the next paragraph.

The commencing was done by configuring the NodeJS environment. The express module aids in the creation of a server in a more assured manner; for example, with the express module, one can design a desirable server capable of processing thousands of HTTP requests. On the client/user side, Express allows developers to construct a page management engine. The uploading page will be the first thing that will appear when the user will upload his or her JSON/GeoJSON file. When visitors upload a file, it will be validated for the file type. It will show the user the file name and size to ensure that they are uploading the correct file.

After posting/submitting the file, the server will double-check all parameters, such as the file mime type, which must be JSON, and the file size, which must be under 5MB (for a small amount of area). If all these conditions are met, the server will send the client the file details page, which includes the feature type, the type of object it contains, the name of the object (if it is a building, show it as a building, and if it is a polygon, show it as a polygon), and the total number of objects in the file. Including this information, the screen also prompts the user to select the file's height attribute from the object properties within the file. Correct attribute selection will aid in the accurate visualization of a 3D city. Geospatial visualization and cutting-edge technology are demonstrated: CesiumJS works on the client side since it requires less time for processing and GPU calculations because it uses the Cesium precompiled JavaScript library, which aids in the creation of 3D models/objects from 2D datasets. Cesium will use an uploaded JSON data file as the main data source for this. On the 3D globe, users can project data or file polygons, which will appear on the projected region as a rectangle or polygon, flat. The user-selected height attribute for the 3D elevation of such objects/buildings can be made. Giving each object's attributes height to the extruded height of the cesium object/entity will give 2D shapes alive and generate level-1 detailed 3D models. To make the scene more realistic, the coloring of the buildings according to their height is done, which will lead to a new set of visual effects. To demonstrate how a shadow is projected on a building in real life, we will introduce a shadow tool that replicates shadow on these buildings and allows users to see how their shadow will appear or be projected on the selected buildings over time.

B. Analysis of Photovoltaic Potential on Building Rooftop

India is a nation that is a densely populated country, with a population of over 1.3 people as of 2021. Solar energy appears to be the most promising form of renewable energy in the country, despite the few renewable options available. India has a better potential for solar PV deployment in residential neighborhoods because the residential built stock accounts for a considerable portion of the overall building footprint in urban locations. As a result, government organizations are attempting to speed up the implementation of solar PV systems on public residential blocks around the country in order to generate green electricity on a yearly basis. Because efficient solar PV installation is largely determined by the broader urban form, the very first question addressed in this study is whether there are ways that can provide urban design guidelines for improving PV deployment and hence renewable energy production potential. Despite the fact that India receives favorable solar irradiation throughout the year, not all building surfaces are ideal for PV installation due to the large concentrations of residential neighborhoods and the range of building shapes and orientations. The goal of this research is to determine how urban morphology characteristics at the neighborhood scale ultimately affect PV potential as well as to provide urban citizens with solar panel installation guidelines that maximize PV generation potential while meeting the standards of high population residences. The general overall process of the Analysis is represented in figure 3. Using a JSON file (vector file) uploaded by the user, a raster file in the GeoTIFF format will be created. Provide the user with a raster file. The user will enter a date and time, and the sun's azimuth and height will be determined every hour. Provide the user with this output. Using the above outputs, create a shadow file for the building. Process all of the files and construct energy, shadow, and sun-light analysis. The detailed methodology is explained in the same section.

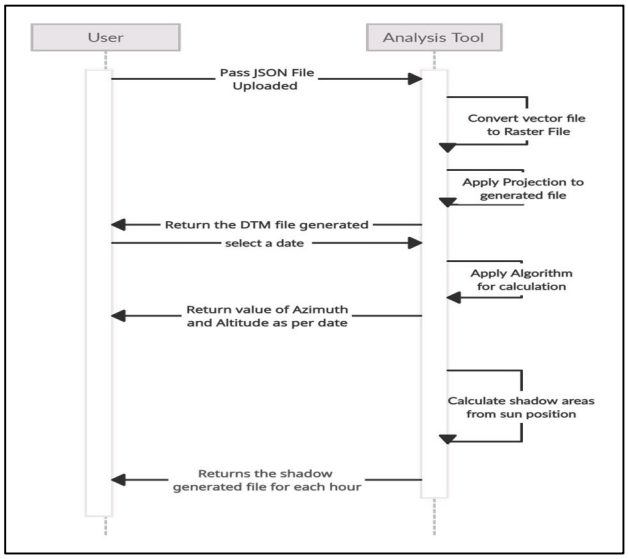


Figure 3: General Analysis Process

This approach begins by converting the user-uploaded GeoJSON/JSON (Vector) file to a Digital Terrain Model (Raster) file, with burn value as a user-selected height value/attribute. The projection must be set appropriately during this process.

C. Rasterization

The GDAL rasterize function is a simple and quick technique to automate polygon conversion in Python. GDAL is written in C++ and has Python bindings, making it computationally efficient and scalable to supercomputing tasks. This study covered both typical and basic gdal implementations. user may want to save a value from the vector attribute, such as a MEAN value or a user-supplied height attribute, on occasion. With gdal, you can simply accomplish this. The options or attribute argument is passed to RaterizeLayer(). As concentration is put on unique development demands, there's more possibility for customization. A Digital Terrain Model (DTM) is a file that provides additional data defining the terrain in locations where Lidar data itself is unable to complete the job successfully. The passing of the GeoJSON as vector files is done during this operation. By using geodata in the files, the building of a blank raster file with row and column numbers, as well as projection is done. Afterward, using gdal's RasterizeLayer() function with settings such attributes as MEAN and Band = 1, the DTM file is created in the same way as a TIFF file. Figure 4 depicts the vector file (left) and raster file (right) comparison.

ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538
Volume 9 Issue XII Dec 2021- Available at www.ijraset.com

Figure 4: Vector file vs Raster file

Following rasterizing the image, the Sun's Azimuth and Altitude must be determined for the file's latitude-longitude position using the user-supplied date and time. After obtaining these two outputs, the solar shadow on these structures must be estimated for each hour between dawn and sunset using azimuth and altitude. Zonal Statistics will be applied to all the sun shadowed files once they have been obtained. This leads to an examination of the shadow details on each building. Details such as the shadow region in a certain file and the sun-light area are displayed. Then, using mathematical calculations, the energy and the percentage of shadow coverage are determined.

IV. RESULTS

Following the selection of a specific building and completion of all visualization and analysis methods, figure 5 exhibits the view and displays the attribute values of building ID 2946, with MEAN as the eighth attribute.

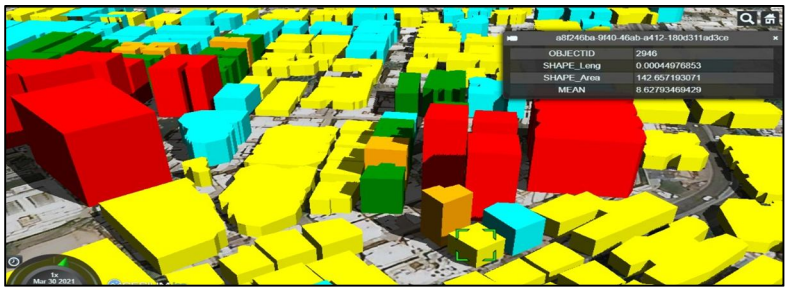


Figure 5: Visualization of 3D city

Thereafter, by clicking on any of the building structures, the user can see the values of all that building's properties. Buildings are differentiated based on their height attributes, with lower heightened buildings being yellow, average heightened buildings being green, and tallest buildings being red, as shown in figure 5. In addition, the user may view the actual shadow over time by using the time bar on the bottom part. The conversion of vector file to rater file is depicted in figure 4. Figure 6 shows the output for the shadow files generated for March 21, June 21, September 21, and December 21.

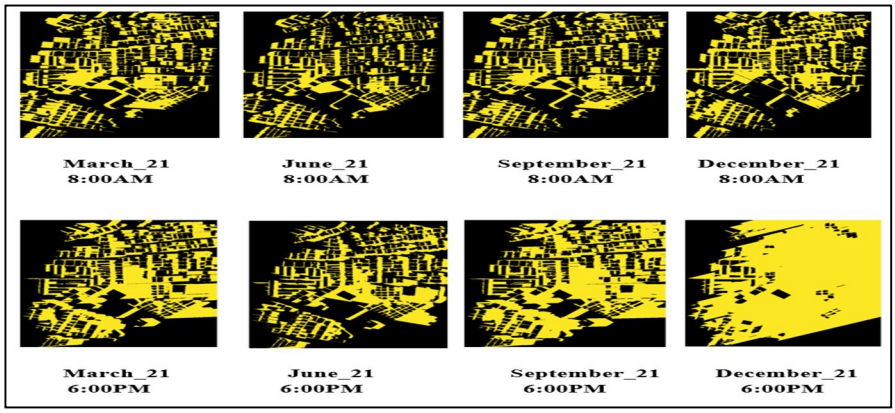


Figure 6: Shadow file

Table 1, Table 2, and Table 3 provide the findings for zonal statistics and calculations for buildings visualized with building ID 2946 for all four dates in 2020: March 21, June 21, September 21, and December 21. The calculations are carried out using zonal statistics on the created shadow file.

Table 1: March-Zonal Statistics

Time	Object ID	Shape Length	Shape Area	Mean	Count	Sum	Shadow Percent	Sunlight Percent	Shadow Free Area	Energy
Sunrise	2946	0.00045	142.66	8.63	69	69	48	52	74.18	0.51
6:00	2946	0.00045	142.66	8.63	84	84	59	41	58.49	0.22
7:00	2946	0.00045	142.66	8.63	69	69	48	52	74.18	5.50
8:00	2946	0.00045	142.66	8.63	42	42	29	71	101.29	31.11
9:00	2946	0.00045	142.66	8.63	11	11	8	92	131.24	63.58
10:00	2946	0.00045	142.66	8.63	4.56	4.56	3	97	138.38	96.42
11:00	2946	0.00045	142.66	8.63	2.82	2.82	2	98	139.80	115.93
12:00	2946	0.00045	142.66	8.63	3.22	3.22	2	98	139.80	122.52
13:00	2946	0.00045	142.66	8.63	3.22	3.22	2	98	139.80	125.47
14:00	2946	0.00045	142.66	8.63	3.13	3.13	2	98	139.80	119.98
15:00	2946	0.00045	142.66	8.63	3.19	3.19	2	98	139.80	100.22
16:00	2946	0.00045	142.66	8.63	3	3	2	98	139.80	71.61
17:00	2946	0.00045	142.66	8.63	12	12	8	92	131.24	36.17
18:00	2946	0.00045	142.66	8.63	19	19	13	87	124.11	10.10
Sunset	2946	0.00045	142.66	8.63	141	141	99	1	1.43	0.002

Table 2: June-Zonal Statistics

Time	Object ID	Shape Length	Shape Area	Mean	Count	Sum	Shadow Percent	Sunlight Percent	Shadow Free Area	Energy
Sunrise	2946	0.00045	142.66	8.63	131	131	92	8	11.41	0.251
6:00	2946	0.00045	142.66	8.63	4.72	4.72	3	97	138.38	5.459
7:00	2946	0.00045	142.66	8.63	119	119	83	17	24.25	3.822
8:00	2946	0.00045	142.66	8.63	52	52	36	64	91.30	30.480
9:00	2946	0.00045	142.66	8.63	11	11	8	92	131.24	70.080
10:00	2946	0.00045	142.66	8.63	4.73	4.73	3	97	138.38	86.230
11:00	2946	0.00045	142.66	8.63	4.73	4.73	3	97	107.88	70.120
12:00	2946	0.00045	142.66	8.63	2.92	2.92	2	98	139.80	101.50
13:00	2946	0.00045	142.66	8.63	0	0	0	100	142.66	111.10
14:00	2946	0.00045	142.66	8.63	4.12	4.12	3	97	138.38	102.70
15:00	2946	0.00045	142.66	8.63	19	19	13	87	124.12	82.920
16:00	2946	0.00045	142.66	8.63	42	42	29	71	101.29	52.440
17:00	2946	0.00045	142.66	8.63	64	64	45	55	78.46	25.860
18:00	2946	0.00045	142.66	8.63	78	78	55	45	64.20	9.393
Sunset	2946	0.00045	142.66	8.63	141	141	99	1	1.43	0.002

ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538
Volume 9 Issue XII Dec 2021- Available at www.ijraset.com

Table 3: September-Zonal Statistics

Time	Object ID	Shape Length	Shape Area	Mean	Count	Sum	Shadow Percent	Sunlight Percent	Shadow Free Area	Energy
Sunrise	2946	0.00045	142.66	8.63	69	69	48	52	74.18	1.509
6:00	2946	0.00045	142.66	8.63	81	81	57	43	61.34	0.635
7:00	2946	0.00045	142.66	8.63	64	64	45	55	78.46	5.976
8:00	2946	0.00045	142.66	8.63	33	33	23	77	109.85	26.342
9:00	2946	0.00045	142.66	8.63	4	4	3	97	138.38	61.301
10:00	2946	0.00045	142.66	8.63	4.58	4.58	3	97	138.38	79.582
11:00	2946	0.00045	142.66	8.63	2.40	2.40	2	98	139.80	91.901
12:00	2946	0.00045	142.66	8.63	3.22	3.22	2	98	139.80	97.284
13:00	2946	0.00045	142.66	8.63	3.22	3.22	2	98	139.80	95.991
14:00	2946	0.00045	142.66	8.63	3.67	3.67	3	97	138.38	86.928
15:00	2946	0.00045	142.66	8.63	2.83	2.83	2	98	139.80	71.333
16:00	2946	0.00045	142.66	8.63	6	6	4	96	136.95	50.464
17:00	2946	0.00045	142.66	8.63	13	13	9	91	129.82	24.199
18:00	2946	0.00045	142.66	8.63	23	23	16	84	119.83	5.289
Sunset	2946	0.00045	142.66	8.63	144	144	99	1	1.43	0.004

The overall energy on the building is computed from the energy that is obtained for the entire day from sunrise to sunset after computing shadow-free area and energy at specific times on specific days. The merged_file_18 CSV file contains the energy values for all times of a specific day. The figures in Table 4 are the results for total energy and the shadow for the month of March-21, June-21, September-21, and December-21.

Table 4: Results for Energy and Shadow

Date	March-21	June-21	September-21	December-21
Total Energy (KW)	959.319	772.3018	692.7451	592.7319
Shadow (m^2)	111.5579	94.66324	112.4139	108.039

V. CONCLUSION

The author concludes the following points from the present study:

A. As solar potential models evolve into online maps, the current study aids in analyzing the profits from a building installed solar system, making it available to all citizens, decision-makers, and local governments.

B. The diversity created by combining a 3D city visualization with a PV potential analysis using CesiumJS and Python is extremely useful in the academic sphere. The current study has the ability to liberate a student's potential that has been stifled by the high cost of creativity.

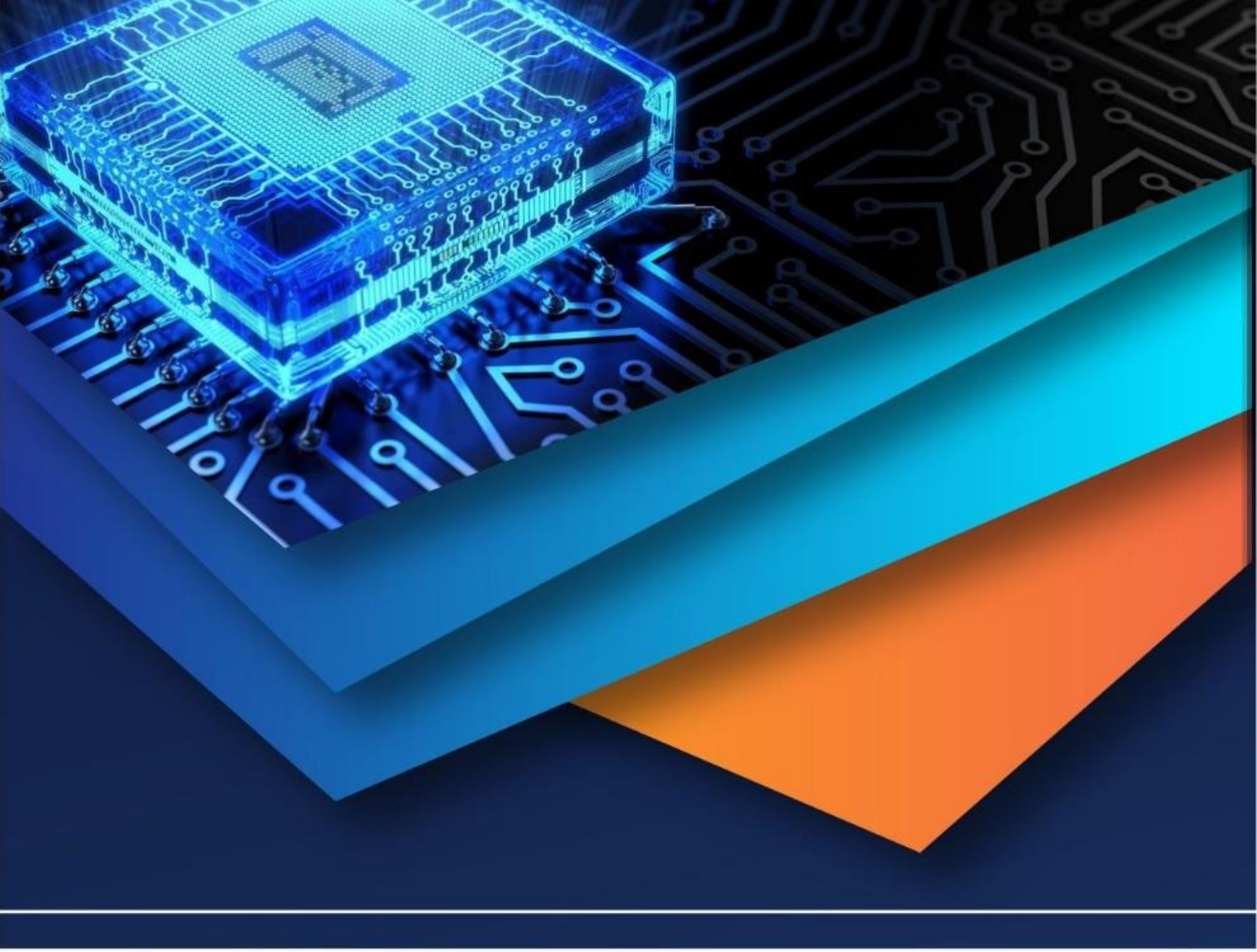
C. This will also give colleges a leg up on traditional study methods by allowing them to be more connected to technology. As new discoveries are uncovered, content updates can be delivered on a regular basis.

D. There is a lot of room for improvement, as there is a great ocean of knowledge waiting to be discovered, and improved software evolution can give end consumers a better experience.

E. Once the present study is commercialized, it will give an elevated platform for quick growth in the education and practical sectors.

REFERENCES

[1] Scherer, R.J., Schapke, S.E., 2011. A distributed multi-model- based management information system for simulation and decision-making on construction projects. Advanced Engineering Information, 25(4), pp. 582-599.
[2] Kurakula, V., 2007. A GIS-based approach for 3D noise modeling using 3D city models. Master Thesis, International Institute for Geo-Information Science and Earth Observation, Enschede, The Netherlands.
[3] Glander, T., Dollner, J., 2009. Abstract representation s for interactive visualization of virtual 3D city mode ls. Computers, Environment and Urban Systems, 33(5), pp. 375-387.
[4] Wagen, J., Rizk, K., 2003. Radiowave propagation, building database s, and GIS: anything in common? A radio engineer ' sviewpoint. Environment and Planning B: Planning and Design, 30(5), pp. 767-787.
[5] You, S., Hu, J., Neumann, U., Fox, P., 2003. Urban site modeling from Lidar. International Conference on Computational Science and Applications, Vol. 3, pp. 579-588.
[6] Verma, V., Kumar, R., 2006. 3D building detection and modeling from aerial lidar data. Computer Vision and Pattern Recognition, Vol. 2, pp.2213-2220
[7] Teo, T.A., 2008. Parametric reconstruction for complex building from LiDAR and vector maps using a divide-and- conquer strategy. International Archives of Photogrammetry and Remote Sensing, Vol. XXXVII, Part B3a, pp. 133-138.
[8] Rottensteiner, F., 2003. Automatic generation of high-quality building models from LiDAR data. IEEE Computer Graphics and Applications, 23(6), pp. 42-51.
[9] Alharty, A., Bethel, J. 2004. Detailed building reconstruction from airborne la ser data using a moving surface method. International Archives of Photogrammetry and Remote Sensing. Vol. XXXV, Part 83, pp. 213-218.
[10] Sampath, A., Shan, J., 2007. Building Boundary Tracing and Regularization from Airborne Lidar Point Clouds. Photogrammetric Engineering & Remote Sensing, 73(7), pp. 805-812.
[11] Sohn, G., Huang, X., Tao., V, 2008. Using binary space partitioning tree for reconstructing 3d building models from airborne lidar data. Photogrammetric Engineering & Remote Sensing, 74(11), pp. 1425-1440.
[12] Weidner, U., Forstner, W., (1995). Towards Automatic Building Extraction from High Resolution Digital Elevation Models, ISPRS Journal, 50(4), pp. 38-49.



10.22214/IJRASET

45.98

IMPACT FACTOR: 7.129

IMPACT FACTOR: 7.429

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY