International Journal of
INTELLIGENT SYSTEMS AND APPLICATIONS IN ENGINEERING

ISSN:2147-6799 www.ijisae.org Original Research Paper

A Strategy for Empowering Multi-Child Semantic Maps through Alpha C-Means Algorithm

Saritha Dasari[1], Dr. A. Rama Mohana Reddy[2], Dr. B. Eswara Reddy [3]

Submitted: 27/05/2023 **Accepted :** 16/07/2023 **Accepted:** 27/07/2023

Abstract*:* Image technology is a developing field that involves harnessing the photovoltaic impact of each image and transforming it into data for segregation. The utilization of this imaging system holds significant importance across diverse industries. Computer science research about image processing has experienced significant growth and advancement, exhibiting a dynamic and progressive nature. The advancement of data in image processing is intricately linked to the image itself. Image analysis pertains to extracting concealed information and interpreting photographs that may lack explicit depiction of the depicted scenario. The image integrates several elements from machine learning, data management, application autonomy, and image processing. Semantic maps serve as visual representations that encapsulate stored image data in extensive databases. A prior investigation was dedicated to the utilization of the K-C Means Clustering Algorithm, specifically recognized as the MCSMK-C Algorithm. This algorithm was applied to facilitate a two-path clustering approach on Multi-Child Semantic Maps. This algorithm facilitates the formation of image clusters and enables the system to analyze the final section of the image. This work presents the Alpha-C Means Algorithm, which focuses on enhancing accuracy. The heightened aversion towards the image prompts the establishment of optimal criteria for visual comparison and repulsion. The Multi Child Semantic Maps algorithm enhances the precision of generating image outputs. The experimental results demonstrate the statistical importance of the expected and actual output.

Keywords: *Clustering, Image, Semantic Maps, Spanning Trees*

1. Introduction

Image, also known as image data or image analysis, is extracting useful information and knowledge from large collections of digital images. It is a subfield of data and machine learning that uses computer algorithms to automatically analyze, classify, and understand the content of images. Image aims to identify patterns and relationships within image data that are not easily visible to the human eye. Image applications include object recognition, face detection, image segmentation, image retrieval, and image classification. Image is used in various fields, including healthcare, manufacturing, retail, and security. Image techniques include feature extraction, clustering, classification, and association rule. Feature extraction involves identifying important characteristics of an image, such as edges, shapes, and textures. Clustering involves grouping similar images, while classification involves labeling images based on their content. The association rule involves identifying patterns and relationships between elements within an image [1] [2]. Overall, image plays an important role in the analysis and interpretation of visual data and has numerous practical applications in various industries.

Image analysis is a crucial process of extracting valuable insights from extensive image datasets, employing various techniques encompassing image processing, machine learning, and computer vision. These techniques collectively enable the comprehension and interpretation of images, and within this realm, various algorithms play pivotal roles [3].

Feature extraction techniques are the cornerstone for revealing significant visual attributes within images, including color, texture, and geometry. Widely utilized feature extraction algorithms in computer vision consist of Scale-Invariant Feature Transform Speeded Up Robust Features. These algorithms systematically identify distinctive image components, contributing to subsequent analysis [4].

Accurate object recognition is crucial in the field of picture analysis. R-CNN, YOLO, and SSD are just some object identification systems that excel at locating specific items in a picture. These methods use complex models to track down the things of interest, allowing for a more nuanced comprehension of their contexts.

Segmenting images into meaningful regions forms another critical aspect of image analysis. Image segmentation algorithms, like watershed segmentation, mean-shift segmentation, and graph-based segmentation, divide images into coherent segments according to attributes like color, texture, or intensity. This partitioning enhances subsequent analysis by focusing on localized regions of interest [5].

Clustering is used in photographs to group similar ones based on characteristics like color, texture, or shape. K-means clustering, hierarchical clustering, and DBSCAN are notable clustering algorithms that group images into clusters using similarity measures. This clustering is useful for unearthing previously unseen patterns in large image databases.

1Research Scholar, JNTUA, Ananthapuramu, Andhra Pradesh, India.
e-mail: sarithadasari123@gmail.com
2Professor, Department of CSE, S.V. University, Tirupati, Andhra Pradesh, India.
e-mail: ramamohansvu@yahoo.com
3Professor, Department of CSE, JNTU, Kalikiri, Andhra Pradesh, India
e-mail: eswarcsejntua@gmail.com

Reducing the dimensionality of high-dimensional image data simplifies processing and analysis. Dimensionality reduction techniques like PCA, t-distributed Stochastic Neighbor Embedding (t-SNE), and LLE facilitate this by projecting data into lower-dimensional spaces. This streamlining accelerates subsequent computations while retaining critical information.

Artificial neural network-based deep learning techniques provide game-changing potential for various image-related tasks, including classification, recognition, and segmentation. Basic models in this area include convolutional and recurrent neural networks and generative adversarial networks. These models use stacked networks of nodes to automatically learn complex visual characteristics, expanding their range of interpretation [11], [12].

Selecting the appropriate algorithm hinges on the distinct requirements of the image task, as each algorithm possesses specific strengths and weaknesses tailored to different challenges. By orchestrating these techniques, image analysis is a powerful tool for gleaning meaningful insights from visual data [13], [14].

Semantic Maps

Semantic maps are visual representations that illustrate the relationship between different concepts or ideas. These maps organize information to make it easier to understand and remember. Semantic maps can take many forms but typically include a central concept or idea surrounded by related concepts or ideas.

One of the most common types of semantic maps is a concept map. Concept maps are a visual tool employed to illustrate the interconnectedness and associations of many concepts or ideas. Typically, they feature a primary concept or theme linked to multiple additional concepts that are somehow related. These cartographic representations have the potential to facilitate the acquisition of novel knowledge among students, aid in the structuring of thoughts for writing or research endeavors, and enhance comprehension of intricate subjects for individuals.

Mind maps represent a further category of semantic mapping. Mind maps share similarities with concept maps, although they tend to exhibit a more unrestricted and flexible structure. Mind maps establish a central concept or theme from which several interconnected thoughts or subtopics emanate. Mind maps are a versatile tool employed to generate ideas, strategize tasks, or structure thoughts and information [6].

Semantic maps are a powerful tool for organizing and understanding complex information. By visually representing the relationships between different ideas or concepts, semantic maps can help individuals to understand better and retain new information.Several types of semantic maps are used for different purposes. Here are some common types of semantic maps:

Concept Maps: Concept maps illustrate the relationship between different concepts or ideas. They typically include a central concept or theme connected to several related concepts. These maps can help students learn new information, organize ideas for writing or research projects, or help individuals understand complex topics.

Mind Maps: Mind maps are similar to concept maps but are typically more freeform in structure. Mind maps often start with a central idea or theme and branch out into related ideas or subtopics. Mind maps can be used to brainstorm ideas, plan projects, or organize thoughts and information.

Spider Maps: Spider maps organize information around a central topic or theme. They typically start with a central idea or theme and branch into related subtopics or categories. Spider maps are often used to plan writing projects or to organize research.

Flow Charts: Flow charts illustrate a process or series of steps. They typically include boxes or nodes representing different stages of a process and arrows connecting the boxes to show the flow of information or actions. Flow charts are often used to plan workflows or to illustrate decision-making processes.

Venn Diagrams: Venn diagrams are employed as visual tools to depict the interconnections and associations among distinct data groups. Commonly, Venn diagrams consist of circles or ellipses that symbolize distinct sets, and the region where these circles meet signifies the shared elements between the sets. Venn diagrams find frequent applications in mathematics, statistics, and data analysis.

In general, semantic maps serve as a potent instrument for arranging and comprehending intricate information. Moreover, several forms of semantic maps can be employed to cater to distinct objectives.

2. Multi-Child Semantic Maps (MCSM)

MCSM also known as cluster maps, organizes information around a central theme or concept. They are similar to spider maps, but instead of branching out into subtopics or categories, they group related concepts or ideas into clusters. Multi-child semantic maps are particularly useful for brainstorming or generating new ideas, as they allow individuals to organize their thoughts around a central concept in a way that is visually appealing and easy to understand [7]. They are also useful for organizing research or planning writing projects, as they allow individuals to group related concepts or ideas and identify connections between them.

To create a multi-child semantic map, start with a central concept or theme and write it in the center of the page. Then, identify related concepts or ideas and group them into clusters around the central concept. Use lines or arrows to connect the clusters to the central concept, and use color coding or different shapes to differentiate between the clusters.

MCSM can be created using pen and paper or with digital tools like mind mapping software. They are flexible tools used in various settings, from classrooms to boardrooms, and can be adapted to suit various learning styles and preferences [8].

3. Proposed Research Area

This algorithm is proposed to horizontally verify the area from the zero barrier to the null barrier, which carries a thorough structure-based verification methodology for successfully plotting the Tumor in the given MINING Image. This algorithm gives the exact substantial area for the x-axis area of the Tumor, which is blotted in the specific area.

Algorithm 1:

Objective: The objective of the algorithm is to verify the presence of tumors in a medical image dataset (A1) and determine their coordinates along the y-axis for further analysis. The algorithm aims to efficiently process the dataset, identify patterns indicative of tumors, and accurately locate their positions.

Input:

A1: Raw medical image dataset containing images labeled as {x1, x2, x3, ..., xn}

Output:

Coordinate Axis B1: The y-axis coordinates of detected tumor regions.

Initialization:

Initialize the algorithm by providing Dataset A1 (MINING Image Dataset) as the input.

Steps:

Step 1: Initialize the algorithm to begin the iterative process through the images in dataset A1.

Step 2: For each image xi in A1, perform the following steps:

a. Check if image xi indicates the presence of a tumor:

i. If the pattern associated with a tumor is detected in image xi:

- Print "Pattern Matched."
- Return the y-axis coordinate for verification and analysis (Coordinate Axis B1).
- Store the y-axis coordinate for future reference.

ii. Else (if the pattern associated with a tumor is not detected):

- Return a modulated axis value indicating no tumor presence.

Step 3: After analyzing the current image, continue with the iteration:

a. If the total number of images processed (i) is less than the total number of images (n) in A1:

Continue the loop to process the next image.

b. If all images in A1 have been processed:

Stop processing and exit the loop, as the entire dataset has been analyzed.

This algorithm follows a systematic approach to analyze each image within the dataset A1 for tumor presence. It iterates through the images, and for each image, it checks whether the specific tumor pattern is present. If the pattern is found, it confirms the tumor's presence, stores the y-axis coordinate, and proceeds with the analysis of the next image. If the pattern is not detected, it indicates the absence of a tumor. Once all images have been processed, the algorithm continues with the next image if available or concludes the process, having completed tumor detection for the dataset.

The algorithm begins by initializing the input dataset A1, which contains medical images to be analyzed for tumor presence. It iterates through each image in A1 using a loop, aiming to identify a specific pattern indicating a tumor's presence. For each image xi in A1, the algorithm checks if the pattern associated with a tumor is present.

If the pattern is detected, the algorithm confirms the presence of a tumor, prints a confirmation message, and returns the y-axis coordinate indicating the position of the tumor. This coordinate is stored for further analysis. If no tumor pattern is detected in the image, the algorithm returns a modulated axis value to indicate the absence of a tumor.

The algorithm continues this process until all images in A1 have been processed. It keeps track of the number of processed images (i) and compares it to the total number of images (n) to determine whether more images are needed. The algorithm stops processing once all images have been analyzed [9] [10].

Progressive Updating of the system, which results in the final approach of the x-axis verification process and the detection methodology, is executed.

Calculating the Area

The given input A1 is the MINING Image Dataset. The repository consists of more datasets and is presented in Repository I. Subjected I is the total repositories for substantial data verification.

I→{a1,a2,a3,a4,a5,a6,a7,a8,a9,a10}

Given

A1(x1) →I
x1→a1 to a10 = Fail
x2→a1 to a10 = Fail
x3→a1 to a10 = Fail
x7→a1 to a10 = Success

In the given input, the Row Value x7 matches with the existing systematic input repositories, resulting in the initial plotting at the x7 and verification prolonged up to the x20 coordinate axis to find where else the success node is presented.

x7→a1 to a10 = Success
x8→a1 to a10 = Success
x9→a1 to a10 = Success
x10→a1 to a10 = Success

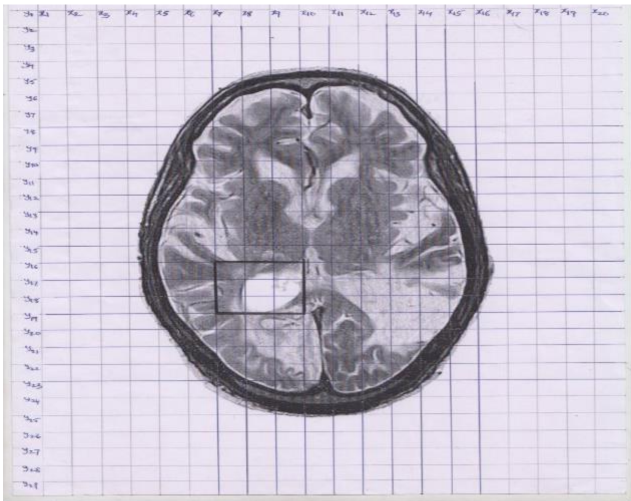

Fig 1: X-Axis plotting of the Inputted MINING Image

Within the aforementioned group, the coordinated action denoted as x7→x10 yields the consequent vector as success, but the action x11→a1 to a10 is deemed a failure.
The vector x8→x20, which represents the existence of a tumor in the given mining image, consistently yields a failure outcome.
The collapse value is represented in a hierarchical structure of coordinates from the initial to the final state.

x1→a1 to a10
Fail

x7→a1 to a10
Success

x10→a1to a10
Fail

x20→a1 to a10

The Refracted area x7→x10 is plotted for the x-axis presence of the Tumor.

The present approach is suggested to verify the augmentation from zero to null barriers vertically. The present study employs a structure-based enhanced technique to validate the accurate representation of the Y-axis in the provided MINING Image. The technique presented in this study precisely determines the spatial arrangement of tumor existence within mining images.

Algorithm 2:

Input: Initialize the algorithm by providing the input B1, representing a substantial analysis result.

Output Objective: The algorithm aims to determine the exact Y-coordinates representing the location of the tumor's existence.

Input B1: The algorithm takes the input B1, which appears to be the output or result obtained from some substantial analysis. It seems that this result is linked to the presence or characteristics of a tumor.

Fuzzy C Means Logic: The algorithm utilizes a Fuzzy C Means (FCM) logic for further analysis. FCM is a clustering technique often used for data partitioning and categorizing.

Initialization: Initialize the Fuzzy C Means Logic. This involves setting up the FCM parameters and configurations required for the subsequent analysis.

Return Y-Coordinates: The algorithm analyzes the data using the FCM logic. Once the analysis is completed, the algorithm returns the exact Y-coordinates corresponding to the specific location(s) where the presence of the tumor has been identified. These Y-coordinates precisely indicate where the tumor exists within the analyzed dataset.

Progressive Updating of the system results in the final approach of the y-axis verification process, and the detection methodology is executed.

Calculating the Area
The provided input A1 refers to the MINING Image Dataset. The collection of datasets is more extensive and is documented in Repository I.

The subject of interest pertains to the comprehensive collection of repositories utilized to verify structured data, specifically about the y-axis.

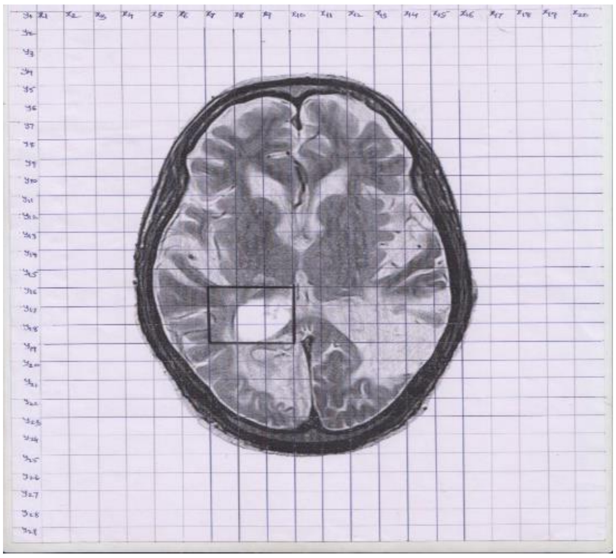

Fig 2: Y-Axis plotting of the Inputted MINING Image

I→{a1,a2,a3,a4,a5,a6,a7,a8,a9,a10}
Given

A1(y1) →I
y1→a1 to a10 = Fail
y2→a1 to a10 = Fail
y3→a1 to a10 = Fail

y16→a1 to a10 = Success

In the given input, the Row Value y16 matches with the existing systematic input repositories, resulting in the initial plotting at the y16 and verification prolonged up to the y30 coordinate axis to find where else the success node is presented.

y16→a1 to a10 = Success
y17→a1 to a10 = Success
y18→a1 to a10 = Success
y19→a1 to a10 = Success

In the group mentioned above, the coordinated y16→y30 gets the resultant vector as success,
y20→a1 to a10 = Fail

The resultant vector y20→y30 remains fails, which plots the failure of the presence of the Tumor in the given MINING Image. The collapsed value is presented in the hierarchy from the initial to the final coordinates.

y1→a1 to a10

Fail

y16→a1 to a10

Success

y19→a1to a10

Fail

y30→a1 to a10

The Refracted area y16→y30 is plotted for the y-axis presence of the Tumor.

4. Plotting the Presence of Tumor

The presence of the Tumor was calculated using the plots overlapped from the given input image and repository image.

In X-axis Coordinate Plotting

A1→x7>x8>x9>x10

A1={x7(I),x8(I),x9(I),x10(I)}

Resultant Vector R

R={x7(a1,a2,a3,a4,a5,a6,a7,a8,a9,a10)}

R={x8(a1,a2,a3,a4,a5,a6,a7,a8,a9,a10)}

R={x9(a1,a2,a3,a4,a5,a6,a7,a8,a9,a10)}

R={x10(a1,a2,a3,a4,a5,a6,a7,a8,a9,a10)}

Point Scale plotting,

R=x7(Plotted), x8(Plotted), x9(Plotted), x10(Plotted)

R→$[x7 \quad x8 \quad x9 \quad x10]$

In Y-axis Coordinate Plotting

A1→y16>y17>y18>y19

A1={y16(I),y17(I),y18(I),y19(I)}

Resultant Vector R

R={y16(a1,a2,a3,a4,a5,a6,a7,a8,a9,a10)}

R={y17(a1,a2,a3,a4,a5,a6,a7,a8,a9,a10)}

R={y18(a1,a2,a3,a4,a5,a6,a7,a8,a9,a10)}

R={y19(a1,a2,a3,a4,a5,a6,a7,a8,a9,a10)}

Point Scale plotting,

R=y16(Plotted), y17(Plotted), y18(Plotted), y19(Plotted)

$$R\rightarrow\begin{bmatrix} y16 \\ y17 \\ y18 \\ y19 \end{bmatrix}$$

Plotting the whole scale tumor detecting method

$$R\rightarrow\begin{bmatrix} x7y16 & x8y16 & x9y16 & x10y16 \\ x7y17 & x8y17 & x9y17 & x10y17 \\ x7y18 & x8y18 & x9y19 & x10y18 \\ x7y19 & x8y19 & x9y19 & x10y19 \end{bmatrix}$$

Scaling with real-time MINING image, which is given as input**.**

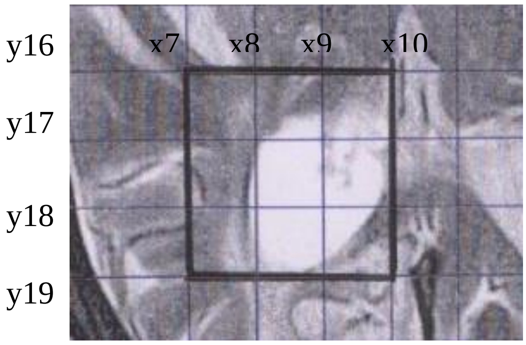

Fig 3 – Exact Plotting of the Tumor area in the given input image.

5. Resultant Vector

Step 1: Begin by initializing the algorithm. Initially, the first image (x1) data is added for the result finding.

Input:

I = R1 (Initialized with data from x1)

Efficient dataset Y = {x1, x2, x3, ..., x10}

Step 2: Create a checklist using the efficient dataset Y, containing images {x1, x2, x3, ..., x10}.

Step 3: Input x1 into the dataset Y.

Step 4: Check the inputted data (x1) with the dataset Y.

Input I after adding x1:

I = X1 {x1, x2, x3, ..., x10}

Step 5: Compare each element in X1 (x1, x2, x3, ..., x10) with the elements in the dataset Y.

For each x in X1:

If x is present in Y:

Result: Tumor Location (Indicates that a tumor is detected at the location of image x)

Else:

Result: Change the Log (Indicates that there is no tumor at the location of image x)

Step 6: Calculate the Image for X1 using the R1 dataset, considering both the Horizontal and Vertical Planes.

Horizontal Plane Calculation:

Perform iterations to process each element in X1.

Calculate image R1 for each element x in X1.

Step 7: Repeat the above steps for other images in the dataset Y (x2, x3, ..., x10).

The algorithm's main objective is to detect the presence of tumors in a dataset of medical images. It begins by initializing the input, where R1 contains data from the first image, x1. An efficient dataset Y is prepared, consisting of images {x1, x2, x3, ..., x10}.

A checklist is created using the dataset Y to aid in the verification process. The algorithm then adds the data from the first image, x1, into the dataset Y. Next, it compares each element in X1 (constructed from R1) with the elements in Y to determine if any of the images match.

If a match is found, it signifies the presence of a tumor at the location of the matched image. If no match is found, the algorithm indicates no tumor at the location of the unmatched image and advises a change in the log.

The algorithm calculates the image for each element in X1 using the R1 dataset. This calculation takes into account both the Horizontal and Vertical Planes.

The same process is repeated for the remaining images in the dataset Y (x2, x3, ..., x10), following the same steps of comparison, calculation, and recording of results.

Calculation:- ITERATION 2

Iteration 1 has no significance on the identification of Tumor. We shift to the second iteration.

Xm→Crm,Crn(Horizontal)

In this research context, the proposed algorithm outlines a method for detecting and assessing the significance of tumors within a given image dataset. The approach involves iterative analysis of distinct subspace segments, each represented by I2. These segments are characterized by fixed cue points (Crm and Crn) and corresponding coordinates (Pm, Pn), with a focus on logical pointer comparisons to discern variable differences. By evaluating the presence or absence of specific variables (Xm and Xn) at these cue points, the algorithm determines the presence or absence of tumors. Tabulated results in Table 2 offer a succinct representation of these evaluations. Furthermore, the algorithm concludes the detectable tumor areas based on horizontal bias, highlighting specific images (X6, X7, X8, X9, and X10) that exhibit tumor presence through logical combinations of cue points. This nuanced approach contributes to localized and context-aware tumor detection, offering potential insights for comprehensive image analysis.

6. Conclusion

Image mining is an emerging and influential field of study within computer programming. picture mining is a discipline that advances data mining techniques within picture analysis and processing. Image mining encompasses extracting concealed data and analyzing additional instances not explicitly depicted within the images. Image mining involves analyzing and extracting relevant information from visual data, specifically focusing on structures such as Image Arrangement, information handling, mechanical independence, and AI. Semantic guides are utilized to visualize the information contained inside image datasets. To produce the semantic guides, we propose the utilization of Multi-Kid Semantic Guides, which effectively present comprehensive visual representations. This study introduces a novel approach called the Multi-Youngster Semantic Guides with the K-C Means Grouping Algorithm, referred to as MCSMK-C Algorithm. The proposed algorithm facilitates two-way gathering, enabling the creation of picture bundles and conducting mining operations until the final stage of the image. The MCSMK-C Calculation incorporates the consideration of X and Y Coordinates to implement the mining process. The inclusion of transformational calculations facilitates efficient booking for all groups. The algorithm systematically examines the dataset's contents, assessing the quantity of items inside each category and determining whether it exceeds a predetermined threshold.

References

[1] Vaidya, Jaideep, and Chris Clifton. "Privacy preserving association rule mining in vertically partitioned data." *Proceedings of the eighth ACM SIGKDD international conference on Knowledge discovery and data mining*. 2002.

[2] Li, Lichun, et al. "Privacy-preserving-outsourced association rule mining on vertically partitioned databases." *IEEE transactions on information forensics and security* 11.8 (2016): 1847-1861.

[3] Chen, Jingxiang, et al. "A distributed decision tree algorithm and its implementation on big data platforms." *2016 IEEE International Conference on Data Science and Advanced Analytics (DSAA)*. IEEE, 2016.

[4] Beloglazov, Anton, and Rajkumar Buyya. "Optimal online deterministic algorithms and adaptive heuristics for energy and performance efficient dynamic consolidation of virtual machines in cloud data centers." *Concurrency and Computation: Practice and Experience* 24.13 (2012): 1397-1420.

[5] Bazarbayev, Sobir, et al. "Content-based scheduling of virtual machines (VMs) in the cloud." *2013 IEEE 33rd International Conference on Distributed Computing Systems*. IEEE, 2013.

[6] Ngenzi, Alexander, and Suchithra R. Nair. "Dynamic resource management in Cloud datacenters for Server consolidation." *arXiv preprint arXiv:1505.00577* (2015).

[7] Tran, Tony T., et al. "Resource-aware scheduling for data centers with heterogenous servers." (2015).

[8] Dong, Ziqian, Ning Liu, and Roberto Rojas-Cessa. "Greedy scheduling of tasks with time constraints for energy-efficient cloud-computing data centers." *Journal of Cloud Computing* 4.1 (2015): 1-14.

[9] Papalexakis, Evangelos E. "Automatic unsupervised tensor mining with quality assessment." *Proceedings of the 2016 SIAM International Conference on Data Mining*. Society for Industrial and Applied Mathematics, 2016

[10] Thakar, Pooja, and Anil Mehta. "Performance analysis and prediction in educational data mining: A research travelogue." *arXiv preprint arXiv:1509.05176* (2015).

[11] Madapudi, Rudra Kumar, A. Ananda Rao, and Gopichand Merugu. "Change requests artifacts to assess the impact on the structural design of SDLC phases." Int'l J. Computer Applications 54.18 (2012): 21-26

[12] Chalapathi, M. M., et al. "Ensemble Learning by High-Dimensional Acoustic Features for Emotion Recognition from Speech Audio Signal." Security and Communication Networks 2022 (2022)

[13] Ramana, Kadiyala, et al. "Leaf disease classification in

smart agriculture using deep neural network architecture and IoT." Journal of Circuits, Systems and Computers 31.15 (2022): 2240004.
[14] Kumar, V., M. Rudra Kumar, N. Shribala, Ninni Singh, Vinit Kumar Gunjan, and Muhammad Arif. "Dynamic Wavelength Scheduling by Multiobjectives in OBS Networks." Journal of Mathematics 2022 (2022).
[15] Goud, B. ., & Anitha, R. . (2023). Emerging Routing Method Using Path Arbitrator in Web Sensor Networks. International Journal on Recent and Innovation Trends in Computing and Communication, 11(4), 232–237. https://doi.org/10.17762/ijritcc.v11i4.6444
[16] Merwe, M. van der, Petrova, M., Jovanović, A., Santos, M., & Rodríguez, M. Text Summarization using Transformer-based Models. Kuwait Journal of Machine Learning, 1(3). Retrieved from http://kuwaitjournals.com/index.php/kjml/article/view/141
[17] Dhabliya, D. (2021). Delay-tolerant sensor network (DTN) implementation in cloud computing. Paper presented at the Journal of Physics: Conference Series, , 1979(1) doi:10.1088/1742-6596/1979/1/012031 Retrieved from www.scopus.com