



Machine Learning Technique for the Assembly-Based Image Classification System

Ambuj Kumar Agarwal^{1*}, D Angeline Ranjithamani², Pavithra M³, A Velayudham³, Anandaraj Shunmugam⁴ and Mohammed Ismail B⁵

Abstract

Additive manufacturing, or 3D printing, is a vital innovation in the field production processes. Furthermore, the decision to change the filling without influencing the outside creates a different vulnerability for 3D printer technologies. This research includes a clause to identify fraudulent filling problems in the printed object: 1) Look into malevolent faults in the 3D printing process, 2) Remove outliers from modeled 3D printer method photos, and 3) Perform an object detection test with one sample of the non-infill test set and another cluster of fault reinforced test set from the 3D printing process. Layer by layer, the photos are gathered from the isometric perspective of the program model display. The data extracted is provided to the developed algorithms, Naive Bayes method, and J48 decision trees. Among them, the Naive Bayes method shows a higher accuracy rate of 86%, and J48 decision trees show an accuracy of 96%.

Keywords: 3D printing; Machine learning; Image classification

Introduction

The blueprint of future manufacturing, such as cyber manufacturing systems, delineates a vision beneficial in cost, efficiency, and sustainability. However, cyber-attacks could threaten complex safety-critical manufacturing systems. The consequence of the cyber-attack on manufacturing systems could defect parts, cost of poor quality, damaged equipment, and loss of customer loyalty or even create a custom safety issue.

The ability to influence interior levels without influencing the outside of 3D printer devices presents distinct risks [1]. These characteristics may result in the unintentional creation of maliciously faulty parts. Turner conducted an experiment in which 3D printer users were uninformed that STL files had been intentionally modified, resulting in the production of a harmful part [2]. These flaws could lead to a decrease in yield load, a decrease in straining at failing, a variation in natural frequency, and other issues that compromise quality control and operation. Poor printing performance difficulties like an insufficient filling, flaws in thin walls, irregular emission, layer separation and breaking, mattress fall, and other faults render most 3D printers components unsuitable and dangerous to use.

*Corresponding author: Ambuj Kumar Agarwal, Chitkara University Institute of Engineering and Technology, Chitkara University, Punjab, India, E-mail: ambuj4u@gmail.com

Received date: August 31, 2021 Accepted date: September 15, 2021
Published date: September 22, 2021

In production, a recognition system was used to find flaws such as welding faults, visually inspecting of metal particles, and defect diagnosis in the rolling process, among other things. Likewise, finding faults in the production process using smart automation screening can efficiently remove specific filling risks. Simultaneously time, it can detect a publishing problem early on, rather than waiting until the end of a time-consuming manufacturing process to diagnose and destroy the part, wasting time, resource, and operation cost.

Background

By putting a vacuum into an ASTM Standard D638-10 tensile test specimen and conducting the tensile test, Sturm et al. [1] were able to evaluate the impact of gaps on component durability. The results of the survey revealed that all of the void-containing samples cracked at the blank region, while another sample failed properly. The yielding pressure was decreased by 14% on average, and the stress at the breakdown was dropped from 10.4% to 5.8%. The project also defined several terms. In composite materials, there are various vulnerabilities. It also advocated software inspections, hashing, quality control, and worker training as ways to mitigate these risks. The suggested methodology of quality control consists of physical factors such as tool pin temperatures, which are derived implicitly from machine settings. In contrast to Sturm et al., this study also used processes. In comparison to Sturm's real application, this research adopted picture machine learning and categorization, and 2) this work applied a new technique using software models are formulated.

Vincent et al. [3] utilized measurement and control approaches to detect alterations in the inherent behavior of a produced item. This technology might be used with 3D printed parts, but it would necessitate adaptable systems for engineering structures. Thus it emphasized the necessity for innovative methodologies above existing quality assurance detection systems assaults. Unlike Vincent et al. This work can identify modifications in 3D printers without modifying adaptable models for physical health monitoring, and it uses tracking during the production process but rather afterward.

To track the leakage of the cyber database, Chhetri et al. studied the auditory side-channel of Fused Deposition Simulation (FDM) based on rapid prototyping equipment, such as 3D-printers [4]. This research differs from Chhetri's in the following ways: 1) Chhetri concentrated on boundary restoration, but this work focuses on infill problem; and 2) Chhetri employed sounds, whereas this work employs foresight. Furthermore, automatic identification assessment based on image categorization has been used in production processes for defect identification, like weld fault diagnosis, thorough check of metallic surfaces [5], defect object tracking in the hot rolling process [6], and so on. The focus of this research is on additive manufacturing technologies.

Identification of Malware Defects in the Additive Production Line

The CAD model, test file, the tool path file, and the machine itself are the four main vulnerabilities that can be exploited during the fabrication. Corruption/encryption, resizing, indents/protrusions, and vertices displacement and holes could be results of an assault [6-15].

At the start of the industrial process, file corruption/encryption can be discovered or identified. With a volumetric test, you may identify resizing, indents/protrusions, and vertex motion after the 3D printing and before the legislature. A void inside a 3D-printed piece, on the other hand, was seen because it might be encapsulated within the framework. The vacancy may cause a minor shift in density, but it is hard to determine. Modifications in the inner arrangement can lead to improvements in physical properties like strength and natural frequency, which can lead to poor performance. Unexpected vibrations, for example, might cause stiffness to deteriorate or possibly fail. During the fabrication, the infill structural change in the 3D printer has already been proved to be harmful and unnoticed. In this case, visual recognition during the procedure is performed [16-19].

A few of the possible infills are described below to use vision-based assessment during 3D printers to decrease inflammation in infill design. Demonstrates 8 infill variations with a 10% frequency using the Slicer 1.2.9 program. Hexagons include Concentric, Section, Rectilinear, Computing Curves, Elliptic Wire, and Octagram Waves.

There are more infill patterns than any of those illustrated in Figure 1, although the Hexagon (Hexagon) infill pattern is among the most

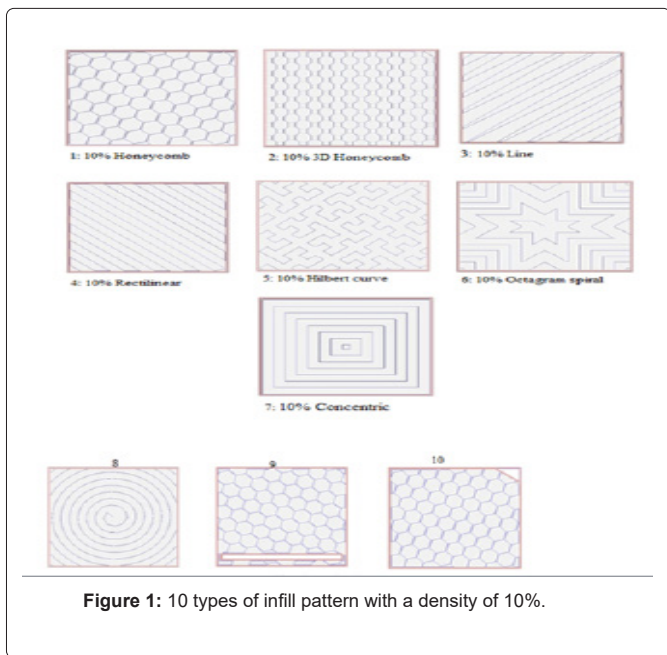


Figure 1: 10 types of infill pattern with a density of 10%.

prevalent in exercise, as Hales proved in 1999 [20,21]. This is also regarded to be the most economical reinforcement and the fastest to produce. Void, Irregular polygon, circle, rectangle, and triangle are 5 different infill defect patterns that can be used to depict deliberately broken portions. Depicts the five problems in the hexagonal reinforcement pattern.

In this paper, the attack method is described as follows: the invasion can cause one or more infill product defects, as seen in Figure 2. The attack could be conducted without eliciting any signals from the firewall or solid wiring harness. The fault could be randomly spread

throughout the component or concentrated in one region. The size of the fault may change automatically depending on the size of the faulty component. There is no manager to oversee the printing process whenever the filler is enclosed, so the issue cannot be assessed after the industrial process.

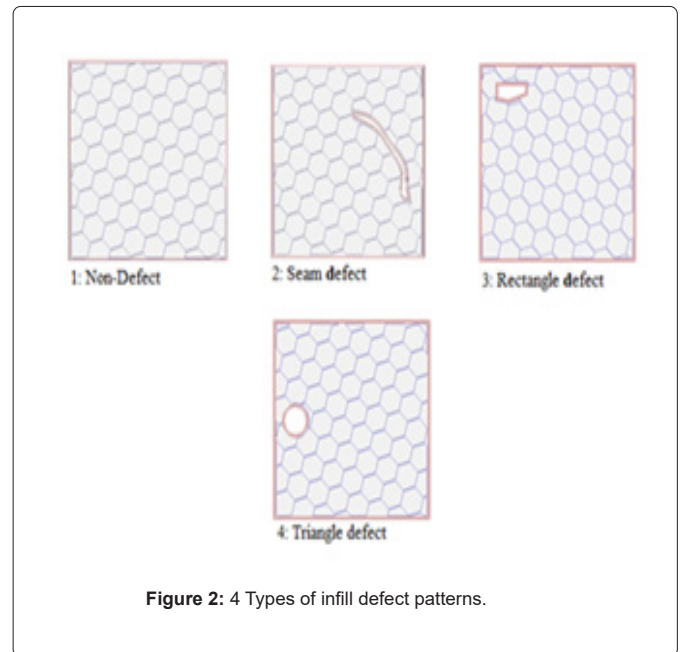


Figure 2: 4 Types of infill defect patterns.

Detection System

Preliminary design

A visual device is installed on a 3D printer to determine infill issues utilizing imagination laboratory tests. In this investigation, the patch flaws also were discovered from the top view of the work piece. On top of the 3D printer 4, we create an automated vision evaluation method.

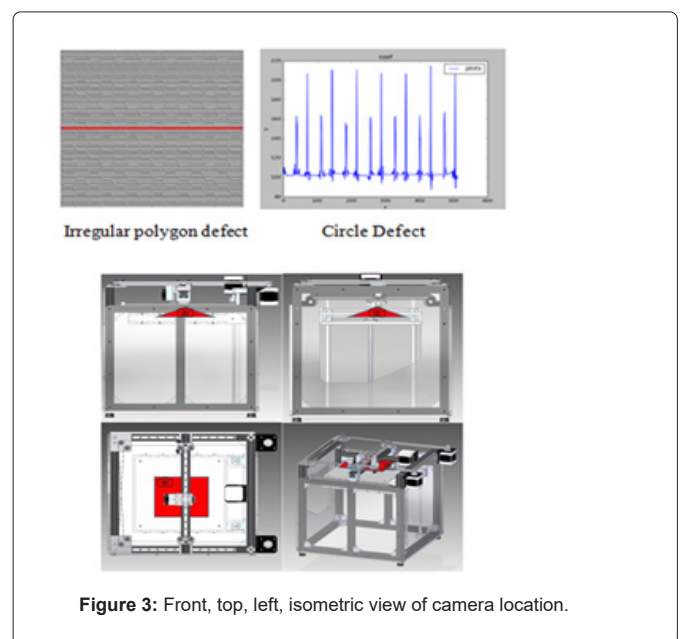


Figure 3: Front, top, left, isometric view of camera location.

A webcam installed on top of the 3D image and placed near the intrusion was used to get a quick glimpse at the infill production process. In a specific context, a supplementary thermal ray can enhance picture quality. In a software simulation viewing scenario, photos can be collected quickly. Every piece of software has its preview, so we selected one that correctly simulates levels and routes. As depicted in Figure 3. The camera's location and viewpoint are depicted by the red triangle region in the front and left views. The camera's view is depicted by the rectangular grid in the center of the top image. As he examines the entire region of infill, the attacker's eyesight may move with him.

Image generation

Images are taken from the MakerBot Desktop 3.9.1 display feature in the 3D printing software. The photos are 512 by 512 pixels in size. A total of 156 photos are taken for the training and testing of the predictor.

A total of 65 non-defective parts photos are taken and classified as group A. From A01 to A65, each representation is obtained a number. To raise the level of training photos, non-defect group A images are recorded every 3-5 stages during the printing process, with infill thickness varying from 8% to 12%. Group B consists of 91 photos of faulty parts that have been labeled. From B01 to B91, each image is given a number. During the manufacturing process, the faulty group B photos are collected every 3-5 levels, with permutations of five separate flaws depicted in Figure 4. For group B, the infill frequency is 10%.

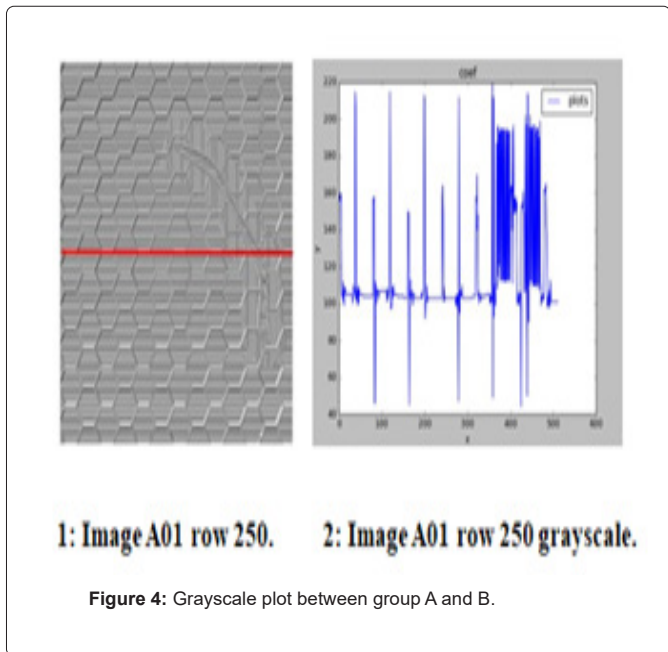


Figure 4: Grayscale plot between group A and B.

Feature Extraction

Python 2.7 is used to perform the image processing techniques. We can see repeating maxima, one moderate rise succeeded by one high peak, in couples, by showing the monochrome impact of non picture A01 row 250 (shown in red in Figure 4-1). By projecting the grayscale value of picture B30 row 250, we can see the repeating peaks in pair from Figure 4-2 at first but substituted by consistent turbulence.

We used a monochrome cutoff of 120 to define peaks. As a result, the peaks in Figures 4-1 and Figures 4-2 can be tallied to classify them. The average greyscale value in is 107, with a normal derivation of 17, and a total of 30 pixels monochrome bigger than the limit. The average greyscale value in is 120, with a normal derivation of 32, and a total of 129 pixels monochrome bigger than the boundary.

Each area is split into eight portions for image retrieval, as illustrated in Fig. 6. Each segment has 64 rows for a total of 32768 pixels. For defect categorization, the following features can be extracted.

- Halftones average in each area.
- Each part has a typical monochrome derivation.
- A monochrome pixel count greater than 120

In a conclusion, each image comprises 24 characteristics, divided into eight sections, each with three features.

Image classification algorithm

The Naive Bayes method is a classification algorithm that counts the occurrence and permutations of entries in a data set to create a set of possibilities. Compared to the value of the target class, the method assigns the Bayes theorem and implies that all qualities are stable. In practical applications, this prediction is rarely valid. In many support vector situations, though, the method continues to work well enough and learn quickly.

In a nutshell, naive Bayes is a feature vector theory in which an issue instance, denoted by a matrix, is categorized. It builds a binary tree for you. In a classification task, the decision tree strategy is by far the most useful. A tree is built to represent the categorization process using this approach. Once the tree is constructed, it is matched to each item in the information, resulting in categorization [8].

$X = (x_1, \dots, x_n)$ representing some n features (independent variables), it assigns to this

instance probabilities $p(C_k | x_1, \dots, x_n)$ for each of K possible outcomes or classes in this

work, the conditional probability model is represented by a vector

$$X = (\text{Mean}_1, \dots, \text{Mean}_8, \text{SD}_1, \dots, \text{SD}_8, \text{NP}_1, \dots, \text{NP}_8) \tag{1}$$

Where

Mean $x_{1..8}$ stands for mean grayscale value for each image section 1 to section 8. Value

SD $x_{1..8}$ stands for grayscale standard derivation value for each image section 1 to section 8.

Value NP $x_{1..8}$ is the number of pixels in a grayscale image

For each picture segment 1 to 8, the value must be greater than the threshold. Also, apply to example possibilities in the same way. $p(A | \text{Mean}_1, \dots, \text{Mean}_8, \text{SD}_1, \dots, \text{SD}_8, \text{NP}_1, \dots, \text{NP}_8)$

$p(B | \text{Mean}_1, \dots, \text{Mean}_8, \text{SD}_1, \dots, \text{SD}_8, \text{NP}_1, \dots, \text{NP}_8)$

The probability distribution can be divided as follows using Bayes' theorem:

- Changed Words
- Structural Changes
- Thesaurus

$$P(CK|X) P(X|CK)P(CK)P(X) \tag{2}$$

Excrete as follows in this job: CK A or B, the conditional probability can

$$P(A|X) P(X|A)P(A) \tag{3}$$

$$P(X) P(B|X) P(X|B)P(B) \tag{4}$$

P(X) In the Weka data mining tool, J48 is an open-source Java implementation of the C4.5 algorithm. It is a commonly accepted inlaying for actual artificial intelligence. Figure 5 flow Chart for system development.

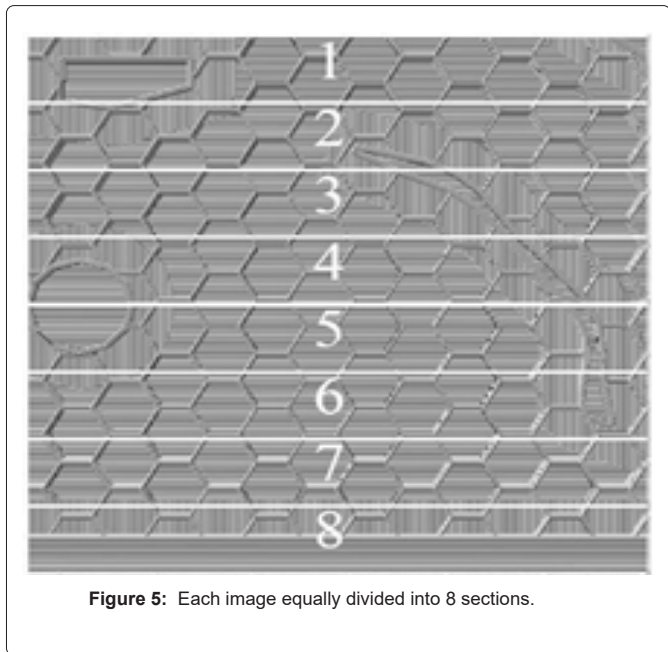


Figure 5: Each image equally divided into 8 sections.

Experiment Results

With tenfold cross-validation, we fed information into a naïve Bayes classifier and J48 Decision trees. The results are summarized in Table 1, which also includes the discriminated function for both techniques. The efficiency of the Naive Bayes classifier is 85.26%, with a strong True Alarm Rate and a fewer detection negative rate. The reliability of the J48 decision trees is 95.51%, with a low error rate and a fewer detection negative rate.

With tenfold cross-validation, we fed information into a Naïve Bayes Classifier and J48 Decision Trees. The results are summarized in Table

Table 1: Performances of Naive Bayes and J48 on testing data.

Algorithm		Navier Bayes		J48 decision trees	
Class		Predicted class			
		A	B	A	B
Actual	A	63	1	61	3
Class	B	22	70	4	88
Accuracy		85.26%		95.51%	

1, which also includes the discriminated function for both techniques. The efficiency of the Naive Bayes classifier is 85.26%, with a strong

true alarm rate and a fewer detection negative rate. The reliability of the J48 decision trees is 95.51%, with a low error rate and a fewer detection negative rate.

The Receiver Operating Characteristic (ROC) Curve for the naive Bayes classifier is shown in Figure 6, as well as the error function for the J48 decision tree is shown in Figure 7. Conspiring the True Positive Rate (TPR) against the False Positive Rate (FPR) yields the curve (FPR). The area under the ROC curve is used to determine accuracy.

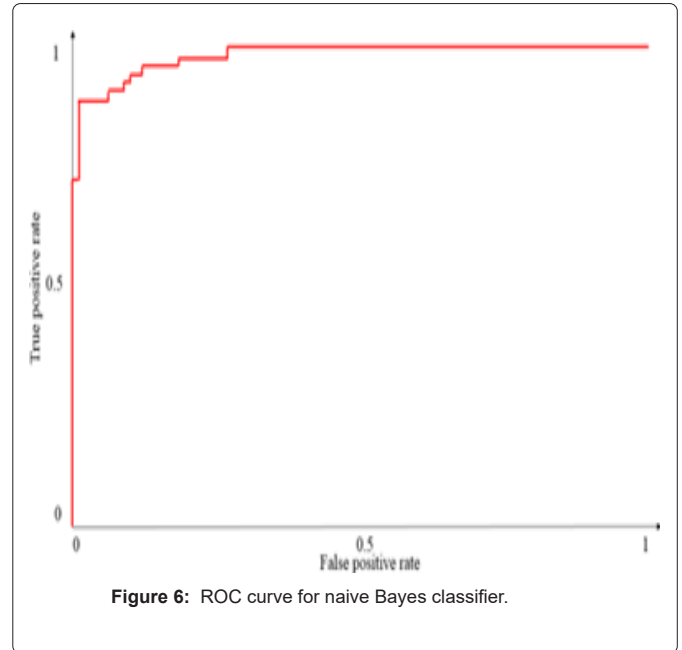


Figure 6: ROC curve for naive Bayes classifier.

The research was conducted out using a Windows 7 64-bit PC with an Intel Core i5-2520 M 2.5 GHz processor and 8 GB RAM. The image retrieval phase takes 47.17 seconds for 156 photos, averaging

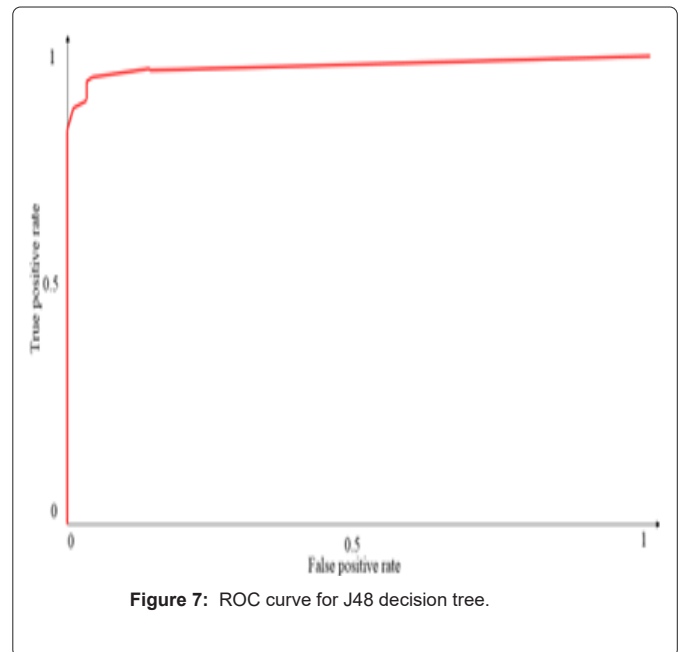


Figure 7: ROC curve for J48 decision tree.

0.3 seconds per picture, according to the processing time. The J48 Decision Trees and the Naive Bayes Classifier are both incredibly fast and can be used in real-time identification.

Conclusion

This paper proposes an artificial intelligence picture categorization approach for detecting 3D printing infill defects. The approach was thoroughly investigated in terms of functionality, feature selection, and implementation of the Naive Bayes classifier and J48 decision trees, respectively, with the efficiency of 85.26% and 95.51%. In the coming, the three goals can be achieved: 1) Use of this approach on some other types of filling, 2) Use of this technique in a physical world with 3D printers and sensors, 3) Use of this technique for 3D printing low-quality errors, and 4) Use of other methods to discover best Instagram.

References

1. Sturm LD, Williams CB, Camelio JA, White J, Parker R (2014) Cyber-physical vulnerabilities in additive manufacturing systems. (SFF) Symposium pp: 951–963.
2. Turner H, White J, Camelio JA, Williams C, Amos B, et al. (2015) Bad parts: are our manufacturing systems at risk of silent cyberattacks?. IEEE Secur Priv 13: 40–47.
3. Vincent H, Wells L, Tarazaga P, Camelio J (2015) Trojan detection and side-channel analyses for cyber- security in cyber-physical manufacturing systems. Procedia Manuf 1: 77–85.
4. Chhetri SR, Canedo A, Abdulla M, Faruque A (2016) Poster: exploiting acoustic side-channel for attack on additive manufacturing systems. 41: 56-61.
5. Pernkopf F, O'Leary P (2003) Image acquisition techniques for automatic visual inspection of metallic surfaces. NDT E Int 36: 609–617.
6. Jia H, Murphey YL, Shi J, Chang TS (2004) An intelligent real-time vision system for surface defect detection. Proceedings of the 17th International Conference on pattern recognition, 2004. ICPR 2004 3: 239-242.
7. Hales TC (1999) The honeycomb conjecture. Discr Comput Geom 25: 1-22.
8. Deepthi T, Balamurugan K, Uthayakumar M (2021) Simulation and experimental analysis on cast metal runs behaviour rate at different gating models. IJESMS 12:156-64.
9. Devaraj S, Malkapuram R, Singaravel B (2021) Performance analysis of micro textured cutting insert design parameters on machining of Al-MMC in turning process. Int J Lightweight 4: 210-7.
10. Garigipati RK, Malkapuram R (2020) Characterization of novel composites from polybenzoxazine and granite powder. SN Applied Sciences. 2: 1-9.
11. Yarlagaadada J, Malkapuram R (2020) Influence of carbon nanotubes/graphene nanoparticles on the mechanical and morphological properties of glass woven fabric epoxy composites. INCAS Bull 12: 209-18.
12. Rama Krishna M, Tej Kumar KR, DurgaSukumar G (2018) Antireflection nanocomposite coating on PV panel to improve power at maximum power point. Energ Source Part 40: 2407-14.
13. Yarlagaadada J, Malkapuram R, Balamurugan K (2021). Machining studies on various ply orientations of glass fiber composite. In advances in industrial automation and smart manufacturing. pp: 753-769.
14. Ezhilarasi TP, Kumar NS, Latchoumi TP, Balayesu N (2021) A secure data sharing using IDSS CP-ABE in cloud storage. In advances in industrial automation and smart manufacturing. pp: 1073-1085.
15. Mishra P, Jimmy L, Ogunmola GA, Phu TV, Jayanthiladevi A, Latchoumi TP (2020) Hydroponics cultivation using real time IOT measurement system. J Phys Conf Ser 1712: 012-040.
16. Sridharan K, Sivakumar P (2018) A systematic review on techniques of feature selection and classification for text mining. Int J Bus Inf Syst 28: 504-518.
17. Vemuri RK, Reddy PCS, Kumar BP, Ravi J, Sharma S, et al. (2021) Deep learning based remote sensing technique for environmental parameter retrieval and data fusion from physical models. Arab J Geosci 14:1-10.
18. Rishu C, Verma S, Rama Krishna C (2017) A survey on driver behavior detection techniques for intelligent transportation systems. In 2017 7th International conference on cloud computing, data science and engineering-confluence. 10: 36-41.
19. Parul D, Sharma B (2017) A survey on IOT architectures, protocols, security and smart city based applications. In 2017 8th International Conference On Computing, Communication and Networking Technologies (ICCNT). pp: 1-5.
20. J Gholap (2012) Performance tuning of J48 algorithm for prediction of soil fertility. AJCST 2: 8.
21. Witten IH, Frank E, Trigg L, Hall M, Holmes G, Cunningham SJ (1999) Weka : practical machine learning tools and techniques with java implementations. Dept of Computer Science pp: 192–196.

Author Affiliations

Top

¹Chitkara Institute of Engineering and Technology, Chitkara University, Punjab, India

²Department of Computer Applications, Francis Xavier Engineering College, Vannarpet, Tirunelveli, Tamil Nadu

³Department of Computer Science and Engineering, Jansons Institute of Technology, Karumathampatti, Coimbatore

⁴Department of Computer Science and Engineering, Jansons Institute of Technology, Karumathampatti, Coimbatore, Tamil Nadu, India

⁵School of CS and IT, DMI St. John the Baptist university, Mangochi, The Republic of Malawi, Southeastern Africa

⁶Department of Information Technology, Kannur University, Mangattupramba Campus Kannur, Kerala

Submit your next manuscript and get advantages of SciTechnol submissions

- ❖ 80 Journals
- ❖ 21 Day rapid review process
- ❖ 3000 Editorial team
- ❖ 5 Million readers
- ❖ More than 5000 
- ❖ Quality and quick review processing through Editorial Manager System

Submit your next manuscript at • www.scitechnol.com/submission