OCL Based Approach for Sustainable ML Model Development

Gautam Hanumanthu[1], Venkateswarlu Lendale[2] *Ramesh* G[3*], *Prabhakar* Kandukuri[1], *Mohd.*Amir[4]

[1]Department of Computer Science and Engineering, Koneru Lakshmaiah Education Foundation Guntur, India.
[2]Department of Computer Science and Engineering, Geethanjali College of Engineering and Technology, Cheeryal, Keesara, India.
[3]Department of CSE, Gokaraju Rangaraju Institute of Engineering and Technology, Hyderabad, India.
[4]Uttaranchal Institute of Management, Uttaranchal University, Dehradun, India

Abstract. It became a bottleneck for the Machine Learning (ML) researchers to select/develop a sustainable model for a particular problem. Hence, there is a need for an approach to prepare a model with all constraints of the software system. The proposed approach is based on Object Constraint Language (OCL) which is a declarative language for writing constraints on software artifacts, it is widely used for effective representation of Functional Requirements (FR's) and Non-Functional Requirements (NFR's). In the proposed system, the paddy leaf disease identification system is considered and proposed Leaf Identification Constraints (LIC) and Leaf Disease Identification Constraints (LDIC) based on OCL, for the proposed constraints the Convolutional Neural Network (CNN) is chosen, as it can handle diverse range of input data and large volume of concurrent requests. To satisfy other constraints of the system, the Auto encoders are used along with CNN and the input data was take in the form of thermal imaging. This system was evaluated with test data and validation data and obtained the accuracy of 90.6%. And 84.8 was attained by earlier researchers before this approach.

1. Introduction

The Machine Learning uses mathematical models and statistical analysis to enables computers to learn and improve their performance without explicit program. In the modern era the machine learning techniques became an emerging solutions for the real-time problems. As these ML models are rowing day by day, it become a tough task for the model developers to identify a suitable model or modifying the existing model or to developing a new model.

*Corresponding author: ramesh680@gmail.com

Hence, this research proposed a mythology for solving the above mentioned problem. It is based on OCL [1] to identify the constraints of problem and identifying the suitable model or finding a model and modify according to the requirements of the system. Basically the OCL is a constraint based language provides a way to express precise and unambiguous constraints on Software artifacts. These constraints can be applied to ML systems during its model preparation.

The proposed approach provides twenty constraints for leaf identification and twenty constraints for leaf disease identification. Based on these OCL constraints the ML model is prepared, and tested.

2. Literature Survey

In the field of machine learning, most of the developers are exercising different models and fining the suitable one at the end for the given problem. Chiu, focused on different regression models and compared the performance of the differential evolution algorithm to that of the genetic algorithm [2]. Using three benchmark datasets, the results show that the differential evolution method beats the genetic algorithm. The algorithms' performance is evaluated using R-squared, MAE, and RMSE. But this approach may not suitable for other algorithms like clustering and classification. Differential evolution (DE) algorithms [3] provides the most general method for optimizing the large and challenging problems. It was experimented with agriculture problems.

PK Ram [4] have suggested the work based on the evolutionary algorithm with the multilayer perceptron for gene analysis using microarray-based health care data. Analyzed genes are responsible for the disease prediction. Here, vectors are designed in efficient way and the fitness function is derived to measure the quality of each vector. They also have designed the genetic algorithm based model to select the good features for disease analysis. Here, chromosomes are designed and to evaluate the fitness of chromosomes, new fitness function is derived. Basically, the fitness function is evaluated using the conflicting objective function. Afterwards, machine learning classifiers are used to measure the accuracy of selected feature subsets [5].

Taskin [6] proposed feature-selection technique for hyperspectral image analysis for handling the problems of classification and dimensionality reduction. This research improved the classification accuracy, stable feature selection, and effective computational performance are all attributes of the approach. Kumar [7] proposed a system that leverages CNN techniques to analyze medical images and accurately detect the presence of brain tumors. The study demonstrates the effectiveness of deep learning in medical image analysis and highlights the potential of cloud-based solutions for efficient and secure tumor detection. Sasank [8] proposed a novel framework that combines CNN and LSTM machine learning approaches to improve evaluation and enable systematisation in diverse cervical spondylosis-related applications. In [9, 10] demonstrated how deep learning techniques, in especially CNNs and RNNs, have the ability to solve privacy and security problems. It analyses the use of DL approaches in attack classification, highlights the significance of intrusion and malicious detection at the node or peer level, and promotes the use of modern IDS systems to reduce harm and false alarms. In [11] the authors explored a novel approach in the form of an enhanced communication paradigm, introducing the Energy Aware Smart Home (EASH) framework. Within this study, we delve into the investigation of communication failures and various types of network attacks occurring in the context of EASH. By harnessing the power of machine learning techniques, we effectively distinguish the sources of abnormalities within the communication paradigm.

JS Kumar [12] proposed a method for analysing patient data using eXtreme Gradient Boosting (XGB) machine learning algorithm to provide diabetes risk predictions. This research highlights the usefulness of AI in healthcare by displaying machine learning's potential in diabetes screening and prevention. Thulasi [13] Providing the projections for Bitcoin prices based on previous data, allowing participants to make wise choices and possibly generate extra income. It is stressed how crucial market analysis and capitalization are, and it is acknowledged that Bitcoin is a substantial digital store of value. Ram et al [14], have proposed the novel genetic algorithm based on the autoencoder with ensemble classifiers for imbalanced health care data. Initially, the imbalanced data is balanced using the novel approach called as GAAE method. Here, genetic algorithm is evaluated through the autoencoder. Each chromosome of GA represents as an autoencoder. To measure the quality of chromosome, an error function is also designed by the authors. After balancing the dataset, feature selection is performed using the correlation coefficient approach. They proposed the feature clustering strategy using particle swarm optimization (PSO) technique for disease analysis. Here, the particles are designed in an efficient manner. Afterwards, the clustering scenario is developed using the correlation coefficient approach during the optimization of PSO process [15]. Accurate classification and identification of plant diseases are achieved through the implementation of computer-based image recognition schemes. An advanced classification approach based on Back Propagated Artificial Neural Networks (ANN) is employed to implement feature-based matching operations in artificial intelligence [16-20].

OCL is the one of the declarative language for writing constraints on the FR’s and NFR’s of a software system at design level. Bolognesi [21] emphasizes the advantage of using constraint-oriented approach for system decomposition. This approach is consistent with object oriented reasoning added advantages in terms of enabling conditions and validation time.

3. OCL Based Constraints

The declarative language OCL is recognized by Object Management Group (OMG) and used along with Unified Modeling Language for specifying the constraints for software systems. This research extending the use of OCL for ML model preparation. As a part of model preparation the paddy leaf disease identification problem is taken into the consideration. Here, there are twenty leaf identification constraints and twenty leaf disease constraints are considered and tabulated in Table1. The leaf disease description is tabulated in Table 2.

Table 1. Constraints for Leaf Identification and Leaf Diseases.

S.No.	Leaf Identification Constraints	Leaf Disease (Constraints)
1.	Healthy leaf images identification	Blast
2.	Diseased leaf image identification	Brown spot
3.	Invalid input Handling	Sheath blight
4.	Invalid image data	Bacterial leaf streak
5.	Incorrect dimensions of image data	Tungro
6.	images with invalid file formats	False smut
7.	Images with corrupted data	Rice tungro disease
8.	Images with invalid labels	Narrow brown leaf spot
9.	Multiple concurrent requests	Leaf scald
10.	Exceeding the maximum concurrent requests	Leaf spot
11.	Images with invalid dimensions	Rice blast
12.	Images with large dimensions	Sheath rot

13.	Images with invalid data type	Bacterial leaf blight
14.	Images with very low contrast	Curvularia leaf spot
15.	Images with very high contrast	Leaf scorch
16.	Grayscale images	Rhizoctonia leaf spot
17.	Images with low resolution	Brown stripe
18.	Images with high resolution	Rice leaf folder
19.	Images with low quality	Rice leaf mite
20.	Images with high quality	Magnaporthe oryzae

Table 2. List of twenty possible paddy leaf Diseases

S.No.	Leaf Disease	Leaf Disease Description
1.	Blast	A fungal disease that can cause oval or diamond-shaped lesions with gray centers and brownish borders on the leaves. It can also affect other parts of the plant, including the stem and grain.
2.	Brown spot	A bacterial disease that can cause small brown spots with yellow halos on the leaves. The spots can merge and form larger patches, leading to leaf yellowing and premature senescence.
3.	Sheath blight	A fungal disease that can cause brown lesions with yellow halos on the leaf sheaths, which can spread to the leaves and cause shredding of the leaf tissue.
4.	Bacterial leaf streak	A bacterial disease that can cause yellowish-brown streaks on the leaves, which can become necrotic and cause lesions that can spread throughout the leaf blade.
5.	Tungro	A viral disease that can cause stunted growth, yellowing of the leaves, and reduced yield. It is spread by a leafhopper insect and can affect both the leaves and the grains.
6.	False smut	A fungal disease that can cause the development of false smut balls on the rice panicles, which are made up of spores. The leaves can also show symptoms such as pale yellow spots, brownish lesions, and dark brown veins.
7.	Rice tungro disease	A viral disease that is transmitted by green leafhoppers and can cause yellowing and stunting of the plants, as well as necrosis and wilting of the leaves.
8.	Narrow brown leaf spot	A fungal disease that can cause long, narrow brown spots on the leaves, which can eventually merge and cause drying and death of the leaf tissue.
9.	Leaf scald	A bacterial disease that can cause yellowing of the leaves, followed by the development of water-soaked lesions that can turn brown and dry up. The disease can spread rapidly during periods of high humidity and rainfall.
10.	Leaf spot	A fungal disease that can cause circular to oval spots with dark brown borders on the leaves. The spots can merge and cause extensive leaf damage, leading to yield loss.
11.	Rice blast	A fungal disease that can cause leaf spots, leaf blight, and blast lesions on the leaves. The lesions may appear as gray, brown, or black spots with a dark ring around them. In severe cases, the lesions can merge and cause death of the leaf tissue.
12.	Sheath rot	A fungal disease that can cause water-soaked lesions on the leaf sheaths, which can lead to the rotting and shredding of the

		sheath tissue. The disease can spread to the leaves and stems, causing severe damage to the plant.
13.	Bacterial leaf blight	A bacterial disease that can cause yellowing of the leaves, followed by the development of water-soaked lesions that can turn brown and dry up. The disease can spread rapidly during periods of high humidity and rainfall.
14.	Curvularia leaf spot	A fungal disease that can cause irregular brown spots on the leaves, which may have a yellow halo. The spots can merge and cause extensive leaf damage, leading to yield loss.
15.	Leaf scorch	A fungal disease that can cause brown necrotic lesions on the leaves, which can lead to leaf curling and drying. The disease can spread rapidly during periods of high temperature and low humidity.
16.	Rhizoctonia leaf spot	A fungal disease that can cause circular or oblong lesions with dark brown centers and yellow borders on the leaves. The lesions can merge and cause extensive damage to the leaf tissue.
17.	Brown stripe	A bacterial disease that can cause yellowing of the leaves, followed by the development of brown stripes that run along the length of the leaf blade. The disease can cause severe yield loss, particularly in wet conditions.
18.	Rice leaf folder	A pest that can cause the folding and rolling of the leaves, which can result in reduced photosynthesis and yield loss. The pest can also transmit viruses and other pathogens.
19.	Rice leaf mite	A pest that can cause the yellowing and drying of the leaves, which can result in yield loss. The pest can also transmit viruses and other pathogens.
20.	Magnaporth e oryzae	A fungal disease that causes blast symptoms on the leaves. It can also affect other parts of the plant, including the stem and grain, and can cause significant yield loss.

The OCL constraints are written for the paddy leaf diseases are tabulated in table 3. Here, for the give twenty diseases OCL expressions and Description is given. Table 4. represents the leaf identification constraints which includes the expression of the artefact, pre-condition, post-condition and its description of all the leaf identification constraints.

Table 3. Leaf Disease Constraints

S.No.	Leaf Disease Constraints	OCL Expressions	Description
1.	Blast	self.disease = 'Blast' implies self.spots->exists(spot | spot.color = 'gray' or spot.color = 'white') and self.spots->exists(spot | spot.border = 'dark')	Leaf must have at least one spot that is either gray or white in color, and at least one spot with a dark border.
2.	Brown spot	self.disease = 'Brown spot' implies self.spots->exists(spot | spot.color = 'brown' and spot.border = 'yellowish-brown')	Leaf must have at least one spot that is brown in color with a yellowish-brown border
3.	Sheath blight	self.disease = 'Sheath blight' implies self.sheaths-> exists(sheath | sheath.color = 'brown' and sheath.lesions->size() >=	Leaf must have at least one brown sheath with three or

		3)	more lesions.
4.	Bacterial leaf streak	self.disease = 'Bacterial leaf streak' implies self.streaks->exists(streak | streak.color = 'yellowish-green' and streak.width < 2)	Leaf must have at least one yellowish-green streak on the leaf with a width less than 2.
5.	Tungro	self.disease = 'Tungro' implies self.veins->exists(vein | vein.color = 'yellow' and vein.spots->size() >= 2) and self.spots->exists(spot | spot.color = 'greenish-yellow')	Leaf must have at least one yellow vein with two or more spots and at least one greenish-yellow spot
6.	False smut	self.disease = 'False smut' implies self.spots->exists(spot | spot.color = 'brown' or spot.color = 'gray') and self.spots->exists(spot | spot.shape = 'globular')	Leaf must have at least one spot that is either brown or gray in color, and at least one spot that is globular in shape.
7.	Rice tungro disease	self.disease = 'Rice tungro disease' implies self.spots->exists(spot | spot.color = 'red' and spot.size > 5) and self.spots->exists(spot | spot.color = 'yellow' and spot.size > 5)	Leaf must have at least one red spot and one yellow spot, both of which are larger than 5 units.
8.	Narrow brown leaf spot	self.disease = 'Narrow brown leaf spot' implies self.spots->exists(spot | spot.color = 'brown' and spot.width < 2 and spot.length > 5)	Leaf must have at least one brown spot with a width less than 2 units and a length greater than 5 units.
9.	Leaf scald	self.disease = 'Leaf scald' implies self.spots->exists(spot | spot.color = 'yellow' and spot.border = 'red') and self.spots->exists(spot | spot.color = 'brown' and spot.border = 'light-brown')	Leaf must must have at least one yellow spot with a red border and at least one brown spot with a light-brown border.
10.	Leaf spot	self.disease = 'Leaf spot' implies self.spots->exists(spot | spot.color = 'brown' or spot.color = 'gray') and self.spots->exists(spot | spot.border = 'dark')	Leaf must have at least one spot that is either brown or gray in color and at least one spot with a dark border.
11.	Rice blast	self.severity >= 0 and self.severity <= 100 and self.infected implies self.hasVisibleSymptoms = true and self.affectedArea > 0 and self.reported = true and (self.severity > 50 implies self.preventiveMeasuresImplemented = true	Leaf conditions related to the severity, infection status, visible symptoms, affected area, reporting, and implementation of preventive measures for rice blast

<table>
<tr><td>12.</td><td>Sheath rot</td><td>self.disease = 'Sheath rot' implies self.sheaths->exists(sheath | sheath.color = 'brown' and sheath.lesions->size() >= 5)</td><td>Leaf must have at least one brown sheath with five or more lesions.</td></tr>
<tr><td>13.</td><td>Bacterial leaf blight</td><td>self.disease = 'Bacterial leaf blight' implies self.spots->exists(spot | spot.color = 'yellow' and spot.border = 'brown') and self.spots->exists(spot | spot.color = 'brown' and spot.border = 'yellow')</td><td>Leaf must have at least one yellow spot with a brown border and at least one brown spot with a yellow border.</td></tr>
<tr><td>14.</td><td>Curvularia leaf spot</td><td>self.disease = 'Curvularia leaf spot' implies self.spots->exists(spot |spot.color = 'brown' and spot.shape = 'irregular' and spot.length > 2 and spot.width > 2) andself.spots->exists(spot | spot.color = 'gray' and spot.shape = 'circular' and spot.length > 1)</td><td>Leaf must have at least one brown spot with irregular shape and size larger than 2 units in length and width, and at least one gray circular spot with length greater than 1 unit.</td></tr>
<tr><td>15.</td><td>Leaf scorch</td><td>elf.disease = 'Leaf scorch' implies self.spots->exists(spot | spot.color = 'brown' and spot.border = 'dark') and self.spots->exists(spot | spot.color = 'gray' and spot.shape = 'oval' and spot.width > 2)</td><td>Leaf must have at least one brown spot with a dark border and at least one gray oval-shaped spot with width greater than 2 units.</td></tr>
<tr><td>16.</td><td>Rhizoctonia leaf spot</td><td>self.disease = 'Rhizoctonia leaf spot' implies self.spots->exists(spot | spot.color = 'brown' and spot.border = 'light-brown' and spot.shape = 'irregular' and spot.width > 2) and self.spots->exists(spot | spot.color = 'gray' and spot.shape = 'circular' and spot.length > 2)</td><td>Leaf must have at least one brown spot with a light-brown border, irregular shape, and width greater than 2 units, and at least one gray circular spot with length greater than 2 units.</td></tr>
<tr><td>17.</td><td>Brown stripe</td><td>self.disease = 'Brown stripe' implies self.spots->exists(spot | spot.color = 'brown' and spot.border = 'dark' and spot.shape = 'striped') and self.spots->exists(spot | spot.color = 'gray' and spot.shape = 'circular' and spot.width > 2)</td><td>Leaf must have at least one brown striped spot with a dark border and at least one gray circular spot with width greater than 2 units.</td></tr>
<tr><td>18.</td><td>Rice leaf folder</td><td>self.disease = 'Rice leaf folder' implies self.insects->exists(insect | insect.type = 'leaf-folder' and insect.color = 'green' and insect.length > 5)</td><td>Leaf must have at least one green leaf-folder insect with length greater than 5 units.</td></tr>
<tr><td>19.</td><td>Rice leaf</td><td>self.disease = 'Rice leaf mite' implies</td><td>Leaf must have at</td></tr>
</table>

	mite	self.insects->exists(insect | insect.type = 'leaf-mite' and insect.color = 'yellow' and insect.length > 2)	least one yellow leaf-mite insect with length greater than 2 units.
20.	Magnaporthe oryzae	self.disease = 'Magnaporthe oryzae' implies self.spots->exists(spot | spot.color = 'brown' and spot.border = 'dark'	Leaf must have at least one brown spot with a dark border.

Table 4. Leaf Identification Constraints

S. No.	**OCL Constraint**	**OCL Expression**		**Description**
		Context: *LDIM::ClassifyLeafImg (image: Image)*		
		Pre-Condition	**Post-Condition**	
1	Healthy leaf images identification	image.isHealthy = true	self.classifyLeaf(image) = 'Healthy'	The input image must have the "isHealthy" attribute set to true, and the system's "classifyLeaf" function should return 'Healthy' after processing the input image.
2	Diseased leaf image identification	image.isHealthy = false	self.classifyLeaf(image) = image.diseaseType	The input image must have the "isHealthy" attribute set to false, and the system's "classifyLeaf" function should return the disease type of the input image after processing.
3	Invalid input Handling	image = null	self.classifyLeaf(image) = null	The input image must be null, and the system's "classifyLeaf" function should also return null after processing the input image
4	Invalid image data	image.data = null	self.classifyLeaf(image) = null	The input image data must be null, and the system's "classifyLeaf" function should also return null after processing the

				input image.
5	Incorrect dimensions of image data	image.width < > 224 or image.height < > 224	self.classifyLeaf(image) = null	The input image width and height must not be equal to 224, and the system's "classifyLeaf" function should also return null after processing the input image.
6	images with invalid file formats	image.fileFormat < > 'jpg' and image.fileFormat < > 'jpeg' and image.fileFormat < > 'png'	self.classifyLeaf(image) = null	The input image file format must not be equal to any of the supported formats, and the system's "classifyLeaf" function should also return null after processing the input image.
7	Images with corrupted data	image.data = corruptedData	self.classifyLeaf(image) = null	The input image data must be equal to a variable "corruptedData", and the system's "classifyLeaf" function should also return null after processing the input image.
8	Images with invalid labels	image.isHealthy = true and image.diseaseType < > null or image.isHealthy = false and image.diseaseType = null	self.classifyLeaf(image) = null	The input image must have an inconsistent label, and the system's "classifyLeaf" function should also return null after processing the input image.
9	Multiple concurrent requests	self.numberOfRequests < self.maxConcurrentRequests	self.classifyLeaf(image) < > null and self.numberOfRequests = self.numberOfRequests@pre + 1	The input image should be processed correctly and the number of requests should increase by 1 after processing.
10	Exceeding the maximum concurrent	self.numberOfRequests = self.maxConcurre	self.classifyLeaf(image) = null and self.numberOfReques	The number of current requests is equal to the

	requests	ntRequests	ts = self.numberOfRequests@pre	maximum allowed concurrent requests.
11	Images with invalid dimensions	image.width < 100 or image.height < 100	self.classifyLeaf(image) = null	The input image must have invalid dimensions, and the system's "classifyLeaf" function should also return null after processing the input image.
12	Images with large dimensions	image.width > 5000 or image.height > 5000	self.classifyLeaf(image) = null	The input image must have large dimensions, and the system's "classifyLeaf" function should also return null after processing the input image.
13	Images with invalid data type	image.data.getType() < > ImageDataType.RGB	self.classifyLeaf(image) = null	The input image data type must not be RGB, and the system's "classifyLeaf" function should also return null after processing the input image.
14	Images with very low contrast	image.contrast < 0.5	self.classifyLeaf(image) = null	The input image must have very low contrast, and the system's "classifyLeaf" function should also return null after processing the input image
15	Images with very high contrast	image.contrast > 10.0	self.classifyLeaf(image) = null	The input image must have very high contrast, and the system's "classifyLeaf" function should also return null after processing the input image.
16	Grayscale images	image.data.getType() = ImageDataType.	self.classifyLeaf(image) = null	The input image data type must be grayscale, and the

 https://doi.org/10.1051/e3sconf/202343001072

		GRAYSCALE		system's "classifyLeaf" function should also return null after processing the input image.
17	Images with low resolution	image.width < 500 or image.height < 500	self.classifyLeaf(image) = null	The input image must have low resolution, and the system's "classifyLeaf" function should also return null after processing the input image.
18	Images with high resolution	image.width > 4000 or image.height > 4000	self.classifyLeaf(image) = null	The input image must have high resolution, and the system's "classifyLeaf" function should also return null after processing the input image.
19	Images with low quality	image.quality < 0.5	self.classifyLeaf(image) = null	The input image must have low quality, and the system's "classifyLeaf" function should also return null after processing the input image.
20	Images with high quality	image.quality > 0.9	self.classifyLeaf(image) < > null	the input image must have high quality, and the system's "classifyLeaf" function should return a non-null value after processing the input image.

4. Results and Discussion

The OCL constraints for paddy leaf detection and paddy leaf disease detection are written from the requirements of the system. And from the constraint tables (i.e. Table 3 and Table

4) the required model has been prepared and executed. Based on the trained data set the model is identifying the disease leaf and healthy leaf.

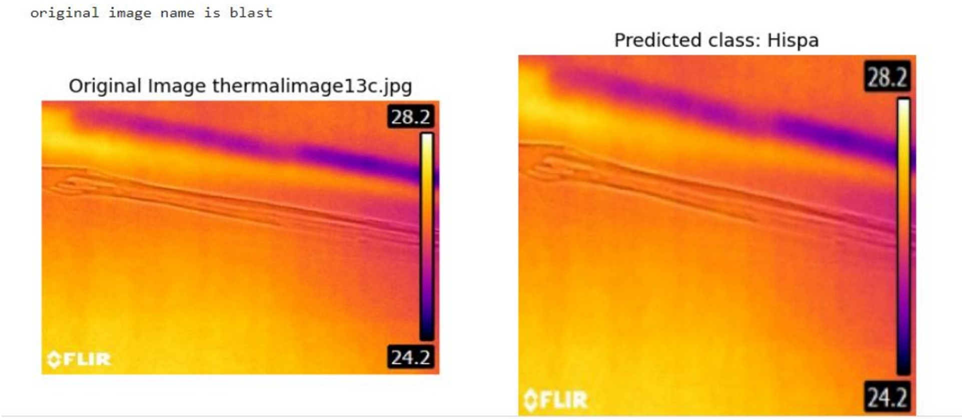

Figure 1. Image predicted as Hispa Disease

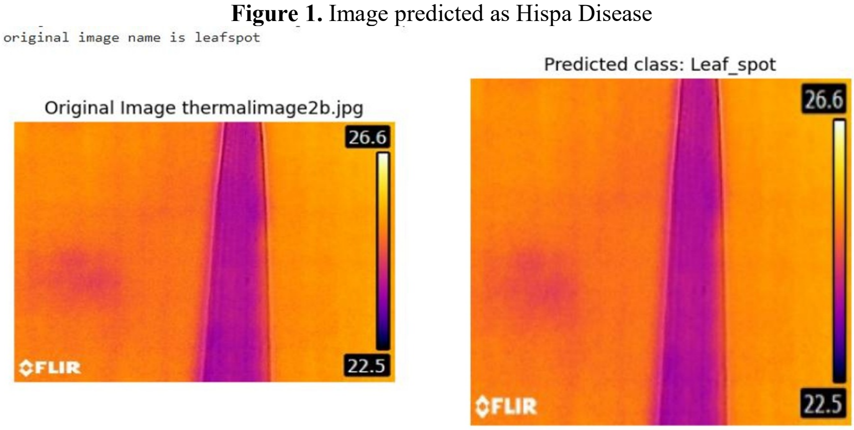

Figure 2. Image predicted as Leaf-Spot Disease

The Fig.1 shows the one sample result identifying Hispa disease and Fig. 2 shows as identifying Leaf-Spot Disease. The Fig. 3 show, the proposed model identifying healthy leaf, and Fig. 4 shows the comparison of the accuracy with existing model.

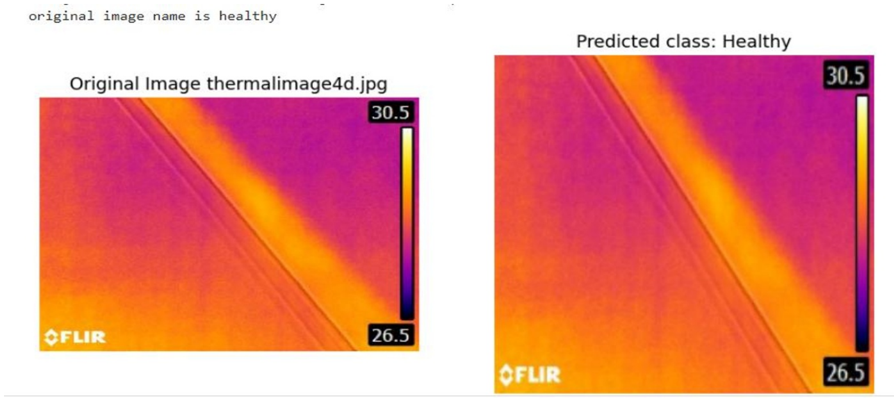

Figure 3. Image predicted as Healthy

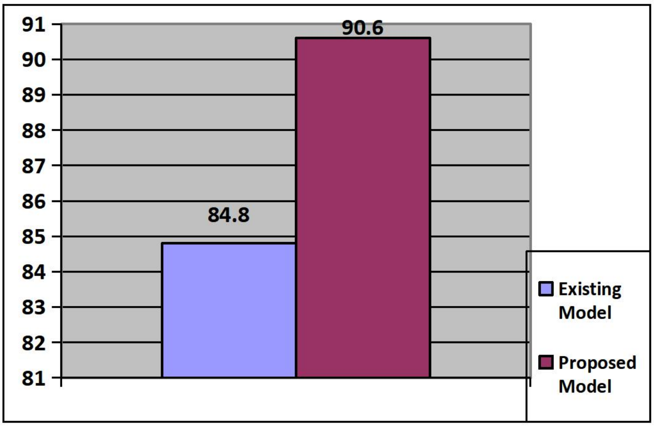

Figure 4. Accuracy between existing and proposed models.

The proposed model produced satisfactory results when compared with existing model.

5. Conclusion and Future Directions

This research introduced OCL based ML model preparation approach in cost effective way. Based on OCL there were 40 constraints are generated for choosing a problem and prepared a model. This model gave satisfactory results the early approaches. This approach leads to accurate classification of training data, testing data and validation data. It maximizes the test coverage, accurate prediction.

In the feature the OCL can be promoted to another level of models like Deep Learning (DL) system. And specific applications like vehicular Internet of Thing (IoT), Health care IoT, Agriculture IoT, where there is a great need of ML models.

References

1. Davis, James P., and Ronald D. Bonnell. IEEE Transactions on Knowledge and Data Engineering **19**(3), 427-440, (2007).
2. Chiu, Yi-Chuan, Hsing-Hung Lin, and Yung-Tsan Jou, *A Model Selection Method for Machine Learning by Differential Evolution.* Proceedings of the 4th International Conference on Big Data and Computing, (2019).
3. Mayer, David G., B. P. Kinghorn, and Ainsley A. Archer, Agricultural Systems **83**(3), 315-328, (2005).
4. Ram, Pintu Kumar, and Pratyay Kuila, Multimedia Tools and Applications, 82(9), 13453-13478, (2023).

5. Ram, Pintu Kumar, and Pratyay Kuila., Journal of Information and Optimization Sciences, 40(8), 1599-1610, (2019).
6. Taşkın, Gülşen, Hüseyin Kaya, and Lorenzo Bruzzone, IEEE Transactions on Image Processing, 26.6, 2918-2928, (2017).
7. Kumar, JNVR Swarup, et al. Secured Cloud Application for Detection of Brain Tumor using Deep Learning Algorithms. 2022 4th International Conference on Inventive Research in Computing Applications (ICIRCA). IEEE, (2022), (2022).
8. Sasank, V. V. S., et al. Executing CNN-LSTM Algorithm for Recognizable Proof of Cervical Spondylosis Infection on Spinal Cord MRI Image: Machine Learning Image. Handbook of Research on Innovations and Applications of AI, IoT, and Cognitive Technologies. IGI Global, 468-484, (2021) (2021).
9. Kranthi kumar Singamaneni, Journal of Green Engineering , (2020).
10. Ch. Mallikarjuna Rao, G. Ramesh, D. V. Lalitha Parameswari Karanam Madhavi, K. Sudheer Babu, International Journal of Recent Technology and Engineering (IJRTE), 8(1),(2019).
11. Dhanke Jyoti Atul, R. Kamalraj, G. Ramesh, K. Sakthidasan Sankaran, Sudhir Sharma, Syed Khasim Microprocessors and Microsystems, 82, 103741, (2021). https://doi.org/10.1016/j.micpro.2020.103741.
12. Kumar, JNVR Swarup, et al.," 2022 International Conference on Applied Artificial Intelligence and Computing (ICAAIC). IEEE,(2022).
13. S. K. K. B. Thulasi, International Journal of Advanced Science and Technology, 29 , no. 3, (2020) .
14. Ram, Pintu Kumar, and Pratyay Kuila, The Journal of Supercomputing, 79(1), 541-572, (2023)
15. P. K. and. P. K. Ram, "FCPSO: Evaluation of Feature Clustering Using Particle Swarm Optimization for Health Data," Intelligent Data Engineering and Analytics: Proceedings of the 10th International Conference on Frontiers in Intelligent Comp, (2023).
16. Dr. Gajula Ramesh, Dr. D. William Albert, Dr. Gandikota Ramu, International Journal of Advanced Science and Technology, 29(8), 1656 – 1664, (2020).
17. Chandrika Lingala, and Karanam Madhavi et.al, "A Survey on Cardivascular Prediction using Variant Machine learning Solutions. E3S Web of Conferences 309, 01042, ICMED 2021, (2021).
18. B.Sankara babu, International Journal of Advanced Science and Technology(IJAST), 29 (1), (2020).
19. B.Sankara Babu " Medical Disease Prediction using Grey Wolf optimization and Auto Encoder based Recurrent Neural Network", Periodicals of Engineering and Natural Sciences, 6(1), 229-240, (2018).
20. Chandrika Lingala, and Karanam Madhavi, A Hybrid Framework for Heart Disease Prediction Using Machine Learning Algorithms ", E3S Web of Conferences 309, 01043, ICMED 2021, (2021).
21. Bolognesi, Tommaso, IEEE Transactions on Software Engineering, 26(7) ,594-616 ,(2000).