International Journal of Engineering Applied Sciences and Technology, 2024
Vol. 8, Issue 11, ISSN No. 2455-2143, Pages 153-160
Published Online March 2024 in IJEAST (http://www.ijeast.com)

INTELLIGENT MEDICATION CLASSIFIER: HARNESSING MACHINE LEARNING FOR PHARMACY MEDICATION SORTING

Sunitha P, Margaret R E
Department of CS & E
Malnad College of Engineering, Hassan
Karnataka, India.

Anragya Y S, Anaga Dixith, Anu Mohan, Amulya D J
Students, Department of CSE
Malnad College of Engineering, Hassan
Karnataka, India

Abstract—**The conventional approach to medication sorting and organization in healthcare settings is being revolutionized by the automated tablet segregation system. This innovative solution focuses on developing an automated system that utilizes a machine learning algorithm to accurately and efficiently segregate tablets based on the first letter of their name. The system employs advanced machine learning algorithms to precisely identify and categorize different types of tablets and an audio module to announce the respective rack number for each tablet. Healthcare professionals can easily sort tablets into designated compartments or racks through a user-friendly interface. The designed Automated Tablet Segregation System (ATSS) is scalable and adaptable across various healthcare environments, including hospitals, clinics, and pharmacies. Its seamless integration into existing medication management systems provides a versatile solution tailored to meet the specific needs of different healthcare facilities. The ATSS undergoes training using a carefully curated dataset, which is divided into 70% for training and 30% for testing. The system's performance is evaluated based on key parameters such as accuracy, loss, and precision. Notably, the implemented model attains an impressive accuracy of 98%.**

The training of the dataset is conducted in Teachable Machine, a user-friendly web-based tool developed by Google's Creative Lab. This tool allows individuals, even those without extensive coding experience, to create machine learning models.

Keywords —**Automated Tablet Segregation System, Machine Learning, Healthcare, Teachable Machine, Audio Module**

I. INTRODUCTION

Pharmacies have become an essential component of our healthcare system, offering a wide range of medications and medical supplies to patients. Tablet segregation is the practice of categorizing and organizing tablets in pharmacies based on factors such as medication type, strength, dosage form, and manufacturer. This process is crucial for efficient inventory management, error reduction, and ensuring patient safety. Traditionally, tablet segregation has been a manual and time-consuming task, which can result in human errors and inefficiencies. Managing and organizing these products can also be overwhelming. This is where Machine Learning (ML) comes into play, providing a promising solution to the challenge of tablet segregation in pharmacies. ML algorithms have the ability to analyse large amounts of data and learn patterns from it. In the context of tablet segregation, ML can be trained on historical and real-time data from pharmacies to identify these patterns and accurately predict how tablets should be segregated.

The designed model integrates a GUI user interface, a machine learning model and an audio system. Machine learning model is designed using Teachable machine. Teachable Machine is a user-friendly and accessible tool developed by Google's Creative Lab. The images are trained in the Image section of the web interface, where users can teach the machine to recognize different objects by providing examples and labelling them. Once trained, Teachable Machine generates a machine-learning model that can recognize these patterns. This model can then be applied in various applications, such as creating interactive experiences, controlling devices, or analysing specific parameters.

While Machine Learning cannot replace the expertise and knowledge of pharmacists, it can complement their work by providing valuable insights and predictions. As technology continues to advance, Machine Learning should be embraced as a powerful tool to revolutionize the field.

II. LITERATURE SURVEY

Advancements in machine learning and computer vision are boosting interest in automating textual information extraction and classification from images, especially in tablet image analysis. The integration of audio modules into ML systems is also gaining popularity. The main objective of this literature review is to examine the current body of research on extracting and categorizing text from tablet images, with a specific emphasis on machine learning approaches. The aim is to gain insights into the current methodologies, recognize the obstacles faced, and propose new and inventive solutions. The research indicates that by integrating machine learning techniques based on images with audio modules, new and groundbreaking applications can be realized.

The main focus of this research is to develop a system that can accurately recognize text from natural images, particularly handwritten or printed bills. M. Geetha and colleagues [1] propose deep learning techniques using the EAST algorithm to analyse letters and words, and Open CV with RNN to automatically recognize and update the text in the database. The text extraction process involves scanning an image, detecting text using EAST and RNN algorithms, and comparing it with a trained dataset using TensorFlow.

Likewise, Sunil Kumar Dasari et. al [2] developed an optimal architecture Fusion Neural Network (FNN) for text identification and recognition, combining convolutional and recurrent neural networks. The FNN architecture extracts features and predicts feature classification. The proposed model achieves 98.67 percent script identification accuracy, 84.65 percent word recognition rate, and 92.93 percent character recognition rate. This approach enhances classification accuracy and improves data mining from street view images.

Afgani Fajar Rizky and their research team [3] explored the use of CNN-based neural networks for image classification and text recognition. The Chars74K dataset was utilized to train the model, while the IIIT-5K-Dataset was used for testing. The highest performing model achieved an accuracy of 97.94 percent for validation data, 98.16 percent for test data, and 95.62 percent for IIIT5K-Dataset samples. The findings of the study has suggested that pre-trained CNNs offer precise text recognition, thereby serving a valuable resource for upcoming text detection systems.

In their research, Enze Xie and collaborators [4] proposed a Supervised Pyramid Context Network (SPCNET) for scene text detection, based on Feature Pyramid Network (FPN) and instance segmentation. Drawing inspiration from Mask R-CNN, SPCNET effectively identifies text regions while minimizing false positives. By incorporating semantic information guidance and sharing FPN, SPCNET surpasses existing methods, achieving impressive F-measures of 92.1 percent on ICDAR2013, 87.2 percent on ICDAR2015, 74.1 percent on ICDAR2017 MLT, and 82.9 percent on Total-Text.

Likewise, in their study R Deepa and colleagues [5] summarized the text extraction from images using Deep Vision techniques. In this process, an image is classified utilizing a Convolutional Neural Network (CNN), while the text extraction is accomplished through Tesseract’s LSTM-based recognition engine. CNN has shown superior performance when applied to large datasets, and its effectiveness can be further enhanced by incorporating a substantial dataset and increasing the number of epochs.

Twana Mustafa and colleagues [6] explored the use of deep learning techniques like OpenCV, YOLO, PaddleOCR, and Tesseract OCR in Python programming to develop Automatic Number Plate Recognition (ANPR) systems. The study evaluates the effectiveness of these techniques in various environmental conditions and presents experiments. The findings have significant implications for future development of efficient and accurate ANPR systems.

Weina Zhou and their research team [7] presented an Adaptive Double Pyramid Network (ADPNet) for real-time detection of arbitrary-shaped text. It uses a Double Feature Enhancement Pyramid with Packet Downsampling Units to enhance feature maps. Validating on three benchmark datasets, ADPNet achieves state-of-the-art performance in speed and accuracy, with an F-measure of 85.7 percent.

Sagar Janokar et. al [8] propose a model that uses Automatic Speech Recognition (ASR) to recognize user’s speech and convert it into text format. For converting text-to-speech, google’s text-to-speech (gTTS) engine and Microsoft’s SAPI5 are being used which provide voice to the model. This voice assistant can not only be used to read a Word document or a PDF file but also search content on Google or Wikipedia.

Baek et. al. [9] have proposed a new scene text detection method using neural networks to effectively detect text areas by exploring each character and their affinity. The method uses both synthetic and real character-level annotations, and uses a new affinity representation. Experiments on six benchmarks show the method outperforms state-of-the-art detectors, ensuring high flexibility in detecting complex scene text images.

Likewise, Zobeir and colleagues [10] have made survey reviews on recent advancements in scene text detection and recognition, focusing on challenges such as in-plane-rotation, multi-oriented and multi-resolution text, perspective distortion, illumination reflection, partial occlusion, complex fonts, and special characters. Current methods have shown superior accuracy on benchmark datasets, but still face challenges in generalizing to unseen data and insufficient labeled data. The paper also presents insights into potential research directions to address these challenges in scene text detection and recognition techniques.

Authors [11] have proposed a Focusing Attention Network (FAN) method for scene text recognition in computer vision. The FAN uses a focusing attention mechanism to draw back drifted attention, recognizing character targets and adjusting attention based on the attention network's performance. The method uses a ResNet-based network to enrich deep representations of scene text images, outperforming existing methods on benchmarks like the IIIT5k, SVT, and ICDAR datasets.

The paper [12] presents an arbitrary orientation network (AON) for recognizing text from natural images, addressing the challenge of irregular arrangements in scene texts. The network captures deep features of irregular texts and uses an attention-based decoder to generate character sequences. Experiments on various datasets show the AON-based method achieves state-of-the-art performance in irregular datasets and is comparable to existing methods.

The literature survey emphasizes the importance of text extraction and classification of images using machine learning techniques. Machine Learning approaches like Convolutional Neural Networks (CNN) and Natural Language Processing (NLP) have been explored for image analysis and integration of audio module respectively.

The work in this paper is divided in two stages.

- Text- Detection
- Inpainting

Text detection is done by applying morphological open- close and close-open filters and combines the images. There- after, gradient is applied to detect the edges followed by thresholding and morphological dilation, erosion operation. Then, connected component labelling is performed to label each object separately. Finally, the set of selection criteria is applied to filter out non text regions. After text detection, text inpainting is accomplished by using exemplar based Inpainting algorithm. Inpainting refers to the process of reconstructing missing or corrupted parts of an image. It’s like digital ”filling in the blanks. Inpainting algorithms analyze the surrounding pixels or features in an image to predict and generate plausible values for the missing or damaged areas.

III. DATASET

A. Dataset Source

The dataset utilized for this project is the tablet image dataset. The tablet images were gathered from various sources. A portion of the tablet images were sourced from Google, with the remaining images captured using a mobile camera during manual visits to pharmacies. Tablets from all categories were photographed at varying resolutions. The captured images have resolutions of 9248*6936 pixels, 1080*2400 pixels, and 64MP resolution. The tablet images, containing tablet names starting with each of the 26 alphabets, were downloaded from Google Images by specifying the tablet names. In total, 2600 images were collected, with each class containing 100 images. For instance, if the first class is “A,” then all 100 images within this class correspond to tablet names starting with the letter “A.” The same methodology was followed for all other classes, from “A” to “Z.”

B. Dataset Preprocessing steps

Images of the curated dataset needs to be prepossessed to make it ready for the designed Machine Learning model. Preprocessing undergoes three phases namely

- Removal of the background for all the tablet images
- Cropping of images
- Splitting the dataset into training and testing dataset

Removal of the background for all the tablet images: Removing the background while training an image dataset can be beneficial for the following reasons:

- Noise reduction: It is an essential step in model training as backgrounds often consist of irrelevant information or noise. By eliminating the background, the noise is reduced, allowing the model to concentrate on the pertinent features required for the given task.
- Focus on Object Recognition: The elimination of the background allows the model to concentrate exclusively on the object to be analyzed. This feature proves especially valuable in tasks such as object detection and recognition, as it enables accurate identification and categorization of objects present in an image.
- Reduced Complexity: Backgrounds have the potential to add unnecessary complexity and variability to the learning task. By eliminating backgrounds, the input data is streamlined, enabling the model to focus on learning the fundamental characteristics of the objects more effectively.

Cropping of images:

Images are cropped removing all the unnecessary information and leaving behind just the name of the tablet. The reasons for cropping the image are:

- Mitigation of Occlusions: Cropping is an effective technique to reduce the influence of occlusions or other objects in the image that could disrupt text recognition. By isolating the specific text area, the model’s accuracy is enhanced as it is less prone to confusion caused by surrounding elements.
- Focus on Region of Interest (ROI): Cropping enables the extraction and concentration of the region of interest (ROI) that encompasses the text, thereby minimizing the presence of irrelevant information in the image. This facilitates the accurate identification and processing of the text-by-text recognition models.
- Improved Model Performance: Cropping plays a crucial role in enhancing the performance of text recognition models by providing a more concentrated and relevant dataset for training. This ultimately leads to improved model performance as the model can effectively learn from a more focused set of examples.

Splitting the dataset into training and testing dataset: The dataset is organized into two distinct folders, namely the train dataset and the test dataset. The train folder encompasses approximately 70% of the tablet images available in the dataset. This subset of tablet images is specifically designated for training purposes. It is utilized to develop and refine machine learning model. The train dataset plays a crucial role in enabling the model to learn patterns, features, and relationships within the tablet images, ultimately enhancing its ability to make accurate predictions or classifications. On the other hand, the test folder comprises the remaining 30% of tablet images. This subset of tablet images is reserved for testing the performance and generalization capabilities of the trained model. By evaluating the model's predictions on the test dataset, it is possible to assess its effectiveness in handling new, unseen tablet images. The test dataset serves as a benchmark to measure the model's accuracy, precision and recall.

There are 26 sub-folders inside the train folder, each containing tablet images. These sub-folders are named as A, B, C,...Z. The tablet images with names starting from A to Z are stored in their respective sub-folders. For instance, sub-folder A contains all the tablet images with the initial letter "A". Similarly, the remaining tablet images are stored in their respective sub-folders. Each sub-folder contains a total of 70 images. The Figures 1-4 below depict the images contained within the different subfolders located in the train folder.

Similarly, the test folder contains images of tablets that are not used during the training phase but are reserved to evaluate the performance of the trained model. It helps to assess how well the model generalizes to new and unseen data.

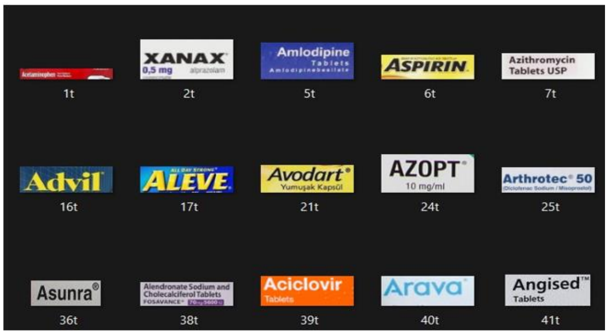


Fig. 1. Images inside folder A

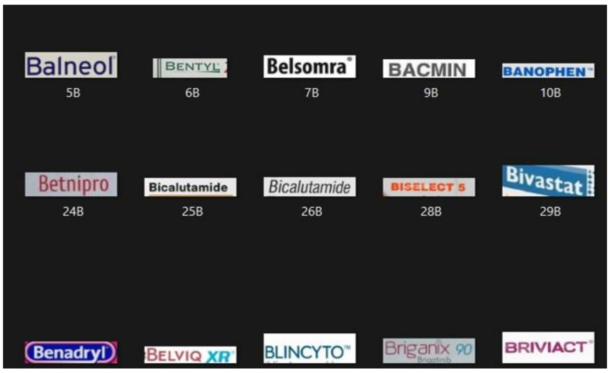


Fig. 2. Images inside folder B

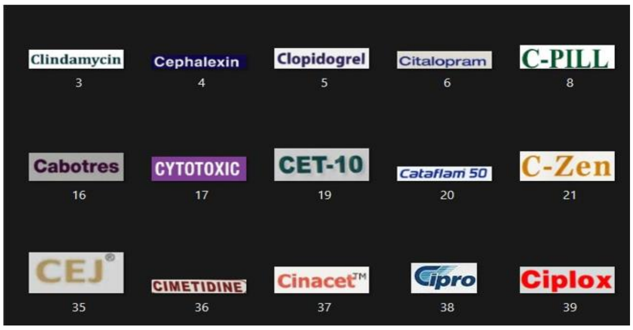


Fig. 3. Images inside folder C

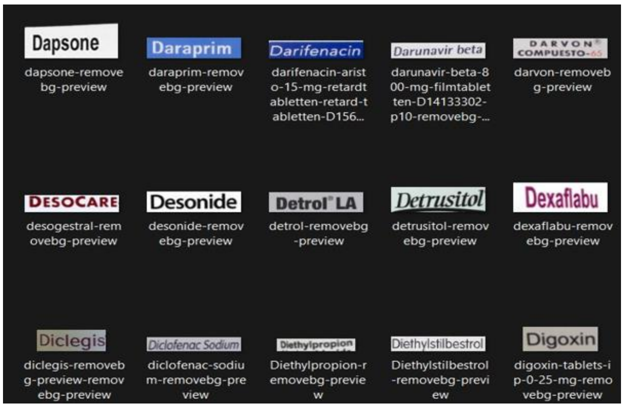

Dapsone
dapsone-removebg-preview
Daraprim
daraprim-removebg-preview
Darifenacin
darifenacin-aristo-15-mg-retardtabletten-retard-tabletten-D156...
Darunavir beta
darunavir-beta-800-mg-filmtabletten-D14133302-p10-removebg-...
DARVON COMPUESTO-65
darvon-removebg-preview
DESOCARE
desogestral-removebg-preview
Desonide
desonide-removebg-preview
Detrol LA
detrol-removebg-preview
Detrusitol
detrusitol-removebg-preview
Dexaflabu
dexaflabu-removebg-preview
Diclegis
diclegis-removebg-preview-removebg-preview
Diclofenac Sodium
diclofenac-sodium-removebg-preview
Diethylpropion
Diethylpropion-removebg-preview
Diethylstilbestrol
Diethylstilbestrol-removebg-preview
Digoxin
digoxin-tablets-ip-0-25-mg-removebg-preview

Fig. 4. Images inside folder D

IV. METHODOLOGY

The flowchart for the Intelligent Medication Classifier represents a systematic and comprehensive approach to harness the power of machine learning for the purpose of efficiently sorting pharmacy medications. This innovative system aims to enhance the accuracy and speed of medication classification, addressing the challenges faced in pharmacy workflows.

The tablets with initial letters as "A","B","C" and "D" are collected as the image dataset. The classes are labelled as the initial letters. The classes are mapped to rack numbers in the following order: A-1,B-2,C-3,D-4. The phases of the model depicted in Fig. 5 are as follows

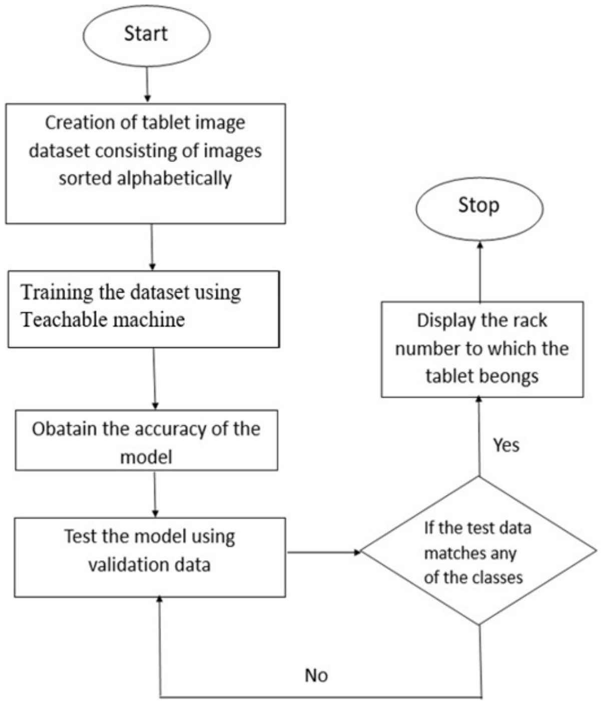

Start
Creation of tablet image dataset consisting of images sorted alphabetically
Training the dataset using Teachable machine
Obatain the accuracy of the model
Test the model using validation data
If the test data matches any of the classes
Yes
No
Display the rack number to which the tablet beongs
Stop

Fig. 5. Process Flowchart

- Initially, the dataset is partitioned into a 70 % Train dataset and 30% Test dataset.
- The dataset is loaded into the predefined classes of the Teachable machine, and the classes are labeled as “A”, “B”, “C”, and “D”.
- Teachable machine trains the model with the training dataset
- Once the model is trained, it is exported in the .h5 file format.
- To visualize the machine learning model, Gradio, a web interface, is used.
- The saved model is taken as input for the Gradio interface, which displays the percentage of the input image that is classified.
- Additionally, a voice module is integrated with Gradio, which converts the given text into speech.
- In the testing phase, an image from the test dataset is provided as input to the Gradio tool.
- If the input image corresponds to a tablet with the initial letter “A”, it will be classified as class A. Similarly, other input images will be classified into their respective classes.
- The voice module will then announce the rack numbers of the recognized classes.
- Finally, the classified label and the confidence of the classification will be displayed in the Gradio interface.

The Fig. 6 below illustrates the GUI input component of the teachable machine.

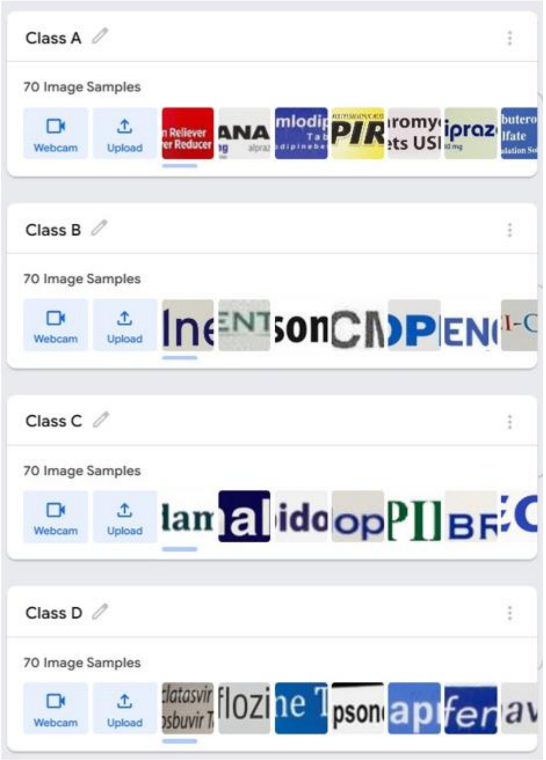
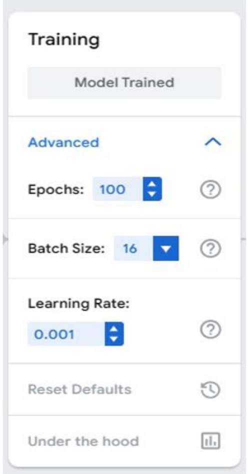


Fig. 6. Images uploaded under different classes

The model underwent experimentation with diverse parameters across a spectrum of values.

- Epochs : 10 to 100
- Batch size : 8 to 128
- Learning rate : 0.0001 to 0.1

Fig. 7 below depicts the diverse parameters that are set and modified within the training controls to train the machine learning model. The performance evaluation of the model is measured using standard metrics namely accuracy, loss, confusion matrix, and precision.


Training

Model Trained

Epochs: 100

Batch Size: 16

Learning Rate: 0.001

Fig. 7. Parameters set to train

V. IMPLEMENTATION

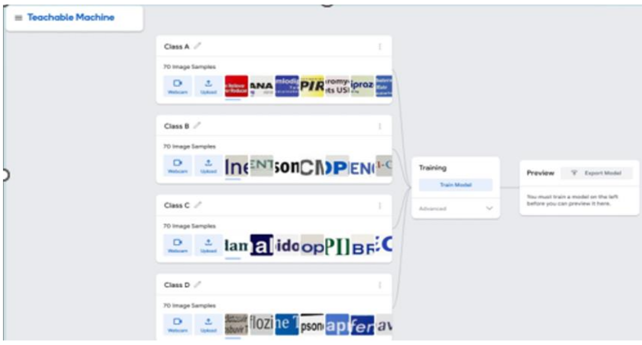
The model was constructed utilizing the web-based tool Teachable Machine, and training was conducted utilizing a curated dataset of tablet images. Additionally, a user interface was developed using the Gradio tool from the Python library. This interface was seamlessly integrated with the model to transform it into an accessible tool for users of all levels of expertise.

A. Teachable Machine

A web-based tool known as Teachable Machine is used to create a machine learning model and train the model using tablet image dataset. The Train datasets of each class are given as input and this is depicted in the Fig 8.

The components of Teachable Machine are listed below:

- Training Interface: The platform provides a user-friendly training interface for uploading images and assigning labels. Utilizing cutting-edge machine learning algorithms, the system analyzes the images to discern patterns and features that differentiate one class from another.
- Training Parameters: The platform allows users to specify crucial training parameters, such as the number of training iterations or epochs, within the range of 100 to 1000. This capability holds significance in machine learning, as it dictates how many times the model undergoes training on the dataset. Each iteration or epoch constitutes a comprehensive pass through the entire dataset, facilitating the model's learning and adjustment of parameters based on the provided examples. The adaptability in configuring training parameters empowers users to experiment and refine their models. Through iterative adjustments of these parameters, an optimal balance is achieved between training duration and model accuracy, ensuring the machine-learning model aligns with specific requirements.
- Export Options: Teachable Machine provides flexibility to export the trained model in a format that is compatible with various deployment scenarios. This trained model can be integrated to a wide range of applications. Whether it is for web applications, mobile apps, specialized hardware, or collaboration purposes, this export functionality empowers users to extend the reach and impact of their trained models. Furthermore, the exported model can be shared with others, allowing for collaboration and knowledge sharing.



You must train a model on the left before you can preview it here.

Fig. 8. Teachable Machine

B. Gradio

Gradio is an open-source Python library that simplifies the process of creating user interfaces for machine learning models. Gradio provides a high-level API for building interactive UIs for machine-learning models.

The main elements of Gradio as depicted in Fig. 9 are as follows:

- User Interface: The primary component of Gradio is the 'Interface' class, which is responsible for creating a user interface for your machine learning model. It defines the input and output components of the interface.
- Input Component: Gradio provides support for an input component that allows users to input data. In this case, an image is used as the input.
- Output Component: Gradio also offers an output component that displays the predictions or results generated by the model.

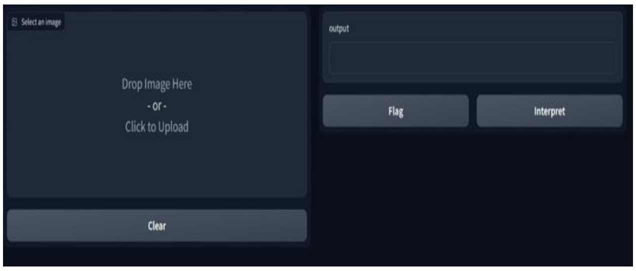


Fig. 9. Gradio Interface

VI. EXPERIMENTAL RESULTS

The model underwent training using a training dataset across epochs ranging from 10 to 100, with batch sizes varying from 8 to 128. Additionally, the learning rate was adjusted within the range of 0.001 to 0.1, and image sizes ranged from 64x64 to 225x225. Subsequently, it was noted that the model achieved optimal performance with an image size of 100x100, 100 epochs, a learning rate of 0.001, and a batch size of 16. The performance analysis of the model primarily focused on evaluating loss, accuracy, and the confusion matrix.

A. The Loss and Accuracy graph

The loss graph serves as a visual tool for assessing the alignment between predicted and target output values, quantifying the neural network's ability to effectively represent the training data. The objective is to minimize the discrepancy between the predicted and target outputs, and this relationship is graphically illustrated in Fig. 10.

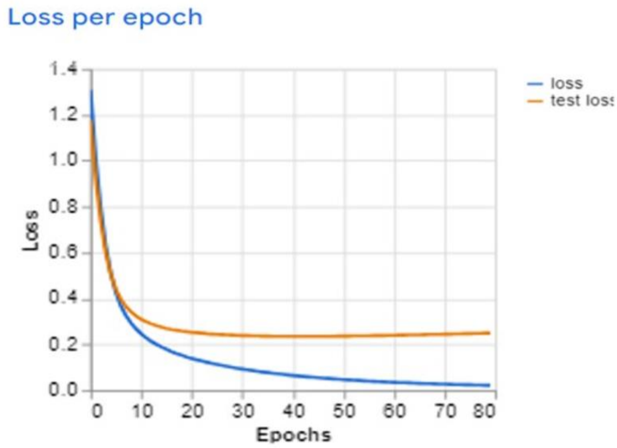


Fig. 10. Loss graph

Similarly, the accuracy chart is employed to illustrate the proportion of accurate classifications achieved by a trained machine learning model. The model is designed to enhance its precision in categorizing various image classes. The accuracy chart showcased in Fig. 11 illustrates the model's attainment of 99% precision.

A. Confusion Matrix

The confusion matrix plays a crucial role in evaluating models and provides insights into the capabilities and limitations of a classification model. It serves as a valuable tool in machine learning and data analysis tasks. The confusion matrix provides a concise summary of prediction outcomes in a classification task. Out of the testing images, 12 were accurately classified as "A", 15 were correctly classified as "B", 14 were correctly classified as "C", and 14 were correctly classified as "D". The Confusion Matrix is illustrated in Fig.12, it clearly demonstrates that the suggested model has attained a praiseworthy degree of precision.

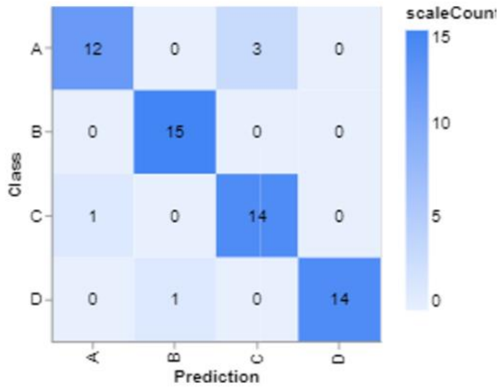
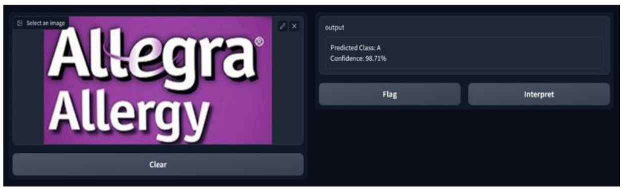


Fig. 12. Confusion Matrix

B. Gradio

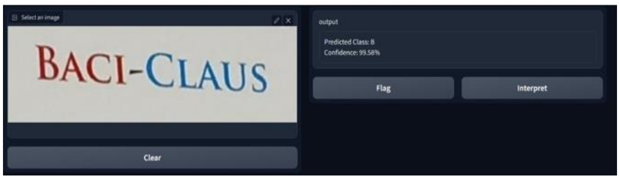
Gradio is utilized to upload an image from the test dataset. It simplifies the process of creating user interfaces (UIs) for machine learning models and enhances the visualization of tablet classification accuracy in their respective racks. Figures 13 through 16 exemplify the categorization of medication in an engaging and interactive format for users.



Predicted Class: A
Confidence: 98.71%

Fig. 13. Gradio output-A

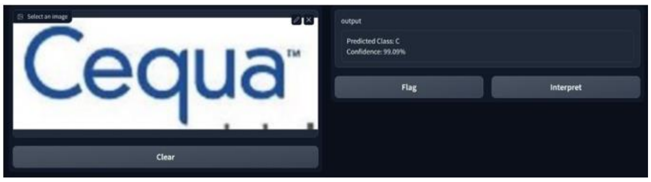
Test-case 1: Tablet starting with 'A' is uploaded. Tablet name: Allegra Allergy
Accuracy: 98.71%
Voice command: Rack 1



Predicted Class: B
Confidence: 99.58%

Fig. 14. Gradio output-B

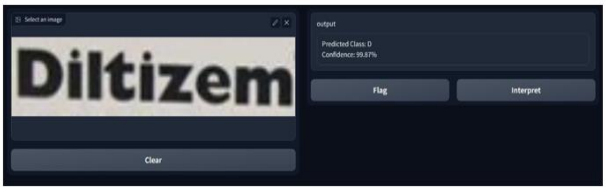
Test-case 2: Tablet starting with 'B' is uploaded.
Tablet name: Balneol
Accuracy: 99.58%
Voice command: Rack 2



Predicted Class: C
Confidence: 99.09%

Fig. 15. Gradio output-C

Test-case 3: Tablet starting with ‘C’ is uploaded.
Tablet name: Colchicine Accuracy: 99.09%
Voice command: Rack 3



Predicted Class: D
Confidence: 99.87%

Fig. 16. Gradio output-D

Test-case 4: Tablet starting with ‘D’ is uploaded.
Tablet name: Descar
Accuracy: 99.87%
Voice command: Rack 4

VII. CONCLUSION

The implementation of a machine learning-based automated tablet segregation system represents a noteworthy progress in pharmaceutical and healthcare procedures. By employing sophisticated image classification models, this system effectively categorizes and organizes tablets, thereby simplifying manual duties. The user-friendly interface of the system, supported by tools such as Gradio, ensures accessibility for individuals with diverse technical expertise. The advantages of this system encompass decreased time spent on manual sorting, heightened operational effectiveness, and the potential to minimize errors in tablet segregation. Moreover, its adapt- ability enables seamless integration into existing workflows, while its interpretability empowers users to comprehend and have confidence in the system’s decision-making processes. The summarized findings are as follows:

- Creating the tablet dataset
- Training the data-set using Teachable machine.
- The use of Gradio tool to visualize the saved model.

The innovative solution exemplifies the practical application of machine learning in pharmaceuticals and sets the stage for future intelligent automation in the healthcare industry.

VIII. REFERENCE

[1] M Geetha, RC Pooja, J Swetha, N Nivedha, and T Daniya. Implementation of text recognition and text extraction on formatted bills using deep learning. Int J Control Automat, 13(2):646–651, 2020.

[2] Sunil Kumar Dasari and Shilpa Mehta. Text detection and recognition using fusion neural network architecture. In 2022 8th International Conference on Advanced Computing and Communication Systems (ICACCS), volume 1, pages 2067–2071. IEEE, 2022.

[3] Afgani Fajar Rizky, Novanto Yudistira, and Edy Santoso. Text recognition on images using pre-trained cnn. arXiv preprint arXiv:2302.05105, 2023.

[4] Enze Xie, Yuhang Zang, Shuai Shao, Gang Yu, Cong Yao, and Guangyao Li. Scene text detection with supervised pyramid context network. In Proceedings of the AAAI conference on artificial intelligence, volume 33, pages 9038–9045, 2019.

[5] R Deepa and Kiran N Lalwani. Image classification and text extraction using machine learning. In 2019 3rd International conference on Electronics, Communication and Aerospace Technology (ICECA), pages 680–684. IEEE, 2019.

[6] Twana Mustafa and Murat Karabatak. Deep learning model for automatic number/license plate detection and recognition system in campus gates. In 2023 11th International Symposium on Digital Forensics and Security (ISDFS), pages 1–5. IEEE, 2023.

[7] Weina Zhou and Wanyu Song. Real-time accurate text detection with adaptive double pyramid network. Neural Processing Letters, 55(4):5055– 5067, 2023.

[8] Sagar Janokar, Soham Ratnaparkhi, Manas Rathi, and Alkesh Rathod. Text-to-speech and speech-to-text converter—voice assistant. In Inven- tive Systems and Control: Proceedings of ICISC 2023, pages 653–664. Springer, 2023.

[9] Youngmin Baek, Bado Lee, Dongyoon Han, Sangdoo Yun, Hwalsuk Lee ,” Character Region Awareness for Text Detection”, Computer Vision and Pattern Recognition , 2019.

[10] Zobeir Raisi, Mohamed A. Naiel, Paul Fieguth, Steven Wardell, John Zelek, “Text Detection and Recognition in the Wild: A Review”, Computer Vision, arXiV,2020.

[11] Zhanzhan Cheng, Fan Bai, Yunlu Xu, Gang Zheng, Shiliang Pu, Shuigeng Zhou, “Focusing Attention: Towards Accurate Text Recognition in Natural Images”, IEEE International Conference on Computer Vision (ICCV), 2017.

[12] Zhanzhan Cheng, Yangliu Xu, Fan Bai, Yi Niu, Shiliang Pu, Shuigeng Zhou, “AON: Towards Arbitrarily-Oriented Text Recognition”, CVF, 2018.