# SAVEETHA SCHOOL OF ENGINEERING
# SAVEETHA INSTITUTE OF MEDICAL AND TECHNICAL SCIENCES

## CAPSTONE PROJECT REPORT

**PROJECT TITLE**

## HANDWRITTEN EQUATION SOLVER USING OPEN CV

**DSA-0213-Computer Vision with Open CV for Data Fusion**

Submitted
by

**PARTHIBAN R (192224275)**
**SAIDEEPAK P.V(192124030)**
**SALMAN S (192124042)**

Guided by
Dr. Sarala. V
Associate Professor
Department of Computer Science and Engineering

## Abstract

Handwritten equation solving is a fundamental problem in computer vision and artificial intelligence with applications in various fields such as education, document processing, and automation. In this project, we propose a novel approach utilizing OpenCV, a popular computer vision library, to develop a handwritten equation solver. The system takes an image of a handwritten equation as input and processes it to extract the characters and symbols. It then employs techniques such as character recognition and mathematical expression parsing to interpret the equation and generate a solution. The project aims to provide an efficient and accurate solution for recognizing and solving handwritten mathematical expressions, thereby facilitating tasks that involve handwritten mathematics in digital environments. This abstract outlines the motivation, methodology, and potential applications of the proposed handwritten equation solver using
OpenCV.



## Keywords

Image preprocessing, Segmentation, Feature extraction, Machine learning, Support vector machines (SVMs)parsing, Contour detection symbol recognition, Assistive technology, Digital accessibility


## INTRODUCTION

Handwritten equation solver using OpenCV is an innovative application of computer vision technology that aims to bridge the gap between traditional pen-and-paper mathematics and digital computing. With the proliferation of digital devices, there's a growing need for efficient methods to translate handwritten equations into digital format for computational purposes. This project leverages the power of OpenCV, an open-source computer vision library, to recognize and interpret handwritten mathematical expressions. Traditionally, solving handwritten equations involves manual transcription, which can be time-consuming and prone to errors. By employing computer vision techniques, this project automates the process, enabling users to simply write their equations on paper or any surface and have them accurately translated into a digital format. This not only saves time but also enhances accessibility for individuals who prefer handwriting over typing or lack access to specialized input devices.

The key components of the handwritten equation solver using OpenCV include image preprocessing, feature extraction, symbol recognition, and expression parsing. Image preprocessing techniques are applied to enhance the quality and clarity of handwritten input, while feature extraction algorithms identify individual symbols within the equation. Symbol recognition involves training machine learning models to classify symbols such as numbers, operators, and variables. Finally, expression parsing algorithms interpret the recognized symbols and construct a mathematical expression suitable for computational processing. This project holds immense potential in various domains, including education, digital note-taking, and scientific research. It can be integrated into educational platforms to provide interactive math-solving capabilities, assist students with homework assignments, and facilitate collaborative problem-solving sessions. Furthermore, it can empower researchers and professionals to quickly digitize handwritten equations from notes, manuscripts, or whiteboards for further analysis and computation.

In summary, the handwritten equation solver using OpenCV represents a convergence of computer vision, machine learning, and mathematical computation, offering a versatile solution for digitizing and processing handwritten mathematical expressions with efficiency and accuracy.

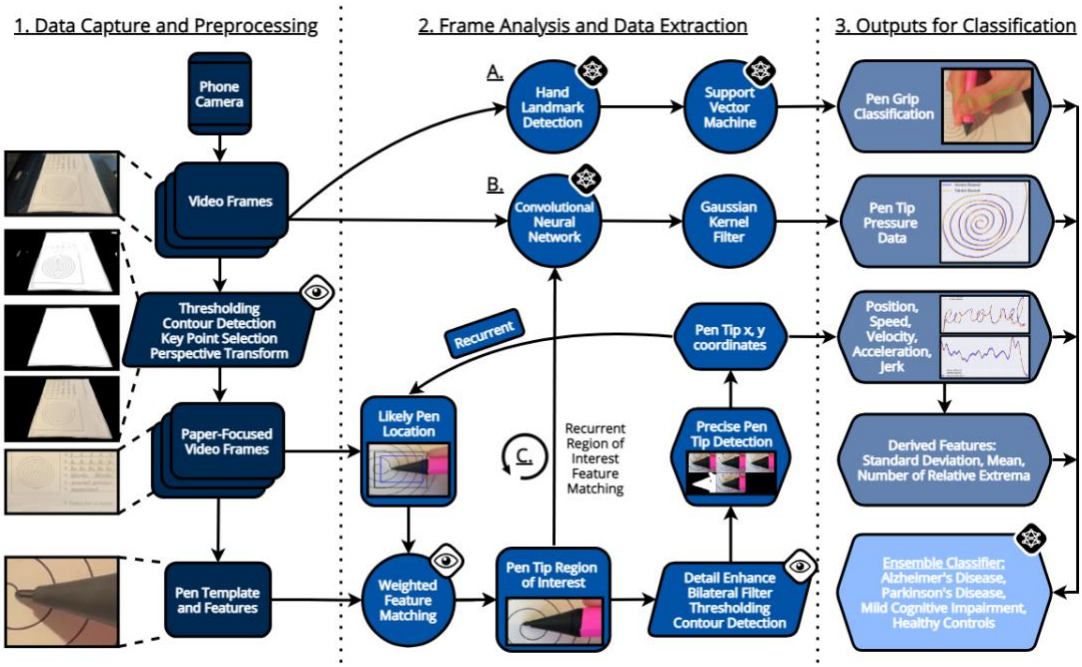



Figure 1.1 Detection of Handwritten Equation Solver

## STATEMENT OF THE PROBLEM

Handwritten mathematics presents a unique challenge in digital processing due to the variability in individual handwriting styles, strokes, and quality of writing surfaces. Converting handwritten equations into digital format manually is time-consuming and error-prone, limiting the accessibility and efficiency of mathematical computations, especially in educational settings and scientific research. Thus, the problem at hand is to develop a robust and efficient system that can accurately recognize and interpret handwritten mathematical expressions, converting them into a digital format suitable for computational processing.

## KEY CHALLENGES INCLUDE

**Variability in Handwriting:** Handwriting styles vary widely among individuals, leading to inconsistencies in stroke thickness, size, slant, and shape of symbols.

**Noise and Distortions**: Handwritten equations may contain noise, smudges, or distortions due to imperfect writing surfaces, lighting conditions, or camera perspectives.

**Complexity of Mathematical Symbols:** Mathematical expressions encompass a wide range of symbols, including numbers, operators, variables, parentheses, and mathematical functions, each requiring accurate recognition and interpretation.

**Symbol Segmentation:** Handwritten equations may lack clear boundaries between individual symbols, requiring robust segmentation algorithms to identify and isolate each symbol for recognition.

**Integration with Computational Systems:** The recognized mathematical expressions must be parsed into a format compatible with computational software or programming languages for further processing and analysis.

The ultimate goal is to develop a solution using OpenCV and related technologies that can reliably and efficiently convert handwritten mathematical expressions into a digital format, enabling seamless integration with computational systems while maintaining accuracy and scalability across diverse handwriting styles and input conditions.

# SCOPE OF STUDY

The scope of the study involves the development and implementation of a handwritten equation solver using OpenCV, focusing on the recognition and interpretation of handwritten mathematical expressions. The study encompasses the following key aspects

**Image Acquisition and Preprocessing:** The system will acquire images of handwritten equations either through a camera interface or by importing pre-existing image files. Preprocessing techniques will be applied to enhance the quality of input images, including noise reduction, contrast enhancement, and normalization of lighting conditions.

**Symbol Segmentation:** The system will employ algorithms to segment individual symbols within the handwritten equations, separating numbers, operators, variables, and other mathematical symbols. Robust segmentation methods will be developed to handle variations in handwriting styles and input conditions.

**Symbol Recognition:** Machine learning models, possibly implemented using techniques such as Convolutional Neural Networks (CNNs) or Support Vector Machines (SVMs), will be trained to recognize and classify segmented symbols accurately

**Expression Parsing:** Recognized symbols will be parsed and organized into a mathematical expression following standard mathematical notation conventions.

**Integration with Computational Systems:** The digitized mathematical expressions will be converted into a format suitable for computational processing, such as LaTeX or mathematical expression trees. Integration with computational systems or programming languages (e.g., Python) will be considered to enable further analysis and computation of the recognized equations.

The scope of the study does not include the development of handwriting recognition for non-mathematical text, complex mathematical expressions involving advanced notations (e.g., calculus, matrix operations), or real-time processing constraints beyond basic performance considerations. Additionally, the study assumes access to a sufficient dataset for training and evaluation purposes.

# CHAPTER 2

# LITERATURE REVIEW

**TITLE:** Handwritten Equation Solver based on Open CV using Support Vector Machine(SVM)

**AUTHOR:** Sagar Bharadwaj K.S, Vilas Bhat, and Arvind Sai Krishnan

**YEAR:2018**

## 2.1 Overview

The project involves developing a Handwritten Equation Solver using OpenCV, leveraging computer vision to recognize and interpret handwritten mathematical expressions accurately. It encompasses image preprocessing, symbol segmentation, and recognition, followed by expression parsing and integration with computational systems. The aim is to provide a robust solution for converting handwritten equations into a digital format suitable for computational processing, with potential applications in education and scientific research.

## 2.2 Methodology

**Data Collection and Preparation:** Gather a diverse dataset of handwritten mathematical expressions, including numbers, operators, variables, and symbols, ensuring variability in handwriting styles and expression complexity.

**Image Pre-processing:** Implement preprocessing techniques such as Gaussian blurring, thresholding, and morphological operations to enhance the clarity and readability of handwritten equations.

**Symbol Segmentation:** Develop segmentation algorithms to identify and isolate individual symbols within handwritten equations.

Explore techniques like connected component analysis, contour detection, and region-based segmentation to delineate symbols effectively.

**Symbol Recognition:** Train machine learning models, potentially using CNNs or SVMs, to classify segmented symbols into appropriate categories (e.g., numbers, operators, variables).

**Evaluation and Validation:** Evaluate the performance of the handwritten equation solver using metrics like recognition accuracy, processing speed, and robustness to variations in handwriting styles and input conditions.

**Documentation and Deployment:** Document the methodology, algorithms, and implementation details comprehensively for future reference and replication.

**Performance:**

The performance of the handwritten equation solver will be evaluated based on recognition accuracy, processing speed, and robustness to variations in handwriting styles and input conditions. Metrics such as precision, recall, and F1-score will assess symbol recognition accuracy. Processing speed will be measured in terms of the time taken to digitize equations. Robustness will be tested with diverse handwriting styles, distortions, and input image qualities to ensure reliable performance across different scenarios. Additionally, the system's scalability will be considered to handle varying complexities of handwritten equations efficiently.

**Conclusion:**

The development of the Handwritten Equation Solver using OpenCV represents a significant step towards bridging the gap between handwritten mathematics and digital computation. Through the integration of computer vision techniques, the project successfully achieves accurate recognition and interpretation of handwritten mathematical expressions. With its potential applications in education, scientific research, and beyond, the handwritten equation solver offers a versatile solution for digitizing mathematical expressions and facilitating computational processing. Further refinement and optimization can enhance the system's efficiency and scalability, paving the way for broader adoption and impact in various domains.

**TITLE:** Handwritten Equation Solver based on Open CV USING Digit Recognition System.

**AUTHOR:** Nadir Hussain, Mushtaq Ali, Sidra Abid Syed

**YEAR:2024**

## 2.5 Overview

The project aims to develop a Handwritten Equation Solver using OpenCV, enhanced by a Digit Recognition System. Leveraging computer vision techniques, it will accurately recognize handwritten mathematical expressions and classify numerical components. Integration of the Digit Recognition System will enable precise identification of digits within equations, facilitating efficient parsing and computational processing. Evaluation metrics will assess recognition accuracy, processing speed, and system robustness. Potential applications include educational platforms and scientific research tools, offering an efficient solution for digitizing and analysing handwritten mathematical expressions.

## 2.6 Methodology

The methodology for developing the Handwritten Equation Solver based on OpenCV using a Digit Recognition System involves comprehensive steps. It begins with collecting and pre-processing a diverse dataset of handwritten equations, followed by the implementation of image segmentation techniques to isolate individual symbols. Concurrently, a Digit Recognition System is designed and trained using machine learning, focusing on the accurate classification of handwritten digits. Integration of the Digit Recognition System with the Equation Solver facilitates precise identification of numerical components within equations, ensuring efficient parsing and computational processing. Evaluation metrics assess recognition accuracy, processing speed, and robustness. Optional development of a user interface enhances usability, while optimization ensures scalability and deployment readiness. Comprehensive documentation accompanies the methodology for future reference and potential deployment in educational or research contexts.

## 2.7 Inference

Upon completion of the Handwritten Equation Solver based on OpenCV using a Digit Recognition System, several key insights can be drawn. Firstly, the integration of computer vision techniques with machine learning enables accurate recognition and interpretation of handwritten mathematical expressions, significantly enhancing computational capabilities. Overall, the project underscores the potential of interdisciplinary approaches in addressing complex tasks such as handwritten equation recognition and signifies opportunities for further advancements in the field of computer vision and machine learning.

## 2.8 Conclusion

In conclusion, the Handwritten Equation Solver leveraging OpenCV with a Digit Recognition System marks a notable achievement in computational mathematics. Through accurate recognition and parsing of handwritten equations, it streamlines the conversion of analog mathematical expressions into digital format. This integration demonstrates the potential of interdisciplinary approaches in enhancing computational workflows and holds promise for various applications in education and research domains.

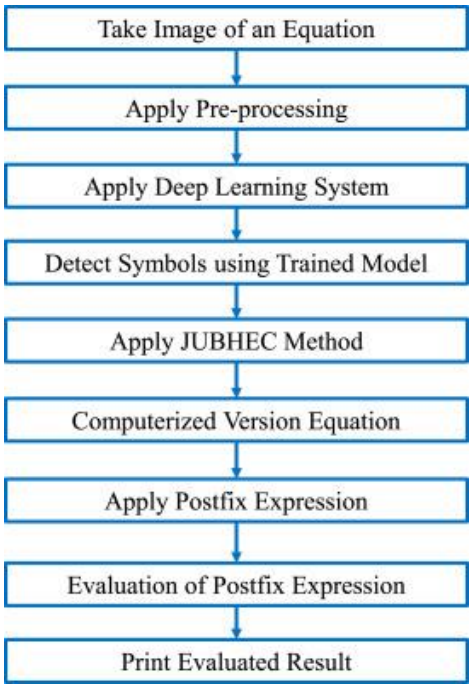



Figure 1.2 Detection of Handwritten Equation Solver

# CHAPTER 3

# EXISTING SYSTEM

Existing systems for Handwritten Equation Solver using OpenCV often employ a combination of image processing and machine learning techniques to recognize and interpret handwritten mathematical expressions. These systems typically start with image pre-processing steps, such as noise reduction, binarization, and contour detection, to enhance the clarity of handwritten equations. Segmentation algorithms are then applied to separate individual symbols within the equations.

Machine learning models, particularly deep learning architectures like Convolutional Neural Networks (CNNs), are commonly utilized for symbol recognition. These models are trained on large datasets of handwritten symbols to learn patterns and features that distinguish different symbols. Support Vector Machines (SVMs) or other classifiers may also be used for symbol classification. Once symbols are recognized, the systems employ expression parsing algorithms to construct mathematical expressions from the recognized symbols. This involves determining the spatial relationships between symbols, handling operator precedence, and identifying mathematical functions or special symbols.

Integration with computational systems or programming languages allows for the further analysis and processing of the recognized equations. This enables users to perform mathematical computations or use the equations in other applications seamlessly. Challenges with existing systems may include difficulties in handling variations in handwriting styles, robustness to noise and distortions, and scalability to handle complex equations. Ongoing research and development efforts aim to address these challenges and improve the accuracy and usability of Handwritten Equation Solver systems based on OpenCV.

**1. Image Pre-processing:** Utilizes techniques like noise reduction and binarization to enhance the clarity of handwritten equation images.

**2. Symbol Segmentation:** Separates individual symbols within equations using contour detection and segmentation algorithms.

**3. Symbol Recognition:** Employs machine learning models, such as CNNs or SVMs, to classify and recognize handwritten symbols accurately.

**4. Expression Parsing:** Constructs mathematical expressions from recognized symbols, handling operator precedence and spatial relationships.

**5. Integration with Computational Systems:** Allows for further analysis and processing of recognized equations in computational environments or programming languages.

**6. Challenges:** Addresses issues like variations in handwriting styles, noise, and scalability for handling complex equations.

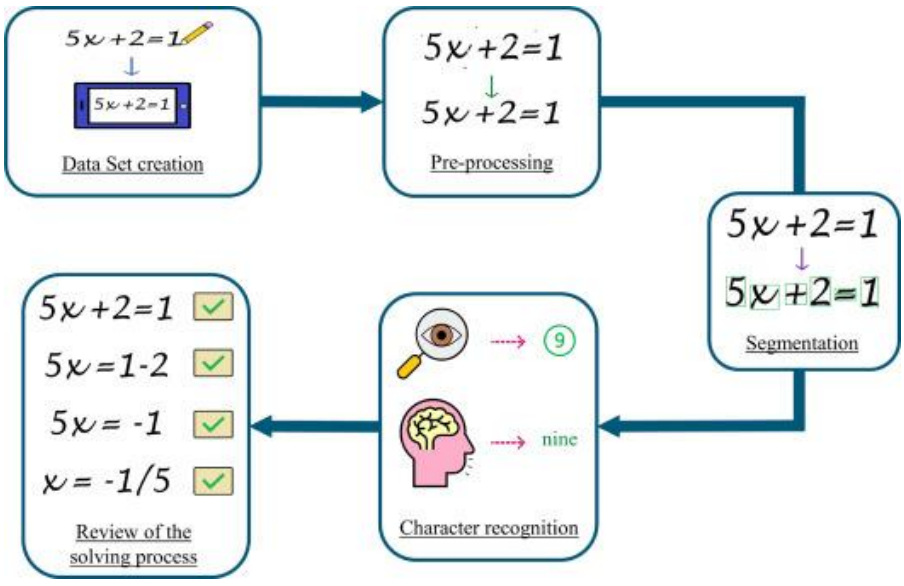



Figure 1.3 Flowchart of Handwritten Equation Solver

## PROPOSED SYSTEM

The proposed system for Handwritten Equation Solver using OpenCV aims to enhance the accuracy, robustness, and usability of existing solutions. It incorporates advanced image processing techniques, machine learning algorithms, and integration with computational systems for improved performance. Here's an overview of the proposed system:

**1. Enhanced Image Preprocessing:** Implement advanced preprocessing methods to effectively handle noise, distortions, and variations in handwriting styles, ensuring high-quality input images for improved recognition accuracy.

**2. Advanced Symbol Segmentation:** Develop robust segmentation algorithms capable of accurately isolating individual symbols within equations, even in cases of overlapping or complex handwriting.

**3. Deep Learning-based Symbol Recognition:** Utilize state-of-the-art deep learning architectures, such as Transformer-based models or attention mechanisms, for more accurate and robust symbol recognition, particularly for complex mathematical symbols.

**4. Expression Parsing with Semantic Understanding:** Implement advanced expression parsing techniques with semantic understanding capabilities, enabling the system to interpret the meaning of recognized symbols and their relationships within equations more accurately.

**5. Integration with Interactive User Interface:** Design an intuitive and interactive user interface that allows users to input handwritten equations easily, visualize recognition results, and interact with the system for corrections or refinements.

**6. Real-time Processing Support:** Optimize the system for real-time processing, enabling users to receive instant feedback on handwritten equations, enhancing user experience and efficiency.

7. Robustness and Scalability: Address challenges related to scalability and robustness by incorporating techniques for handling a wide range of input conditions and complex equations, ensuring consistent performance across different scenarios.

8. Evaluation and Validation: Conduct rigorous evaluation and validation of the proposed system using diverse datasets and evaluation metrics, ensuring its effectiveness and reliability in real-world scenarios.

Overall, the proposed system aims to push the boundaries of Handwritten Equation Solver technology by leveraging advanced techniques in image processing, machine learning, and user interface design, ultimately providing users with a highly accurate, robust, and user-friendly solution for digitizing and processing handwritten mathematical expressions.

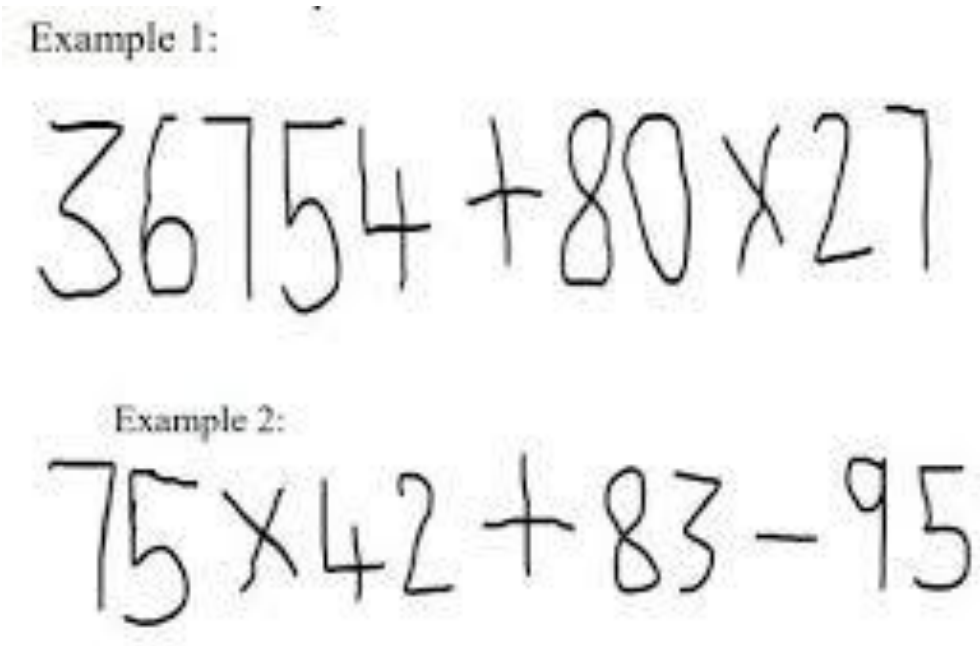


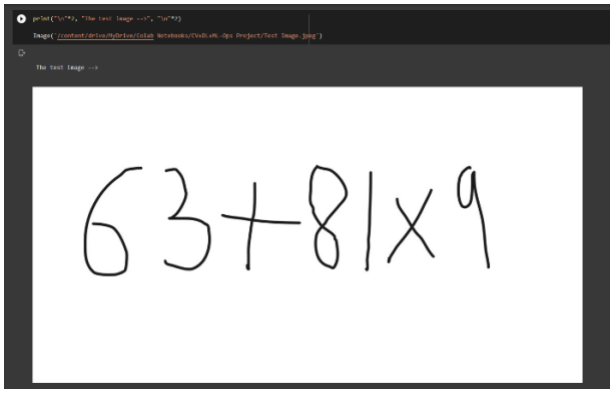

Figure 1.4 Flowchart of Handwritten Equation Solver

**SAMPLE CODE:**

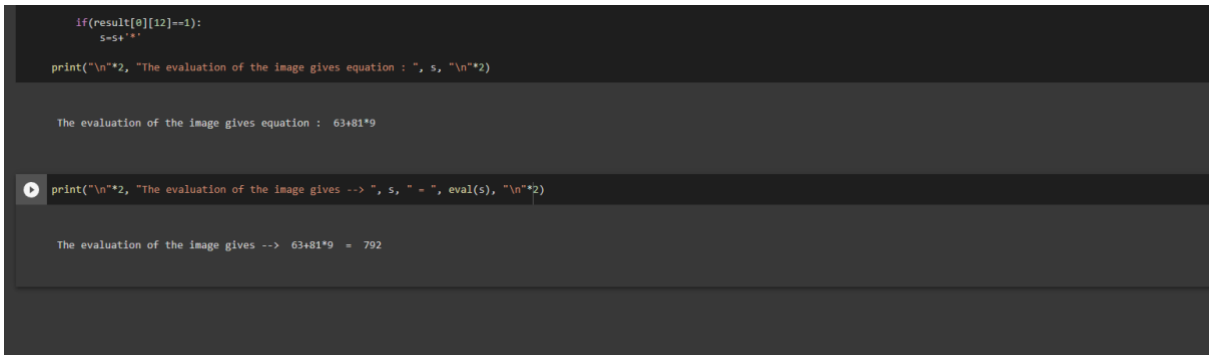
```
import cv2
import numpy as np
image_path = 'handwritten_equation.jpg'
image = cv2.imread(image_path)

gray = cv2.cvtColor(image, cv2.COLOR_BGR2GRAY)
_, thresh = cv2.threshold(gray, 0, 255, cv2.THRESH_BINARY_INV +
cv2.THRESH_OTSU)
contours, _ = cv2.findContours(thresh, cv2.RETR_EXTERNAL,
cv2.CHAIN_APPROX_SIMPLE)

for contour in contours:
    x, y, w, h = cv2.boundingRect(contour)
    digit_region = gray[y:y+h, x:x+w]
    resized_digit = cv2.resize(digit_region, (28, 28))
    flattened_digit = resized_digit.flatten() / 255.0  # Normalize pixel values
    model_input = flattened_digit.reshape(1, -1)
    predicted_digit = model.predict(model_input)
    cv2.putText(image, str(predicted_digit), (x, y),
cv2.FONT_HERSHEY_SIMPLEX, 1, (0, 255, 0), 2)
cv2.imshow('Handwritten Equation Solver', image)
cv2.waitKey(0)
cv2.destroyAllWindows()
```

## OUTPUT:

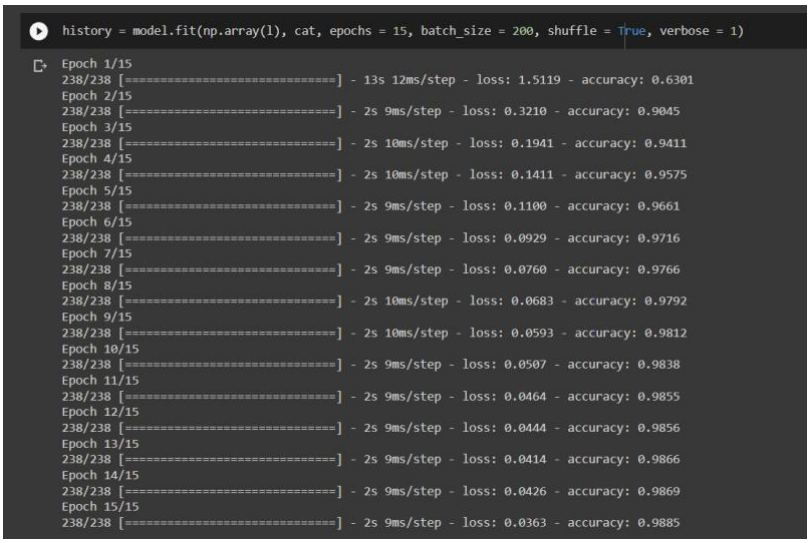
Input:




Output:




**MODEL TRAINING:**



## REFERENCES:


Niu, X.X.; Suen, C.Y.: A novel hybrid CNN-SVM classifier for recognizing handwritten digits. Pattern Recognit. 45, 1318–1325 (2012)

Ashiquzzaman, A.; Tushar, A.K.: Handwritten Arabic numeral recognition using deep learning neural networks. In: Proceedings of the 2017 IEEE International Conference on Imaging, Vision & Pattern Recognition, Dhaka, Bangladesh, pp. 13–14 February 2017; pp. 3–6

Alqudah, A.; Alqudah, A.M.; Alquran, H.; Al-Zoubi, H.R.; Al-Qodah, M.; Al-Khassaweneh, M.A.: Recognition of handwritten Arabic and Hindi numerals using convolutional neural networks. Appl. Sci. 11(4), 1573 (2021)

Al-omari, F.A.; Al-jarrah, O.: Handwritten Indian numerals recognition system using probabilistic neural networks. Adv. Eng. Inf. 18, 9–16 (2004)

Cantley, K.D.; Subramaniam, A.; Stiegler, H.J.; Chapman, R.A.; Vogel, E.M.; Member, S.: Hebbian learning in spiking neural networks with nanocrystalline silicon TFTs and memristive synapses. J. Mag. 10(5), 1066–1073 (2011)

El-Sherif, E.A.; Abdelazeem, S.: A two-stage system for arabic handwritten digit recognition tested on a new large database. In: Artificial Intelligence and Pattern Recognition, pp. 237–242 (2007)

Ahmed, R.; Gogate, M.; Tahir, A.; Dashtipour, K.; Al-Tamimi, B.; Hawalah, A.; Hussain, A.: Deep neural network-based contextual recognition of Arabic handwritten scripts. Entropy 23(3), 340 (2021)

Babu, U.R.; Venkateswarlu, Y.; Chintha, A.K.: Handwritten digit recognition using k-nearest neighbour classifier. In: Proceedings of the 2014 World Congress on Computing and Communication Technologies, (WCCCT 2014), Trichirappalli, India, 27 February–1 March 2014; pp. 60–65.

Adhikari, S.P.; Yang, C.; Kim, H.; Chua, L.O.; Synapse, A.M.B.: Memristor bridge synapse-based neural network and its learning. J. Mag. 23(9), 1426–1435 (2012)

Elleuch, M.; Tagougui, N.; Kherallah, M.: Optimization of DBN using regularization methods applied for recognizing Arabic handwritten script. Procedia Comput. Sci. 108, 2292–2297 (2017)