Automated Data Extraction and Character Recognition for Handwritten Test Scripts Using Image Processing and Convolutional Neural Networks

D. I. Agbemuko[1*], I. P. Okokpujie[1,2], M. J. E. Salami[1], L. K. Tartibu[2]

[1]Department of Mechanical and Mechatronics Engineering, Afe Babalola University, Ado-Ekiti, Ekiti State, Nigeria.
[2]Department of Mechanical and Industrial Engineering Technology, University of Johannesburg Doornfontein Campus Johannesburg, South Africa.

ABSTRACT: Evaluating students through examination scripts in educational environments is crucial, particularly with 'Mastery Feedback' from educators, enhancing student understanding and self-regulation. However, it has remained a hectic exercise requiring some innovative solutions. This study proposes integrating robotics to automate recording and collating marked scripts to reduce the burden on lecturers and improve productivity. Key objectives include developing a data extraction pipeline using methods like Oriented FAST and Rotated BRIEF (O.R.B.) for image alignment and adaptive thresholding for lighting variations. Additionally, a character recognition model using a Single Input Convolutional Neural Network (SICNN) was designed with three preprocessing techniques—binarisation, thinning, and gradient magnitude calculation—tailored to different image requirements. Training on the 'EMNIST by_merge' dataset showed varied validation accuracies, with the gradient input SICNN model achieving the highest at 89.24% overall and the binary input SICNN model excelling with 99.39% on custom scripts. This approach aims to enhance educational administrative processes and efficiency and thus achieve sustainable education.

KEYWORDS: Data Extraction, Character Recognition, Handwritten Test Scripts, Image Processing, Convolutional Neural Networks.

[Received Aug. 10, 2024; Revised Oct. 3, 2024; Accepted Oct. 26, 2024] Print ISSN: 0189-9546 | Online ISSN: 2437-2110

I. INTRODUCTION

In any educational setting, evaluating student test scripts is essential, mainly when conducted by a senior educator, as it completes the students' entire learning cycle (Prilop et al., 2021). This evaluation method significantly enhances students' comprehension and promotes maturity, leading to better academic performance. With evaluating and recording grades for visualisation, the learning cycle would be complete and of higher quality (Kickert et al., 2019). To ensure a high-quality learning cycle, instructors invest significant time and effort in various activities, including designing a curriculum tailored to students' needs, selecting effective teaching methods, assessing each student's progress individually, and processing student results. These efforts are crucial for creating an engaging and supportive learning environment that fosters academic growth and well-being (Abedi et al., 2021). However, processing students' results is labour-intensive, time-consuming, and routine, especially for large student populations. As a result, incorporating robotics, technology, and artificial intelligence (AI) to reduce or eliminate these routine tasks has become highly desirable, allowing instructors to focus on other areas (Añulika et al., 2014).

Artificial intelligence (AI) is a field that enhances the development of intelligent machines or programs that simulate human behaviours with minimal human involvement. AI is derived from the Latin word “artificial intelligentsia, and the word robot is the Czech word "robota," referring to bio-artificial devices used for forced labour. Initiated in 1956, AI includes methods like machine learning and robotics, enabling computers to perceive, reason, learn, and adapt (Hamet et al., 2017). In the context of education, AI has the potential to transform processes by automating routine tasks, thus freeing educators to concentrate on creative teaching strategies, personalised student engagement, and customised learning pathways. This leads to greater efficiency, more teacher-student interaction, and a student-centred approach that enhances subject mastery (Kolbjørnsrud et al., 2016). For instance, Chen describes a scenario where a professor in 1950 struggles to verify plagiarism without reliable tools, while a modern professor in 2019, equipped with A.I., can easily detect plagiarism, enforce rules, and manage larger classes, thereby improving the learning environment (Chen et al., 2020). Ouyang and Jiao categorise AI in education into three paradigms: AI-directed cognitive learning, collaborative AI-supported learning, and student-controlled, data-driven, personalised learning (Ouyang & Jiao, 2021).

*Corresponding author: davidagbemuko.ad@gmail.com
doi:http://dx.doi.org/10.4314/njtd.v21i4.2829

Machine learning, a promising and impactful subset of A.I., aims to emulate human intelligence by learning from the environment. It has revolutionised intelligent systems across various fields due to its ability to analyse large datasets and provide valuable insights (Ghazal et al., 2023). Machine learning, particularly supervised and unsupervised learning, offers techniques for accurate predictions and uncovering hidden patterns in data (Mahesh, 2020, El Naqa & Murphy, 2015). Deep learning, a subset of machine learning, enhances computational models to learn data representations with multiple levels of abstraction, transforming fields such as speech recognition, visual object identification, and drug discovery (LeCun et al., 2015).

Computer vision, a multidisciplinary area combining computer science and AI, aims to enable machines to interpret and understand visual data (Paneru& Jeelani, 2021). It uses machine learning techniques and neural networks to analyse images and videos, extracting insights and identifying patterns beyond human capability. Applications range from self-driving cars to medical diagnostics (Forsyth & Ponce, 2002). For example, computer vision technology has revolutionised quality control in the agrifood industry by enabling objective and comprehensive inspections of fruits and vegetables, detecting issues beyond the visible spectrum (Saldana et al., 2013). Image processing, a related field, involves manipulating and enhancing digital images to improve visual information for human understanding and machine perception (Wiley & Lucas, 2018).

Image processing is a longstanding and valuable field that involves transforming and enhancing digital images for better comprehension and efficient storage, transmission, and representation (Toennies & Toennies, 2012). It encompasses various techniques for improving raw images from cameras or sensors for applications in remote sensing and medical imaging (Chitradevi & Srimathi, 2014). Image processing mainly focuses on digitising and coding images for efficient transmission and storage, enhancing image quality for better clarity, and segmenting and describing images for further analysis using computer vision systems (Petrou & Petrou, 2010).

II. RELATED WORKS

In recent years, some research has been done on image processing, ROI detection, and character recognition. A detailed study checks the proposed method(s), findings and results, relevance, strengths, and weaknesses of some of these works. Gupta et al. (2019) investigated the efficacy of handcrafted features, precisely Oriented FAST and Rotated BRIEF (O.R.B.), and Scale Invariant Feature Transform (SIFT) in object recognition tasks. It employs the Locality Preserving Projection (L.P.P.) algorithm for dimensionality reduction and evaluates the performance using k-NN, decision tree, and random forest classifiers on an 8000-sample dataset. The findings reveal that O.R.B. and SIFT achieve precision rates of 69.8% and 76.9%, respectively. Combining O.R.B. and SIFT features notably results in an improved precision rate of 85.6%. The research underscores the relevance of handcrafted features in scenarios where deep learning may be impractical due to computational limitations. The study's strengths include a comprehensive evaluation of feature descriptors and classifiers and the effective use of L.P.P. for dimensionality reduction. However, the study's limitations include using a relatively small dataset, the absence of a comparative analysis with deep learning methods, and potential generalisation issues. This work demonstrates that integrating multiple handcrafted features can significantly enhance object recognition performance, providing a valuable alternative to deep learning approaches in specific contexts.

Tareen et al. (2018) provided a detailed comparative analysis of various feature-detector-descriptor algorithms—SIFT, SURF, KAZE, AKAZE, O.R.B., and BRISK—focusing on their performance in image registration tasks. By examining these algorithms under different transformations (scale, rotation, viewpoint changes) using benchmark datasets, the study assesses key metrics such as feature detection and description time, matching time, outlier rejection and model fitting time, repeatability, and error rates. Results indicate that SIFT and BRISK are the most accurate algorithms, while O.R.B. and BRISK are the most efficient. This comprehensive evaluation is particularly relevant for vision-based applications requiring robust image registration, such as medical imaging and remote sensing. The study's strengths lie in its extensive experiments and practical recommendations for algorithm selection. However, the computational complexity of some algorithms and the limited diversity of tested datasets are notable weaknesses. Overall, the article provides valuable insights and sets a benchmark for future research, aiding practitioners in making informed decisions about the most suitable algorithms for their needs.Song et al. (2022) noted that image binarisation is the first and most challenging step in image recognition, introducing a 2-stage adaptive binarisation approach to improve image recognition, particularly in scenarios involving complex morphological structures and intense background noise. Building on Sauvola's binarisation algorithm, the method aims to reduce the over-segmentation of organisms with intricate structures. Tested on underwater images of jellyfish collected with the PlanktonScope system, the new approach demonstrates enhanced integrity of extracted jellyfish compared to traditional methods. The first stage effectively suppresses redundant information and reduces the number of ROI, while the second stage preserves weak and low-intensity signals to ensure target integrity. This dual approach enhances hardware resource utilisation, computational efficiency, and robustness in sub-optimal conditions, leading to more accurate analytical results in marine organism studies. Despite its strengths, including improved efficiency and empirical validation on real-world data, the study's reliance on a specific dataset and the potential complexity of the algorithm may limit its broader applicability. Nonetheless, this innovative method is promising for enhancing image recognition in various

fields, particularly those requiring precise segmentation of complex structures in noisy environments.

Cao and Liu et al. (2024) introduced an improved two-dimensional maximum Shannon entropy median filter (TSETMF) to address the challenge of high-density salt and pepper noise in grayscale image processing. Traditional media filters often fail to preserve image details and lack robustness under strong noise conditions. The proposed TSETMF algorithm enhances performance by adapting a threshold using two-dimensional maximum Shannon entropy, thereby improving noise suppression and detail retention. Empirical validation through comparative experiments on classical grayscale images, such as Lena and Cameraman, demonstrates the algorithm's superiority over traditional filters. TSETMF achieves a higher peak signal-to-noise ratio (PSNR) of 24.97 dB at 95% noise density and exhibits stable denoising performance with minimal PSNR decline of 10.78% across noise densities ranging from 5% to 95%. While the study's strengths lie in adaptive threshold selection, superior noise suppression, and denoising stability, its limitations include reliance on specific datasets and potential computational complexity. The TSETMF algorithm offers a robust solution for improving image quality in noisy environments, making it highly relevant for medical imaging, remote sensing, and security surveillance applications. Elen and Dönmez (2024) presented a novel approach to global thresholding for grayscale images, focusing on segmenting regions with different homogeneity based on pixel intensity levels. The method defines alpha and beta regions using the mean and standard deviation of the image histogram and calculates the optimum threshold by averaging the grayscale values of these regions. Experiments conducted on datasets from H-DIBCO'14 (Document Image Binarization Competition), Human HT29 colon-cancer cells (BBBC008), and C. elegans live/dead assay (BBBC010) demonstrate the method's promising performance compared to traditional and state-of-the-art techniques. This innovative approach enhances segmentation accuracy and is validated across diverse applications, including document binarisation and biomedical image analysis. The study's strengths include its novel thresholding method, rigorous empirical validation, and comprehensive comparisons with existing techniques. However, the method's complexity and specificity of datasets used for validation may limit its broader applicability. Despite these limitations, the proposed global thresholding method offers significant improvements in image segmentation, making it a valuable tool for various image processing tasks. Its effectiveness in handling different image types and maintaining high performance across challenging datasets underscores its potential for wide-ranging applications in image analysis.

Michalak and Okarma (2020) presented a novel approach to image binarisation to enhance Optical Character Recognition (O.C.R.) accuracy, particularly for non-uniformly illuminated document images. Recognising the limitations of single-method binarisation techniques, the authors propose a combined approach that integrates various methods, including recent advances in entropy filtering and multi-layered region stacks. This robust combined measure leverages the strengths of individual techniques to improve overall binarisation performance. Experimental validation on the WEZUT OCR Dataset, consisting of 176 non-uniformly illuminated images, shows significant improvements in O.C.R. accuracy, confirming the approach's effectiveness. The study is highly relevant for applications involving document analysis under challenging conditions, such as historical document digitisation and mobile scanning. Its strengths include a focus on enhancing O.C.R. performance, adaptability to different image types, and validation on a real-world dataset. However, the combined method's computational complexity and the need for broader validation across diverse datasets are notable challenges. Overall, this research offers a valuable contribution to image processing, providing a robust solution for improving text extraction from non-uniformly illuminated images and advancing the capabilities of O.C.R. systems. Bloechle et al. (2023) introduce an innovative approach to address the challenge of O.C.R. from document photos taken by mobile phones, particularly under varying and often challenging lighting conditions. Traditional global thresholding methods often struggle in such environments, prompting the development of a new binarisation algorithm. This algorithm utilises two complementary local adaptive passes and incorporates colour components to enhance the binarisation process. The approach aims to improve O.C.R. accuracy by effectively mitigating the effects of shadows and uneven illumination that commonly affect document images. Experimental validation at the DocEng'22 competition demonstrates significant advancements over state-of-the-art methods, showcasing superior performance in document binarisation tasks. The algorithm's success underscores its relevance in practical O.C.R. applications where image quality variability is expected. Key strengths include its innovative design, competitive performance in a competitive setting, and alignment with current mobile technology trends.While promising, the algorithm's implementation complexity and potential computational overhead may warrant further optimisation for real-time applications. Moreover, broader validation across diverse datasets beyond the competition scope could enhance confidence in its applicability across various real-world scenarios. Overall, this research presents a valuable contribution to advancing O.C.R. capabilities on mobile platforms, paving the way for more robust and reliable text extraction from diverse document images.

Anggraeny et al. (2023) investigated the impact of preprocessing methods—dilation, skeletonisation, and noise reduction—on recognising handwritten Javanese characters, leveraging the Aksara Jawa dataset from Kaggle. Focusing on cultural preservation and scientific inquiry, the study applies segmentation and various

preprocessing techniques to enhance character recognition accuracy. Experimental results reveal significant improvements, particularly achieving up to 98% accuracy on black backgrounds, underscoring the efficacy of these methods in optimising recognition outcomes. Notably, configurations utilising black backgrounds consistently outperform white backgrounds, highlighting the importance of contrast-enhancing preprocessing techniques in this context. While the study provides valuable insights into Javanese character recognition and cultural heritage preservation, challenges remain in validating findings across diverse datasets and optimising algorithm complexity for broader applications. Overall, this research contributes to advancing the understanding and practical application of preprocessing techniques in enhancing O.C.R. systems for handwritten Javanese characters, offering promising avenues for future research in digital preservation and cultural heritage studies. Rahman and Rinty (2023) explored O.C.R. systems for extracting text from digital image documents, addressing challenges in text variation, capture modes, and image types. Emphasising the Tesseract O.C.R. engine implemented in C/C++ on Linux, the study details algorithms for text detection, localisation, segmentation, and feature categorisation essential for semantic understanding and structural analysis of image-based textual content. Experimental findings underscore the system's robust performance across standard document images and scanned book/magazine pages, validating its effectiveness in real-world applications. The research contributes significantly to document management, digital archives, and information retrieval systems by enabling precise text extraction from diverse digital sources. However, algorithmic complexity and dataset diversity challenges highlight areas for further research to optimise O.C.R. systems for broader scalability and real-time application demands. This study provides valuable insights and methodologies for advancing O.C.R. technologies to enhance text information extraction capabilities from digital image documents.

Sambath (2023) presented a novel O.C.R. approach that recognises handwritten digits, integrating advanced preprocessing techniques and Convolutional Neural Network architecture. The proposed eight-step preprocessing pipeline enhances O.C.R. accuracy and efficiency, encompassing normalisation, grayscaling, binarisation, noise removal, skew correction, skeletonisation, line separation, and character segmentation. Notably, the model revitalises deprecated algorithms and introduces contour-based feature extraction methods, demonstrating superior performance metrics on the Balanced EMNIST dataset (Crawford, 2017). The study positions itself as a competitive alternative in textual image classification tasks by surpassing current architectures in accuracy, adjustability, and computational simplicity. However, algorithmic complexity and dataset specificity require further exploration for scalability and applicability across diverse handwritten digit datasets. Overall, this research contributes significantly to advancing O.C.R. technologies, offering promising avenues for enhancing digit recognition in practical applications requiring efficient and precise data extraction from handwritten documents. Agrawal et al. (2021) explored CNNs for handwritten digit recognition using the MNIST dataset, comparing their performance with Support Vector Machine (SVM) and K-nearest Neighbor (K-NN) techniques. The proposed CNN model, implemented on Keras with RMSprop optimisation, achieves an impressive training accuracy of 99.06% and testing accuracy of 98.80% after 10 epochs. Enhanced with additional convolutional layers, pooling, and dropout mechanisms, the CNN demonstrates superior effectiveness over traditional methods. Parameter tuning, including filter size, kernel size, and neuron count, significantly optimises model performance for digit classification tasks. Results underscore CNN's capability to outperform SVM and K.N.N. in accuracy and efficiency, validating its prominence in pattern recognition applications. However, challenges such as computational demands and dataset specificity highlight areas for further research to enhance scalability and generalizability across varied handwritten digit datasets. Overall, this research contributes significantly to advancing deep learning techniques in handwritten digit recognition, offering robust methodologies for real-world applications requiring precise and efficient pattern classification.

In work done by Ghazal (2022), image acquisition is done by the user scanning and uploading the handwritten character image as an input, after which preprocessing methods such as greyscaling and binarization are used to improve the quality of the image before segmentation techniques are applied to them in order to extract the needed features. Just like proposed in this methodology, image segmentation was done systematically until individual characters were selected for the image selection process, as this is a significant factor for correct recognition. However, the proposed methodology in this research considers checking for correctness right from image acquisition by employing an image alignment method to correct noises that the user upload errors could have introduced. Overall, the work done by the Ghazal (2022) resulted in an accuracy of 93%.

III. MATERIALS AND METHODOLOGY

A. Conceptual Framework

The frameworks presented in this section, as described in Figure 1, help summarise the entire process involved in the script's image alignment and text extraction, all from the point of image acquisition.

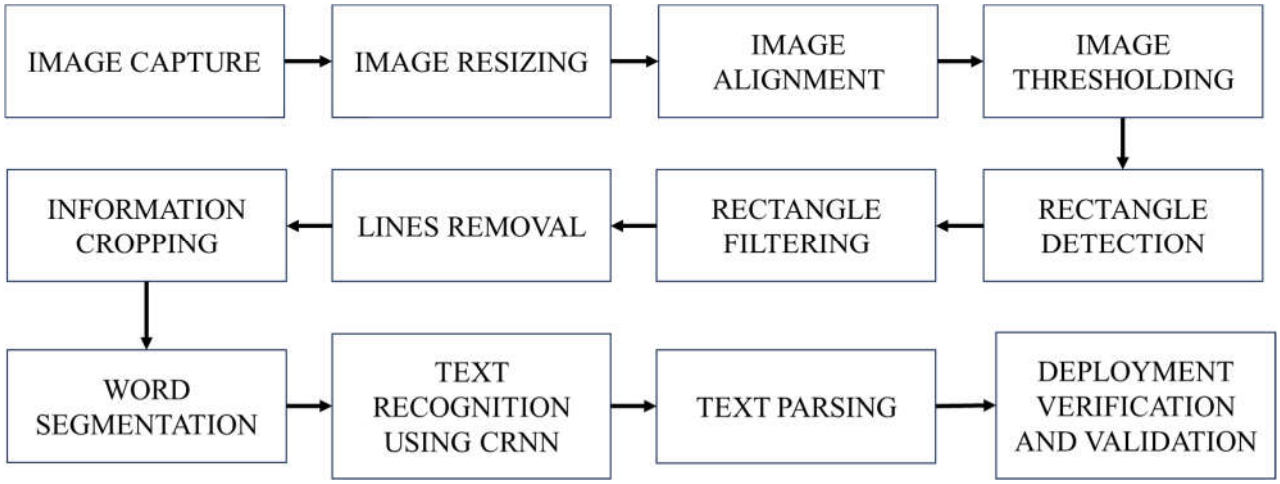


Figure 1: Chart Describing Methodology From Image Acquisition to Text Parsing

B. Experimental Procedure

1) Image alignment

As depicted in Figure 2, the image alignment process is crucial for accurately processing custom student script images Figure 3(a) against a template Figure 3(b) in educational settings. It initiates with preprocessing steps applied to both the script image and the template. Initially, the images are converted to grayscale to simplify subsequent operations and enhance contrast, which helps in improving the visibility of handwritten characters. Subsequently, a binarisation step is employed to convert the grayscale images into binary images, where pixel values are thresholded to distinguish foreground (inked characters) from background. Next, the O.R.B. algorithm is employed for keypoint detection and description. O.R.B. identifies distinctive points (key points) in the script image and the template that are robust to rotations and scale changes. These key points are then matched using a brute-force matcher, which assesses the similarity between corresponding vital points in both images based on their descriptors. To ensure accurate alignment despite perspective differences between the script image and the template, a homographic matrix is calculated using the Random Sample Consensus (RANSAC) algorithm. RANSAC iteratively estimates the best-fit homography by selecting a subset of critical points that consistently agree on the transformation, discarding outliers that could distort the alignment. The resulting homographic matrix describes the perspective transformation needed to accurately align the script image with the template. This image alignment process is essential for establishing precise correspondence between the script image and the template, enabling subsequent automated processing steps such as character recognition and data extraction. By ensuring alignment through these detailed preprocessing and transformation techniques, the system enhances the reliability and accuracy of extracting student information from handwritten scripts in educational environments.

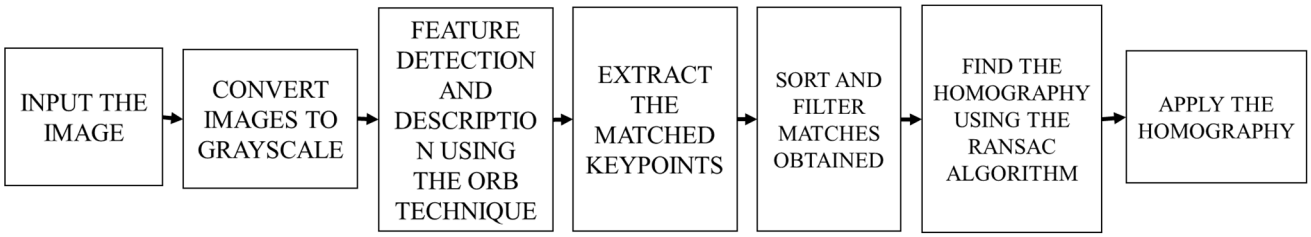
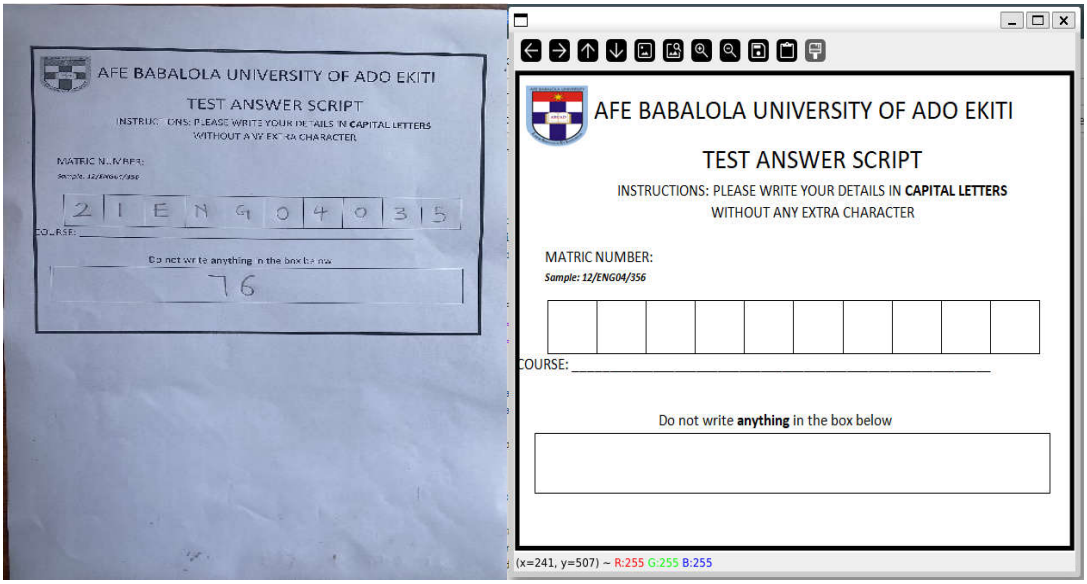


Figure 2: Illustration of the image alignment methodology



(a) (b)

Figure 3: (a) Example Image of a Custom Test Student Script Requiring Alignment and (b)Template Image to which Figure 3(a) should be aligned

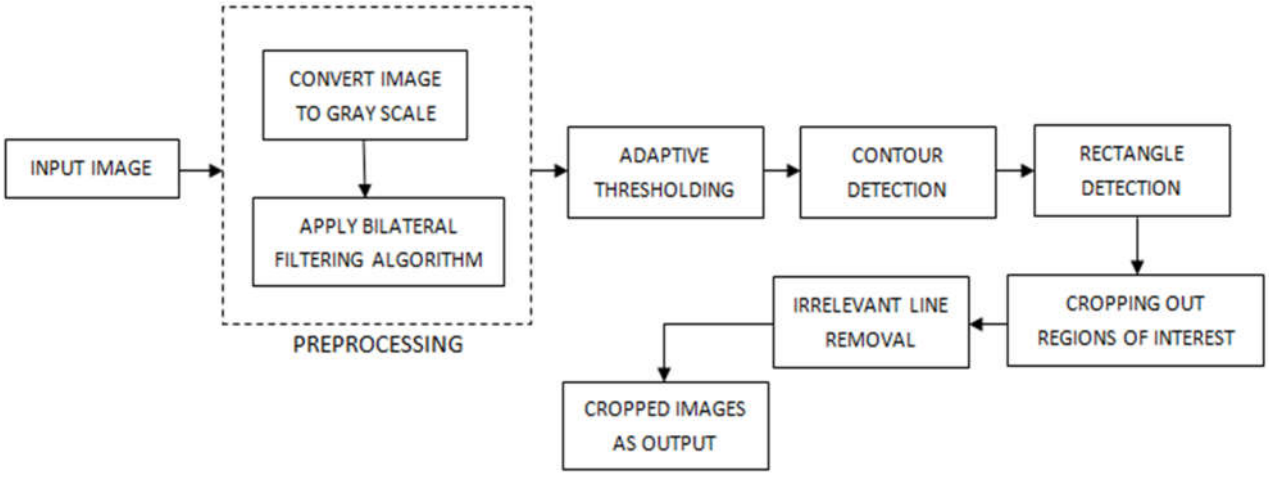


Figure 4: Image Processing Pipeline for Student Information Extraction

2) Student information extraction process

The data extraction process, illustrated in Figure 4, plays a pivotal role in this study in processing student script images and extracting relevant information efficiently. It commences with adaptive thresholding applied to the preprocessed binary image. Adaptive thresholding is crucial as it adjusts the threshold dynamically based on local pixel intensities, effectively separating foreground (student information) from background noise in varying lighting conditions. This step ensures that handwritten characters and marks are accurately segmented from the rest of the image. Following thresholding, contour detection algorithms are employed to identify potential R.O.I.s within the binary image. Contours are outlines of connected components in the image, representing distinct regions such as text areas or handwritten marks. The contours are then filtered based on specific size criteria to select rectangles that likely contain student information, such as names and scores. These selected R.O.I.s are cropped from the original image to isolate and extract the relevant data.

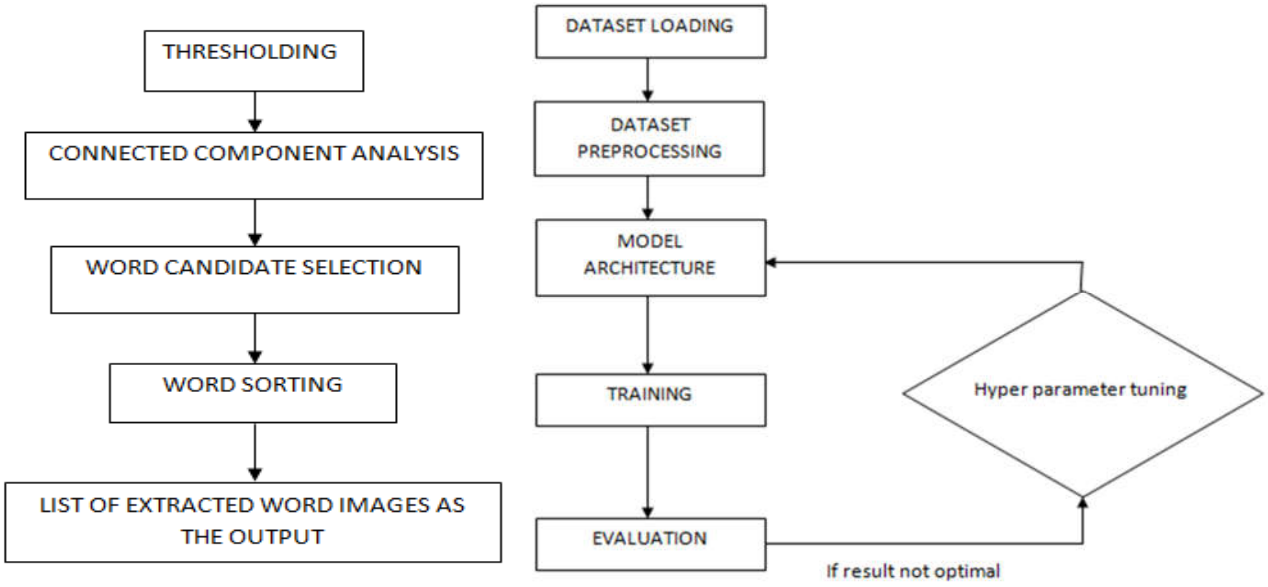
The study refined and extracted data further using Otsu's method, which was applied in conjunction with morphological operations (Ren et al., 2024). Otsu's method is utilised for automatic thresholding to binarise the image, optimising the separation of foreground from background by maximising inter-class variance. Morphological operations, such as erosion and dilation, are subsequently employed to enhance the clarity of extracted data by smoothing rough edges, filling gaps, and removing noise

artefacts that may have persisted through earlier processing stages.This multi-step data extraction process ensures that student information, crucial for assessment and feedback in educational settings, is accurately identified and extracted from handwritten script images. By combining adaptive thresholding, contour detection, Otsu's method, and morphological operations, the system achieves robustness in handling diverse script formats and conditions, enhancing the efficiency and accuracy of automated data processing in educational environments.

3) Word segmentation of the extracted information

The word segmentation process, depicted in Figure 5a, is another critical step in the study to analyse and interpret text from handwritten script images. It begins by converting the input image into grayscale to simplify subsequent processing steps and enhance contrast. Gaussian blur is then applied to the grayscale image to reduce noise and smoothen the edges of characters, improving the quality of subsequent segmentation operations. C.C.A. is employed on the preprocessed image to identify and delineate connected regions or components within the binary image. These components typically correspond to individual characters or groups of connected characters (words). During this stage, components are filtered based on size and aspect ratio criteria to isolate word candidates from noise and non-text elements. This filtering ensures that only regions likely to contain meaningful textual information are retained for further processing.

The extracted word images are then sorted based on their spatial positions within the image. Sorting by position helps maintain the reading order of words, ensuring that segmented text regions are processed sequentially and accurately. This step is crucial for downstream tasks such as O.C.R. or semantic analysis, where the sequential order of words is essential for understanding and interpreting the content of handwritten scripts. By integrating grayscale conversion, Gaussian blur for noise reduction, C.C.A. for component identification, and spatial sorting of word images, the word segmentation process facilitates accurate and efficient textual content extraction from handwritten script images. This systematic approach enhances the capabilities of automated systems in educational environments, enabling robust analysis and interpretation of student responses and textual data.



(a) (b)

Figure 5: (a) Segmentation Methodology for Text Extraction from Images and (b) Workflow illustrating methodology to develop recognition for students' details

4) Implementation of the character recognition model used to recognise the student's details

The character recognition model methodology (described in Figure 5(b))employed in this study focused on evaluating three distinct preprocessing techniques—binarisation, thinning, and gradient magnitude—to enhance the recognition accuracy of handwritten character images. Each technique targeted specific aspects of character representation: binarisation emphasised the shape and contrast between foreground (characters) and background, and thinning aimed to simplify and preserve the structural details of characters. At the same time,

gradient magnitude highlighted fine details and edges within the character images.

A dedicated ICNN architecture was developed and optimised for each preprocessing technique. The SICNNs were designed to accommodate the unique characteristics introduced by each preprocessing method, ensuring that the models effectively leveraged the enhanced image features for accurate recognition. The CNN model architecture is standardised across all input streams to ensure consistency in feature extraction and learning processes, except for using a (5, 5) kernel size specifically for the gradient magnitude input stream. This larger kernel size is chosen to enhance the capture of significant edge features and details emphasised by gradient magnitude images, highlighting contours and boundaries within characters. Figure 6 illustrates the unified model architecture, emphasising the consistent approach across input types.

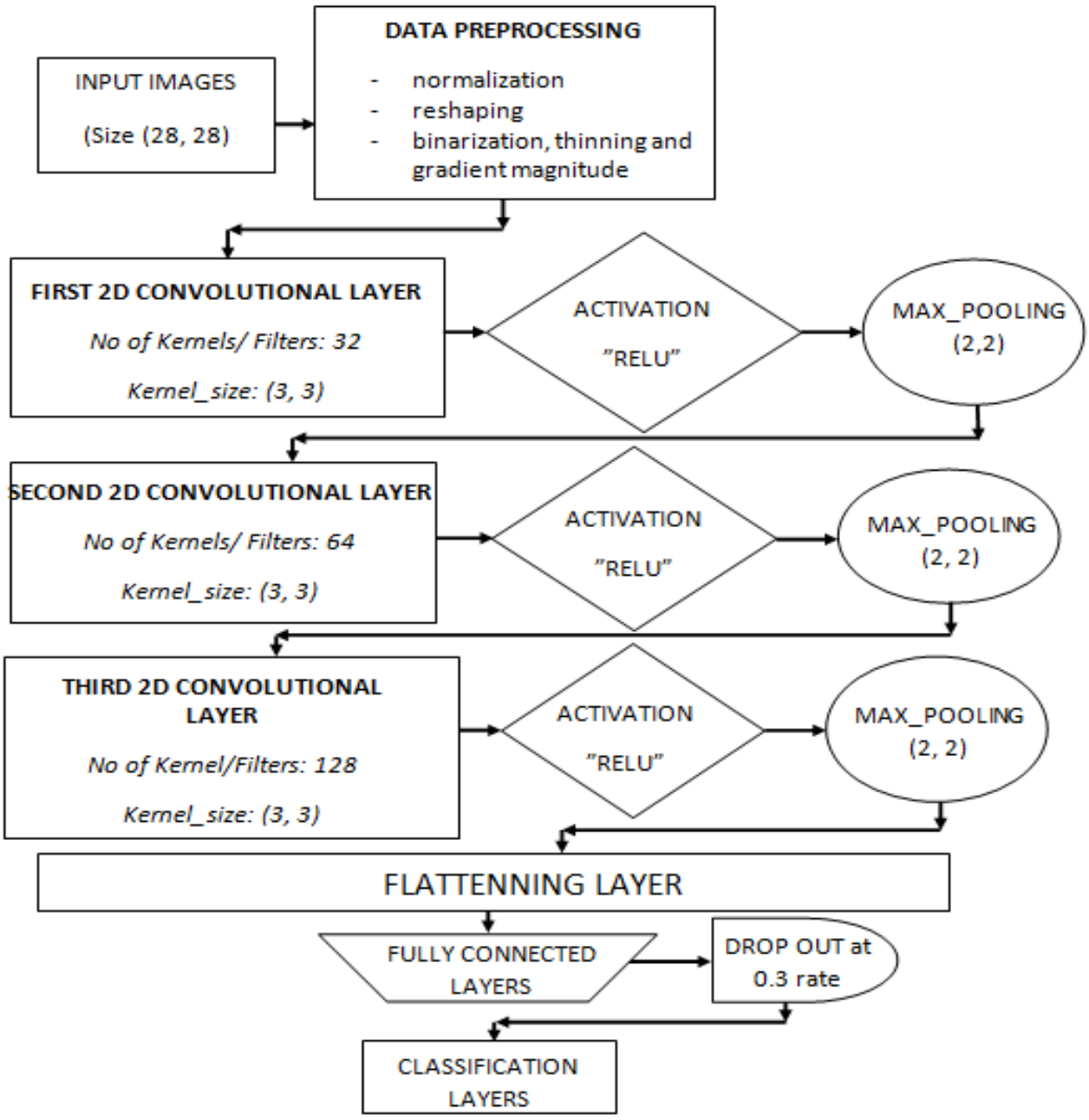


Figure 6: Convolutional Neural Network Architecture Used

The architecture includes normalisation and data reshaping, followed by three Convolutional layers (32, 64, and 128 kernels) to progressively capture complex patterns and features. ReLU activation introduces non-linearity, and max pooling (2, 2) reduces spatial dimensions after each Convolutional layer to prevent overfitting. To enhance robustness, a flattened layer converts outputs to a one-dimensional array, followed by dropout (rate: 0.3). Three fully connected layers (128, 64, and 32 units) refine features, leading to a classification layer tailored to the dataset's class requirements (47 units for the full dataset and 13 units for the engineering faculty subset). This

architecture typically consists of convolutional layers for feature extraction, pooling layers for spatial downsampling, and fully connected layers for classification, optimising feature extraction, minimising overfitting, and enhancing classification accuracy. The study validated the performance of the SICNN models, and extensive testing was conducted using datasets specifically tailored to assess recognition accuracy and robustness across different preprocessing techniques. The models were trained on these datasets, and their performance metrics, including accuracy rates and computational efficiency, were evaluated. This rigorous evaluation process enabled the identification of the most effective preprocessing method and corresponding SICNN model for accurately recognising handwritten characters in diverse educational script contexts. By integrating specialised preprocessing techniques with tailored CNN architectures, the methodology effectively enhanced the capabilities of automated character recognition systems in achieving its objectives.

C. System Development

The experimental setup divided the dataset into 80% for training and 20% for testing, with an additional 30% split from training data for validation to prevent overfitting. Hyperparameters included training for 20 epochs with a learning rate 0.001 to ensure stable convergence. The training utilised regularisation techniques like a dropout rate of 0.3 to mitigate over-reliance on specific features. The hardware involved a 6GB RAM machine with CPU capabilities, and TensorFlow libraries were used for implementation. Key training procedures included employing callbacks such as EarlyStopping with patience of 5 based on validation accuracy, ModelCheckpoint to save the best-performing model, and TrainLogger to monitor and log training progress and metrics effectively (Figure 7).

```
93
94    # Define callbacks
95    earlystopper = EarlyStopping(monitor="val_accuracy", mode='max', patience=5, verbose=1)
96    checkpoint = ModelCheckpoint(f"model.keras", monitor="val_accuracy", verbose=1, save_best_only=True, mode="max")
97    trainLogger = TrainLogger('train_log')
98    tb_callback = TensorBoard(f"logs", update_freq=1)
99    reduceLROnPlat = ReduceLROnPlateau(monitor="val_accuracy", factor=0.9, min_delta=1e-10, patience=4, verbose=1, mode="auto")
100
101   model.fit(
102       X_trainr,
103       y_train,
104       epochs=50,
105       validation_split= 0.3,
106       callbacks=[earlystopper, checkpoint, trainLogger, reduceLROnPlat, tb_callback],
107       )
108
```

Figure 7: Code Snippet showing the callbacks written in Python Language.

IV. THEORETICAL ANALYSIS

A. Mathematical Principle Underpinning Image Alignment Using O.R.B. Algorithm in Student Script Analysis

In implementing the objectives of this research, one of the mathematical principles employed is intensity summation and orientation calculation, which are crucial theories for image alignment requiring the O.R.B. algorithm. Eqns (1), (2), and (3) describe the calculation of horizontal and vertical moments, m_x and m_y, respectively, determining the centroid of detected features. Eqn. (3) computes the orientation θ of corners based on the intensity gradients, which is essential for understanding the directional alignment of image key points. These calculations underpin the O.R.B. algorithm's ability to detect and match image features efficiently, ensuring accurate alignment and robust correspondence between student scripts and template images.

$$m_x = \sum_{x=-r}^{r} xI(x,y) \quad (1)$$

$$m_y = \sum_{y=-r}^{r} yI(x,y) \quad (2)$$

$$\theta = \arctan\left(\frac{m_y}{m_x}\right) \quad (3)$$

Where I(x,y) is the pixel's intensity at the coordinate (x,y), r defines the radius/range around the central point of the R.O.I., and θ is the orientation of the corner.

B. Optimal Image Threshold Using Otsu's Algorithm for Enhanced Edge Detection and Shape Recognition

Another mathematical principle employed is Otsu's algorithm for image thresholding, intending to find the optimal threshold value to convert a grayscale image into a binary format. Eqn.(4) calculates the intra-class variance σ^2 which measures the spread of pixel intensities within each class (foreground and background). Otsu's method effectively determines the threshold that maximally separates the two classes based on their intensity distributions by minimising this variance. This mathematical approach ensures efficient segmentation of image features, enhancing subsequent edge detection and shape recognition tasks in image processing workflows.

$$\sigma^2 = w_1(\mu_1 - \mu_T)^2 + w_2(\mu_2 - \mu_T)^2 \quad (4)$$

Where:σ^2 is the intra-class variance, w_1 and w_2 are the probabilities of the two classes (foreground and background),

μ_1 and μ_2 are the mean intensities of the two classes,

μ_T is the overall mean intensity of the image

B. Foundations of Convolutional Neural Networks for Character Recognition: Mathematical Principles and Operations

A CNN for character recognition leverages fundamental mathematical operations to process input images and predict character labels. The network begins with the Convolutional operation, where a filter W.W.W. is applied to an input image XXX to generate a feature map Z, as defined by Eqn. (5).

$$Z_{i,j} = (X * W)_{i,j} = \sum_m \sum_n X_{i+m,j+n} \cdot W_{m,n} \quad (5)$$

Where:X: input image matrix, W: Filter (kernel) matrix, Z: Output feature map, i,j: indices of the output feature map

Non-linearity is then introduced through activation functions like ReLU (as defined by Eqn. 6), after which pooling operations reduce spatial dimensions while retaining essential features (P) as defined by Eqn. 7. ReLU ensures that only positive values pass through, effectively addressing the vanishing gradient problem and accelerating convergence during training. Following Convolutional and activation, pooling operations refine feature maps by reducing spatial dimensions while preserving essential features. Max pooling is a critical operation in CNNs designed to reduce the spatial dimensions of feature maps generated by Convolutional layers while retaining essential features. The process involves dividing the feature map into non-overlapping regions, typically squares of a defined size, and selecting the maximum value within each region.

$$A_{i,j} = \sigma(Z_{i,j}) = \max(0, Z_{i,j}) \quad (6)$$

$$P_{i,j} = max_{(m,n)\in R} Z_{i+m,j+n} \quad (7)$$

Where: R is the pooling region

After Convolutional and pooling layers, the feature maps are flattened to create a one-dimensional vector known as the flattened feature map. This transformation collapses the spatial dimensions of the feature maps into a single continuous vector, F, which serves as the input to the subsequent fully connected layers. The process involves reshaping the 2D or 3D feature maps into a linear sequence of values. For instance, if a feature map has dimensions H × W × D (height, width, depth), where H, W, and D represent the spatial dimensions and depth, the flattened feature map F will have a length of H × W × D. This step is crucial as it allows the network to interpret and process the extracted features from the previous layers in a manner conducive to making final predictions. Flattening bridges the gap between the convolutional layers, which capture hierarchical features, and the fully connected layers, which perform classification based on these features.

The Flattened feature maps then pass through fully connected layers, where outputs O (as computed in Eqn. 8) are optimised via backpropagation and gradient descent, guided by a loss function L, as described in Eqn. 9. These Eqn. (5) through Eqn. (9) encapsulate the foundational operations enabling CNNs to effectively recognise characters from handwritten scripts.

$$O = \Phi(W.F + b) \quad (8)$$

Where:F: Flattened feature map vector, W: weight matrix, b: Bias vector

$$\Phi: Activation\ function$$

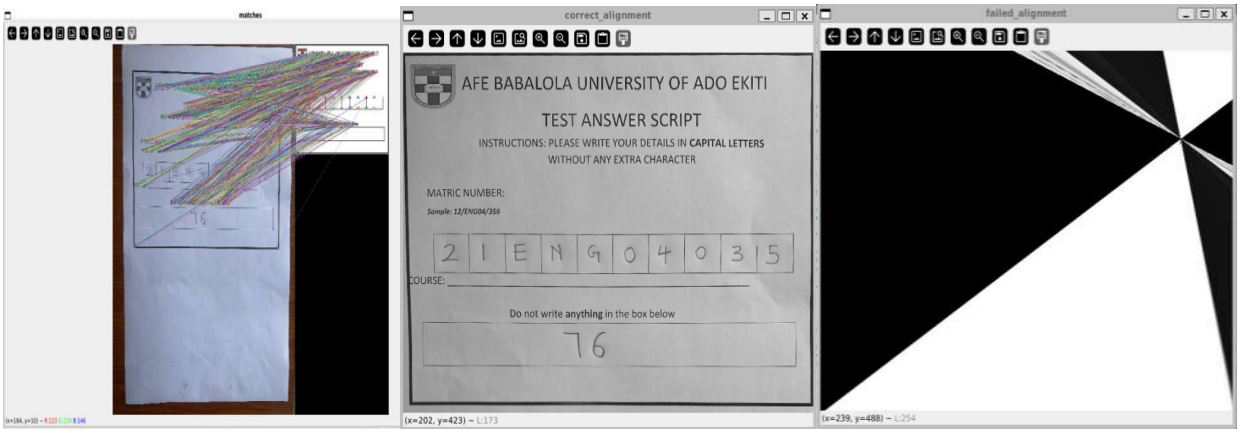
$$L(Y, \hat{Y}) = cross_{entropy}(Y, \hat{Y}) \quad (9)$$

Where:Y: Actual labels, $\hat{Y}$: Predicted labels. The mathematical principles or theories used should be included as appropriate.

V. RESULTS AND DISCUSSION

A. Results from the Image Processing and Data Extraction

Under the hood, Figure 8(a) exposes how the computer aligns in Figures 3(a) and (b). Both images must be resized to a similar ratio for proper alignment. When alignment fails due to wrong sizes or an actual different image from the template, the output would be unpleasant, as described by Figures 8(b) and (c) are successful and unsuccessful, respectively.

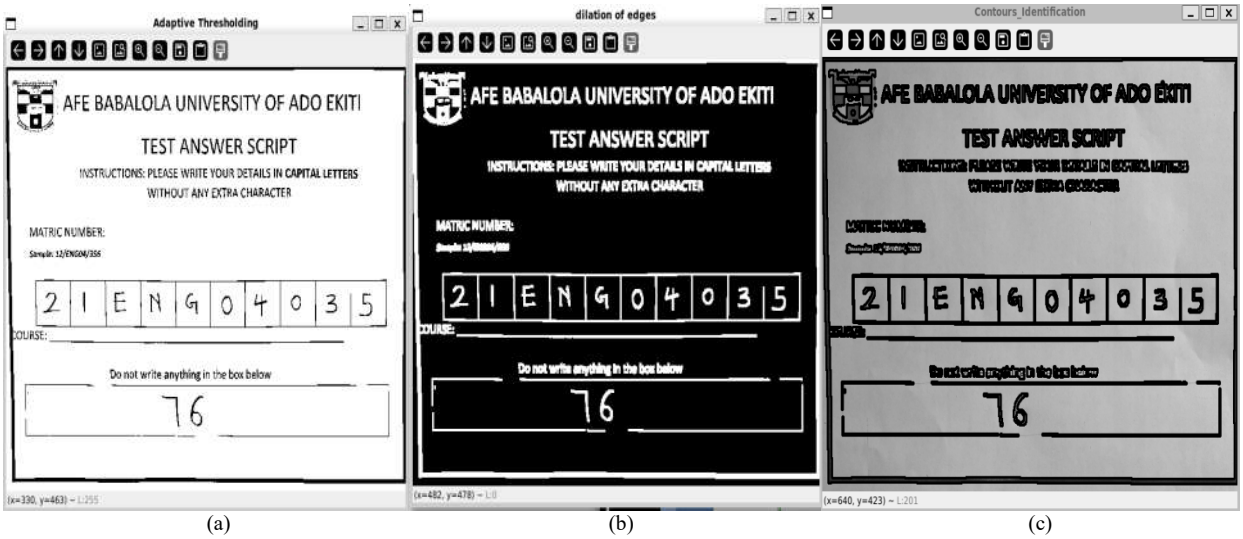


(a) (b) (c)

Figure 8: (a) Image Alignment Process Between Live Script and Template, (b) Result of successful image alignment for further processing, and (c) Result of unsuccessful image alignment (i.e., when the image taken is not a script)

After successfully aligning the script images, the system proceeds with subsequent operations to extract specific student details—primarily the matriculation number and scores—from the script images (Bose et al., 2024). This process involves several key stages illustrated in Figures 9(a) through Figure 10.

1. Adaptive Thresholding (Figure 9(a)): Initially, adaptive thresholding is applied to the aligned script image. This technique dynamically adjusts the threshold for each pixel based on its local neighbourhood, effectively converting the grayscale image into a binary image. This step enhances the contrast and clarity of the text and markings on the script, making it easier for subsequent processing steps to identify and extract relevant information.
2. Dilation for R.O.I. Detection (Figure 9(b)): A dilation operation is performed on the binary image following adaptive thresholding. Dilation expands the boundaries of the foreground objects, effectively thickening the edges of characters and other features. This process helps emphasise the edges and contours within the script, making them more distinct and aiding in detecting R.O.I.s.
3. Contour Detection and Rectangle Filtering (Figure 9(c)): The next step involves detecting contours within the processed image. Contours are outlines of connected components in the binary image, typically representing characters, numbers, or other markings on the script. These contours are then filtered based on specific criteria, such as size and aspect ratio, to identify rectangular regions that likely contain the student's matriculation number and scores. This filtering step helps to isolate potential R.O.I.s for further analysis.
4. R.O.I. Extraction (Figure 10): Finally, the identified rectangles from the previous step are filtered to extract only the regions containing the desired student information—precisely, the matriculation number and scores. This final stage ensures the system accurately identifies and isolates the relevant data points from the script image, preparing them for subsequent processing or storage.



(a) (b) (c)

Figure 9: (a) Result of Adaptive Thresholding during Image Thresholding, (b) Result of Dilation during image thresholding, and (c) Result of contour and rectangle detection on the image

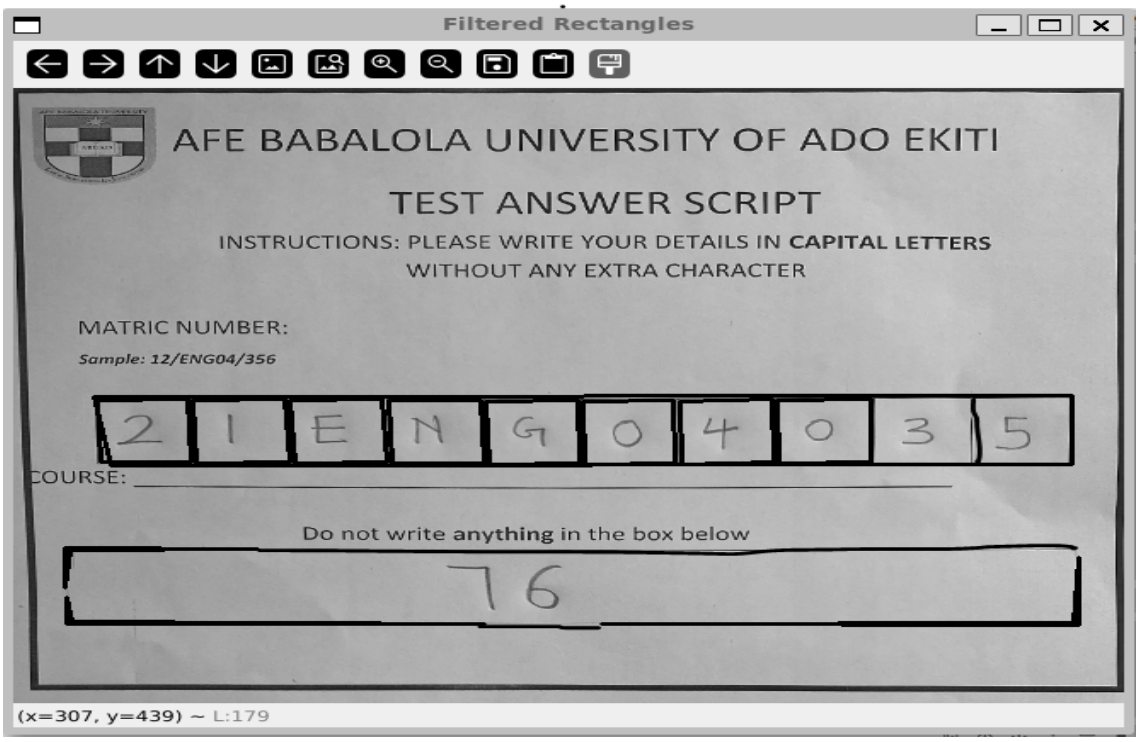


Figure 10: Image showing only the important rectangles needed for information extraction

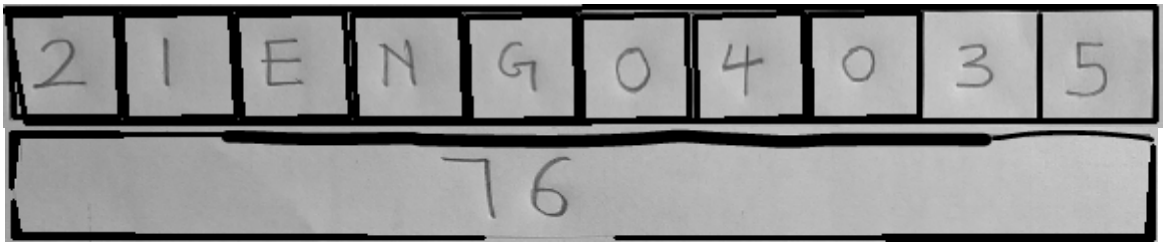


Figure 11: Image of cropped-out information of all the necessary details

After identifying the necessary R.O.I.s from the student script, it crops out these areas, as shown in Figure 11. The system knows an expected ground truth of 11 rectangles, which helps determine if the initial alignment was successful. It then applies thresholding to the cropped images to enhance clarity, ensuring the text is distinct for segmentation (Ying et al., 2024). Figure 12 shows the difference between a thresholded and non-thresholded cropped image. Next, the system segments the images into the student's matriculation number (Figure 13) and scores (Figure 14). This segmentation isolates each character, preparing them for the subsequent character recognition process.

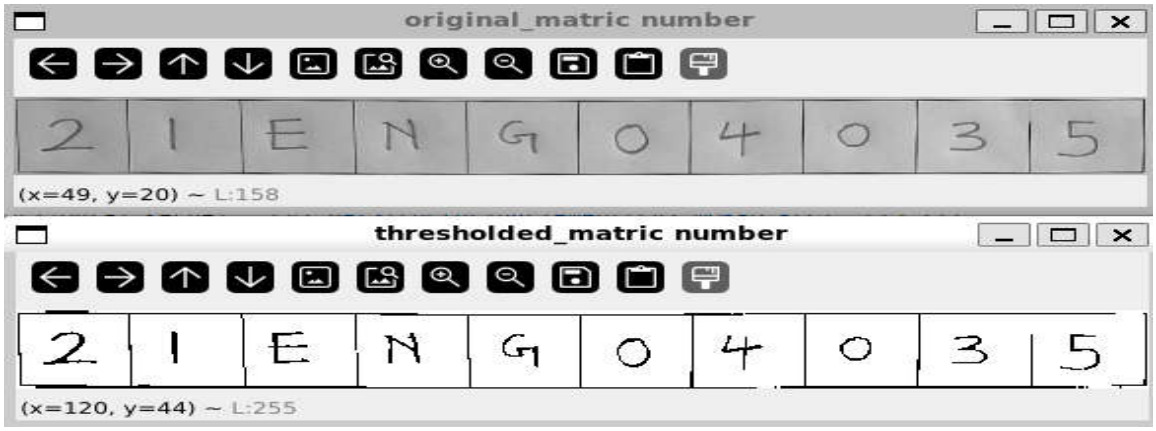


Figure 12: Image showing the difference between a matric number before and after thresholding operation

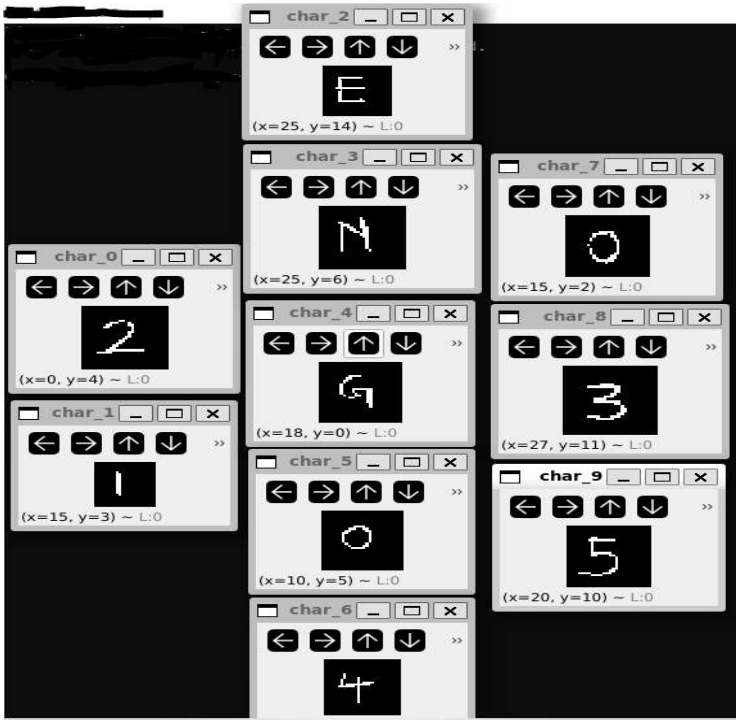


Figure 13: Image showing the segmented characters of the student's matric number

Figure 14: Image showing the segmented characters of the marks allocated to the student

B. Results from the SICNN Development

The final stage of results is the performance of the Convolutional neural networks that have been developed. This section will describe its performance in 3 stages:

1. A comparative analysis of the 3 SICNN models on the entire EMNIST by_merge dataset

2. A comparative analysis of the 3 SICNN models on the part of the dataset needed for the Engineering Faculty, with a deeper look into its confusion matrix

3. Results obtained from system testing the results for four student custom scripts (written with different handwritings)

1) Implementation of the character recognition model used to recognise the student's details

When all three models were implemented on the entire dataset, the binarised model demonstrated consistent performance across training, validation, and testing phases, with train, validation, and test accuracies of 87.13%, 87.69%, and 87.50%, respectively. This indicates that the model generalises well to unseen data without significant overfitting. In contrast, the thinning model shows the lowest performance among the three models. Its train accuracy is 84.83%, validation accuracy is 85.30%, and test accuracy is 85.16%. The reduced accuracy suggests that thinning might remove some crucial information, resulting in a less effective model for character recognition. The gradient model excels in the training and validation phases with the highest accuracy, achieving 89.02% for training and testing and 89.24% for validation. Its test performance remains strong, matching its training

accuracy, indicating excellent generalisation without overfitting. Comparatively, the gradient model demonstrates the highest performance across all phases, suggesting its superiority in capturing fine details and edges crucial for character recognition (Zheng et al., 2024).

The binarised model shows balanced and consistent performance, indicating robust generalisation capabilities. However, the thinning model underperforms in all phases, suggesting that the thinning process may eliminate essential information needed for accurate recognition. The gradient model has the highest accuracy, which indicates its effectiveness in character recognition tasks. The binarised model also performs well with consistent accuracy, while the thinning model shows the lowest performance, indicating potential limitations in its preprocessing approach. Table 1 shows a results table comparing the three models on the entire dataset.

Table 1: Comparative results of all three models on the entire EMNIST by_merge dataset

	Train Accuracy	Val Accuracy	Test Accuracy
Binarised	87.13%	87.69%	87.50%
Thinning	84.83%	85.30%	85.16%
Gradient	89.02%	89.24%	89.02%

2) Analysis of the models on the extracted emnistby_merge dataset

In order to achieve an analysis of the models on needed characters, the study extracted the following characters from the entire dataset: the digits and the characters E, N, and G. The three models were then tested on the extracted dataset. The binarised model demonstrates impressive performance across all metrics. With a training accuracy of 99.49%, validation accuracy of 99.39%, and test accuracy of 99.18%, it shows strong generalisation capabilities. The precision, recall, and F1 scores all stand at 99.18%, indicating that the model is highly accurate and consistent in its predictions without significant overfitting. Figure 15 shows the confusion matrix of the model designed for the binarised input. On the other hand, the thinning model shows the lowest performance among the three models (Bappi et al., 2024). Its train accuracy is 99.12%, validation accuracy is 98.85%, and test accuracy is 94.62%. The precision is 94.47%, recall is 94.62%, and the F1 score is 94.54%. These results suggest that the thinning process might remove some crucial information, resulting in a less effective model for character recognition. Figure 16 shows the confusion matrix of the model designed for the thinned-out input. The gradient model exhibits solid performance with a training accuracy of 99.25%, validation accuracy of 99.40%, and test accuracy of 95.04%. The precision is 94.98%, recall is 95.04%, and the F1 score is 95.01%. These metrics indicate that the gradient model effectively captures fine details and edges crucial for character recognition (Bennour et al., 2024). However, it slightly underperforms compared to the binarised model in terms of test accuracy. Figure 17 shows the confusion matrix of the model designed for the gradient input.

Comparatively, as described in Table 2, the binarised model has the highest performance across all metrics, demonstrating its effectiveness and robustness in character recognition tasks. The gradient model performs well, particularly in capturing detailed features, but shows slightly lower test accuracy. While still performing reasonably well, the thinning model shows the lowest accuracy and may need to be more effective due to the potential loss of essential information during the thinning process.

Table 2: Comparative results of the three models for the extracted dataset

	Train Accuracy	Validation Accuracy	Test Accuracy	Precision	Recall	F1 Score
Binarised	99.49%	99.39%	99.18%	99.18%	99.18%	99.18%
Thinning	99.12%	98.85%	94.62%	94.47%	94.62%	94.54%
Gradient	99.25%	99.40%	95.04%	94.98%	95.04%	95.01%

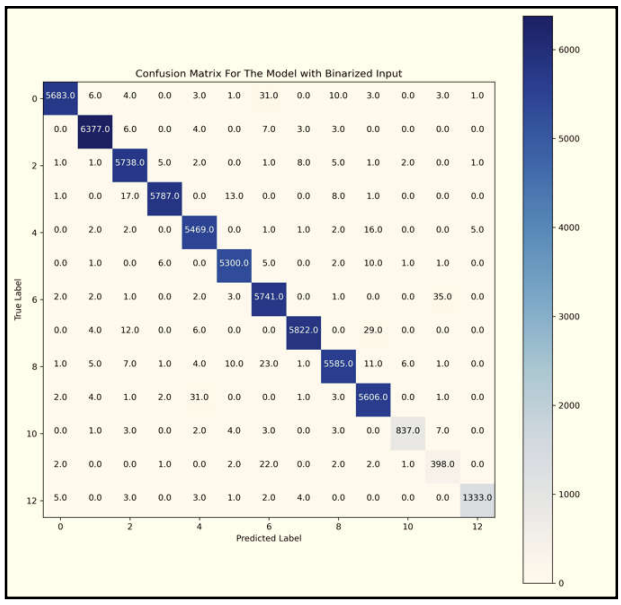


Figure 15: Confusion Matrix for the Binarized Input SICNN model

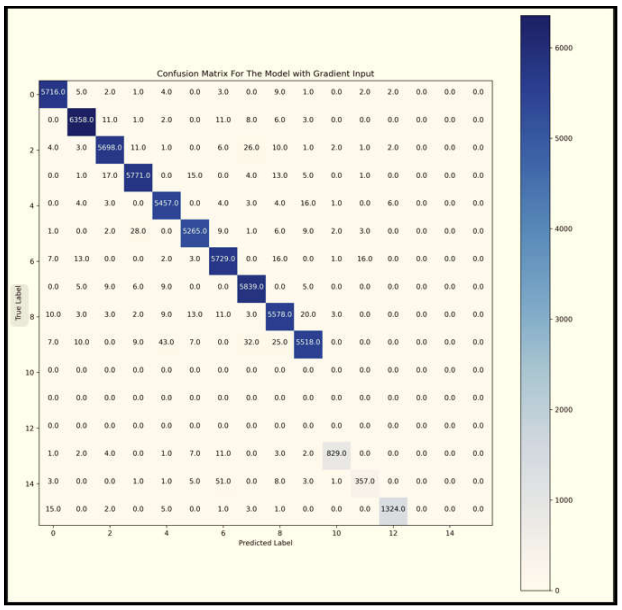


Figure 17: Confusion Matrix of the Gradiented Input SICNN model

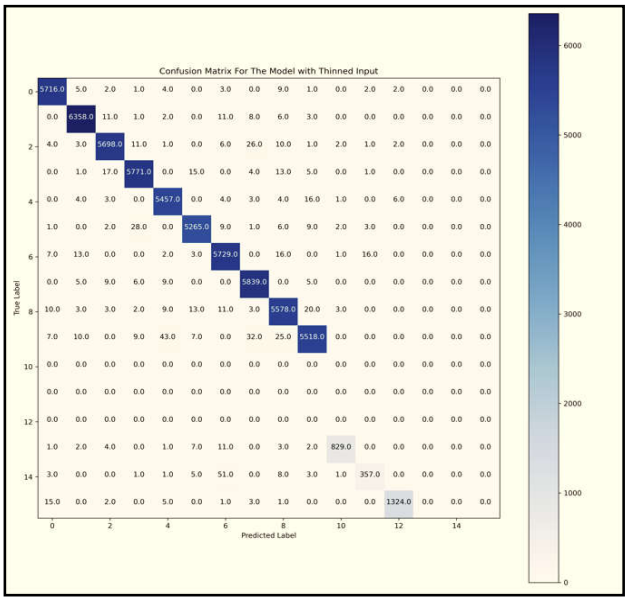


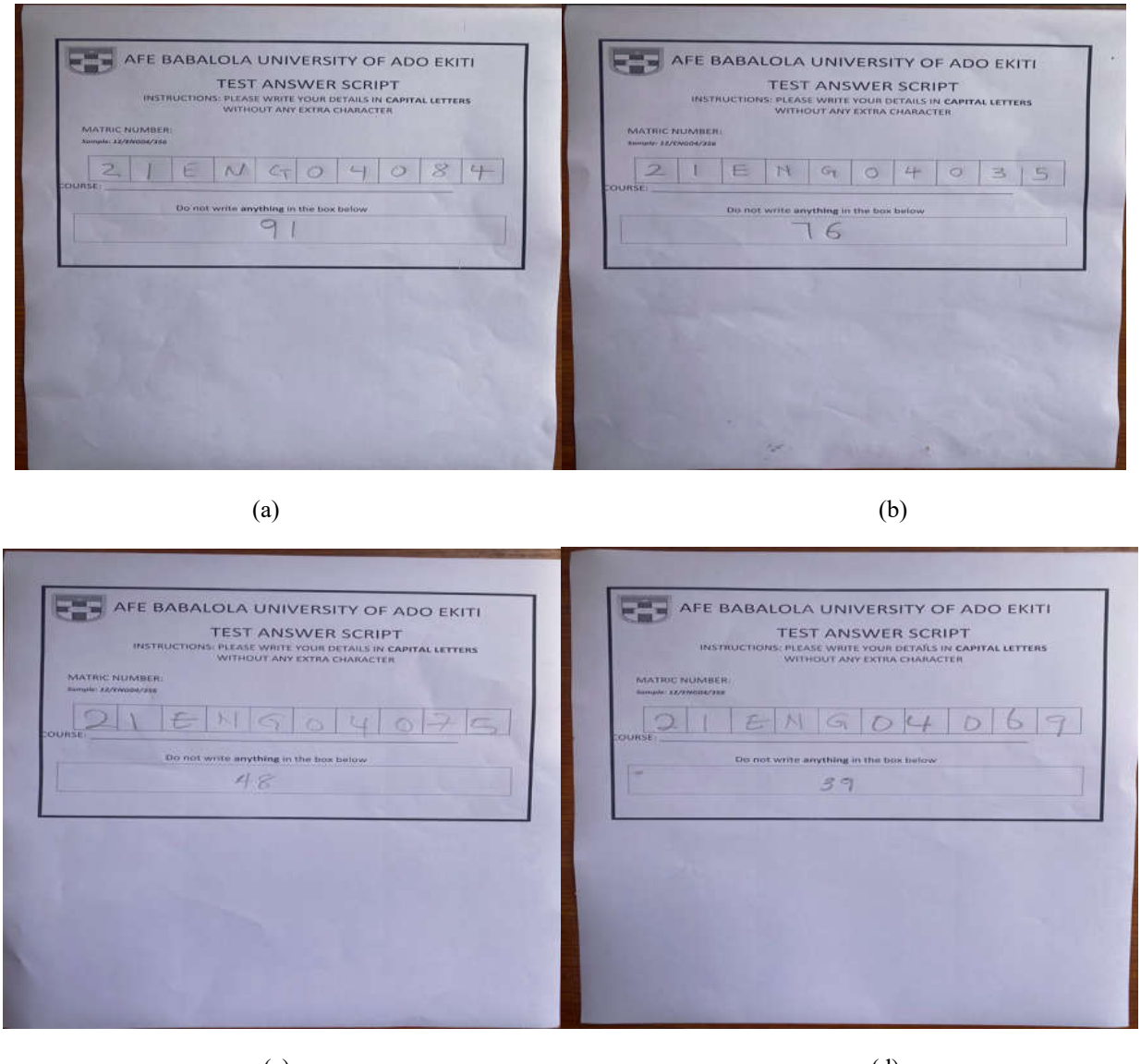
Figure 16: Confusion Matrix of the Thinned Input SICNN model

3) System testing of the models on four foreign and custom student scripts

The custom scripts used to validate the models are 4 test scripts written by four people with different handwriting. These scripts were used to test the entire pipeline of the image-based system, that is, the image preprocessing and data extraction pipeline and the recognition pipeline. Figures 18(a) through 18(d) show the four custom scripts used for the systems, while Tables 3 to 5 and Figures 19 through 22 show the comparative results of each model on these custom scripts. The cells highlighted green in the tables indicate that the model got all the characters from the script correctly. The tables show that the gradient input model has the highest level of correctness, while the thinned input provided the least.

Table 3: The performance analysis of the Binarised Input SICNN model on the four scripts is based on the model's confidence and correctness

	Model One (Binarized Input)		
	Matric (Model's Confidence)	Marks (Model's Confidence)	Correctness?
SCRIPT 1	21 ENG04 084 (94.609%)	91 (99.65%)	100%
SCRIPT 2	21 E4909935 (75.331%)	76 (78.553%)	67%
SCRIPT 3	21 GNG04 025(74.876%)	48 (99.95%)	83.33%
SCRIPT 4	21 ENG04 059 (81.533%)	39 (99.986%)	91.67%
Average Correctness			85.50%



(a) (b)

(c) (d)

Figure 18: (a) Custom Test Script 1, (b) Custom Test Script 2, (c) Custom Test Script 3, and (d) Custom Test Script 4

Table 4: The performance analysis of the Thinned Input SICNN model on the four scripts based on the model's confidence and the model's correctness.

	Model Two (Thinned Input)		
	Matric (Model's Confidence)	Marks (Model's Confidence)	Correctness?
SCRIPT 1	21 ENG04 084 (91.7529)	91 (71.012%)	100%
SCRIPT 2	26 E4909 035 (75.331%)	76 (89.312%)	75.00%
SCRIPT 3	21 GNG04 025 (79.783%)	48 (99.998%)	83.33%
SCRIPT 4	21 GNG04 059(74.892%)	39 (99.894%)	83.33%
Average Correctness			85.42%

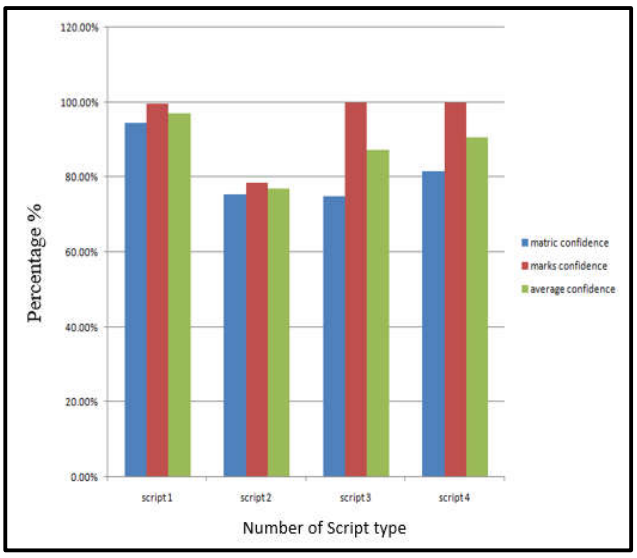


Figure 19: Graph Describing the Confidence of the Binarized Input SICNN Model on the 4 Custom Scripts

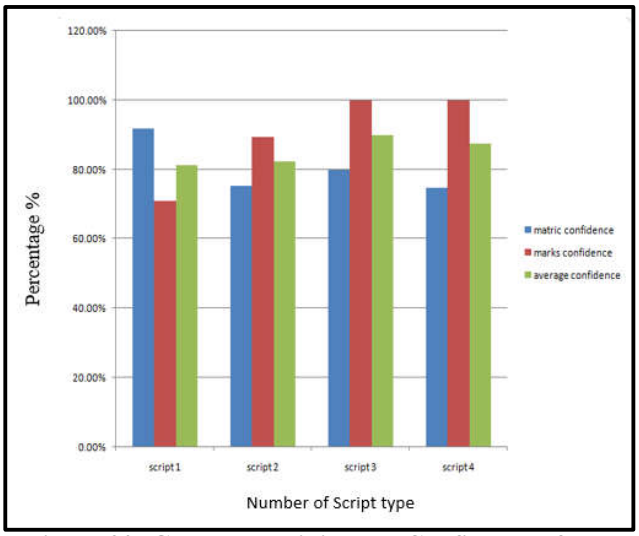


Figure 20: Graph Describing the Confidence of the thinned Input SICNN Model on the 4 Custom Scripts

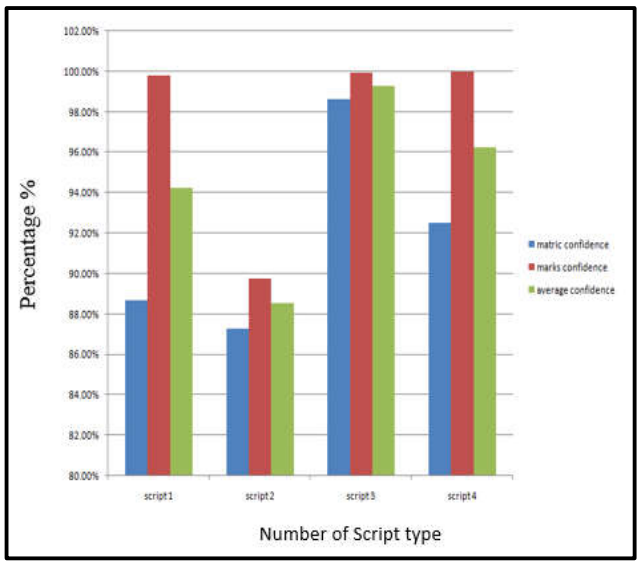


Figure 21: Graph Describing the Confidence of the Gradiented Input SICNN Model on the 4 Custom Scripts

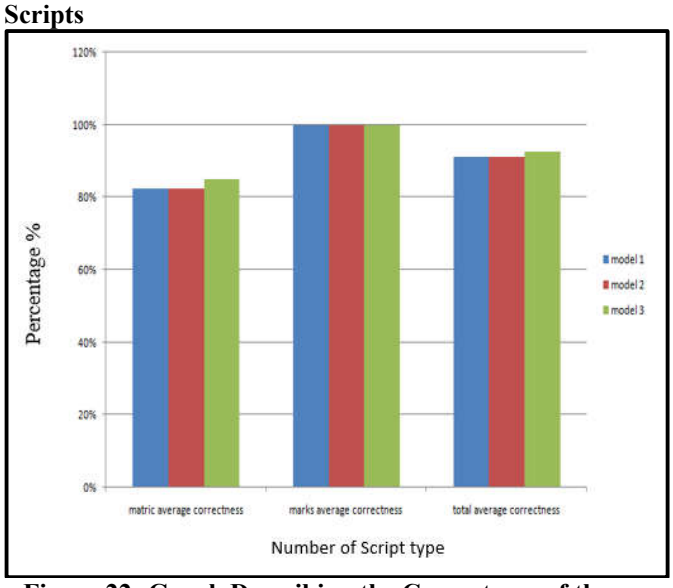


Figure 22: Graph Describing the Correctness of the Models Across the details of the 4 Custom Scripts

Table 5: The performance analysis of the Gradient Input SICNN model on the four scripts based on the model's confidence and model's correctness

	Model Three (Gradient Input)		
	Matric (model's confidence)	**Marks (model's confidence)**	**Correctness**
SCRIPT 1	21 EEG04 084 (88.771%)	91 (99.808%)	91.67%
SCRIPT 2	21 E1509935 (87.297%)	76 (89.753%)	66.67%
SCRIPT 3	21 ENG04 075 (98.656%)	48 (99.966%)	100%
SCRIPT 4	21 ENG04 069(92.514%)	39 (99.980%)	91.67%
Average Correctness			87.50%

VI. CONCLUSION

This research was driven by the need to enhance the efficiency of educational administrative processes, particularly in the context of the rigorous demands placed on lecturers for grading and recording student scripts. The study proposed an innovative solution integrating robotics and advanced image processing techniques to automate the extraction and collation of student information and scores from examination scripts. Key findings revealed that O.R.B. feature detection was highly effective for image alignment, ensuring standardised input images despite variations in orientation and lighting. The adaptive threshold technique proved superior for precise and accurate text extraction, highlighting its importance in the data extraction pipeline, which is crucial for feeding character recognition models. The study further examined the performance of Single-Input Convolutional Neural Network (SICNN) models across different preprocessing techniques and datasets. Results indicated that the gradient magnitude preprocessing technique provided robust performance on diverse datasets, while the binary input technique excelled on specific, targeted datasets.

Conversely, the thinning technique was found to remove essential information, leading to poorer performance. The gradient magnitude SICNN model demonstrated the highest average correctness on custom scripts, underscoring its effectiveness in handling varied data. This comprehensive approach addressed the challenge of automating the grading process. It paved the way for future advancements in educational technology, ensuring that lecturers can focus more on teaching and less on administrative burdens.

DATA AVAILABILITY

The handwritten character digits dataset supporting this study is from previously reported studies and datasets, which have been cited. The datasets are available at https://www.kaggle.com/datasets/crawford/emnist.

AUTHOR CONTRIBUTIONS

D.I. Agbemuko and I.P. Okokpujie: Conceptualization, Methodology, Writing – original draft. **L.K. Tartibu**: Data curation, Project administration, **I.P. Okokpujie and M.J.E. Salami**: Supervision. **D.I. Agbemuko and I.P. Okokpujie**: Visualization, Software. **D.I. Agbemuko, I.P. Okokpujie, and M.J.E. Salami**: Writing – review & editing. **D.I. Agbemuko and I.P. Okokpujie**: Investigation, Resources. **I.P. Okokpujie**: Formal analysis. **L.K. Tartibu**: Validation.

REFERENCES

Abedi, R., Nili Ahmadabadi, M. R., Taghiyareh, F., Aliabadi, K., &PourroustaeiArdakani, S. (2021). The effects of personalised learning on achieving meaningful learning outcomes. *Interdisciplinary Journal of Virtual Learning in Medical Sciences*, *12*(3), 177–187.

Agrawal, A. K., Shrivas, A. K., & Kumar Awasthi, V. (2021, May). A Robust model for handwritten digit recognition using machine and deep learning techniques. In *2021 2nd International Conference for Emerging Technology (INCET)*, IEEE, pp. 1-4

Anggraeny, F. T., Via, Y. V., &Mumpuni, R. (2023). Image preprocessing analysis in handwritten Javanese character recognition. *Bulletin of Electrical Engineering and Informatics*, *12*(2), 860-867.

Añulika, E. A., Bala, E., &Nyap, C. D. (2014). Design and Implementation of result processing system for public secondary schools in Nigeria. International Journal of Computer and Information Technology, 3(01), p. 1.

Bappi, J. O., Rony, M. A. T., & Islam, M. S. (2024). BNVGLENET: Hypercomplex Bangla handwriting character recognition with hierarchical class expansion using Convolutional Neural Networks. *Natural Language Processing Journal*, *7*, 100068, p. 10.

Bennour, A., Boudraa, M., Siddiqi, I., Al-Sarem, M., Al-Shabi, M., &Ghabban, F. (2024). A deep learning framework for historical manuscripts writer identification using data-driven features. *Multimedia Tools and Applications*, 1-27.

Bloechle, J. L., Hennebert, J., & Gisler, C. (2023, August). YinYang a fast and robust adaptive document image binarisation for optical character recognition. In *Proceedings of the A.C.M. Symposium on Document Engineering 2023*, pp. 1-4.

Bose, S. R., Singh, R., Joshi, Y., Marar, A., Regin, R., & Rajest, S. S. (2024). Light weight structure texture feature analysis for character recognition using progressive stochastic learning algorithm. In *Advanced Applications of Generative A.I. and Natural Language Processing Model*s, ISBN13: 9798369305027, I.G.I. Global, pp. 144–158.

Cao, N., & Liu, Y. (2024). High-noise grayscale image denoising using an improved median filter for the adaptive selection of a threshold. *Applied Sciences*, *14*(2), 635.

Chen, L., Chen, P., & Lin, Z. (2020). Artificial intelligence in education: A review. IEEE Access, 8, 75264-75278.

Chitradevi, B., & Srimathi, P. (2014). An overview on image processing techniques. International Journal of Innovative Research in Computer and Communication Engineering, 2(11), 6466–6472.

Crawford, C. (2017). EMNIST (Extended MNIST). https://www.kaggle.com/datasets/crawford/emnist

El Naqa, I., & Murphy, M. J. (2015). What is machine learning? (pp. 3–11). Springer International Publishing.

Elen, A., & Dönmez, E. (2024). Histogram-based global thresholding method for image binarisation. *Optik*, *306*, p. 171814.

Forsyth, D. A., & Ponce, J. (2002). Computer vision: a modern approach. Prentice Hall Professional Technical Reference.

Ghazal, T. M. (2022). Convolutional neural network-based intelligent handwritten document recognition. *Computers, Materials & Continua*, *70*(3), 4563-4581.

Ghazal, T. M., Hasan, M. K., Ahmad, M., Alzoubi, H. M., &Alshurideh, M. (2023). Machine Learning Approaches for Sustainable Cities Using Internet of Things. In The Effect of Information Technology on Business and Marketing Intelligence Systems, Cham: Springer International Publishing, pp. 1969-1986

Gupta, S., Kumar, M., & Garg, A. (2019). Improved object recognition results using SIFT and O.R.B. feature detector. *Multimedia Tools and Applications*, *78*(23), 34157–34171.

Hamet, P., & Tremblay, J. (2017). Artificial intelligence in medicine. Metabolism, 69, S36-S40.

Kickert, R., Meeuwisse, M., M. Stegers-Jager, K., V. Koppenol-Gonzalez, G., R. Arends, L., &Prinzie, P. (2019). Assessment policies and academic performance within a single course: The role of motivation and self-regulation. *Assessment & Evaluation in Higher Education*, *44*(8), 1177-1190.

Kolbjørnsrud, V., Amico, R., & Thomas, R. J. (2016). How artificial intelligence will redefine management. Harvard Business Review, 2(1), 3-10.

LeCun, Y., Bengio, Y., & Hinton, G. (2015). Deep learning. nature, 521(7553), 436-444.

Mahesh, B. (2020). Machine learning algorithms a review. *International Journal of Science and Research* (IJSR), 9(1), 381–386.

Michalak, H., & Okarma, K. (2020). Robust combined binarisation method of non-uniformly illuminated document images for alphanumerical character recognition. *Sensors*, *20*(10), 2914.

Ouyang, F., & Jiao, P. (2021). Artificial intelligence in education: The three paradigms. Computers and Education: Artificial Intelligence, 2, p. 100020.

Paneru, S., & Jeelani, I. (2021). Computer vision applications in construction: Current state, opportunities & challenges. Automation in Construction, 132, p. 103940.

Petrou, M. M., & Petrou, C. (2010). Image processing: the fundamentals. John Wiley & Sons.

Prilop, C. N., Weber, K. E., & Kleinknecht, M. (2021). The role of expert feedback in developing pre-service teachers' professional vision of classroom management in an online, blended learning environment. Teaching and Teacher Education, 99, p. 103276.

Rahman, M. M., & Rinty, M. R. (2023). Text Information Extraction from Digital Image Documents Using Optical Character Recognition. In *Computational Intelligence in Image and Video Processing*, Chapman and Hall/C.R.C, pp. 1–31.

Ren, J., Wang, J., Chen, R., Li, H., Xu, D., Yan, L., & Song, J. (2024). Remote sensing identification of shallow landslide based on improved Otsu algorithm and multi-feature threshold. *Frontiers in Earth Science*, *12*, 1473904.

Saldaña, E., Siche, R., Luján, M., & Quevedo, R. (2013). Computer vision is applied to the inspection and quality control of fruits and vegetables. Brazilian journal of food technology, 16, 254-272.

Sambath, S. (2023). Novel Approach to High Accuracy and Efficiency Optical Character Recognizer for Handwritten Digits.

Song, J., Jiao, W., Lankowicz, K., Cai, Z., & Bi, H. (2022). A two-stage adaptive thresholding segmentation for noisy low-contrast images. *Ecological informatics*, *69*, p. 101632.

Tareen, S. A. K., & Saleem, Z. (2018, March). A comparative analysis of sift, surf, kaze, akaze, orb, and brisk. In *2018 International Conference on Computing, Mathematics and Engineering Technologies (iCoMET)*, IEEE, pp. 1-10.

Toennies, K. D., & Toennies, K. D. (2012). Image Storage and Transfer. Guide to Medical Image Analysis: Methods and Algorithms, 83-109.

Wiley, V., & Lucas, T. (2018). Computer vision and image processing: a paper review. International Journal of Artificial Intelligence Research, 2(1), 29–36.

Ying, S., Fang, J., Tang, S., & Bao, W. (2024). Improved YOLOV5 Angle Embossed Character Recognition by Multiscale Residual Attention with Selectable Clustering. *Electronics*, *13*(13), 2435.

Zheng, T., Chen, Z., Fang, S., Xie, H., & Jiang, Y. G. (2024). Cdistnet: Perceiving multidomain character distance for robust text recognition. *International Journal of Computer Vision*, *132*(2), 300-318.