Automating Vehicle Number Plate Recognition: Leveraging Machine Learning Techniques

Naman Sharma[*1], *Meenakshi Sharma*[2]
GNDU, Amritsar, Punjab, India[1], GGI, Amritsar, Punjab, India[2]
Naman.ptk30@gmail.com, sharma.minaxi@gmail.com*

Abstract

In the current landscape, machine learning techniques have gained considerable importance in the field of computer vision. These techniques are particularly synonymous with recognition and classification processes, especially within optical character recognition systems. Many applications, ranging from processing postal addresses to validating bank checks and verifying vehicle license plates, rely heavily on optical character recognition systems to achieve fast data processing. Over the years, researchers have harnessed machine learning algorithms to create high-speed processing systems that address challenges such as fast tagging, e-challan traffic management, and image search. Techniques such as statistical classifiers, artificial neural networks, kernel methods and various classification systems have been used to improve the accuracy of license plate detection, thereby reducing congestion in parking lots and checkpoints. This article explores a range of machine learning techniques aimed at overcoming the challenges ahead. The authors analyze the unique features and advantages of different methods and discuss performance evaluation methodologies for comparing classifiers using machine learning for efficient vehicle license plate recognition. The ultimate goal remains the swift and efficient detection of vehicle number plates, driving progress within this vital field.

Keywords: Machine learning, classifier, license plate recognition, optical character recognition, deep learning

1. INTRODUCTION

In an era defined by the inexorable growth of vehicular populations, the challenges of efficiently orchestrating traffic flow at toll booths and parking facilities have become increasingly daunting. The market of technology, ceaseless in its innovation, has yielded a transformative solution to this intricate issue. Through the strategic integration of machine learning, deep learning, and the progress of Artificial Neural Networks (ANN), a new era of traffic management is dawning. At the heart of this paradigm shift lies the evolution of Automatic License Plate Recognition (ALPR) systems, poised to revolutionize the pace of number plate identification and processing. These systems don't merely promise respite from the congestion plaguing shopping complexes and theatres; they also grapple with the intricacies presented by an array of license plate designs.

Engaging in the realm of optical character recognition, machine learning techniques decipher the characters imprinted on license plates, drawing upon statistical methods, artificial neural networks, kernels, and an assortment of classification techniques .This introductory narrative not only establishes the landscape where technology and urban mobility intersect but also underscores the role of algorithms in not only reading license plates but also paving the way for the seamless movement of vehicles through our bustling urban centres.

1.1 Character Recognition Methods

Character recognition systems can be categorized into two main types: online systems and offline systems as shown in Figure 1. In online systems, 2D coordinates are utilized to represent characters. Conversely, offline systems involve the scanning of images, followed by character segmentation and the detection and correction of skewness. These processes are tailored to the requirements of application-based Optical Character Recognition (OCR) systems. Within the realm of Character Recognition, a significant focus area relevant to the recognition of handwritten scripts. This serves as a means to identify both written content and signatures. The recognition process entails several core tasks: segmentation, feature extraction, and classification.

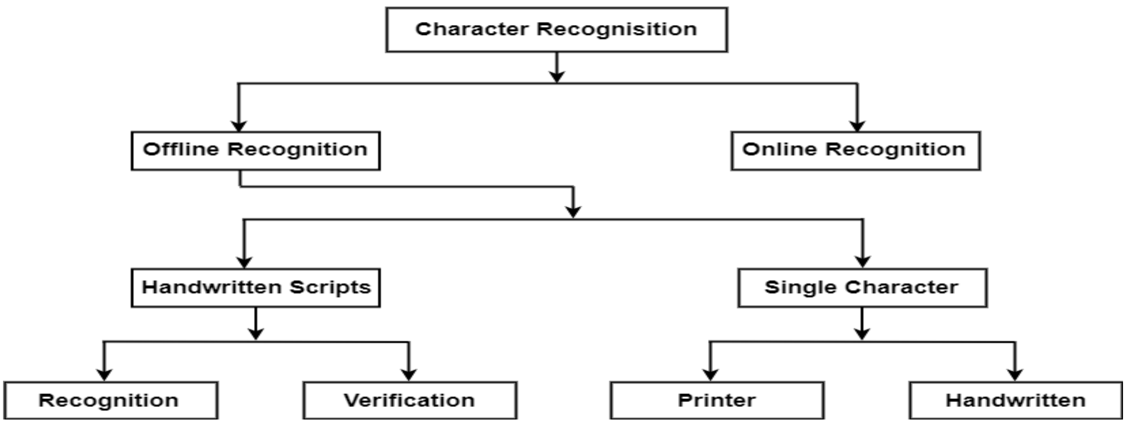

Figure1. Domains of character recognition

The paramount importance is the analysis of documents containing vehicle number plates, with the ultimate aim of transforming them into digital formats. Segmentation involves the separation of words from sentences and characters from words, achieved by defining character boundaries. Subsequently, the next step is character recognition, involving individual character training specific to each license plate. To facilitate this, a Feature

Extraction process is implemented, identifying relevant features while eliminating extraneous ones. Following extraction, the pivotal task remaining is classification, which categorizes data into distinct class labels [2].

1.2 Phases of Recognition System in ALPR

The process of character recognition in automatic number plate systems involves a series of distinct phases, each playing a pivotal role in achieving accurate results. These phases collectively form a comprehensive framework that transforms license plate data into a machine-readable format. The various phases include image acquisition, ROI definition, pre-processing (including character segmentation, skew detection and correction, binarization, noise removal, and thinning), representation, feature extraction, training, recognition, post-processing, and performance evaluation [3]. A detailed explanation of these phases follows:

Image Acquisition:

At the outset, the foundation for the machine learning model is laid by preparing a dataset, often consisting of images in formats like JPG, PNG, and JPEG. This process involves using cameras and scanners, yielding grayscale, colored, or binary images. To optimize memory usage, a thresholding process converts images into binary images, known as image preprocessing. Additionally, for swift processing, license plates undergo compression.

Defining ROI

Simplifying the machine learning model is achieved through the delineation of the Region of Interest (ROI), ensuring focus on relevant areas.

Preprocessing

For compatibility with feature extraction, various preprocessing tasks are executed, encompassing character segmentation, skew detection and correction, binarization, noise removal, and thinning.

Character Segmentation

Just as sentences are separated from multimedia within the scope of ROI, character segmentation isolates individual characters for effective recognition. This phase mitigates challenges arising from character overlap.

Skew Detection and Correction

Skew angle, the deviation from the horizontal axis, often results from image capture with non-parallel planes. Calculation of the skew angle allows for the subsequent rotation of the text by that angle.

Binarization

Binarization streamlines the captured image by retaining only relevant data. This division results in two regions: the text region and the background region.

Noise Removal

Image properties like intensity can be disrupted by noise, which refers to absent or erroneous data. This phase is essential for purifying the image and enhancing its quality.

Thinking

Thinking is akin to a compression technique, extracting the structural essence of an image while retaining its core meaning. The variability in stroke styles, even for the same alphabet, is accounted for to achieve accurate analysis.

Training and Recognition

The crux of recognition lies in pattern recognition techniques, encompassing template matching, statistical methods, structural techniques, and Artificial Neural Networks (ANNs). To address the challenges posed by these techniques, two overarching strategies, Holistic and Analytic, are employed.

Holistic Strategy: This top-down approach focuses on recognizing characters comprehensively.

Analytic Strategy: Employing a bottom-up approach, this strategy dissects characters for recognition.

The meticulous execution of these phases synergizes technology and algorithmic ingenuity, enabling Automatic License Plate Recognition (ALPR) systems to effectively decipher and process license plate information.

2. Neural Network in Recognition of License Plates

2.1 Physical or Geometrical Layout Analysis

Within the realm of license plate recognition, Artificial Neural Networks (ANN) prove invaluable, primarily in the domain of pre-processing tasks and pattern recognition, particularly in the segmentation of characters. In this context, the Multilayer Perception (MLP), a subset of ANN, takes center stage, adeptly handling various preprocessing responsibilities. Furthermore, the Self-Organizing Map (SOM), another component of ANN, collaborates with MLP for pixel-level image classification [4]. This conjunction proves especially effective in character separation.An innovative utilization of SOM and MLP is observed in the realm of rejected characters. By training SOM for each such character and subsequently associating it with MLP, a synergy is established to rectify rejected characters.

This integration showcases the versatility of ANN in document image recognition, rendering it a potent option in the spectrum of recognition techniques [5].

2.2 Logical Structure Analysis:

In the realm of Logical Structure Analysis, a predominant and effective approach involves the utilization of a model-driven framework. This approach hinges upon a model, an intricate representation of the document's physical components, intertwined with their logical labels. The crux of recognition within the model-driven paradigm is to accurately identify the correlations between these physical elements and their corresponding logical designations. Models manifest in two principal forms: tree-based or grammar-rule-based. Irrespective of the model type, syntactic analysis is harnessed to execute structural labeling, thereby enhancing precision [6].

However, the model-driven approach, while conceptually robust, may fail when faced with complex documents or inherent noise within input images. In such scenarios, the data-driven methodology emerges as a pragmatic alternative. This technique, characterized by its immunity to noise, is well-suited to address these challenges. Within this context, the potency of Artificial Neural Networks (ANN) comes to the forefront. ANN-based solutions offer a means to circumvent the limitations of model-driven techniques, particularly the need for integrated prior knowledge.The deployment of Multi-Layer Perceptrons (MLPs) alone might fall short in mitigating these challenges. In response, an innovative avenue surfaces, combining MLPs with structural problem facets. This fusion precipitates two distinct types of ANN: static and dynamic. These variants offer a comprehensive strategy, capable of not only addressing the limitations of a singular approach but also harmonizing structural intricacies, thereby transcending the constraints of traditional methods [7].

3. Neural Network for Structure Patterns

Neural Networks demonstrate proficiency in handling static information, while for unstructured challenges, Artificial Neural Networks (ANN) are often referenced. However, the latter faces limitations when it comes to managing structured data forms such as trees or graphs. Yet, certain models adeptly incorporate structural patterns, whether within static networks or dynamic counterparts.

3.1 Static Network:

Among the mechanisms suitable for static networks, the Multilayer Perceptron (MLP) takes precedence. This choice is driven by its ease of implementation, versatility across various data types, and straightforward applicability to training processes.

3.2 Dynamic Network:

For addressing real-world complexities, the Dynamic Network emerges as a compelling counterpart to the Static Network.

Time Delay Neural Network (TDNN):

Introducing a solution for temporal sequence concerns within the static model, the TDNN employs a Tapped Delay Line mechanism.

Dynamic Feedback Methods:

These approaches integrate feedback mechanisms distinct from the conventional feedforward and output feedback systems. The network output from the second TDNN plays a pivotal role in this architecture.

Time Hopfield Networks (THN):

Used as a monolayer network, THN adeptly manages a multitude of interconnections. The Continuous THN (CTHN) variation caters to oscillatory scenarios, while the Discrete THN (DTHN) variant is akin to its predecessor, albeit with limitations on the activation function, such as a hard limiter rather than a sigmoid function.

Continuous Time Recurrent Neural Network:

Similar to CTHN, this architecture primarily differs in its utilization of differential equations to oversee dynamic processes [8].

In summation, the landscape of Neural Networks is not confined to static or unstructured challenges alone. By embracing the duality of static and dynamic networks, and harnessing a spectrum of specialized methodologies, Neural Networks demonstrate their prowess in tackling a diverse array of problems, from structured data to real-world intricacies.

4. Classification and Learning Methods

4.1 Statistical Methods:

Within the realm of classification, Statistical methods leverage Bayes' decision rule to categorize input patterns. These methods are grouped into two categories based on probability density estimation: parametric and non-parametric classifiers. Parametric classifiers presuppose a known density function form, such as the Gaussian function, and estimate unknown parameters using maximum likelihood techniques. In contrast, non-parametric classifiers directly approximate arbitrary density functions through a posteriori probability estimation from samples. Remarkably, parametric methods often outperform their non-parametric counterparts, even when assuming restrictive density functions [1][9].

4.2 Artificial Neural Networks (ANNs):

In the classification and learning domain of Artificial Neural Networks (ANNs), the pivotal focus lies on adjusting and refining connecting weights iteratively to achieve optimal approximations. The process involves minimizing regression errors, typically achieved using the back propagation method. In supervised learning scenarios, weight adjustments are based on minimizing the squared error between obtained outputs and target values, striving to keep this squared error at a minimum. Pattern recognition phases employ various ANN architectures, such as feed-forward neural networks, radial basis function networks, and higher-order neural networks [10]. Classification decisions are rendered by determining the maximum output among the output nodes for each defined class.

The incorporation of local connections, weight sharing, and structure selection further enhances MLP performance, culminating in what is known as a Convolutional Neural Network (CNN). CNNs function as trainable feature extractors, directly operating on character images while utilizing local connections within hidden nodes [11].Higher-Order Neural Networks, also referred to as polynomial networks or classifiers, produce outputs as weighted amalgamations of pattern features and their polynomial expansions. However, the proliferation of polynomial terms in high-dimensional features can lead to complexity. To counter this, a dimensionality reduction method is often implemented prior to polynomial expansion, streamlining the overall process [12].

4.3 Kernel Methods:

Kernel methods, including Support Vector Machines (SVM), kernel PCA, and kernel FDA, play a pivotal role in pattern recognition. These techniques are instrumental in capturing the essence of patterns within expanded nonlinear feature spaces. Kernel functions are introduced to perform transformations into these nonlinear spaces indirectly. By computing the product of patterns and employing kernel functions, training and classification tasks are conducted. .SVM, a binary classifier, computes a discriminant function by amalgamating kernel functions for all training data with weighted combinations. Shown in Figure 2. Maximizing these weights enhances performance. The identified samples with non-zero weights post-learning are termed support vectors, pivotal for classification [13].

Research has delved into the performance nuances of linear and nonlinear kernels. While the pair-wise method showed superiority with linear kernels, the one-versus-all method demonstrated better performance with nonlinear kernels. Furthermore, SVM classification exhibited superior accuracies when compared to statistical and neural classifiers [14]. Nevertheless, it's noteworthy that the storage and computational demands of support vectors are substantial, prompting exploration into strategies like statistical and neural classifiers for candidate class selection, aiming to manage computation costs [15].

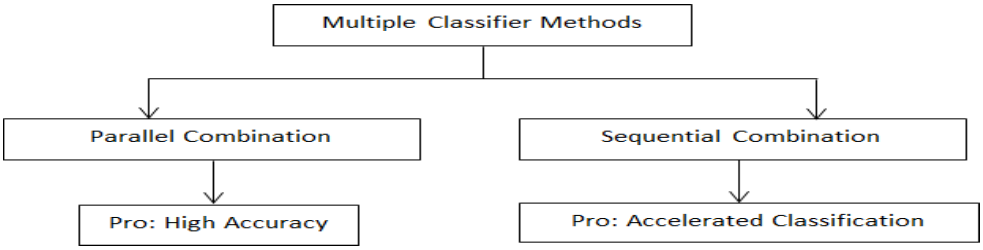

Figure 2: Combination methods used in multiple classifiers

4.4 Multiple Classifier System:

In the pursuit of increased performance accuracy, the realm of multiple classifier systems has emerged, wherein hybrid classifiers combine diverse classifiers. This endeavor has been explored extensively by researchers to bolster single classifier performance. Rahman et al.'s survey [16] encompasses diverse structures of classifier organization, revealing two primary approaches to classifier combination.Parallel or horizontal combinations, resulting in heightened accuracy, and sequential or vertical combinations, adopted for rapid classification of extensive category sets, define the two primary ways of combining classifiers.Parallel combinations harness decision fusion methods to glean insights from classifier outputs. These methods are classified into abstract-level, rank-level, and measurement-level combinations. Among these, measurement-level combination yields the most accurate classifiers, prompting researchers to propose fusion methods. For instance, Suen and Lam applied various classifier combinations at varying levels for character recognition tasks [16]. Enhancements in classification performance attributed to multiple classifiers extend beyond fusion strategies. Techniques like Bagging [17] and Boosting [18] exemplify performance exploration through the diversity of training samples. Single classifiers are repurposed to classify multiple deformations, commonly referred to as virtual test samples derived from input patterns.

5. Performance Comparison of Classification Methods

A multitude of research experiments have been undertaken in the realm of character recognition, aiming to delineate the performance disparities across distinct classification methods. However, these experiments often vary across diverse facets including feature representation, sample datasets, learning algorithms, and pre-processing techniques. Given the variability in these elements across researchers' implementations, comparing reported accuracies across the spectrum of classification and learning algorithms proves to be an intricate endeavour.

5.1 Parameter-Based Comparison:

In the pursuit of evaluating classification and learning models, fundamental parameters within Optical Character Recognition (OCR) systems are scrutinized. As OCR accuracy is profoundly relating on image quality and lacks specific test sets, three core parameters guide system evaluation:

Recognition Rate: This metric represents the percentage of accurately classified characters. However, it falls short in identifying the specific nature of errors.

Rejection Rate: Indicating the percentage of characters unrecognizable by the system, this parameter provides insights into instances where classification is unsuccessful.

Error Rate: Reflecting the percentage of characters that OCR incorrectly classifies, the error rate serves as a comprehensive gauge of classification accuracy.

5.2 Statistical vs. Discriminative Classifiers:

The classification landscape encompasses both Statistical and Discriminative classifiers, each rooted in distinct principles. Statistical classifiers hinge on parametric and nonparametric density estimates. In contrast, Discriminative classifiers prioritize error minimization through classification or regression. Within the realm of Discriminative classifiers, parameters of one class are estimated using samples from all classes, whereas statistical classifiers base parameter estimations for a specific class solely on samples from that class.

Factors	Statistical	Discriminative
Complexity of training	Training time is linear with the number of classes	Training time is proportional to square of the number of classes
Density parameter Adaptation	Adapting the density parameters of a class to new samples is possible	Need re-training with samples
Classification accuracy	Higher with enough samples	Lower with enough samples
Storage and execution Complexity	Statistical classifiers are less Economical	Discriminative classifiers are more economical
Rejection capability	Resistant to outliers	Susceptible to outliers

Table 1: Comparison of statistical and discriminative classifiers

A comprehensive comparison of statistical and discriminative classifiers, considering various factors, is encapsulated within Table 1 [19][20].

In essence, the comparison of classification methods navigates through intricate dimensions, encompassing parameter-based evaluations and the distinct characteristics of statistical and discriminative classifiers. As the field evolves and diversifies, these analyze form the bedrock for informed algorithm selection and performance benchmarking.

5.3 Neural Networks vs. Support Vector Machines (SVMs)

Neural networks and Support Vector Machines (SVMs) are both representative of the discriminative classifier category, sharing fundamental traits within this framework. Nevertheless, their comparison unveils a nuanced interplay of similarities and differences, as outlined in Table 2 [22]. While both models seek to decipher complex patterns and relationships, their distinct approaches and performance nuances contribute to the diverse landscape of classification methodologies.

Factors	Neural Networks	SVMs
Complexity of training	Training time is linear with the number of samples	Training time is proportional to the square of number of samples
Flexibility of training	The parameters of neural classifiers c anbe adjusted in string-level or layout-level training	The parameters of neural classifiers can be adjusted in only layout-level training
Model Selection	The selection of an appropriate structure relies on cross-validation	It depends on the selection of kernel type and kernel parameters but not so influential
Classification Accuracy	Low accuracy	High accuracy

Table 2: Comparison of Neural Networks and SVMs classifiers

6. Conclusion and future directions

In the realm of Automatic License Plate Recognition (ALPR) systems, significant advancements have emerged, propelled by the diverse typography of printed text and the intricate strokes inherent in handwritten text recognition. This evolution has seen the integration of machine learning techniques into the very fabric of ALPR systems, enriching their capabilities. Through contextual analysis, these systems have evolved to tackle the segmentation of characters, a task further complicated by deformations. The integration of machine learning-based classifiers with diverse characteristics has yielded multifaceted functionalities, expanding the scope of ALPR.. Here, the fusion of Artificial Neural Networks (ANNs) with noise-tolerant features, employing feed-forward networks for learning and feed-backward networks for error propagation, represents a robust approach.

Furthermore, the promising domain of deep learning holds potential as an alternative to traditional neural networks, opening doors for heightened character precision outcomes. The journey from ALPR's inception to its current prowess reflects a trajectory of continual enhancement, hinting at a future where technological convergence and machine learning innovations will pave the way for even more accurate, adaptive, and comprehensive license plate recognition systems.

References

1 Barmaki, R. (2015) ‘Multimodal assessment of teaching behavior in immersive rehearsal environment - TeachLivE™’, *ICMI 2015 - Proceedings of the 2015 ACM International Conference on Multimodal Interaction*, 139, pp. 651–655. doi: 10.1145/2818346.2823306.

2 Baumann, S. *et al.* (2018) ‘Pitfalls in accelerometer-based measurement of physical activity: The presence of reactivity in an adult population’, *Scandinavian Journal of Medicine and Science in Sports*, 28(3), pp. 1056–1063. doi: 10.1111/sms.12977.

3 Breiman, L. (1996) ‘Bagging predictions’, *Machine Learning*, 24, pp. 123–140.

4 Brugger, R., Zramdini, A. and Ingold, R. (1997) ‘Modeling documents for structure recognition using generalized N-grams’, *Proceedings of the International Conference on Document Analysis and Recognition, ICDAR*, 1(4), pp. 56–60. doi: 10.1109/icdar.1997.619813.

5 Brunner, C. *et al.* (2012) ‘Pairwise support vector machines and their application to large scale problems’, *Journal of Machine Learning Research*, 13, pp. 2279–2292.

6 Chaudhuri, A. et al. (2017) Optical Character Recognition Systems for Different Languages with Soft Computing, Springer. doi: 10.1007/978-3-319-50252-6.

7 Chi, Z. and Wong, K. W. (2001) ‘A two-stage binarization approach for document images’, *Proceedings of 2001 International Symposium on Intelligent Multimedia, Video and Speech Processing, ISIMP 2001*, pp. 275–278. doi: 10.1109/isimp.2001.925387.

8 Gilloux, M., Gallinari, P. and Bellili, A. (2003) ‘An MLP-SVM combination architecture for offline handwritten digit recognition’, *International Journal on Document Analysis and Recognition*, 5(4), pp. 244–252. doi: 10.1007/s10032-002-0094-4.

9 Gori, M. and Scarselli, F. (1998) ‘Are multilayer perceptrons adequate for pattern recognition and verification?’, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 20(11), pp. 1121– 1132. doi: 10.1109/34.730549.

10 Hamza, H., Smigiel, E. and Belaid, A. (2005) ‘Neural based binarization techniques’, *Proceedings of the International Conference on Document Analysis and Recognition, ICDAR*, 2005, pp. 317– 321. doi: 10.1109/ICDAR.2005.168.

11 Hinduja, D., Dheebhika, R. and Prem Jacob, T. (2019) ‘Enhanced character recognition using deep neural network- A survey’, *Proceedings of the 2019 IEEE International Conference on Communication and Signal Processing, ICCSP 2019*.

IEEE, pp. 438–440. doi: 10.1109/ICCSP.2019.8698008.

12 Jain, A. K., Duin, R. P. W. and Mao, J. (2000) 'Statistical pattern recognition: A review', *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 22(1), pp. 4–37. doi:10.1109/34.824819.

13 Kawatani, T. and Shimizu, H. (2002) 'Handwritten Kanji recognition with the LDA method', pp. 1301–1305. doi: 10.1109/icpr.1998.711940.

14 Li, H., Liu, J. and Zhou, X. (2018) 'Intelligent Map Reader: A Framework for Topographic Map Understanding with Deep Learning and Gazetteer', *IEEE Access*, 6(c), pp. 25363–25376. doi: 10.1109/ACCESS.2018.2823501.

15 Liu, C. L. and Marukawa, K. (2004) 'Handwritten numeral string recognition: Character-level vs. string-level classifier training', *Proceedings - International Conference on Pattern Recognition*, 1, pp. 405–408. doi: 10.1109/icpr.2004.1334137.

16 Liu, C. L., Sako, H. and Fujisawa, H. (2002) 'Performance evaluation of pattern classifiers for handwritten character recognition', *International Journal on Document Analysis and Recognition*, 4(3), pp. 191–204. doi: 10.1007/s100320200076.

17 Liu, C. L., Sako, H. and Fujisawa, H. (2004) 'Effects of classifier structures and training regimes on integrated segmentation and recognition of handwritten numeral strings', *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 26(11), pp. 1395–1407. doi: 10.1109/TPAMI.2004.104.

18 Narendra, K. S. and Parthasarathy, K. (1990) 'Identification and Control of Dynamical Systems Using Neural Networks', *IEEE Transactions on Neural Networks*, 1(1), pp. 4–27. doi: 10.1109/72.80202.

19 Oh, I. S. and Suen, C. Y. (2002) 'A class-modular feedforward neural network for handwriting recognition', *Pattern Recognition*, 35(1), pp. 229–244. doi: 10.1016/S0031-3203(00)00181-3.

20 Rahman, A. F. R. and Fairhurst, M. C. (2003) 'Multiple classifier decision combination strategies for character recognition: A review', *International Journal on Document Analysis and Recognition*, 5(4), pp. 166–194. doi: 10.1007/s10032-002-0090-8.

21 Rangoni, Y. and Bela, A. (2006) 'Document logical structure analysis based on perceptive cycles', *Document Analysis Systems*, 1(7), pp. 117–128.

22 Simard, P. Y., Steinkraus, D. and Platt, J. C. (2003) 'Best practices for convolutional neural networks applied to visual document analysis', *Proceedings of the International Conference on Document Analysis and Recognition, ICDAR*, 2003-

Janua, pp. 958–963. doi: 10.1109/ICDAR.2003.1227801.