

A custom-built deep learning approach for text extraction from identity card images

Geerish Suddul, Jean Fabrice Laurent Seguin

Department of Business Informatics and Software Engineering, School of Innovative Technologies and Engineering,
University of Technology, Port Louis, Mauritius

Article Info

Article history:

Received Dec 5, 2023

Revised Feb 21, 2024

Accepted Feb 28, 2024

Keywords:

Computer vision

Data augmentation

Deep learning

Optical character recognition

Text detection and recognition

ABSTRACT

Information found on an identity card is needed for different essential tasks and manually extracting this information is time consuming, resource exhaustive and may be prone to human error. In this study, an optical character recognition (OCR) approach using deep learning techniques is proposed to automatically extract text related information from the image of an identity card in view of developing an automated client onboarding system. The OCR problem is divided into two main parts. Firstly, a custom-built image segmentation model, based on the U-net architecture, is used to detect the location of the text to be extracted. Secondly, using the location of the identified text fields, a (CRNN) based on long short-term memory (LSTM) cells is trained to recognise the characters and build words. Experimental results, based on the national identity card of the Republic of Mauritius, demonstrate that our approach achieves higher accuracy compared to other studies. Our text detection module has an intersection over union (IOU) measure of 0.70 with a pixel accuracy of 98% for text detection and the text recognition module achieved a mean word recognition accuracy of around 97% on main fields of the identity card.

This is an open access article under the [CC BY-SA](#) license.



Corresponding Author:

Geerish Suddul

Department of Business Informatics and Software Engineering

School of Innovative Technologies and Engineering, University of Technology

Port Louis, Mauritius

Email: g.suddul@utm.ac.mu

1. INTRODUCTION

The identity card is among the most important documents for an individual and its purpose is to link streaming data such as name, surname, date of birth or identity number with a person [1]. Throughout the world, various institutions manually verify the identity of their customers against an identity card. Although time consuming, it is an important verification process. Automating this process using the image of an identity card by traditional computing approaches is virtually impossible, as it requires an exact set of requirements to be respected regarding lighting conditions, orientation or background among others. With the increasing interest in deep learning and especially computer vision, few research works are applying deep learning techniques to retrieve text information from images including identity cards [2]-[5]. In computer vision, the field of study related to extracting and recognising text information from images is termed optical character recognition (OCR) [6]. In most cases, an OCR framework involves two tasks. Firstly, it detects the locations of the text of interest and secondly, it extracts and identifies the text in these regions of interest. The fast evolution in OCR using deep learning techniques created a vast array of models that have been used to extract and recognise text from documents. Zhu *et al.* [7] used techniques like dilation, and vertical and horizontal projects to isolate characters on 55000 Chinese ID cards and these isolated characters were passed

through a Le-Net 5 [8] Convolutional neural network (CNN), achieving an accuracy of 99.2% in recognition and extraction of isolated characters. A further experiment by Pratama *et al.* [9], demonstrated that it was possible to locate text regions on 10000 Indonesian ID Card by applying morphological operations such as dilation and erosion and then passing these text regions into a CNN to extract and recognize the characters in the text regions with an accuracy of 91%.

A complete deep learning approach was adopted, by Ge *et al.* [10], introducing a framework with two deep learning models for OCR on bank cards where the object detection algorithm YOLOV3 [11] with a DarkNet-53 backbone was used to predict bounding boxes corresponding bank identity number location on 1024 bank card images. A convolutional recurrent neural network (CRNN) has been used to extract and recognize the information within the predicted region. Their experiment achieved a recall of 75.20% and a precision of 86.20% on recognizing the bank card number. Similarly, Hoai *et al.* [12] investigated the use of two deep learning models to tackle the OCR problem on 2500 Vietnamese Identification cards where a pre-trained connectionist text proposal network (CTPN) was used to detect important text regions on the ID cards and a CRNN with gated recurrent units (GRU) cells was used to recognize the text within the detected regions. Their approach reached an accuracy of 96.7%, a character error rate of 8% and a word error rate of 9%. A deep learning model with transfer learning with a GoogLE-Net [13] architecture was used to detect and recognize text regions on 13,070 Chinese ID cards and they obtained an accuracy of 95% in recognizing the Chinese characters. Even though there are some techniques that have been used to tackle the OCR problem, there is still a need for further study since the existing deep learning frameworks make use of models trained on image recognition data and are complex to reproduce.

A new identity card system for the citizens of Mauritius was introduced in 2013. It contains biometric features like the storage of fingerprints minutiae of the card bearer and a digital certificate on a chip. The information provided on the identity card is the photograph, surname, first name, surname at birth, gender, signature and identity number of the card holder. It is to be noted that on the back of the card, we can again find the identity number followed by the date of issue, a barcode and a card control number. A detailed specimen version of the identity card is available on the web portal of the Government of Mauritius [14].

The main aim of this study is to design and implement an OCR technique using deep learning to locate and extract textual information from the Mauritian identity card so as to automate the client onboarding process for a FinTech company. Our study focuses on the extraction of text data that corresponds to six main fields: surname, first name, surname at birth, date of birth, gender and identity number. While this problem can be virtually impossible to solve using the traditional programming approach, we focus on applying deep learning computer vision techniques. Our automated approach is divided into two main parts, firstly detecting and locating textual information for the fields of interest on identity card images and secondly extracting and recognising the detected text information.

2. RESEARCH METHOD

2.1. Identity card dataset

A set of 215 images of the Mauritian identity card has been used for this study. Most of the images have been taken by the cardholders, mainly through the camera of their mobile phone. This dataset belongs to a private company registered in the Republic of Mauritius, which is investigating the possibility of developing an automated client onboarding system. To respect the privacy and confidentiality of the cardholder's information, as well as the current legislations, the dataset was accessible only at the physical location of the company on their internal computer system. Therefore, all the necessary data preparation phases and experiments have been physically conducted at the company.

2.2. Overview of proposed method

The proposed method for the deep learning-based OCR solution to extract text values from an identity card consists of three main steps: ID card image pre-processing, Text Detection Module and Text Recognition Module. The solution consists of two separate deep learning models responsible for text detection and recognition. The text values extracted from the ID card images are the surname, name, surname at birth, gender, date of birth and identity number. The first step in the pipeline is to load the dataset and then pre-process the images. The pre-processing step consists of binarizing and resizing the ID card images. Note that in Figure 1, the details of the card holder presented for input have been deliberately hidden for privacy reasons. In this study, our approach reads the card with all the detail therein. Once the images have been pre-processed, they are passed through the text detection module where the coordinates of the text regions of interest are located on the document. The coordinates are then used to crop the text from the images. The cropped images are pre-processed and passed to the text recognition module which identifies the text present at a character level. The characters found for each cropped text image are then arranged into words for the output.

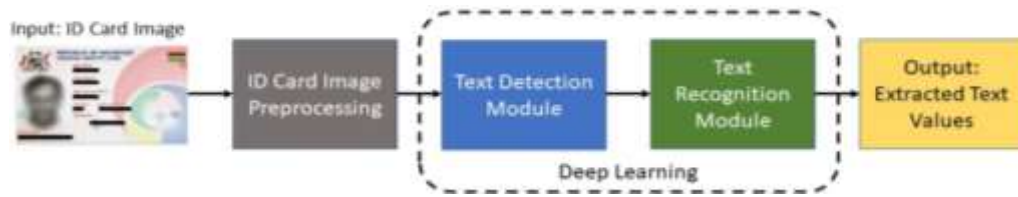


Figure 1. Proposed OCR method to extract text values from image of identity cards

2.3. Image pre-processing phase

Before going through the text detection module, the ID card images are manually cropped to remove unnecessary background. Subsequently, they are pre-processed by: i) applying a grayscale operation to each image ii) thresholding the images with a value set at 127 where any pixel intensity in the image above the threshold is set to 0 and others set to 255, iii) resizing the images to 512 x 512 pixels, and iv) binarizing the images.

2.4. Text detection module

Once an image has been pre-processed it is passed through the text detection module to retrieve coordinates of bounding boxes that correspond to the regions of interest where the text is found on the card, as can be depicted in Figure 2. The text detection algorithm is an image segmentation algorithm adapted from the U-net architecture of Ronneberger *et al.* [15]. Originally designed to detect biomedical image segmentations, we modified and trained a U-Net convolutional network algorithm to detect text regions. The algorithm is trained to predict a segmented mask where all the text regions of interest are identified and represented by white pixels regions. Therefore, the identity card images have been manually masked to specify the text regions of interest. These masked images have been used as a training set for our custom-built deep learning computer vision model. The predicted segmented mask is then used to retrieve the bounding box coordinates of the text regions by converting the predicted image into grayscale, applying the Otsu thresholding operation [16] followed by the find contour operation. All the coordinates are stored in a list, and used to crop the image and create smaller images which contain the text regions of interest. These smaller images are further pre-processed before being passed to the text recognition module. The pre-processing consists of converting to grayscale, resizing and transposing.

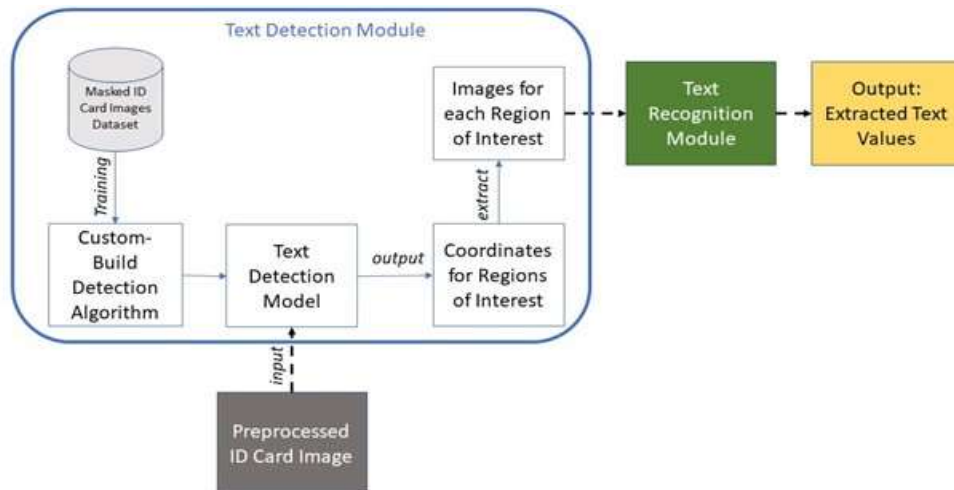


Figure 2. Steps involved in the text detection module

2.4.1. Custom-built text detection algorithm

The proposed algorithm is based on the U-net architecture consisting of a contracting path and an expanding path. The contracting path extracts feature maps from the image, allowing the network to understand the context found in the image. The contracting path decreases the resolution of the image and increases the number of channels in the image, both by a factor of two. The expanding path does the opposite

action by decreasing the number of channels by two and increasing the resolution of the image by two. The expanding path allows the network to use the feature maps obtained from the contracting path and construct a high-resolution segmented output mask which can localise the region of interest. Throughout the contracting path, every convolution operation used a ReLU activation function and the weights were initialised using He-Normalization. The last 1x1 convolution from the expansion path uses a sigmoid activation function (1).

$$f(z) = \frac{1}{1 + e^{-z}} \quad (1)$$

The energy function brings together the cross-entropy loss with a softmax function based on a pixel-by-pixel approach. The softmax function is provided in (2) where, a is the activation, c is the channel and z is the pixel position $z \in \Omega$. The position of each deviation of $f_{l(z)}(z)$ is penalised by the cross-entropy function using the maximum activation of $a_c(z)$ for c , in (3). The ground truth label for each pixel $l: \Omega \rightarrow \{1, \dots, k\}$ and the weight map $\theta: \Omega \rightarrow \mathbb{R}$.

$$f_c(z) = \frac{\exp(a_c(z))}{\sum_{c'=1}^c \exp(a_{c'}(z))} \quad (2)$$

$$E = \sum_{z \in \Omega} \theta(z) \log(f_{l(z)}(z)) \quad (3)$$

2.4.2. Custom-built text detection model configuration

Table 1 shows the hyperparameters configuration used to train the text detection algorithm. These hyperparameters are crucial settings that dictate how the algorithm learns and performs during training and validation phases.

Table 1. Text detection algorithm hyperparameters configuration

Hyper-parameters	Values
Training Batch Size	2
Number of Epochs	30
Steps per Epochs	100
Validation Batch Size	2
Validation Steps	10
Optimizer	Adam
Learning Rate	10-4
Loss Function	Binary Cross-entropy
Metric	Accuracy

2.4.3. Dataset for custom-built text detection model training

The dataset that has been used for the training of the custom-built text detection algorithm was built from the 215 identity card images. It is made up of the cropped images containing the identity card, with as little background as possible, and their corresponding segmented mask version. The segmented masks have been manually created using a Pixel Annotation Tool [17] by masking the text regions on the identity card images. The regions of interest are values for each of the following fields: surname, first name, surname at birth, gender, date of birth and identity card number. The segmented mask image consists of black background with and grey masks on the fields of interest.

Before the segmented masks are used for training, they are converted to grey-scale, threshold, resized to 512 by 512 pixels and binarised. After these operations, the segmented images are black and white images where the white regions represent the regions of text of interest. A pair of actual images and its segmented version is used in a supervised learning approach to train the custom-built deep learning algorithm. Using the actual identity card image as input, and the segmented version as output label/ground truth, the algorithm predicts the mask. The predictions were compared with the ground truth segmented masks and network parameters adjusted.

2.4.4. Evaluation metrics for custom-built text detection model

The intersection over union (IOU) is a metric used to evaluate the accuracy of object detection, instance segmentation or image segmentation algorithm. It may be referred to as the Jaccard Index or Jaccard Similarity [18] and is represented in (4). Pixel accuracy is an evaluation metric that can be used to evaluate semantic segmentation algorithms [19]. It represents the percentage of pixels that have been correctly classified in the predicted image. However, using pixel accuracy alone may not be the most appreciated

solution since in certain cases with high score values, some pixels may be wrongly classified. In (5) represents pixel accuracy (PA), with $k + 1$ classes, p_{ij} being the number of pixels of class i inferred to be of class j and p_{ii} are true positives. Binary cross entropy (BCE) is a loss function that is used to evaluate the performance of binary classification algorithms [20]. It is a popular loss function and is often referred to as log loss. It compares the prediction of the model to the actual class and calculates a score. This score is then used to penalise and adjust the parameters learned by the model, to improve its performance. In (6) represents the BCE, where y_i : Represents actual class (1 or 0), p_i : Represents the probability of the class belonging to class 1 (which is always between 0 and 1) and N Represents the number of training examples in the dataset.

$$J(A, B) = \frac{|A \cap B|}{|A \cup B|} \quad (4)$$

$$PA = \frac{\sum_{i=0}^k p_{ii}}{\sum_{i=0}^k \sum_{j=0}^k p_{ij}} \quad (5)$$

$$BCE = \frac{1}{N} \sum_{i=1}^N -(y_i \log(p_i) + (1 - y_i) \log(1 - p_i)) \quad (6)$$

2.5. Text recognition module

The text recognition module is based on the CRNN algorithm [21] designed for text sequence recognition from images. The CRNN takes the pre-processed cropped text images and extracts the collection of characters from them. The CRNN is firstly made up of a CNN which extracts feature maps from the images, a recurrent neural network (RNN) which provides a probability over the vocabulary for each time step, and a CTC decoder layer, sometimes called the transcript layer, to reconstruct the text.

The coordinate of the text regions found by the text detection module is used to input small crops of text from the original images to be fed into the CRNN. The cropped images represent the text to be extracted, in the size of 200 x 50 pixels, which are passed through a CNN consisting of two convolutional blocks. Each convolutional block consists of a 3x3 convolution operation and a 2x2 max pooling operation. When the cropped text images are passed through the first convolutional block, 32 feature maps are extracted and the size of the resulting images is halved to 100 pixels in width and 25 pixels in height. The resulting images are then passed through the second convolution block, extracting 64 feature maps and again reducing the width and height by half, that is 50 pixels in width and 12 pixels in height. The resulting images are reshaped in a sequential way such that it fits to the RNN. The images are further reshaped to 50 pixels in width and 1 pixel in height, with 768 feature maps.

In the next phase, the time steps are sequentially fed into the RNN which consists of two Bidirectional long short-term memory (LSTM) cells with 25% dropout to avoid overfitting. The two Bidirectional LSTMs are used to extract features along a receptive field/time steps. Through the first Bidirectional LSTM, 256 features are extracted for each of the 50-time steps. The resulting feature is then passed through another Bidirectional LSTM, extracting 128 features for each of the 50-time steps. With the features found in each time step, we need to get the probability that it corresponds to each label in the vocabulary by connecting a neural network consisting of 65 neurons to the output of the RNN. This network calculates the probabilities of the character in a time step that corresponds to each of the labels in the vocabulary using a softmax function. The use of 65 neurons results from 10 numbers (0 to 9), 26 lowercase alphabets (a-z), 26 uppercase alphabets (A-Z), 2 special characters (space “ ” and a hyphen “-”) and a blank character, used by the CTC decoder to reconstruct words. The CTC decoder layer uses the probabilities at each timestep to determine which character is most likely to be present in that timestep. Once it has predicted the characters for each 50-time steps, it goes through the process of grouping consecutive repeating characters and removing the blank characters to form a word or a sentence in our case a name, a date or an ID number. The resulting word or sentence is placed into a string which terminates the text recognition and extraction process.

2.5.1. Text recognition algorithm hyperparameters configuration

The hyperparameters configuration utilized for training the text detection algorithm is displayed in Table 2. Table 2 outlining the settings for the text recognition algorithm.

2.5.2. Dataset for text recognition algorithm training

To train the text recognition algorithm, another dataset has been created using the original raw data. Text regions of interest were manually cropped from each identity card image to create a set of six small images corresponding to the text value for each field of interest, that is surname, first name, surname at birth, gender, date of birth and identity number. Each small image is given as name, its own specific value and saved in the portable networks graphics (PNG) format. The name of each cropped image is used as a label in

the supervised learning approach. The dataset initially consisted of 600 samples of cropped text images. We further devised a data augmentation approach to artificially increase the amount of training data available to 1500 cropped text images. The existing cropped text images were trimmed to form new text images containing new words. For example, the original image containing the name “Jean Fabrice Laurent” was further cropped into five new image containing text “Jean”, “Laurent”, “Fabrice”, “Jean Fabrice” and “Fabrice Laurent”. Once the trained CRNN model recognises a text, it is compared with the file name and the difference was used to adjust the network parameters.

Table 2. Text recognition hyperparameters configuration

Hyper-parameters	Values
Training Batch Size	1
Number of Epochs	100
Validation Batch Size	1
Optimizer	Adam
Early Stopping Patience	10
Loss Function	CTC Loss

2.5.3. Evaluation metrics for text recognition algorithm

The evaluation metrics used for the text recognition module are the mean character accuracy (MCA) and the mean word accuracy (MWA) [22], based on the Levenshtein distance [23]. The evaluation of text recognition solutions is more than a match between the ground truth and the predicted output text. Three main types of errors can occur during the recognition process, namely the substitution error, where characters/words are replaced with other characters/words, deletion error, where characters/words are missing and the insertion error, where additional characters/words are included in the prediction.

$$\text{MCA} = 1 - \text{CER} \quad (7)$$

$$\text{CER} = \frac{S+D+I}{N} \quad (8)$$

$$\text{MWA} = 1 - \text{CWR} \quad (9)$$

$$\text{CWR} = \frac{S_w+D_w+I_w}{N_w} \quad (10)$$

2.6. Hardware setup

All experiments have been conducted on a computer having an i5-4210U CPU, 8 GB Ram and hardware acceleration being a Geforce 820M.

3. RESULTS AND DISCUSSION

3.1. Text detection results

The result obtained for the text detection module using the 215 images of the text detection dataset demonstrates an efficient prediction of the locations of the text of interest. As most of the images have been taken with the card holder’s mobile phone, there are a few degrees of noise, such as variation in lighting conditions or slight blurring, but the model has a mean IOU of 0.70 and pixel accuracy of 98%. The work of Wang *et al.* [24] for text detection on receipt images achieved an IOU of around 0.54. Therefore, our results have been considered satisfactory to link this module into the pipeline, and move on to the text recognition module.

3.2. Text recognition results

It was noted that our custom data augmentation approach, generating a combination of 1500 images, significantly contributed to make the text recognition model efficient. Data augmentation accounts for a performance increase of 9% for MCA and 13% for MWA, corresponding to an average increase of 11%. Combining individual results for the recognition of surname, first name, surname at birth, date of birth, gender and identity number, the model achieves MCA of 87% and MWA of 80%. However, the highest accuracy is achieved individually when recognising the text values for identity card number, date of birth and gender, with MCA and MWA of 97%.

Comparatively, no previous studies focused on the automated detection and recognition of text from the Mauritian identity card. Therefore, we compared our approach with the work done on the identity cards

from other countries. Hoai *et al.* [12] achieved the highest recognition accuracy of 96.7% and Viet *et al.* [25] achieved 91%. Our approach demonstrates a higher accuracy for the recognition of text-based fields on the identity card with a result of 97%.

3.3. Limitations

It has been observed that composite names with more than two words represents a problem for the text recognition model, with wrong predictions for at most two characters from clear images. However, for images which contain noise, although well identified by the text detection module, the text recognition model can make wrong predictions for more than two characters. We have noted that errors can go up to four characters in these situations. It is to be noted that we have not encountered first name values which are beyond 3 words in the dataset. Also, for one specific uppercase character “J”, the model predicts it as lowercase. Another important aspect for the text recognition model is its training and inference time. For our experiments the average training time for the model was appropriately 2 hours and each epoch took about 70 seconds. However, it is necessary to mention that the model was trained on a machine with relatively low performance and resources. Machines with more computing resources are available, and will contribute the training time. However, following our training, the average inference time to make a prediction was around two seconds. This was deemed acceptable for integration into a client onboarding system.

4. CONCLUSION

Using a three-phase approach with two deep learning models has proved to be efficient for detecting and extracting six specific text values from the Mauritian identity card. The first model is a custom-built deep learning model that has been trained to detect the text regions of interest and automatically crops them out of the identity card image, saving them individually. This model is quite effective with an IOU metric of 0.70 and a pixel accuracy of 98%. Our custom approach to data augmentation contributed to an overall increase of around 11% in the performance for text detection. The second model in our system consists of taking as input the detected text regions of interest as individual images, and recognising the characters in each, to build words out of them. The model achieved the highest character and word accuracy of 97%. While the combined approach proved effective for extracting the identity number, gender and date of birth we noted that improvements were required mainly for long composite names greater than two words. The next steps consist of defining a post-processing step to correct errors at character level prediction and integration into a web and mobile based client onboarding system.





REFERENCES

- [1] R. Clarke, “Human identification in information systems: Management challenges and public policy issues,” *Information Technology & People*, vol. 7, no. 4, pp. 6–37, Dec. 1994, doi: 10.1108/09593849410076799.
- [2] N. Tavakolian, A. Nazemi, and D. Fitzpatrick, “Real-time information retrieval from Identity cards,” *arXiv preprint arXiv:2003.12103*, 2020.
- [3] J. Harefa, Alexander, A. Chowanda, E. Haikal, Fedrick, and S. Antonio Wiranata, “ID card storage system using optical character recognition (OCR) on android-based smartphone,” in *Proceedings - IEIT 2022: 2022 International Conference on Electrical and Information Technology*, Sep. 2022, pp. 75–79, doi: 10.1109/IEIT56384.2022.9967874.
- [4] B. J. Bipin Nair, S. Unni Govind, and M. Jose, “Multiple object recognition from smart document images using YOLOv5s,” in *Proceedings - 7th International Conference on Computing Methodologies and Communication, ICCMC 2023*, Feb. 2023, pp. 824–828, doi: 10.1109/ICCMC56507.2023.10084220.
- [5] Y. Q. Li, H. Sen Chang, and D. T. Lin, “Large-scale printed chinese character recognition for ID cards using deep learning and few samples transfer learning,” *Applied Sciences (Switzerland)*, vol. 12, no. 2, p. 907, Jan. 2022, doi: 10.3390/app12020907.
- [6] Microsoft, “OCR - optical character recognition,” 2023.
- [7] J. Zhu, H. Ma, J. Feng, and L. Dai, “ID card number detection algorithm based on convolutional neural network,” in *AIP Conference Proceedings*, 2018, vol. 1955, doi: 10.1063/1.5033788.
- [8] Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner, “Gradient-based learning applied to document recognition,” *Proceedings of the IEEE*, vol. 86, no. 11, pp. 2278–2323, 1998, doi: 10.1109/5.726791.
- [9] M. O. Pratama, W. Satyawan, B. Fajar, R. Fikri, and H. Hamzah, “Indonesian ID card recognition using convolutional neural networks,” in *International Conference on Electrical Engineering, Computer Science and Informatics (EECSI)*, Oct. 2018, vol. 2018-October, pp. 178–181, doi: 10.1109/EECSI.2018.8752769.
- [10] J. Ge, Z. Fang, and Q. Tao, “Bank card number recognition based on deep learning,” in *Proceedings of 2020 IEEE 4th Information Technology, Networking, Electronic and Automation Control Conference, ITNEC 2020*, Jun. 2020, pp. 863–867, doi: 10.1109/ITNEC48623.2020.9085084.
- [11] J. Redmon and A. Farhadi, “YOLOv3: An incremental improvement,” *arXiv preprint arXiv:1804.02767*, 2018.
- [12] D. P. Van Hoai, H. T. Duong, and V. T. Hoang, “Text recognition for Vietnamese identity card based on deep features network,” *International Journal on Document Analysis and Recognition*, vol. 24, no. 1–2, pp. 123–131, Feb. 2021, doi: 10.1007/s10032-021-00363-7.
- [13] C. Szegedy *et al.*, “Going deeper with convolutions,” in *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, Jun. 2015, vol. 07-12-June-2015, pp. 1–9, doi: 10.1109/CVPR.2015.7298594.
- [14] “Government of Mauritius Web Portal.” <https://mnis.govmu.org/Pages/New ID Card/Card-Layout-and-Design.aspx>.





- [15] O. Ronneberger, P. Fischer, and T. Brox, "U-net: convolutional networks for biomedical image segmentation," in *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, vol. 9351, Springer International Publishing, 2015, pp. 234–241.
- [16] N. Otsu, "Threshold selection method from gray-level histograms," *IEEE Trans Syst Man Cybern*, vol. SMC-9, no. 1, pp. 62–66, Jan. 1979, doi: 10.1109/tsmc.1979.4310076.
- [17] A. Breheret, "Pixel annotation tool," 2017, [Online]. Available: <https://github.com/abreheret/PixelAnnotationTool>.
- [18] H. Rezatofighi, N. Tsoi, J. Gwak, A. Sadeghian, I. Reid, and S. Savarese, "Generalized intersection over union: A metric and a loss for bounding box regression," in *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, Jun. 2019, vol. 2019-June, pp. 658–666, doi: 10.1109/CVPR.2019.00075.
- [19] D. Müller, I. Soto-Rey, and F. Kramer, "Towards a guideline for evaluation metrics in medical image segmentation," *BMC Research Notes*, vol. 15, no. 1, Jun. 2022, doi: 10.1186/s13104-022-06096-y.
- [20] S. Jadon, "A survey of loss functions for semantic segmentation," Oct. 2020, doi: 10.1109/CIBCB48159.2020.9277638.
- [21] B. Shi, X. Bai, and C. Yao, "An end-to-end trainable neural network for image-based dequence recognition and its application to scene text recognition," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 39, no. 11, pp. 2298–2304, Nov. 2017, doi: 10.1109/TPAMI.2016.2646371.
- [22] C. Clausner, S. Pletschacher, and A. Antonacopoulos, "Flexible character accuracy measure for reading-order-independent evaluation," *Pattern Recognition Letters*, vol. 131, pp. 390–397, Mar. 2020, doi: 10.1016/j.patrec.2020.02.003.
- [23] K. Leung, "Evaluate OCR output quality with character error rate (CER) and word error rate (WER)," *In Towards Data Science*, 2021. <https://towardsdatascience.com/evaluating-ocr-output-quality-with-character-error-rate-cer-and-word-error-rate-wer-853175297510>.
- [24] X. Wang, X. Zhang, S. Lei, and H. Deng, "A method of text detection and recognition from receipt images based on CRAFT and CRNN," *Journal of Physics: Conference Series*, vol. 1518, no. 1, p. 12053, Apr. 2020, doi: 10.1088/1742-6596/1518/1/012053.
- [25] H. T. Viet, Q. Hieu Dang, and T. A. Vu, "A robust end-to-end information extraction system for vietnamese identity cards," in *Proceedings - 2019 6th NAFOSTED Conference on Information and Computer Science, NICS 2019*, Dec. 2019, pp. 483–488, doi: 10.1109/NICS48868.2019.9023853.

BIOGRAPHIES OF AUTHORS



Geerish Suddul     received his Ph.D. from the University of Technology, Mauritius (UTM). He is currently a Senior Lecturer at the UTM, in the Department of Business Informatics and Software Engineering under the School of Innovative Technologies and Engineering. He has been actively involved in research and teaching since 2005, and currently his research work focuses on different aspects of machine learning such as computer vision and natural language processing. He can be contacted at e-mail: g.suddul@utm.ac.mu.



Jean Fabrice Laurent Seguin     received a BEng Electronic Engineering and MSc in Artificial Intelligence with Machine Learning from the University of Technology, Mauritius (UTM). He has more than 5 years experience in the electronic broadcasting sector. He has been actively involved in research since 2021 focusing on machine learning problems in the field of computer vision and natural language processing. He can be contacted at e-mail: jseguin@umail.utm.ac.mu.