Automated Bank Cheque Verification Using Image Processing and Deep Learning Methods

**Prateek Agrawal · Deepak Chaudhary ·
Vishu Madaan · Anatoliy Zabrovskiy ·
Radu Prodan · Dragi Kimovski ·
Christian Timmerer**

Abstract Automated bank cheque verification using image processing is an attempt to complement the present cheque truncation system, as well as to provide an alternate methodology for the processing of bank cheques with minimal human intervention. When it comes to the clearance of the bank cheques and monetary transactions, this should not only be reliable and robust but also save time which is one of the major factor for the countries having large population. In order to perform the task of cheque verification, we developed a tool which acquires the cheque leaflet key components, essential for the

Prateek Agrawal
University of Klagenfurt, Austria
Lovely Professional University, India
E-mail: prateek061186@gmail.com

Deepak Chaudhary
Lovely Professional University, India
E-mail: deepak3.14159@gmail.com

Vishu Madaan
Lovely Professional University, India
E-mail: vishumadaan123@gmail.com

Anatoliy Zabrovskiy
University of Klagenfurt, Austria
Petrozavodsk State University, Russia
E-mail: anatoliy.zabrovskiy@aau.at

Radu Prodan
University of Klagenfurt, Austria
E-mail: radu.prodan@aau.at

Dragi Kimovski
University of Klagenfurt, Austria
E-mail: dragi.kimovski@aau.at

Christian Timmerer
University of Klagenfurt, Austria
Bitmovin Inc., CA, USA
E-mail: christian.timmerer@aau.at

task of cheque clearance using image processing and deep learning methods. These components include the bank branch code, cheque number, legal as well as courtesy amount, account number, and signature patterns. our innovation aims at benefiting the banking system by re-innovating the other competent cheque-based monetary transaction system which requires automated system intervention. For this research, we used institute of development and research in banking technology (IDRBT) cheque dataset and deep learning based convolutional neural networks (CNN) which gave us an accuracy of 99.14% for handwritten numeric character recognition. It resulted in improved accuracy and precise assessment of the handwritten components of bank cheque. For machine printed script, we used MATLAB in-built OCR method and the accuracy achieved is satisfactory (97.7%) also for verification of Signature we have used Scale Invariant Feature Transform (SIFT) for extraction of features and Support Vector Machine (SVM) as classifier, the accuracy achieved for signature verification is 98.10%.

1 Introduction

Cheques have an intriguing past, as evidently, it is still a mystery as to when exactly the concept of cheque has emerged. Speculations were made that the idea in itself was entertained during the Roman empire, however the very idea could not catch on. Thus, its usage came in the business hubs in the 1500s in regions near Holland and Amsterdam because of their richness in business opportunities during that time. Therefore, with its gradual acceptance in the commerce and enhanced usability it was slowly accepted as a means of monetary transaction. The word "cheque" was evidently originated in the 1700s as a common practice for monetary transactions, certain unique numbers were written on the piece of papers which were checked to keep track of and to proceed with the transaction. Financial transactions using cheques is still one of the widely accepted way across the globe. Almost in every continent, banking institutions have provision for their customers to use cheques for any financial transaction. In 2018, people in United Kingdom used 346 million cheques for payments and to acquire cash with totaling a value of £442 billion [1]. Few sample cheques of different countries (Malaysia, Maldives, USA, Kenya, UAE, and China etc.) are shown for reference purpose in Fig. 1:

Fig. 1 Sample cheque images of different countries [2] [3] [4].

Although cheque systems in banks for transactions is one of the most powerful tool, primarily due to its slow processing time, considerable complexities in cheque clearing process and gradually with an immense growth in the technology, the electronic-transactions have created a big shadow on the cheque system to cater the need of the banking sector. Therefore, we have proposed a mechanism to make the entire process of bank cheque verification not only fast but also reliable by using deep learning methods and image processing techniques. Due to various technical constraints, it was not feasible for the banking sector to implement the automation of the bank cheque clearing which required sophisticated measures for authentication and verification for the transfer of
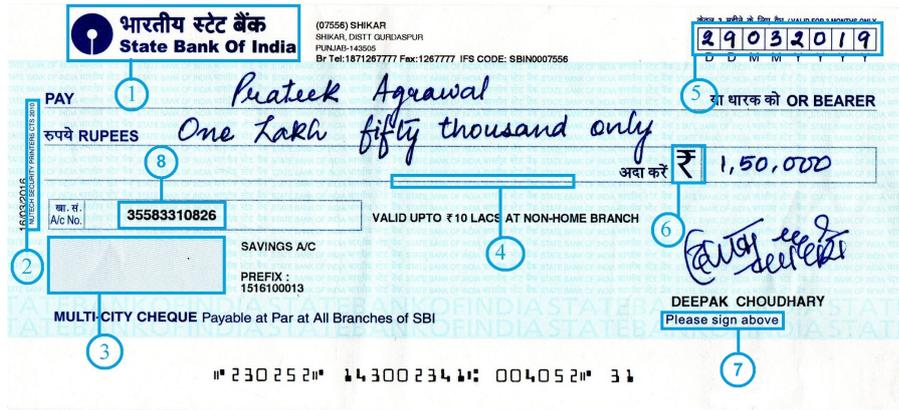
Fig. 2 CTS-2010 compliant sample bank cheque

funds from accounts. Therefore, as a use case example, we have implemented and tested our proposed model on cheque samples of different Indian banks which follow the standard guidelines of the Reserve Bank of India (RBI). Before Jan 2019, the security features used in cheques at the country level by all Indian banks were lacking technical and security edge such as non-availability of void pantograph, Rupee symbol, micro-lettering, bank logo in Ultra Violet (UV) ink etc. However taking a major step in modernization of cheque based transactions, RBI made it mandatory to use the CTS-2010 cheque format as a standard format for all regulated banks [5]. There are certain mandatory security features in the CTS-2010 compliant cheques as shown in Fig 2 and are mentioned below:

① Bank Logo in UV ink
② Printer name with CTS-2010
③ Void Pantograph
④ Micro-lettering
⑤ Date
⑥ Rupee Symbol
⑦ Signature
⑧ Account Number

Cheque Truncation System (CTS) is a way to authenticate the cheque clearing process. The process flow of CTS is shown in Fig. 3. There are two steps authentication of any bank cheque leaflet / form before making any financial transaction using cheques. First step is to validate whether the leaflet used, is genuine or fake. Deepak at al (2019) [6] proposed some methods to validate the bank cheque leaflet. Second step to authenticate the cheque is to verify about the details filled on that leaflet. In our research work, we focused on proposing some novel methods to automate the verification of the bank cheques. For this, we extracted the key information from the cheque leaflet which were required for verification of bank cheque and represented

these features in the highlighted using bounding boxes. There are various techniques already available for the identification, authentication and verification of the bank cheques as shown in the research works presented by Dhanva et al (2018) [7], Singh and Tigga (2012) [8], Gonge and Ghatol (2014) [9], Gonge and Ghatol (2016) [10], Menon et al (2017) [11]. Unfortunately, the efficiency of the works are not up to mark. For the task of verification, we considered matching of Indian Financial System Code (IFSC), cheque number, account number, legal amount, courtesy amount and signature of cheque issuer as input components.
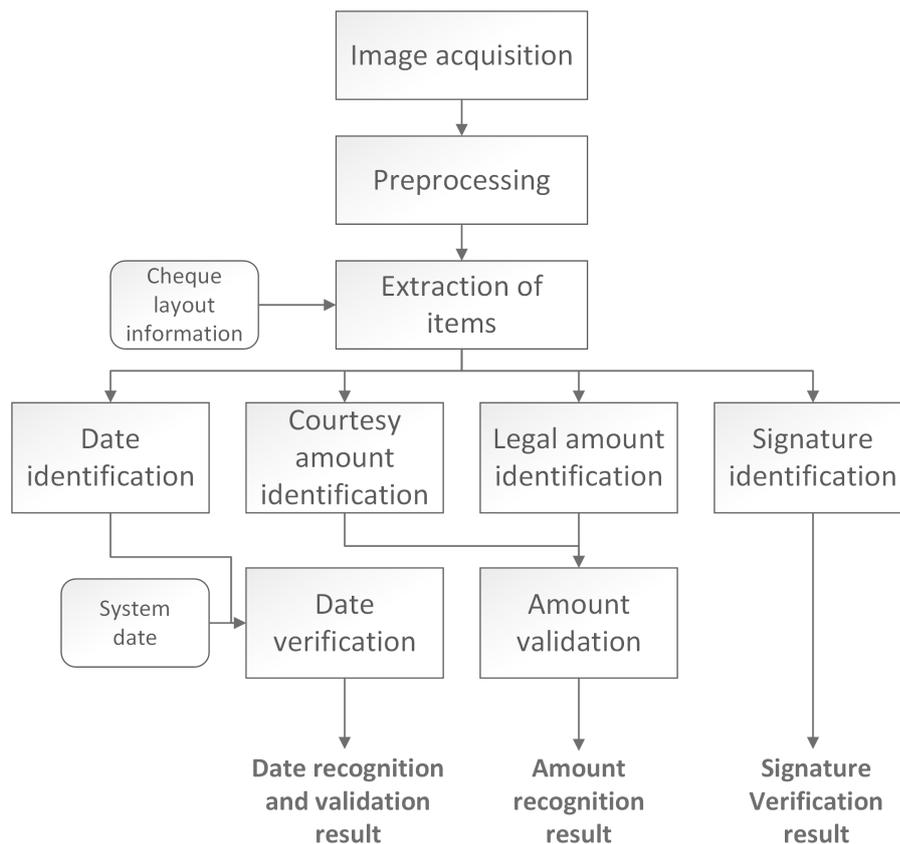
Fig. 3 Process flow diagram of proposed work.

Optical Character Recognition (OCR) is an automated text reading process from images that includes machine printed (typographic) or handwritten character recognition and numeral recognition. Image processing is a process of analysing a digitised image for performing operations such as enhancement, compression, segmentation, or editing. The fundamental stages of image processing which almost all the image has to go through include the following

components which are sequential in nature: acquiring the image, pre-processing of the image, segmenting the image, interpreting the image, and recognising the image. There are many ways to interpret any image. One effective way is using convolution neural networks (CNN). The deep learning approach is an advancement of machine learning with multiple layers allowing it to achieve higher level of accuracy for pattern recognition. However, it demands a large number of datasets for the desired training efficiency of the network. If we have a enough large database, its implementation will assist in precise estimation of the handwritten characters so as to give accurate algorithmic implementation ability for the bank cheques. This may benefit the banking sector by reducing the processing time and saving resources by automating the task of cheque clearance. We implemented our research work using MATLAB-2018a.

The major contribution of our research was to identify the various possible techniques to verify the bank cheque and then apply the best selected image processing and deep learning methods to improve the performance of automated verification in terms of accuracy. After that, we compared our results with existing work and showed the comparative analysis on how our innovation covers all primary aspects to verify a bank cheque in a more efficient and effective way in terms of accuracy. We describe the major contribution of our innovation in the following steps:

- Captured the following key components of a scanned bank cheque image using image segmentation: Date, Account Number, Cheque Number, Signature of account holder, Courtesy amount detail, legal amount detail, associated bank information (IFS Code)
- Validated bank branch information, genuineness of the cheque and customer account number by applying OCR method on IFSC segment, cheque number segment and account number segment respectively.
- Obtained the pattern of signature and match it with the specimen signatures of account holder using SIFT and SVM methods.
- Recognised the characters stated in the legal amount segment of the cheque using CNN method and convert it into the place value system.
- Recognised the digits stated in the courtesy amount segment of the cheque using CNN and verify it with the extracted and converted legal amount information using the proposed algorithm.

This paper is organized as follows: Section 2 summarises the related work. Section 3 presents the complete methodology and proposed algorithm along with the models for verification of bank cheques. Section 4 presents the experimental results and analysis of our implemented model. Finally, section 5 concludes the research work and outlines future work.

2 Related Work

In order to perform the task of getting a logical response for the input of an image, it requires a meaningful interpretation of the characters present in the

image and, thus, we need to perform the character recognition for the various patterns present in an image. Singh and Sachan (2018) [12], Wankhede and Mohod (2017) [13], and Ramanathan et al. (2009) [14] presented their work to retrieve the characters from the image Optical Character Recognition (OCR). The segmentation of digits for which there are several techniques devised is the major challenge faced in the recognition of characters. Chen (2017) [15] proposed a general approach for handwritten digit segmentation using spectral clustering and achieved an accuracy of over 97%.

One of the earliest researches on font recognition was done by Dickinson (1960) [16], Liu and Shelton (1966) [17], Nagy and Shelton (1966) [18], Wang et al. (1972) [19], Zramdini and Ingold (1998) [20] in which they suggested a statistical approach for the Latin Language font recognition at the text line level depending on the global typographical features. They were able to achieve approximately 97% of recognition rate. For English language also, due to its widespread usage and commercial usability, Zhu and Wang (2002) [21], Dhande and Kharat (2017a) [22], Pal et al. (2006) [23], Dhande and Kharat (2017b) [24], and Saenthon and Sukkhadamrongrak (2014) [25] proposed few approaches for character recognition. Font recognition was done by using a wide array of techniques and they were able to achieve a satisfactory accuracy on text blocks.

Similarly, Ramanathan and Sharma (2017) [26] devised a method using Support Vector Machine (SVM) for classification and feature extraction by proposing a SVM-based multi-knowledge-based system, achieving an accuracy rate of 93.54%, which marked the continued focus on the OCR or segmentation for the character recognition in image processing methods. However, Stewart et al. (2018) [27] proposed a segmentation free recognition techniques in which they presented that for small training datasets, it is possible to manipulate either manually or automatically that segmentation and stitching improves handwriting recognition on dataset with few samples. In another attempt by Peres et al. (2018) [28] in 2018 proposed a novel handwritten methodology in which they used a hybrid CNN and SVM classifier with which they were able to achieve an overall accuracy of 96.7% using the MNIST dataset [29].

For intelligent character recognition to create and match dynamic templates as well as the conversion of characters into text, Kajale et al. (2017) [30] implemented the supervised machine learning methodology and achieved the accuracy close to 95% for the alpha numerals and special characters. The presence of the noise in the form of lines, watermarks, diagrams, patterns or all of these at the background is one of the important factors to create the performance variation in different character recognition techniques. Goto and Aso (2002) [31], Brisinello and Gibric (2018) [32], Brisinello et al. (2017) [33], and Feng and Gao (2010) [34] performed research involving extraction of the information from the colored or noisy background with satisfactory level of output. Contrary to the notion that output will drop significantly if there is a change in the background with varying levels of color change in the texts or introduction of noise in the form of patterns and figures. Jia and Wei (2010) [35] extracted the legal amount from the Chinese bank cheques and proposed a

multiple channel binarization approach for extracting legal amount from color Chinese bank check images with a combination of global and local threshold. Sneha Singh et al. (2015) [36] proposed a method to recognise the legal amount of Indian bank cheques using correlation coefficient method. They achieved an average 76.4% word recognition accuracy on a self created dataset of 19215 word samples of 61 different words. Another researchers M Badrul Alam et al. (2015) [37] developed a tool to recognise the handwritten legal amount of Bangladesh bank cheques using ANN and achieved only 93.4% accuracy. Similarly, Jayadevan et al. (2010) [38] proposed a lexicon-driven segmentation recognition technique for the recognition of the legal amount words from the Indian bank checks written in English using Modified Quadratic Discriminant Function (MQDF) and claimed 97.20% accuracy.

Raghevendra and Danti (2015) [39] proposed a method to identify the bank details associated with the cheques using bank after removing the background noise. Researchers used the invariant geometrical features for accomplishing the recognition of the checks with 93.8% of accuracy only. To improve the security for verifying the cheque, another important feature of it is Magnetic Ink Character Recognition (MICR) code. Even though if we are able to perform the pattern recognition as well as the legal amount recognition using the above stated techniques one more component of a bank check that needs to be properly identified for the sake or security as well as determining the validity of the check is the Magnetic Ink Character Recognition (MICR), Shirai et al. (2012) [40] proposed a three pronged solution to the problem starting with the character color estimation and conversion into binary form to remove the background, to separate each character using the character pitch estimation and the removal of the unwanted noise & extraction of the required MICR code using the sign fragments removal by the stroke tracing.

Lu et al. (2017) [41] proposed techniques to deal with noisy images as the image that we have, may be a copy of the original image with some considerable noise such as shadow along with a spatial noise due to improper copying of this image. Researchers used local adaptive threshold algorithm in order to transform the gray-scale image into the binary image to capture the contours of the texts in the image and applied a double-filtering algorithm to remove the noise between the texts and the characters. Wiraatmaja et al. (2017) [42] proposed a deep convolutional de-noising auto-encoder and claimed the improvement in accuracy by 26.78% as compared to a simple OCR in average accuracy testing. Void Pantograph is another important feature which ensures whether an original image of the cheque was taken for verification or not. It refers to the process of creating a pantograph which is otherwise invisible to naked eyes on the original print. However, when copied with a printer, it is clearly visible making it a very important security features for security printing toolkit. Aronoff et al. (2011) [43] formulated a method to automate the optimization of void pantograph settings. Peymani and Soryani (2017) [44] proposed a model to identify machine generated as well as hand written characters for Farsi language images with an accuracy of 68% using an Arabic language dataset [45]. Bhadwal (2019) [46], Singh (2018) [47] and Agrawal

(2012) [48] also developed different applications ranging from object recognition to edge detection by using image processing and computer vision methods. Ignat and Aciobniei (2016) [49] proposed method of feature extraction for offline handwritten digit recognition by using rotation and filtering of edges for the extraction of the digit characteristics. KNN and SVM classifiers were used for classifying the features and claimed to achieve the accuracy about 99% on the MNIST dataset.

Hou et al. (2017) [50] implemented a model using convolutional neural Network for learning the handwritten numeric characters and perform scratchy or non-scratchy decisions with error rate of 0.40% on MNIST database. Li et al. (2018) [51] used grab cut segmentation based on image region by considering image texture and boundary information which helps them to perform segmentation with minimal user interaction. Mityakov et al. (2017) [52] in application of modern architectures of deep neural networks for solving practical problems explained the advent of sophisticated networks like GoogleNet [53] capable of automatic feature extraction and classification. Evidently watershed segmentation is one of the most widely used algorithm in terms of achieving feasible results however in cases of adhesion among components of the image, it becomes difficult to obtain remarkable results. Ji et al. (2015) [54] proposed an improved watershed algorithm based on distance transform for cell image segmentation.

Shenoy et al. (2016) [55] proposed a performance based string matching algorithm for Nvidia GPUs in which they are using Quick Search, Horspool and Brute Force string matching approaches allowing a performance boost. Whereas, Joshi et al. (2017) [56] proposed modified Rabin-Karp method by parallelizing conventional Rabin-Karp algorithm. Zhou and Gao (2017) [57] used double hash in order to perform multi-pattern matching and are able to reduce the number of comparison by two hashes thus giving a slight performance boost. Consequently all these approaches worked towards reducing the space-time complexity and improving the performance for string matching. Dhande & Kharat [22] [24] proposed a method to perform pattern recognition using vertical and horizontal projection methods along with convex hull algorithm for feature extraction for cursive English handwritten characters and implemented support vector machine (SVM) on CCC dataset [58]. Similarly, Dash et al. (2015) [59] proposed a perimetric complexity based on gestalt principle to develop a handwritten numeral recognition system for obtaining a point-wise correspondence from the matching of shape context based on affine transformation on a dataset of Odia numerals. Li et al. (2010) [60], Shen et al. (2012) [61], Shirazi et al. (2017) [62], Kalaiyarasi and Karthikeyan (2014) [63], Pornpanomchai et al. (2015) [64], Sahbi et al. (2013) [65], and Kaur (2016) [66] proposed various approaches to recognize of logos present on the cheque leaflets for identifying the issuing bank of the bank cheque.Few remarkable signature recognition methods are highlighted as shown in table 1:

Table 1 Methods of Signature Recognition.

Authors	Techniques	Datasets	Efficiency
Naseer and Dogru (2017) [67]	SVM classifier based on RBF Kernel using SIFT	CASIA dataset	98.75%
	SVM classifier based on RBF Kernel using SURF	CASIA dataset	96.25%
Ferrer et al. (2018) [68]	HMM	OnOffSignHindi-75	96.78%
	SVM	OnOffSignHindi-75	99.69%
Mondal and Kundu (2017) [69]	Probabilistic Divergence Measure- Kullback-Leiber Divergence	Self-Created	96.50%
Jadhav and Chavan (2018) [70]	Local Binary Pattern Features using K-Nearest Neighbor algorithm	Bangla Offline Signature Dataset	75.34%
	Local Binary Pattern Features using Support Vector Machine algorithm	Bangla Offline Signature Dataset	90.36%
Jain and Patil (2014) [71]	A comparative study of various methods for offline signature verification	N/A	N/A
Handhyani et al (2017) [72]	GLCM using Bootstrap	Self-Created	100%
Morocho et al (2017b) [73]	Local Directional Patterns using Support Vector Machines	Biosecure ID Corpus	96.10%
Beresneva et al (2018) [74]	Discrete Wavelet Transform using SVM	Self-Created	99.40%
	Discrete Wavelet Transform using KNN	Self-Created	98.44%
	Discrete Wavelet Transform using Random Forest	Self-Created	99.41%
Morocho et al (2017a) [75]	Histogram of Templates using Artificial Immune Recognition System	BioSecureID Dataset	95.28%
Boudamous et al (2017) [76]; Gonge and Ghatol (2014, 2016) [9] [10]	Crowdsourcing and Human-assisted Schemes	Benchmark Dataset	99.6%
Karanjkar and Vasambekar (2016) [77]	Artificial Neural Networks	Self-Created	85%
Hafemann et al. (2019) [78]	CNN based model and handcrafted feature extractor (CLBP) SVM : Linear, RBF	MCYT-75, CEDAR, GPDS-160 and the Brazilian PUC-PR	N/A
Hafemann et al. (2020) [79]	Model Agnostic Meta Learning (MAML)	MCYT-75, CEDAR, GPDS-160 and the Brazilian PUC-PR	94.84%

If we summarise the state of the art, many researchers worked on image segmentation, character recognition and signature recognition separately and also achieved remarkable performance. Few researchers considered bank cheque validation as their research application but none of the researcher proposed any complete solution to successfully implementing the automated cheque clear-

ance process without any human intervention. While in our research, we proposed the solution starting from pre-processing of scanned cheque image to verification of cheque amount and signature. In further section, we described the entire methodology to implement the model and presented the results which are found satisfactory.

3 Methodology

The bank cheque verification tool required fulfillment of some key stages in sequential manner. At first, the tool ensured the genuineness of IFSC present on cheque then it verified the cheque number to ensure whether the cheque was from the set of cheque leaflets assigned to the account holder or not.

After that, the amount was verified by checking whether the customer was having sufficient amount in his account for which he released the cheque, and finally the signature of the cheque issuer(s) were verified. Only after the completion of these necessary verification steps, the cheque clearance was passed for transfer or withdrawal of funds.

Table 2 Bank cheque verification key features and their description.

Key feature	Description
IFS code	- Indian financial system code. - It is an eleven character code where first four characters represent bank name and last six characters represents the branch. Fifth character is kept reserved for future use.
Account number	- A unique number assigned to the user of the bank services, used for monetary transaction related services.
Cheque number	- A six digit number on the bottom left corner of a cheque, it helps to identify a cheque and also verify if the cheque is from the list of issued cheques to the user.
Legal amount	- The amount mentioned on the cheque leaflet in handwritten / printed words.
Courtesy amount	- The amount mentioned on the cheque leaflet in handwritten / printed numeric digits.
Signature	- Signature of the account holder. - Every person has a unique signature which depends on the static or dynamic features and it is hard to copy it. - It is used as a key feature in authenticating a user's identity.

The entire process of cheque clearance happened in a sequential manner and each step was mandatory for the next step. All the key features presented in the cheque verification are shown with their description in Table 2.

Different methods were used for extracting such information from the cheque leaflet that depended on the viability and level of accuracy. For exam-

ple, to extract information from the machine typographic script, we use OCR method as it tends to produce results with higher level of accuracy whereas for hand-written words and numerical values, we can use deep learning based CNN. Similarly, for authenticating the signature(s), we extracted the features using SIFT algorithm and for classification, SVM produced optimum results. To implement these methods, we required noise free image therefore the better idea was to segment the image so as to acquire only the relevant information.

Fig. 4 shows the broad steps using a flow-chart, involved in our research process of cheque verification. These steps encompassed certain sub-steps each of which were essential to meet the standards so as to extract the required information from the bank cheque. The entire methodology for bank cheque verification is summarized in Fig. 5.
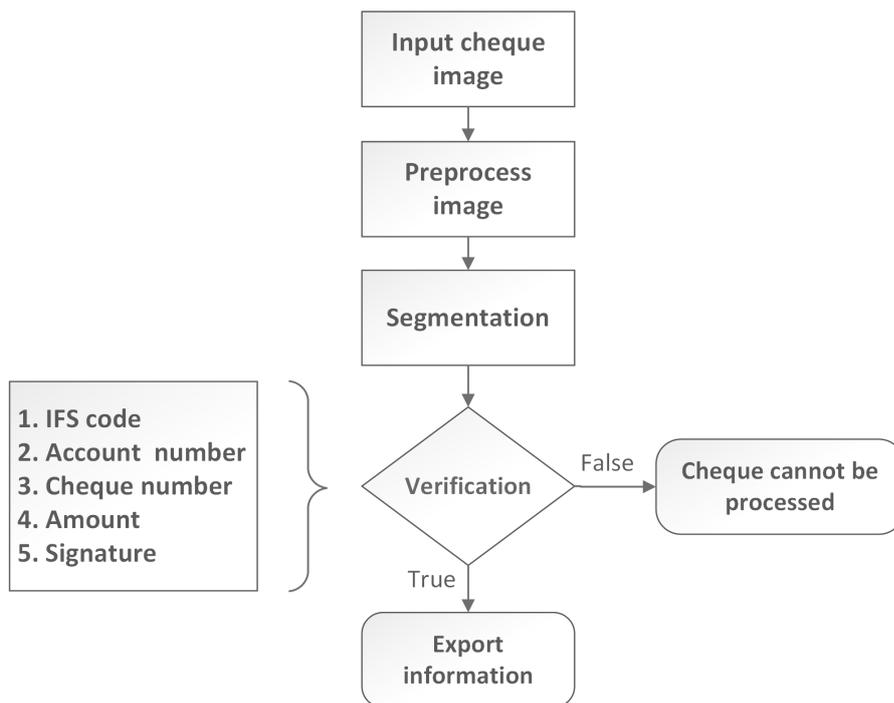

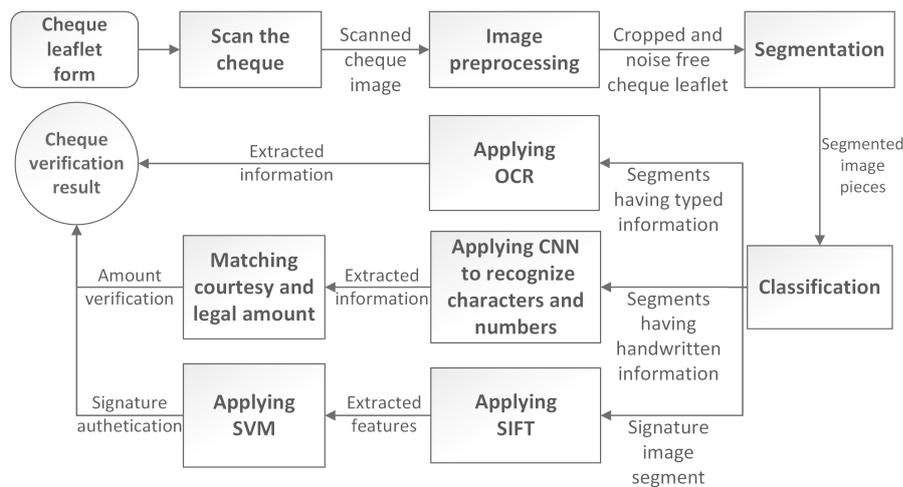**Fig. 4** Flow Chart for Bank Cheque Verification.

Fig. 5 Block diagram of bank cheque verification process

3.1 Image Acquisition

The image acquisition of a bank cheque is crucial for the CTS. Generally, flatbed scanners are used to acquire such images. Due to orientation and irregularities presented in the scanned image(s), we were unable to use the acquired image(s) directly for the image processing operations therefore it required some pre-processing steps.
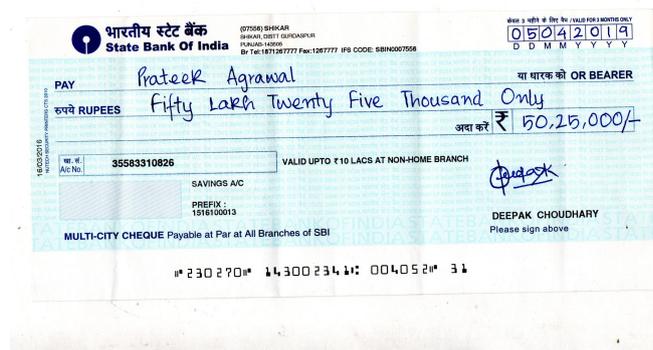

Fig. 6 Raw scanned image of a sample cheque.

3.2 Image Pre-processing

We used the cheque scanned images for our research. A sample scanned image is shown in Fig. 6. As a scanned image obtained from the scanner can not be used directly thus we needed to pre-process it, which involved two primary operations, i.e, rotation and removal of unnecessary background information. In this step, we rotated the scanned image with respect to the 'Date Box' (which is a common feature presented at the same part of every bank cheque) and then removed the background noise and extra information. Efficiency of the parameter identification considerably improves with removal of extra background information.

3.2.1 Rotation

As the scanned images may vary in terms of orientation therefore we used the date box present in all standard cheque leaflets and the relative invariant nature of the position was utilized. In order to perform the rotation of image, we determined the point of rotation and degree of rotation. The primary component essential for rotation to work was contour extraction, as we were able to determine the position values of date box by using it which used as our anchor for any set of operation related to length mapping. The process of contour extraction was explained in the forthcoming segmentation sub-section. Also, in order to perform the rotation, we used the midpoint of the image as the rotation point, and we used date box in order to determine the angle necessary for the rotation. Fig. 7 shows the same process.
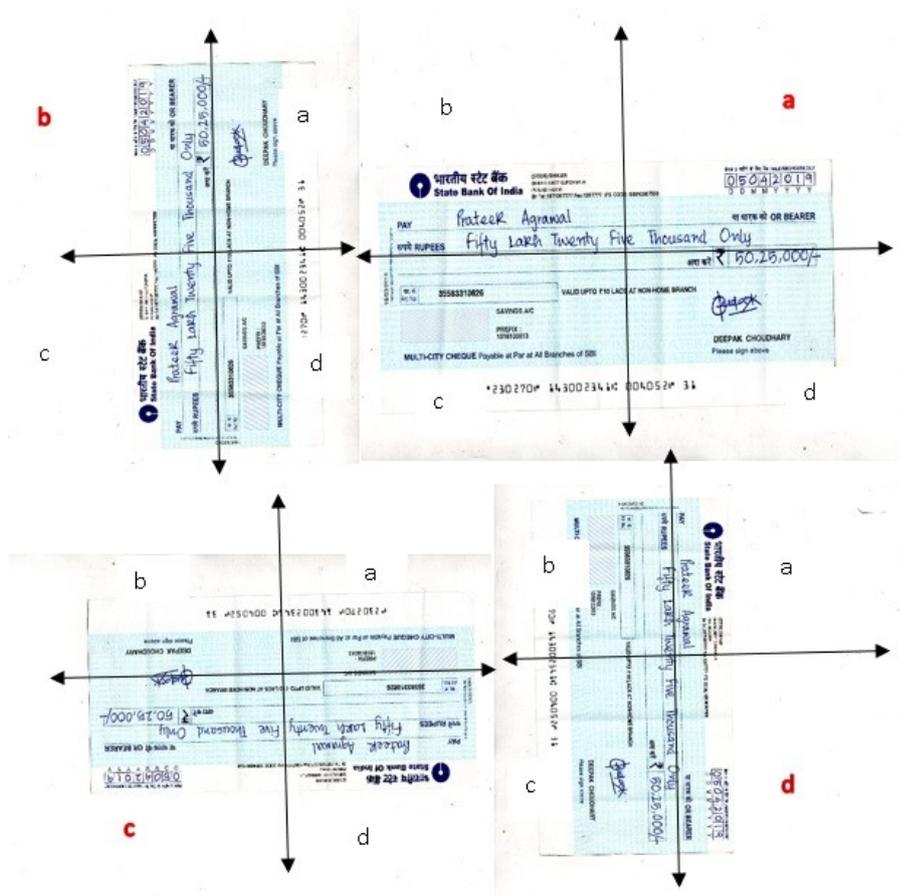
Fig. 7 Possible scenarios for rotation of cheque leaflet.

As shown in the figure among the possibilities for rotation, we represented each quadrant with a, b, c, d and highlighted the quadrant consisting the contour of date box with red color. We performed rotation by 90°, 180°and 270°depending upon the quadrant in which the date box is present. In Fig 7, the scenario the order of rotation for possibilities was 270°, 180°and 90°. Once the cheque orientation became correct, we proceed with the removal of unnecessary region available in the scanned cheque leaflet.

3.2.2 Removing Background Noise

There was lot of extra information in the scanned image which was to be removed. For the task of removing of background information, we used the date box present in the standard cheque template. The cheque dimensions of any bank in a country are standard and are followed by the norms decided by the government financial body. In India, RBI monitors and controls the
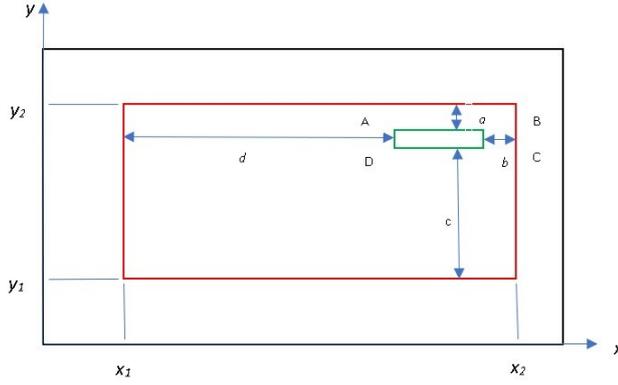
functioning of all Indian banks. A sample cheque leaf with it's standard dimensions regulated and decided by RBI, is shown in Fig. 8 [80]. All Indian banks are bound to follow these standards and geometric dimensions strictly. These geometric dimensions may vary for banks of different countries and can easily be calculated as per the respective country agency standards. Using the date box coordinates range and with the implementation of length mapping between the date box and the whole cheque image, we determined the range of the cheque as per the schematic representation of edge removal as mentioned in Fig. 9. In reference to data box, we extracted the regions of interest for courtesy amount, legal amount, signatures etc.

Fig. 8 CTS-2010 cheque form specification as per RBI guidelines [80].

Fig. 9 Schematic representation for edge removal.

In the schematic representation, the black rectangle represents the scanned image of the bank cheque whereas, the red rectangle represents the edge of the cheque leaflet and the green rectangle represents the edge of the date box. The purpose of the edge removal task was to remove the region between the red and black rectangle and to retain the region in the red rectangle only. We used following mathematical equation for length mapping of bank cheque leaflet as represented below in Eq. (1).

$$
\begin{cases}
\frac{AD}{(AD+a)} = \frac{(y_A - y_D)}{y_2 - y_D} \\
\frac{AD}{(AD+c)} = \frac{(y_A - y_D)}{y_A - y_1} \\
\frac{AB}{(AB+d)} = \frac{(x_B - x_A)}{x_B - x_1} \\
\frac{AB}{(AB+b)} = \frac{(x_B - x_A)}{x_2 - x_A}
\end{cases}
\tag{1}
$$

Further the equation is simplified as Eq. (2).

$$
\begin{cases}
y_2 = \left(\frac{((y_A - y_D) \times (AD+a))}{AD} + y_D \right) \\
y_1 = \left(y_A - \frac{((y_A - y_D) \times (AD+c))}{AD} \right) \\
x_1 = \left(x_B - \frac{((x_B - x_A) \times (AB+d))}{AB} \right) \\
x_2 = \left(\frac{((x_B - x_A) \times (AB+b))}{AB} + x_A \right)
\end{cases}
\tag{2}
$$

The parameters involved in the equations are explained in the Fig. 9. Here, dimension AB and AD are representing the length, breadth of the date box. Whereas, a, b, c, d are the distance from the edges of the cheque leaflet. Also, x and y are the horizontal and vertical coordinates of the date box. The key parameters of these equations are the boundary of cheque leaflet which is the coordinate range for unnecessary background information removal.

3.2.3 Grayscale Operation

After removing the background noise of an *RGB* (a three channel) image, we converted it into grayscale (a single channel) image. The mathematical formula for this conversion is represented in Eq. (3) [81].

$$Gray\ Image = 0.2126 \times R + 0.7151 \times G + 0.0721 \times B \tag{3}$$

Here *RGB* represents the *Red*, *Green* and *Blue* channels of the image and *Gray Image* represents the pixel value of the grayscale image.

3.2.4 Gaussian Filtering

After converting the image into grayscale format, next pre-processing step was to remove the noise. For this, we applied Gaussian de-noising method which is used in a linear form of smoothing filter for a signal. This method of filtering is a form of convolution between the normal arrangement of values and the image. The formula for 2-D Gaussian function is explained in Eq. (4).

$$G(x, y) = (\frac{1}{2\pi\sigma^2}) \times e^{\frac{-(x^2+y^2)}{2\sigma^2}} \tag{4}$$

Here $G(x, y)$ is the normal arrangement of the values. Whereas, σ represents the standard deviation of the arrangement.

We used the binary image format of the gray-scale image for precised and accurate contour extraction, i.e., we converted the image into binary format and then determined the boundary points for the region using the formula as described in the Eq. (5).

$$\begin{cases} f(i, j-1) = 0 \\ f(i, j) = 1 \\ f(i, j) \geq 1 \\ f(i, j+1) = 0 \end{cases} \tag{5}$$

In this equation, $f(i, j)$ function for binary image represents the pixel value of the required point, and the functions $f(i, j-1)$ and $f(i, j+1)$ are representing the pixel values of the neighboring points for $f(i, j)$.

3.2.5 Morphological Operations

After the removal of noise from the image, last step of pre-processing was to perform morphological operations. There are much chances to have certain additional values present in the image even after applying filters which hamper the process of identification by using CNN. Therefore, we performed erosion and dilation for morphological operations which were able to determine the local minimum and local maximum. We applied dilation first followed by erosion. This closing operation was capable of closing small cracks in the image while not changing the shape of the it.

3.3 Segmentation

To use only the desired part of the image for various operations, the good idea is to do image segmentation. For this, we separated each pattern locally in the optimum way. Thus, we performed segmentation to ensure that the tool would use only the required information of image for its processing, while accessing the entire process. We separated each key parameter of the cheque leaflet using segmentation for correct identification and verification. Other than contour extraction in order to determine the position values for the date box as explained in previous section(s), we identified region of interest (ROI) as per the CTS-2010 cheque leaf standard dimensions as shown in 2. In order to remove the extra information from the desired region, we created a standard template using pixel values as per RBI guidelines on CTS-2010 for bank cheques as shown in Fig. 10. After the segmentation of bank cheque image, we separated each segment of key parameters for OCR to determine the patterns for verification. We also performed transfer learning approach in order to recognize the handwritten numeric digits present in the date and courtesy amount of cheque. For Indian banks, image segmentation work is considered automated as the cheque dimensions of different banks are standardised as per CTS-2010 standards while for different country cheques, it should be done either manually due to variation in cheque dimensions or separate dimensions will be defines as per the country or bank guidelines. Therefore we consider our approach for bank cheque leaf image segmentation as semi-automatic.
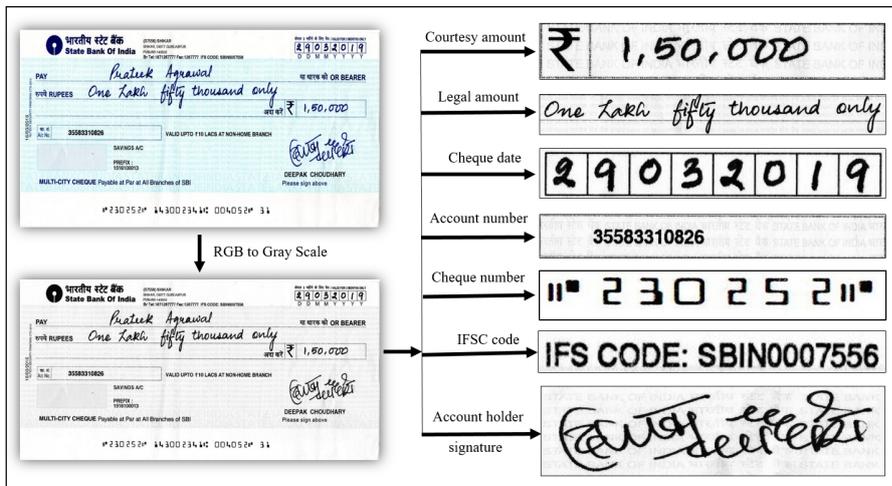

Fig. 10 Sample bank cheque image segmentation

Also, to identify the machine typographic characters of IFSC, cheque number and account number, we used OCR method while for the Similarly, signa-

ture verification, we used SIFT for feature extraction and SVM for classification.

3.4 CNN model for Legal amount and Courtesy amount identification

After extracting the image segments, we used CNN for handwritten numeric digits recognition of courtesy amount as well as to convert the legal amount into string. We used Deep Learning Toolbox (a MATLAB toolbox) for CNN implementation on two convolution layers with six and twelve filters as shown in Fig. 11. Whereas, in max pooling process, we used a window of 2x2 with one thousand epochs and one stride. After that, we received numeric output from the courtesy amount of bank cheques and converted it into string to compare with the legal amount string. We followed IPV system so as to be written in the words IPV format. For handwritten courtesy amount identification, we used transfer learning for digit identification. We used MNIST dataset [82] of handwritten digits which has sixty thousands examples in training set and ten thousands examples in test set. All the images in the dataset are normalized and centered in a fixed size. We used a dataset of two thousands samples (two hundred for each digit number ranging from 0 to 9) from training dataset and five hundred (fifty for each digit) samples from testing dataset. We distributed the entire dataset in 80% for the training and 20% for the testing using CNN. We trained with the training loss occurring with one thousand epochs, 0.01 base learning rate and seven iterations per epoch. Also for handwritten characters used in legal amount recognition, we used EMNIST dataset [83] of English letters identification. EMNIST dataset contains 88800 training examples and 14800 testing examples. We used a dataset of 4160 training samples (160 for each letter with mixed uppercase and lowercase alphabets in twenty six classes) and 1040 (forty for each letter) samples for testing dataset in our work. For CNN training, we distributed this training and testing dataset in a ratio of 80:20. Rest all the CNN training parameters are kept same as it was used for MNIST dataset training. Table 3 shows the training parameters for MNIST and EMNIST sample dataset training.

3.4.1 Proposed transfer learning algorithm for amount verification

After identifying the digits from courtesy amount image segment and characters from legal amount image segment using CNN, we arranged the acquired numeric digits into the Indian Place Value (IPV) system for matching and comparing it with the legal amount identified text. For this, we developed an algorithm "transfer learning method" to identify different nuances of handwritten strings of numerals in accordance to the IPV chart. Then we converted courtesy amount into strings for comparing it with the legal amount. It is very essential to have a clear match between both the strings as only after that the process of funds transfer is supposed to be initiated. For this we convert the

Table 3 CNN model parameters for MNIST and EMNIST dataset.

Parameters	Value
Number of convolution layers	2
Number of Max pooling layer	1
Number of filter at convolution layer	6 and 12
Kernel window size at convolution layer	3X3
Kernel window size at max pool layer	2X2
Number of stride at convolution layer	1
Number of stride at max pool layer	1
Number of neurons in output layer	2
Learning rate	0.01
Number of epochs passed	1000
Number of iteration (per epoch)	7
Number of input attributes for digits data (MNIST dataset)	784
Number of output attributes for MNIST dataset	1
Training data for MNIST dataset	2000
Testing data for MNIST dataset	500
Input attributes for letters data (EMNIST dataset)	784
Output attributes for EMNIST dataset	1
Training data for EMNIST dataset	4160
Testing data for EMNIST dataset	1040
Training / testing data	80%/20%
Activation function (before and after max pooling layer)	ReLU
Batch size	64
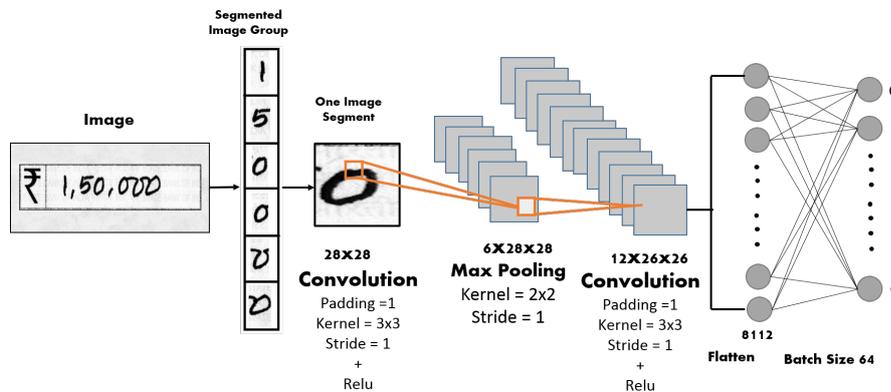
Fig. 11 Proposed CNN architecture for handwritten digit recognition of courtesy amount.

numeric values into strings and then they are compared with the legal amount strings. As the way of referring the amount IPV system be different due to

local influence thus a dictionary function was used which kept all the possible ways, a word can be written.

Algorithm 1: Algorithm for verifying legal amount with courtesy amount.

Input: *Input Courtesy amount (number), Legal amount (legal_amount)*
Output: (1) *Courtesy amount to words (courtesy_in_words)*
Output: (2) *If Courtesy amount in words equal to Legal Amount*
`// Indexing of arrays ones[] and tens[] starts from one`
1 $ones[\,] \leftarrow \{"One","Two","Three",..."Nineteen"\}$
2 $tens[\,] \leftarrow \{"Twenty","Thirty","Forty",..."Ninety"\}$
3 $S \leftarrow null$
4 $output \leftarrow null$
5
6 **Function** convert(*number*):
7   **if** *number* > 19 **then**
8   **if** *number%10 = 0* **then**
9 | $S = tens[(number/10) - 1]$
10   **else**
11 | $S = tens[(number/10) - 1] + "\ " + ones[number\%10]$
12   **else**
13 | $S = ones[number]$
14   **end**
15   **return** S
16 **End Function**
17
18 **Function** num2words(*number*):
19   **if** $len(number) > 7$ **then**
20 | $output \leftarrow output + crore$
21   **if** $len(number) > 5$ **then**
22 | $output \leftarrow output + lakh$
23   **if** $len(number) > 3$ **then**
24 | $output \leftarrow output + thousand$
25   **if** $len(number) > 2$ **then**
26 | $output \leftarrow output + hundred$
27   **if** $number > 100$ *and* $(number\%100)$ **then**
28 | $output \leftarrow output + "and"$
29 | $output \leftarrow output + convert(number\%10^2)$
30   **if** $len(number) = 2$ *and* $(number > 19)$ **then**
31 | $output \leftarrow convert(number)$
32   **if** $(len(number) = 2$ *and* $number \leq 19)$ *or* $(len(number) = 1)$ **then**
33 | $output \leftarrow ones[number - 1]$
34   **return** $output$
35 **End Function**
36
37 **Function** compare(*string_1, string_2*):
38   **if** $string\_1 = string\_2$ **then**
39 | $Result \leftarrow "Matching"$
40   **else**
41 | $Result \leftarrow "Not\ Matching"$
42   **end**
43   **return** $Result$
44 **End Function**
45
46 $crore \leftarrow output + convert(number/10^7) + "Crore"$
47 $lakh \leftarrow output + convert((number/10^5)\%100) + "Lakh"$
48 $thousand \leftarrow output + convert((number/10^3)\%100) + "Thousand"$
49 $hundred \leftarrow output + convert((number/10^2)\%10) + "Hundred"$
50
`// Courtesy amount to words`
51 $courtesy\_in\_words \leftarrow num2words(number)$
52
`// Comparing Courtesy amount in words with Legal Amount`
53 $result \leftarrow compare(courtesy\_in\_words, legal\_amount)$

We followed the algorithm as shown in algorithm 1 to convert the numeric value of the courtesy amount as per the IPV numbering system and then to match with the strings of legal amount. Once we converted the numeric digits into string, next step was to match these strings. For matching the legal amount with courtesy amount, we used a dictionary function that verified the characters between both string values. If the strings match each other, the result was displayed as verified otherwise not-verified. Using this algorithm, we compared the legal amount with the courtesy amount for verification.

3.5 Signature verification

Next, we moved to signature verification step. In the dataset 114 images used for our experiments, four images do not contain signatures. Therefore, we used 110 handwritten offline signatures from scanned cheque images for signature verification task. These 110 signatures were created by 50 different users. Out of these signature samples, 80 were genuine while rest 30 were forged signatures. Each user's signatures were found in one to three different cheque images. Fig. 12 shows the overall signature verification sub-process of bank cheque. For feature extraction, we normalized each signature image segment in order to extract the features like resizing, thinning, rotating and cropping. The normalization experimental results are represented in the Fig. 13. From the figure, it is clearly shown that we identified the region of interest (of actual signature) by normalizing the signature image. We applied SIFT method on these normalized images for feature extraction and contour identification. SIFT works on a principle to reflect the structure of the image regions using key points and preventing the scale and orientation information of these structures. We applied SIFT method into four steps: i) Scale-space extrema counstruction (SSE), ii) Key point localization (KPL), iii) Orientation assignment (OA), iv) Key point descriptor extraction (KDE).

SSE breaks down each image segment into Gaussian pyramid at different levels. It further breaks each level into small sub-levels with different variations using Laplacian of Gaussian (LoG) filters to get the prominent keypoints. Later in next part, secured locations and key points are detected and calculated. OA is responsible for calculating the scales and orientations of finalized keypoints. KDE is responsible to generate a SIFT descriptor against each selected key point.

After extracting *SIFT descriptors* and *scales & orientations* for each signature image, we applied SVM classifier on these two features dataset to match and classify the signature patterns. SVM is a machine learning method under supervised learning. It is used for both classification as well as regression. To verify the genuineness of the signatures, we applied SVM classifiers on SIFT extracted features. We distributed genuine and forged signatures data in uniform manner between training and testing dataset in ration of 80:20. For this, 88 signature data (with 66 genuine and 22 forged signature data) were taken in training dataset while 22 signature data (14 genuine and 8 forged signature

data) were taken as testing dataset. In SVM, a separating hyperplane works as a discriminatory classifier which divides the data using a line or plane into two parts in two-dimensional space [84]. This method produces the classification results as data-points which are categorised ideally using optimal hyperplane.

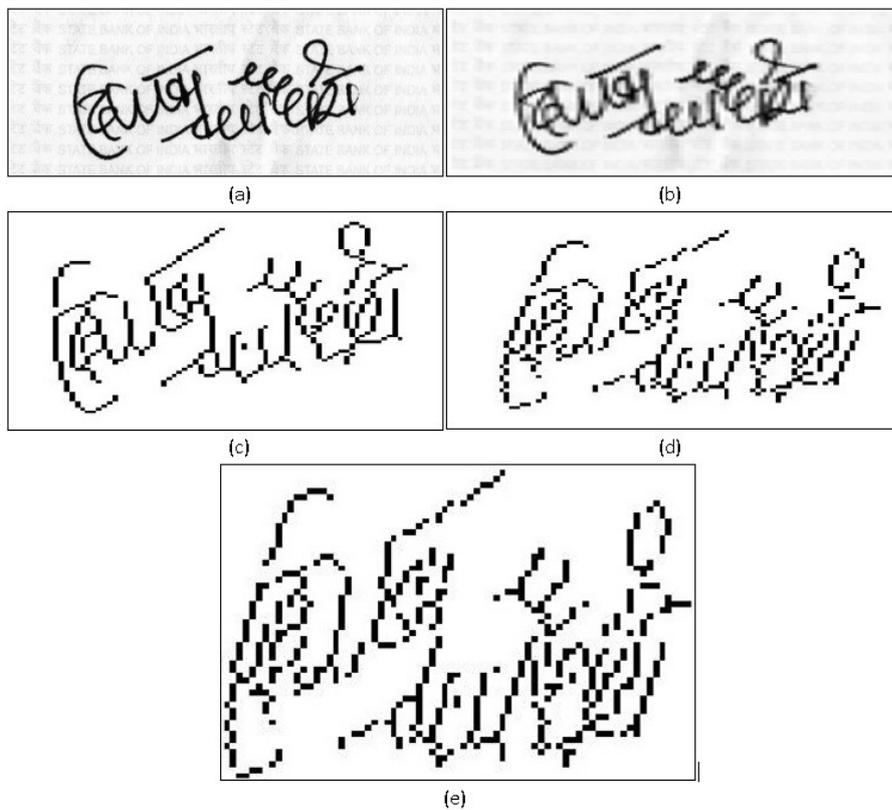**Fig. 12** Account holder Signature classification process for bank cheque verification.

Fig. 13 Normalization of Signature: (a) Original segment of Signature. (b) resized image. (c) morphological Operation: Thinning. (d) rotation of Image. (e) cropping.

4 Results and Analysis

We experimented the problem stated in the research with the procedure elucidated above in terms of performance and accuracy using MATLAB software (deep learning toolbox and image processing toolbox). We used 114 bank cheque images in our research work. Out of which, 112 images were from IDRBT bank cheque dataset and 02 cheque images were scanned by us [85] [86]. We performed training and testing on the developed system with these cheque leaflets. We used different key parameter segments obtained for bank cheque leaflet for training and testing.

For handwritten digit recognition using CNN, we achieved the accuracy with a mean value of 99.14% as shown in Table 4. Similarly, the learning progress of training networks for character recognition on different epochs and iterations are shown Table 5. From the table, we can see that the mini batch loss is reduced up to 0.0077 with 99.94% accuracy at 850 iterations. The experimental results are represented in Fig. 17.

We achieved the collective loss less than 1% for the trained network. After extracting the numbers from the courtesy amount images, we converted these numbers into words using IPV system. Once we acquired the result in words for the courtesy amount, we compared it with the legal amount using the algorithm mentioned above and in case of matching the strings of courtesy amount and legal amount, we proceed with the verification of signature. We first normalized the segmented image, extracted the features using SIFT method and then we applied SVM classifier on extracted features to match and classify the patterns. Using SVM, we were able to achieve 98.1% accuracy.

Table 4 Accurate recognition rate for individual digits.

Digit Types	Test Samples	Recognition Accuracy (%)
0	50	98.10
1	50	98.98
2	50	99.20
3	50	99.13
4	50	98.96
5	50	99.55
6	50	99.23
7	50	99.11
8	50	99.20
9	50	98.96
Total	500	99.14 (average)

In order to acquire the required segments for different cheque leaflets, we assigned different templates using pixel values. We took an example of axis bank cheques and its multiple segments to demonstrate the different key parameters which are represented in the Fig. 14. We used OCR technique and applied in-built OCR method for identifying all machine typographic text and CNN for hand written courtesy amount text. We used SIFT and SVM as re-

Fig. 14 Segmentation of a bank cheque image: (a) Original scanned cheque, (b) cropped cheque leaflet, (c) gray-scale cheque image, (d) IFS code, (e) account number, (f) cheque number, (g) courtesy amount, and (h) signature.

ferred in [87] for recognizing the signature patterns. Another sample cheque with leaflet segmentation is presented in the Fig. 15. Considering a sample cheque leaflet for experimenting the verification procedure, we got the experimental outcome as shown in the Fig. 16. We executed the segmented images from figure 15(a), 15(b), and 15(c) by OCR for text identification while for figure 15(d) and 15(e), we implemented CNN respectively.

There are advances in the field of forensics still handwritten signatures are considered as one of the most reliable feature for certifying authenticity due to its unique style of writing. We can perform online as well as offline verification of signatures and both have their benefits and shortcomings. Some of the highly efficient methods for signature verification are summarized in the Table 1.

Module wise system performance comparison in terms of accuracy with the state of the art methods, is summarised in Table 6.
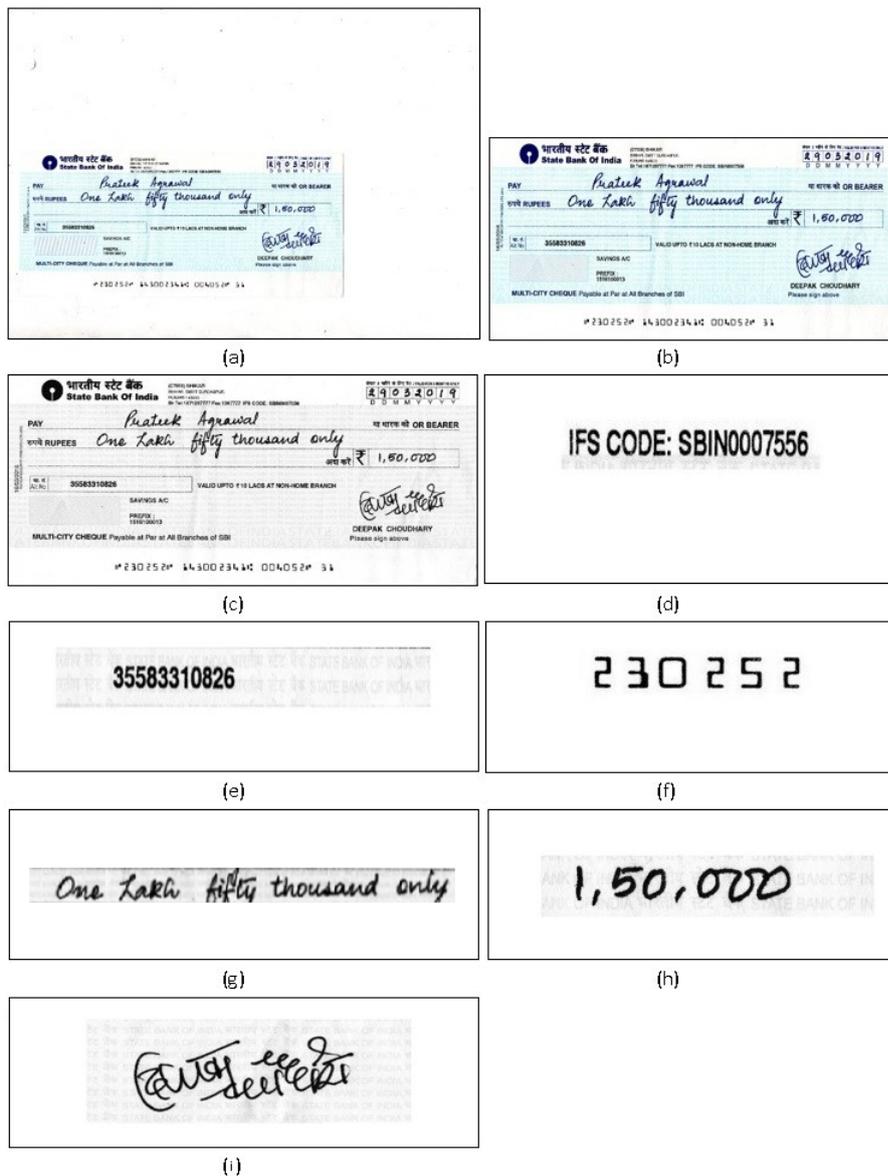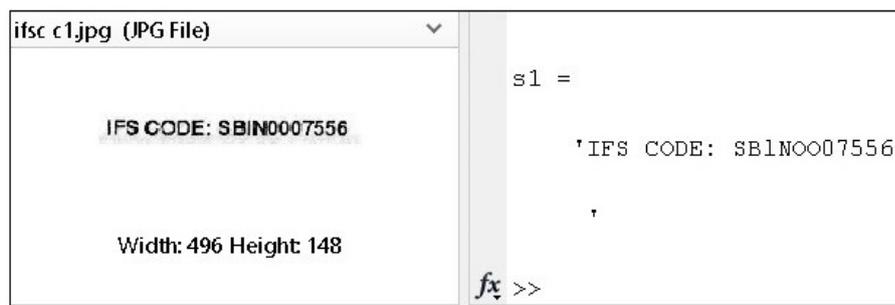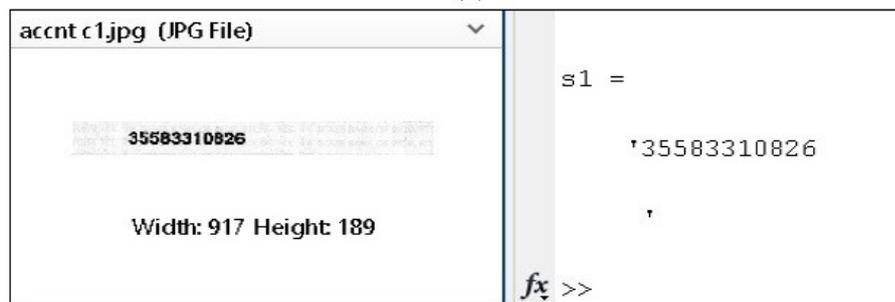
Fig. 15 Segmented regions of SBI bank cheque: (a) Original scanned cheque, (b) cropped cheque leaflet, (c) gray-scale cheque image, (d) IFS code, (e) account number, (f) cheque number, (g) legal amount, (h) courtesy amount, and (i) signature.
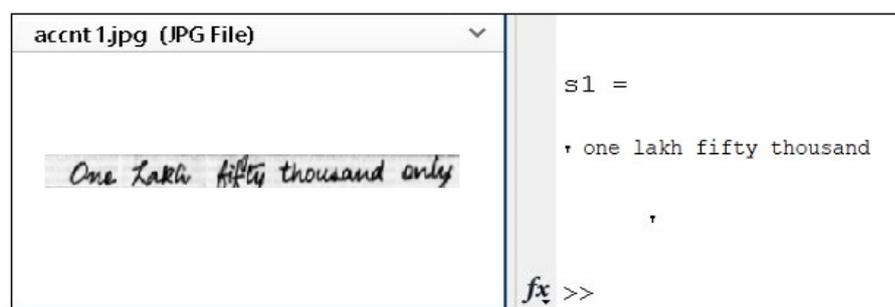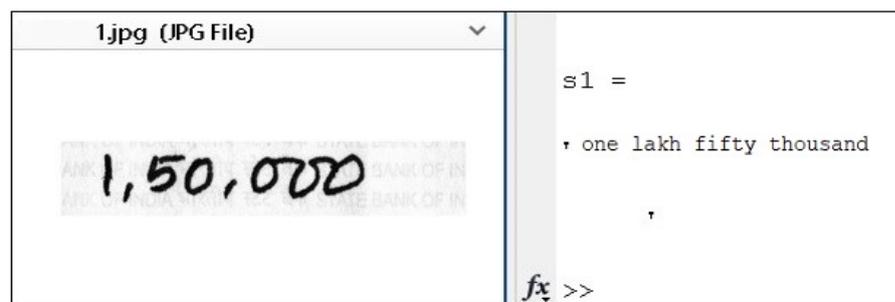
(a)

(b)

(c)

(d)

(e)

Fig. 16 Image segments on left hand size and their OCR results (shown as value of s1) for machine printed and handwritten characters on right hand side for (a) IFS code, (b) cheque number, (c) account number, (d) legal amount, and (e) courtesy amount.

Table 5 Experiment results of CNN for character recognition.

Epoch	Iteration	Time elapsed (hh:mm:ss)	Mini-batch accuracy (%)	Mini-batch loss
1	1	00:00:00	10.89	2.4308
8	50	00:00:22	97.88	0.0755
15	100	00:00:44	98.58	0.0556
22	150	00:01:06	98.96	0.0423
29	200	00:01:29	99.18	0.0298
36	250	00:01:51	99.38	0.0264
43	300	00:02:13	99.34	0.0254
50	350	00:02:35	99.50	0.0212
58	400	00:02:57	99.66	0.0169
65	450	00:03:20	99.56	0.0202
72	500	00:03:42	99.73	0.0168
79	550	00:04:04	99.76	0.0126
86	600	00:04:27	99.84	0.0108
93	650	00:04:49	99.80	0.0122
100	700	00:05:11	99.84	0.0099
108	750	00:05:33	99.90	0.0083
115	800	00:05:56	99.83	0.0103
122	850	00:06:16	99.94	0.0077
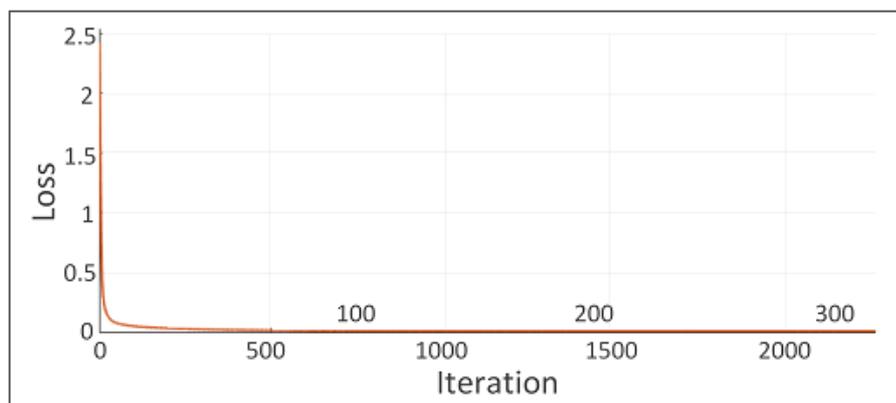
Fig. 17 Experimental loss in training.

5 Conclusion and Future Work

We developed the model to verify the bank cheques using OCR, CNN, SIFT and SVM. We used OCR method to identify the machine typographic characters with desirable accuracy and efficiency, whereas, we performed CNN to give precise output for the handwritten digits written on the cheque leaflet. We proposed and implemented the algorithm to convert numbers into words to verify the cheque which is one of the major reason of its bounce-off and of halt the monetary transaction. In order to achieve this, we applied OCR technique to recognise the machine printing digits and achieved 97.7% accurate matching. Then, we used separate database to train the network and after

Table 6 Module wise performance analysis of proposed system with other existing works.

Work details	Bank cheque leaf image segmentation	Machine typo-graphic text & digit recognition	Handwritten text & digit recognition	Signature verification
Peres et al. [28]	Manual	NA	CNN & SVM method for handwritten text (96.5% accuracy)	NA
Sneha et al. [36]	manual	NA	Template matching & correlation coefficient for handwritten text (76.4% accuracy)	NA
Jayadevan et al. [38]	Manual	NA	Modified quadratic discriminat function for handwritten text (97.04% accuracy)	NA
Ahlawat et al. [88]	manual	NA	Pixel & geometrical based CNN method for handwritten digit (99.89% accuracy)	NA
Mondal & Kundu [69]	Manual	NA	NA	Kullback - Leiber Divergence method (97% accuracy)
Hafemann & et al. [79]	Manual	NA	NA	Model Agnostic Meta Learning method (94.84% accuracy)
Karanjkar et al. [77]	Manual	NA	ANN method (85% accuracy)	NA
Ramanathan et al. [14]	Manual	Gabor Filter & SVM classifier (97% accuracy)	NA	NA
Proposed system	Semi-automatic (standard cheque template dimensions)	OCR method (97.7% accuracy)	CNN method & transfer learning algorithm (99.14% & 99.94% accuracy)	FIST feature extraction & SVM classifier (98.1% accuracy)

achieving a desired level of accuracy after training, we used different data sets to test the trained model for matching the numbers into the words. Evidently, we achieved an accuracy of 99.14% for the digit recognition which is an improvement over the previous accuracy of 99.05% for the CNN used for digit recognition. Similarly, for character recognition using CNN, we were able to achieve the accuracy up to 99.94%. For signature recognition, we used SIFT and SVM classifier to determine the relevancy for verification of signature and achieved 98.1% accuracy. our innovation will help the banking organizations to automate the process of cheques clearance in faster and easier way in comparison to conventional and manual approach by saving time and manpower. In our research work, we performed the verification of bank cheques using the CNN, in an effort to automate the process of verification of the bank cheques based on the identification of the handwritten numeric digits. However, in order to successfully perform the clearance of bank cheque, we need to perform the validation of bank cheques as well as ability to function online using a database resolving the required queries in real time.

In the future, verification process can be clubbed with validation process of cheque leaflets by developing a system capable enough of sequentially performing all the required steps for clearance of bank cheques in real time using a customer-centric database with the required information of a bank customer and also allowing the bank to keep track of all the transactions via cheques in a centralized manner, making the process of cheques clearance completely automated with minimal human intervention and in minimal processing time. Currently, our innovation is applicable for the cheques written in English language only. In future, similar models may be proposed for other languages too.

Acknowledgements

References

1. Cheque Market 2018. `https://www.chequeandcredit.co.uk/information-hub/facts-and-figures/cheque-market-2018`, 2018. [Online; accessed October 4, 2019].
2. Malaysia Bank Template. `https://www.chequepro.com/Cheque-printing-writing-software-for-Malysian-banks.html`, 2019. [Online; accessed October 4, 2019].
3. Chrysanth Cheque Writer. `https://freechequewriter.com/download/cheque-writing-printing-software-for-maldives-banks`, 2019. [Online; accessed October 4, 2019].
4. Malaysia Bank Template. `https://www.finder.com/how-to-write-a-check`, 2019. [Online; accessed October 4, 2019].
5. CTS2010. `https://www.rbi.org.in/scripts/FAQView.aspx?Id=63`, 2018. [Online; accessed October 4, 2019].
6. Deepak Chaudhary, Prateek Agrawal, and Vishu Madaan. Bank cheque validation using image processing. *International Conference on Advanced Informatics for Computing Research (ICAICR'19)*, pages 148–159, 2019.
7. K Dhanva, M Harikrishnan, and P U Babu. Cheque image security enhancement in online banking. *Second International Conference on Inventive Communication and Computational Technologies (ICICCT)*, pages 1256–1260, 2018.
8. HK Singh and AE Tigga. Impact of information technology on indian banking services. *1st International Conference on Recent Advances in Information Technology (RAIT)*, pages 662–665, 2012.
9. S S Gonge and A A Ghatol. Combined dwt-dct digital watermarking technique software used for cts of bank. *International Conference on Issues and Challenges in Intelligent Computing Techniques (ICICT)*, pages 776–783, 2014.
10. S S Gonge and A A Ghatol. An integration of svd digital image watermarking with aes technique for copyright protection and security of bank cheque image. *2nd International Conference on Contemporary Computing and Informatics (IC3I)*, pages 769–778, 2016.
11. D G Menon, S G Sreelakshmi, and A Shivdas. Green banking initiatives: A review of indian banking sector. *International Conference on Technological Advancements in Power and Energy (TAP Energy)*, pages 1–7, 2017.
12. H Singh and A Sachan. A proposed approach for character recognition using document analysis with ocr. *Second International Conference on Intelligent Computing and Control Systems (ICICCS)*, page 190195, 2018.

13. P A Wankhede and W S Mohod. A different image content based retrieval using ocr techniques. *International conference of Electronics, Communication and Aerospace Technology (ICECA)*, pages 155–161, 2017.

14. R Ramanathan, S Ponmathavan, L N V Thaneshwaran, A S Nair, and K P Soman. Optical character recognition for english and tamil using support vector. *International Conference on Advances in Computing, Control, and Telecommunication Technologies*, pages 610–612, 2009.

15. C Chen. A general approach for handwritten digits segmentation using spectral clustering. *14th IAPR International Conference on Document Analysis and Recognition*, pages 547–552, 2017.

16. W E Dickinson. A character-recognition study. *IBM Journal of Research and Development*, pages 335–348, 1960.

17. N Liu, C and L Shelton, G. An experimental investigation of a mixed-font. *IEEE Transactions on Electronic Computers*, EC-15(6):916–925, 1966.

18. G Nagy and G Shelton. Self-corrective character recognition system. *IEEE Transactions on Information Theory*, 12(2):215222, 1966.

19. P Wang and C R Shiau. Machine recognition of printed chinese characters via transformation algorithms. *Pattern Recognition*, 5(4):303–321, 1972.

20. A Zramdini and R Ingold. Optical font recognition using typographical features. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 20(8):877–882, 1998.

21. Zhu Yong, Tan Tieniu, and Wang Yunhong. Font recognition based on global texture analysis. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 23(20):1192–1200, 2002.

22. P Dhande and R Kharat. Recognition of cursive english handwritten characters. *International Conference on Trends in Electronics and Informatics ICEI*, pages 199–203, 2017.

23. U Pal, F Kimura, K Roy, and T Pal. Recognition of english multi-oriented characters. *International Conference on Pattern Recognition (ICPR'06)*, pages 1–6, 2006.

24. P Dhande and R Kharat. Character recognition for cursive english handwriting to recognize medicine name from doctors prescription. *International Conference on Computing, Communication, Control and Automation (ICCUBEA)*, pages 1–5, 2017.

25. A Saenthon and N Sukkhadamrongrak. Comparison the training methods of neural network for english and thai character recognition. *Signal and Information Processing Association Annual Summit and Conference (APSIPA)*, pages 1–4, 2014.

26. TT Ramanathan and D Sharma. Multiple classification using svm based multi knowledge based system. *International Conference on Advances in Computing & Communications, (ICACC'17)*, pages 307–311, 2017.

27. S Stewart, L Pinto, and Barrett B. Segmentation and stitching improves handwriting recognition on datasets with few samples. *16th International Conference on Frontiers in Handwriting Recognition (ICFHR)*, pages 465–470, 2018.

28. A A Peres, S M Vieira, and J R C Pinto. Hybrid neural models for automatic handwritten digits recognition. *International Joint Conference on Neural Networks (IJCNN)*, pages 1–8, 2018.

29. Y. LeCun, L. Bottou, Bengio. Y., and P. Haffner. Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11):2278–2324, 1998.

30. R Kajale, S Das, and P Medhekar. Supervised machine learning in intelligent character recognition of handwritten and printed nameplate. *International Conference on Advances in Computing, Communication and Control (ICAC3)*, pages 1–5, 2017.

31. H Goto and H Aso. Character pattern extraction from colorful documents with complex backgrounds. *Object recognition supported by user interaction for service robots (ORSUISR)*, pages 180–183, 2002.

32. M Brisinello and R Gibric. Optical character recognition on images with colorful background. *IEEE 8th International Conference on Consumer Electronics - Berlin (ICCE-Berlin)*, pages 1–6, 2018.

33. M Brisinello, R Grbic, M Pul, and T Andelic. Improving optical character recognition performance for low quality images. *59th International Symposium ELMAR-2017*, pages 167–171, 2017.

34. Wanli Feng and Shangbing Gao. A vehicle license plate recognition algorithm in night based on hsv. *3rd International Conference on Advanced Computer Theory and Engineering(ICACTE)*, 4:53–56, 2010.

35. H Jia, Shui and G Wei, Dong. Extraction of legal amount from color bank check images. *International Conference on Web Intelligence and Intelligent Agent Technology*, pages 253–256, 2010.

36. Sneha Singh, Tharun Kariveda, Jija Das Gupta, and Kallol Bhattacharya. Handwritten words recognition for legal amounts of bank cheques in english script. *Eighth IEEE International Conference on Advances in Pattern Recognition (ICAPR)*, pages 1–5, 2015.

37. Mohammad B. A. Miah, Mohammad A. Yousuf, Mohammad S. Mia, and Mohammad P. Miya. Handwritten courtesy amount and signature recognition on bank cheque using neural network. *International Journal of Computer Applications*, 118(5):21–26, May 2015.

38. R Jayadevan, U Pal, and K F. Recognition of words from legal amounts of indian bank cheques. *12th International Conference on Frontiers in Handwriting Recognition, pp. 166-171. IEEE*, pages 166–171, 2010.

39. S P Raghevendra and Danti A. A novel recognition of indian bank cheques based on invariant geometrical features. *International Conference on Trends in Automation, Communications and Computing Technology (I-TACT'15)*, pages 1–5, 2009.

40. K Shirai, M Akita, M Okamoto, K Tanikawa, T Akiyama, and T Sakaguchi. Removal of background patterns and signatures for magnetic ink character recognition of checks. *10th IAPR International Workshop on Document Analysis Systems*, pages 190–194, 2012.

41. H Lu, B Guo, J Liu, and X Yan. A shadow removal method for tesseract text recognition. *International Congress on Image and Signal Processing, BioMedical Engineering and Informatics (CISP-BMEI)*, pages 1–5, 2017.

42. C Wiraatmaja, K Gunadi, and IN Sandjaja. The application of deep convolutional denoising autoencoder for optical character recognition preprocessing. *International Conference on Soft Computing, Intelligent System and Information Technology (IC-SIIT)*, pages 72–77, 2017.

43. Jason Aronoff, Steven Simske, and Margaret Sturgill. Automated optimization of void pantograph settings. *HP Laboratories*, pages 690–693, 2011.

44. K Peymani and M Soryani. From machine generated to handwritten character recognition; a deep learning approach. *3rd International Conference on Pattern Recognition and Image Analysis (IPRIA)*, pages 243–247, 2017.

45. Farsi-datatset. `https://github.com/Kianenigma/DeepConvFarsiOCR`, 2017. [Online; accessed October 4, 2019].

46. Neha Bhadwal, Prateek Agrawal, Vishu Madaan, Awadhesh Shukla, and Anuj Kakran. Smart border surveillance system using wireless sensor network and computer vision. *International Conference on Automation, Computational and Technology Management (ICACTM19)*, pages 183–190, 2019.

47. Sandeep Singh Chauhan, Prateek Agrawal, and Vishu Madaan. E-gardener: Building a plant care taker robot using computer vision. *4th International Conference on Computing Sciences (Feynman100-ICCS'18)*, pages 137–142, 2018.

48. Prateek Agrawal, Simranjeet Kaur, Harjeet Kaur, and Amita Dhiman. Analysis and synthesis of an ant colony optimization technique for image edge detection. *International Conference on Computing Science (ICCS)*, pages 127–131, 2012.

49. A Ignat and B Aciobniei. Handwritten digit recognition using rotations. *18th International Symposium on Symbolic and Numeric Algorithms for Scientific Computing*, pages 303–306, 2016.

50. J Hou, H Zeng, L Cai, J Zhu, J Cao, and J Hou. Handwritten numeral recognition using multi-task learning. *International Symposium on Intelligent Signal Processing and Communication Systems (ISPACS)*, pages 155–158, 2017.

51. Y Li, J Zhang, P Gao, L Jiang, and M Chen. Grab cut image segmentation based on image region. *3rd IEEE International Conference on Image, Vision and Computing*, pages 311–315, 2018.

52. A. V. Mityakov, V. K. Varankin, and Y. S. Tatarinov. Application of modern architectures of deep neural networks for solving practical problems. *International Conference on Soft Computing and Measurements (SCM)*, pages 389–390, 2017.

53. Christian Szegedy, Wei Liu, Yangqing Jia, Pierre Sermanet, Scott Reed, Dragomir Anguelov, Dumitru Erhan, Vincent Vanhoucke, and Andrew Rabinovich. Going deeper with convolutions. *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1–9, 2015.

54. X Ji, Y Li, J Cheng, Y Yu, and M Wang. Cell image segmentation based on an improved watershed algorithm. *8th International Congress on Image and Signal Processing (CISP)*, pages 433–437, 2015.

55. Sourabh S. Shenoy, Supriya U Nayak, and Neelima B. Swift - a performance accelerated optimized string matching algorithm for nvidia gpus. *International Symposium on Parallel and Distributed Computing*, pages 80–87, 2016.

56. Omkar Sunil Joshi, Bhargavi R. Upadhvay, and Supriya Musica. Parallelized advanced rabin-karp algorithm for string matching. *International Conference on Computing, Communication, Control and Automation (ICCUBEA)*, pages 1–5, 2017.

57. Zhou Yansen and Gao Cun. Research and improvement of a multi-pattern matching algorithm based on double hash. *3rd IEEE International Conference on Computer and Communications (ICCC)*, pages 1772–1776, 2017.

58. F. Camastra, M. Spinetti, and A. Vinciarelli. Offline cursive character challenge: a new benchmark for machine learning and pattern recognition algorithms. *International Conference on Pattern Recognition (ICPR)*, pages 1–7, 2006.

59. K S Dash, N B Puhan, and G Panda. Gestalt configural superiority effect: A complexity paradigm for handwritten numeral recognition. *Eighth International Conference on Advances in Pattern Recognition (ICAPR)*, pages 1–6, 2015.

60. Z Li, M Schulte, Austum, and M Neschen. Fast logo detection and recognition in document images. *20th International Conference on Pattern Recognition (ICPR10)*, pages 2716–2719, 2010.

61. L Shen, W Wu, S Zheng, Shen Linlin, Wu Wenfeng, and Zheng & Songhao. Tv logo recognition based on luminance variance. *IET International Conference on Information Science and Control Engineering 2012 (ICISCE 2012)*, pages 1–4, 2012.

62. AA Shirazi, A Dehghani, H Farsi, and M Yazdi. Persian logo recognition using local binary patterns. *3rd International Conference on Pattern Recognition and Image Analysis (IPRIA 2017)*, pages 258–261, 2017.

63. C Kalaiyarasi and S Karthikeyan. Enhancing logo matching and recognition using local features. *International Conference on Information Communication and Embedded Systems (ICICES2014)*, pages 1–6, 2014.

64. C Pornpanomchai, P Boonsripornchai, P Puttong, and C Rattananirundorn. Logo recognition system. *ICSEC15*, pages 1–6, 2015.

65. H Sahbi, L Ballan, G Serra, and AD Bimbo. Context-dependent logo matching and recognition. *IEEE Transactions on Image Processing*, 22(3):1–13, 2013.

66. Gurpreet Kaur and Prateek Agrawal. Optimisation of image fusion using feature matching based on sift and ransac. *Indian Journal of Science and Technology*, 9(47):1–7, 2016.

67. T Naseer and N Dogru. Signature recognition by using sift and surf with svm basic on rbf for voting online. *International Conference on Engineering and Technology (ICET)*, pages 1–5, 2017.

68. A Ferrer Miguel, Chanda Sukalpa, Diaz Moises, Chayan Kumar Banerjee, Anirban Majumdar, Carmona Duarte Cristina, Acharya Parikshit, and Pa Umapada. Static and dynamic synthesis of bengali and devanagari signatures. *IEEE Transactions on Cybernetics*, 48(10):2896–2907, 2018.

69. P Mondal and N Kundu. An automated handwritten signature detection approach for e-security purposes. *Third International Conference on Science Technology Engineering & Management (ICONSTEM)*, pages 409–413, 2017.

70. S K Jadhav and M K Chavan. Symbolic representation model for off-line signature verification. *9th International Conference on Computing, Communication and Networking Technologies (ICCCNT)*, pages 1–5, 2018.

71. U A Jain and N N Patil. A comparative study of various methods for offline signature verification. *International Conference on Issues and Challenges in Intelligent Computing Techniques (ICICT)*, pages 760–764, 2014.

72. T Handhyani, A R Yohannis, and L Hiryanto. Hand signature and handwriting recognition as identification of the writer using gray level co-occurrence matrix and bootstrap. *Intelligent Systems Conference (IntelliSys)*, pages 1103–1110, 2017.

73. D Morocho, A Morales, J Firrrez, and J O Garcia. Humans in the loop: Study of semi-automatic signature recognition based on attributes. *International Carnahan Conference on Security Technology (ICCST)*, pages 1–5, 2017.

74. A Beresneva, A Epishkina, and D Shingalova. Handwritten signature attributes for its verification. *IEEE Conference of Russian Young Researchers in Electrical and Electronic Engineering (EIConRus)*, pages 1477–1480, 2018.

75. D Morocho, A Morales, J Fierrez, and R V Rodriguez. Human- assisted recognition based on comparative attributes. *14th IAPR International Conference on Document Analysis and Recognition*, pages 5–9, 2017.

76. F Boudamous, H Nemmour, Y Serdouk, and Y Chibani. Offline handwritten signature identification using histogram of templates and artificial immune recognition system. *IEEE Conference of Russian Young Researchers in Electrical and Electronic Engineering (EIConRus)*, pages 1–5, 2017.

77. Shubhangi L. Karanjkar and P. N. Vasambekar. Signature recognition on bank cheques using ann. *IEEE International WIE Conference on Electrical and Computer Engineering (WIECON-ECE)*, pages 44–47, 2016.

78. Luiz G. Hafemann, Robert Sabourin, and Luiz S. Oliveira. Characterizing and evaluating adversarial examples for offline handwritten signature verification. *IEEE Transactions on Information Forensics and Security*, 14(8):2153 – 2166, January 2019.

79. L. G. Hafemann, R. Sabourin, and L. S. Oliveira. Meta-learning for fast classifier adaptation to new users of signature verification systems. *IEEE Transactions on Information Forensics and Security*, 15:1735–1745, 2020.

80. CTS-2010 cheque leaf dimensons and specifictions. `https://rbidocs.rbi.org.in/rdocs/content/PDFs/SCFR220210.pdf`, 2010. [Online; accessed July 17, 2020].

81. Kanan Christopher and W. Cottrell Garrison. Color-to-grayscale: Does the method matter in image recognition? *PLoS ONE*, 7(1):1–7, 2012.

82. Yann LeCun, Corinna Cortes, and Christopher J.C. Burges. The mnist database of handwritten digits. `http://yann.lecun.com/exdb/mnist/`, 1998.

83. Gregory Cohen, Saeed Afshar, Jonathan Tapson, and Andr van Schaik. Emnist: an extension of mnist to handwritten letters, 2017.

84. V. Madaan and A. Goyal. Predicting ayurveda-based constituent balancing in human body using machine learning methods. *IEEE Access*, 8:65060–65070, 2020.

85. P. Dansena, S. Bag, and R. Pal. Differentiating pen inks in hand-written bank cheques using multi-layer perceptron. *7th International Conference on Pattern recognition and Machine Intelligence*, pages 655–663, 2017.

86. Cheque Market 2018. `http://www.idrbt.ac.in/icid.html`, 2017. [Online; accessed October 4, 2019].

87. Zhang Hongyang, Han Jianfeng, Jia Hui, and Zhang Yan. Extraction and matching of binocular image based on sift algorithm. *International Conference on Intelligent Transportation, Big Data & Smart City (ICITBS)*, pages 665–668, 2018.

88. S Ahlawat, A Choudhary, A Nayyar, S Singh, and B Yoon. Improved handwritten digit recognition using convolutional neural networks (cnn). *Sensors*, 20:3344, 2020.