

Online handwritten Gurmukhi word recognition using fine-tuned Deep Convolutional Neural Network on offline features

Sukhdeep Singh^a, Anuj Sharma^{b,1}, Vinod Kumar Chauhan^{c,2,*}

^a D.M. College (Affiliated to Panjab University, Chandigarh), Punjab, India

^b Department of Computer Science and Applications, Panjab University, Chandigarh, India

^c Department of Engineering, University of Cambridge, Cambridge, United Kingdom

ARTICLE INFO

Keywords:

Online handwriting recognition
Word recognition
Gurmukhi
Deep learning
Convolutional neural network

ABSTRACT

The recognition of online handwriting is a vital application of pattern recognition, which involves the extraction of spatial and temporal information of handwritten patterns, and understanding the handwritten text while writing on the digital surface. Although, online handwriting recognition is a mature but exciting and fast developing field of pattern recognition, the same is not true for many of the Indic scripts. Gurmukhi is one of such popular scripts of India, and online handwriting recognition issues for larger units as words or sentences largely remained unexplored for this script till date. The existing study and first ever attempt for online handwritten Gurmukhi word recognition has relied upon the widely used hidden Markov model. This existing study evaluated against and performed very well in their chosen metrics. But, the available online handwritten Gurmukhi word recognition system could not obtain more than 90% recognition accuracy in data dependent environment too. The present study provided benchmark results for online handwritten Gurmukhi word recognition using deep learning architecture convolutional neural network, and obtained above 97% recognition accuracy in data dependent mode of handwriting. The previous Gurmukhi word recognition system followed the stroke based class labeling approach, whereas the present study has followed the word based class labeling approach. Present Online handwritten Gurmukhi word recognition results are quite satisfactory. Moreover, the proposed architecture can be used to improve the benchmark results of online handwriting recognition of several major Indian scripts. Experimental results demonstrated that the deep learning system achieved great results in Gurmukhi script and outperforms existing results in the literature.

1. Introduction

The technology is advancing at a rapid rate and the technological innovations have also witnessed for enhancement in use of devices like PCs, Tablet PDAs, pen tablets and mobile phones, and the ways of exchanging information between these devices and human beings have also changed over time. The increased support of such devices in free flow natural handwriting has also resulted to get the attention of worldwide pattern recognition researchers for online handwriting recognition (OHWR). OHWR was an open research problem for decades, and a few recent extensive research studies in this direction have led to remarkable success of developed countries scripts, thanks to the evolution of data capture technologies (Arica & Yarman-Vural, 2001; Bilgin-Tasdemir & Yanikoglu, 2018; Carbune et al., 2019; Mandal et al., 2019; Plamondon & Srihari, 2000; Ren & Ganapathy, 2019; Szegedy

et al., 2015; Zouari et al., 2019). Although, a lot of work has been done for Chinese–Japanese–Korean (CJK), Arabic, and Latin scripts, where praiseworthy efforts have been made for noble handwriting recognition accuracy with great reliability and different feature extraction (Abandah & M Malas, 2010; Al-Taani & Al-Haj, 2010; Chakraborty & Chakraborty, 2002; Fink et al., 2010a; Ghods & Kabir, 2010; Ghosh & Roy, 2015a; Hanmandlu et al., 2003; Jaeger et al., 2001b; M. et al., 2001; Rani & Meena, 2011; Verma et al., 2004), classification and recognition techniques as k-nearest-neighbors (*k*-NN) (Altman, 1992; Zanchettin et al., 2012), support vector machines (SVM) (Cortes & Vapnik, 1995; Keshari & Watt, 2008; Simistira et al., 2015) (refer to Chauhan et al., 2019 for a recent review on SVM), neural networks (NN) (Jaeger et al., 2001a; McCulloch & Pitts, 1943), convolutional neural networks (CNN) (Bluche et al., 2013; Ciresan et al., 2011; Dan et al., 2012; Kunihiro, 1980) and hidden Markov models

* Corresponding author.

E-mail addresses: sukha13@gmail.com (S. Singh), anuj@pu.ac.in (A. Sharma), vk359@eng.cam.ac.uk (V. Chauhan).

URLs: <https://sites.google.com/site/sransingh13/> (S. Singh), <https://sites.google.com/view/anujsharma/> (A. Sharma), <http://www.eng.cam.ac.uk/profiles/vk359/> (V. Chauhan).

¹ ORCID: 0000-0003-2219-2528.

² ORCID: 0000-0001-8195-548X.

(HMM) (Artieres et al., 2007; Benouaretha et al., 2008; Bharath & Madhvanath, 2012; Rabiner, 1989) have been used for handwriting recognition. But in comparison to CJK, Latin and other non Indic scripts, the work done for Indic scripts OHWR is limited. A few OHWR studies of some Indic scripts are available, but majority of OHWR studies of Indic scripts considered only smaller units as strokes or characters. Other difficulty with the available studies for OHWR in Indic scripts word recognition is that either major work is done for a writer-dependent system or it requires a huge manual intervention, so it requires a lot to be done for larger units OHWR in Indic scripts.

Irrespective the mode of handwriting recognition, one of the essential steps for successful automated handwriting recognition system is the identification of correct and distinguishable characteristics among distinct individual units as stroke or character classes, and this process is called feature extraction which effects the classification accuracy in handwriting recognition. In existing studies, numerous handcrafted feature representations of handwritten units were used for handwriting recognition. But the most recent decades success of using deep learning (DL) techniques for image recognition has motivated pattern recognition researchers to use DL techniques for improving the state of the art in OHWR also, and DL models as convolutional neural networks (CNN) (Li et al., 2018), recurrent neural networks (RNN) (Zhang et al., 2018) and long-short term memory (LSTM) (Keysers et al., 2017) which is a special case of RNN have attained great success in OHWR. In recent times, hybrid architectures of DL have also been studied for OHWR (Ding et al., 2018). The hybrid architectures based on CNN-RNN (Dutta et al., 2018) and CNN-LSTM (Ding et al., 2018) have been successfully used for offline unconstrained line/word recognition and OCR/character recognition, respectively. It is very useful to use CNNs for capturing the spatial relationship in data, and using memory units of LSTM/RNN for keeping the memory of previous state that makes it more suitable to work with sequential data and time series data. The success of employing DL in OHWR can also be viewed from the LSTM architecture based Google's OHWR system deployed in various products (Keysers et al., 2017).

In the present work, we deal with recognizing online handwritten Gurmukhi words using deep architecture CNN. Generally, OHWR is carried out in two different paradigms-holistic and analytic. In holistic approach which is also called word-based approach, segmentation module is not required for recognition, whereas analytic approach needs segmentation to recognize handwritten words. OHWR based on holistic approach has many advantages over the analytic approach (Madhvanath & Govindaraju, 2001). The need of segmentation for analytic approach based word recognition leads to decompose the OHWR task into the problem of recognizing a sequence of smaller subunits, and it further faces major problems as segmentation ambiguity and determination of the identity of each segment (Kang et al., 2014). In view to see the advantages of holistic approach (Madhvanath & Govindaraju, 2001), the present study relies on features and matching at word level for determination of the identity of the word. Also, since there is no benchmark result available for static representation of online handwritten Gurmukhi word samples, the text as an image is also considered for OHWR of Gurmukhi data after conversion of online handwritten words to static representation of handwriting as a part of present the study.

1.1. Gurmukhi the script of Punjabi language

The Indic scripts that belong to the family of Brahmi scripts are categorized into Southern and Northern Indic scripts. Gurmukhi is the one of these popular Northern Indic scripts which is the script of 14th most widely spoken language(Punjabi) of the world (Kumar & Sharma, 2013a), and it is one of the 10 official Indian scripts (Bharath & Madhvanath, 2010). The Punjabi language speaking people not only reside in North Indian parts such as Punjab, Harayana and Delhi but also reside in rest of the world, including America, Australia, Canada,

England, Germany, Kenya, Singapore, Saudi Arabia, New Zealand and UAE etc. Punjabi is the second most widely-spoken language in Pakistan but has no official status. The word "Gurmukhi" for Gurmukhi script is derived from the "Guramukhi" Punjabi term, and it means "from Guru's mouth" (Kumar et al., 2011). The standardization of Gurmukhi script was done in 16th century and the origin of Gurmukhi script is around 500 years old, but it shares many characters with Sharada and Takari scripts, so Gurmukhi characters are even older than 500 years.

To recognize online handwriting in all scripts, the identification of fundamental units of writing is necessary, and these are called symbols of the script which denote consonants, vowels and vowel modifiers of the script. The symbol set for Gurmukhi script is given in Fig. 1. Gurmukhi script has 35 primary consonants, comprising 3 vowel bearers that form the basis for vowels; 10 independent vowels; 12 vowel modifiers, comprising three auxiliary signs; 2 half characters that are placed in the lower zone; and 6 secondary consonants, which are derived from primary consonants using a dot symbol in the feet. In Fig. 1, the basic consonants are represented by symbols 1 to 35 where first three symbols (vowel bearers) form the basis for vowels, the symbols 36 to 41 represent the multi component characters (secondary consonants derived from basic consonants having dot at the feet), the vowel modifiers and half characters are represented by symbols 42 to 53 and 54 to 55, respectively. In vowel modifiers, the symbols 51 to 53 are auxiliary signs. Gurmukhi script is also called "Pentehi" in Punjabi language because it consists of 35 basic characters. The Gurmukhi words are written from left to right direction in top to bottom approach, but individual Gurmukhi strokes are also written in a right to left direction. Most Gurmukhi characters have a horizontal line at the top and it is called headline. The headline is also used for linking characters together to formulate the words. A Gurmukhi word is divided into three different zones. The region above the headline is called upper zone, where vowels or its subparts are placed. The middle zone is the busiest zone and it lies below the headline, where consonants or subparts of vowels are placed. The lower zone is present below the foot of consonants, where a few vowels and half characters are placed. Though it is said that Gurmukhi is cursive in writing (Kumar & Sharma, 2013a), but it would not be wrong to say that Gurmukhi script is not only cursive in nature, as Gurmukhi writers also write words in mixed cursive or run-on discrete manner (Sharma, 2009). As Gurmukhi is not an isolated script and the problems faced for OHWR in other related Brahmi scripts are also faced for OHWR in Gurmukhi. One of primary challenges for researchers to work in the area of OHWR in Gurmukhi script is that the available studies for handwritten text recognition in Gurmukhi are either on smaller units such as characters and strokes or offline handwriting recognition. The only available study for OHWR in Gurmukhi is Singh and Sharma (2019), it recognizes online handwritten Gurmukhi words in four different writing environments. Further, OHWR in Gurmukhi script also faces other problems such as the isomorphic nature of characters, character/stroke order variation in handwritten words, dissimilarity in writing style of writers (Kumar & Sharma, 2013a), unfamiliarity of Gurmukhi writers to write using a digital pen on a digital surface, a large number of stroke classes, more than one characters in a single stroke and difficulty to segment words into characters. As an unavailability of major benchmarked results for OHWR in Gurmukhi especially using deep learning architecture to date, the present study is a novel effort to provide a set of benchmarks for OHWR in Gurmukhi using different deep architectures.

This article consists of five sections including this section. The rest of this paper is arranged as follows. Section 2 presents the recent decades work done for online handwriting in Gurmukhi script and other related Indic scripts that share many common properties with the Gurmukhi. Section 3 describes the existing deep learning architectures. Section 4 discusses the deep architecture used in the present study. Experimental results have been presented in Section 5, where OHWR results for Gurmukhi script using static representation of handwriting have been presented. Finally, Section 6 concludes the present article.



Fig. 1. Gurmukhi symbol set; the basic consonants are showed by first 35 symbols, the next 6 symbols having dot at the feet represent the multi component characters (here dot distinguish these characters from basic Gurmukhi characters). The vowel modifiers are represented by next 12 symbols and the last 2 symbols represent the half characters in Gurmukhi.

2. Literature review

The study of script family and different writing systems by Ghosh et al. (2010) presented that the Gurmukhi script uses the abugidas writing system and it shares many properties with other Indic scripts, so the comprehensive review presented in context of online handwriting is for Gurmukhi and other related Indic scripts. The major research for online handwriting in Indic scripts is done in recent decades, so the present study presented the work done for Indic scripts online handwriting in recent decade mainly. Further, the literature regarding online handwriting in Indic scripts for smaller units as characters or strokes is also presented here because the reports on it also assist OHWR research greatly. The Table 1 and 2 show the selected recent recognition results for online handwritten characters/digits/strokes and words in Indic scripts, respectively.

2.1. Online handwriting recognition in Indic scripts

One of the major challenges for language technology research for online handwriting in the Indian context has been the lack of the availability of benchmarked datasets for Indic scripts. In 2004, Bhaskarabhatla and Madhvanath (2004) made a vital contribution in this direction, and discussed the experiences in online handwriting recognition for collection of handwriting data in Indic scripts. They described the data collection process at isolated character and word level, and discussed the user interface issues and explicit annotation at word level. In the same year, Deepu et al. (2004) used PCA (Duda et al., 2000) technique to classify online handwritten isolated Tamil characters, where the constraints of traditional classification technique were avoided with the introduction of pre-clustering and modified distance measure.

The preprocessing, feature extraction and classification techniques applied in this study were script independent, and this makes it applicable for other Indic and non-Indic scripts also. In 2007, Bharath and Madhvanath (2007) used the HMM based symbol and word modeling for writer independent OHWR in Tamil script, and presented recognition results with lexicon sizes of 1K, 2K, 5K, 10K and 20K Tamil words and attained best results with 1K lexicon size. In the same year, Swethalakshmi et al. (2007) used SVM classifier to recognize

online handwritten isolated Telugu and Devanagari characters, where strokes were explicitly grouped into characters based on “proximity analysis” and recognition of each stroke in it was done. Roy et al. (2007) recognized online handwritten Bangla characters and numerals based on the sequential and dynamical information of the pen movements in handwriting trajectory. Bhattacharya et al. (2007) used direction code based feature extraction technique to recognize online handwritten isolated Bangla characters. A novel study to recognize online handwritten Bangla character based on stochastic modeling was given by Parui et al. (2008) in 2008, where they recognized online handwritten Bangla strokes using HMM classifier in first stage and look-up tables were used to recognize characters from strokes in second stage. The first ever work for OHWR in Bangla script was done by Bhattacharya et al. (2008). In their work, headline position and busy zone of the word were used to segment Bangla words into strokes, and 8 directional (Bhattacharya et al., 2007) feature vector for every stroke of the word was used for stroke recognition with MQDF (Kimura et al., 1987) classifier, finally characters are formed using rule based script knowledge and words are recognized. In 2009, Sternby et al. (2009) used template matching technique to recognize the Arabic online handwriting, where diacritical marks were dealt dynamically. Although, the availability of benchmarked datasets for OHWR in Indic scripts is very limited, but the freely available benchmarked datasets for online handwritten isolated characters in Indic scripts made it possible for Mondal et al. (2010) to recognize online handwritten isolated characters for Bangla, Devanagari, Tamil and Telugu scripts, and presented benchmarked results in 2010. In their work, they obtained recognition results using three different classifiers as HMM, MLP and NN, and they used chain code histogram and point-float values for feature extraction. They concluded that chain-code histogram feature values give the best recognition accuracy irrespective of classifiers, and the NN classifier gives the best recognition accuracy irrespective of the feature vector used. In the same year, Nethravathi et al. (2010) collected handwriting samples of different writing styles from large number of users, and created dataset for online handwritten Kannada and Tamil words. A set of novel features to recognize unconstrained online Devanagari handwriting was described by Lajish and Kopparapu (2010), and used fuzzy directional features and directional features

Table 1

Reported accuracy results for online handwritten character/digit/stroke recognition in Indic scripts.

Script	References	Dataset	Classifier/Method	Accuracy rate (%)
Gurmukhi	Singh et al. (2020)	Strokes based on minimal set of words	CNN	93.19 to 95.03
	Singh et al. (2016)	Strokes based on minimal set of words	HMM	87.10, 85.43 and 84.33 for different zones
	Kumar et al. (2014)	Multi stroke based aksharas	SVM	93.33 for Gurmukhi Mukta
	Kumar and Sharma (2013b)	Characters	Set theory for post processing	80.3 overall and 95.6 for one character stroke sequencing
	Sharma et al. (2008)	Characters	Elastic matching	90.08
Devanagari	Ghosh and Roy (2015b)	Characters	ZSDP, SVM	90.63
	Ghosh and Roy (2015b)	Characters	Zone wise structural and directional features (ZSD), SVM	85.10
	Chowdhury et al. (2013)	Characters	Levenshtein distance metric	83.95
	Mondal et al. (2010)	Characters	Point-float feature, HMM	82.43
	Mondal et al. (2010)	Characters	Point-float feature, Multi-layer perceptron (MLP)	83.30
	Mondal et al. (2010)	Characters	Chain-code feature, HMM	87.13
	Mondal et al. (2010)	Characters	Chain-code feature, MLP	86.15
	Swethalakshmi et al. (2006)	Characters	SVM	96.69 (42 classes) and 97.27 (82 classes)
Bangla	Ghosh and Roy (2015b)	Characters	ZSDP, SVM	92.48
	Ghosh and Roy (2015b)	Characters	ZSD, SVM	87.48
	Surinta et al. (2013)	Digits	Contour angular technique (CAT), SVM	95.9, 40% train data
	Surinta et al. (2013)	Digits	Hotspot technique (HOT), SVM	92.7, 40% train data
	Surinta et al. (2013)	Digits	Gray pixel-based method (GPB), SVM	95.9, 40% train data
	Surinta et al. (2013)	Digits	Black and white down scaled method (BWS), SVM	94.1, 40% train data
	Chowdhury et al. (2013)	Characters	Levenshtein distance metric	86.24
	Mondal et al. (2010)	Characters	Point-float feature, HMM	81.55
	Mondal et al. (2010)	Characters	Point-float feature, MLP	84.40
	Mondal et al. (2010)	Characters	Chain-code feature, HMM	90.72
	Mondal et al. (2010)	Characters	Chain-code feature, MLP	89.82
	Lajish and Kopparapu (2010)	Characters	Directional Feature (DF)	74.86
	Lajish and Kopparapu (2010)	Characters	Fuzzy Directional Features (FDF)	82.75
	Parui et al. (2008)	Characters	HMM, look-up tables	87.7 (character level accuracy), 84.6 (Stroke level accuracy)
	Bhattacharya et al. (2007)	Characters	Direction code histogram features, MLP	93.90 and 83.61
	Roy et al. (2007)	Characters	Quadratic classifier	91.13
Tamil	Kunwar et al. (2014)	Characters	Bayesian network (BN)	83.85
	Kunwar et al. (2014)	Characters	Random BN (RBN)	86.10
	Kunwar et al. (2014)	Characters	Online RBN (ORBN)	87.80
	Kunwar et al. (2014)	Characters	Semi-supervised ORBN (SSORBN)	88.48
	Kunwar et al. (2014)	Characters	Naive bayesian (NB)	78.26
	Kunwar et al. (2014)	Characters	SVM	90.68
	Kunwar et al. (2014)	Characters	HMM	87.82
	Chowdhury et al. (2013)	Characters	Levenshtein distance metric	85
	N. Murthy and Ramakrishnan (2011)	Characters	Dexterous classifier	90.2
	Mondal et al. (2010)	Characters	Point-float feature, HMM	84.67
	Mondal et al. (2010)	Characters	Point-float feature, MLP	84.98
	Mondal et al. (2010)	Characters	Chain-code feature, HMM	92.10
	Mondal et al. (2010)	Characters	Chain-code feature, MLP	91.80
Telugu	Kinjarapu et al. (2016)	Characters	SVM	90.21
	Chowdhury et al. (2013)	Characters	Levenshtein Distance Metric	87.10
	Mondal et al. (2010)	Characters	Point-float feature, HMM	85.72
	Mondal et al. (2010)	Characters	Point-float feature, MLP	86.97
	Mondal et al. (2010)	Characters	Chain-code feature, HMM	91.82
	Mondal et al. (2010)	Characters	Chain-code feature, MLP	92.37
	Arora and Namboodiri (2010)	Characters	SVM, decision directed acyclic graph (DDAG)	95.12
	Prasanth et al. (2007)	Characters	Dynamic time warping (DTW)	90.6
	Babu et al. (2007)	Symbols	HMM	91.6 (Top-1) and 98.7 (Top-5)
Kannada	Prasad et al. (2012)	Characters	Principal component analysis (PCA)	87.5
	Prasad et al. (2012)	Characters	DTW	63.5
	Rampalli and Ramakrishnan (2011)	Characters	SVM, angles and derivatives	81.2

(continued on next page)

Table 1 (continued).

Script	References	Dataset	Classifier/Method	Accuracy rate (%)
Gujarati	Rampalli and Ramakrishnan (2011)	Characters	Fusion classifier	92.3
	N. Murthy and Ramakrishnan (2011)	Characters	Dexterous classifier	92.6
	Prasad et al. (2009)	Characters	Divide and conquer strategy (DCS)	77.6 and 81.3
	Gohel et al. (2015)	Characters	k -NN, low level stroke (LLS) features	93
	Gohel et al. (2015)	Digits	k -NN, LLS features	95
	Prasad and Kulkarni (2014)	Characters	Adaptive neuro fuzzy classifier (ANFC)	65.67, 62 and 68

Table 2

Reported accuracy results for online handwritten word recognition in Indic scripts.

Script	References	Classifier/Method	Accuracy rate (%)
Gurmukhi	Singh and Sharma (2019)	HMM, step by step approach	82.20 to 85.98 (data/writer dependent and independent modes)
	Sharma et al. (2009)	Rearrangement of strokes, Elastic matching	81.02
Devanagari	Ghosh et al. (2019)	BLSTM	94.34 to 99.50 (for 1K to 10K size)
	Ghosh and Roy (2016)	Zone wise slopes of dominant points (ZSDP) approach, HMM	93.82
	Bharath and Madhvanath (2012)	HMM	87.13
Bangla	Ghosh et al. (2019)	BLSTM	87.38 to 95.24 (for 1K to 10K size)
	Bhattacharya et al. (2018)	HMM, Mod-NPen features	94.10 (250 lexicon size), 76.30 (20K lexicon size)
	Bhattacharya et al. (2017)	HMM	89.29 (All test set words)
	Bhattacharya et al. (2016)	SVM	94.3 (Character level accuracy)
	Ghosh and Roy (2016)	ZSDP, HMM	90.23
	Samanta et al. (2015)	DCEseg, HMM	84.67
	Samanta et al. (2015)	MINseg, HMM	85.69
	Samanta et al. (2015)	ANGseg, HMM	84.85
	Chowdhury et al. (2015)	Fuzzy features	77%
	Srimany et al. (2014)	SVM	87.73
	Samanta et al. (2014)	HMM	89.65
	Frinken et al. (2014b)	Bi-directional long short-term (BLSTM) neural network	55.05 (Complex words)
	Frinken et al. (2014a)	BLSTM neural network	89, 85.9 (Lexicon-driven word recognition)
	Bhattacharya et al. (2013)	Rule based segmentation, SVM	97.45
	Bhattacharya and Pal (2012)	Rule based segmentation, SVM	97.68 (Stroke level accuracy)
	Fink et al. (2010b)	HMM	87.70
	Bhattacharya et al. (2008)	MQDF	82.34
Tamil	Sundaram and Ramakrishnan (2015)	SVM + bigram + DTW	81.6
	Urala et al. (2014)	SVM + bigram	89.2 (Symbol level accuracy, GNote data), 74.5 (Word level accuracy, GNote data)
	Urala et al. (2014)	SVM + bigram	83.22 (Symbol level accuracy, tablet PC data), 54.2 (Word level accuracy, tablet PC data)
	Urala et al. (2014)	SVM	78.52 (Symbol level accuracy, tablet PC data), 40.05 (Word level accuracy, tablet PC data)
	Sundaram and Ramakrishnan (2013)	Dominant overlap criterion segmentation (DOCS), SVM	50.9
	Sundaram and Ramakrishnan (2013)	Attention feedback segmentation (AFS), SVM	64.9
	Bharath and Madhvanath (2012)	HMM	91.8
	Sundaram and Ramakrishnan (2011)	Dominant overlap segmentation (DOS), SVM	86.9 (Symbol level accuracy)
	Bharath and Madhvanath (2007)	HMM	94.49, 93.17 and 92.15 for 5k, 10k and 20k words respectively

for character recognition. Arora and Namboodiri (2010) proposed a hybrid system to recognize online handwriting in Indian scripts, and

it handles ambiguities in segmentation as well as stroke recognition. A lexicon free segmentation technique for OHWR in Tamil script was

proposed by Sundaram and Ramakrishnan (2011) in 2011, and it is also applicable to other non cursive Indic scripts. In their work, they proved that the OHWR accuracy is greatly affected by the segmentation quality. Sundaram and Ramakrishnan (2011) analyzed the challenges faced for text input methods in Indic scripts, and concluded that the existing methods either require huge manual intervention or most of the existing approaches for OHWR in Indic scripts are script dependent. They proposed a data driven and script independent approach for OHWR in Devanagari and Tamil scripts. The best results for Tamil and Devanagari were obtained using lexicon driven approach, and combination of lexicon free and lexicon driven approach, respectively. Further, an algorithm for headline detection that identifies the multiple headline strokes present in a word was also proposed in their work. Rampalli and Ramakrishnan (2011) presented a method to recognize online handwritten Kannada characters by combining offline image based recognition, where the accuracy of online handwriting recognizer was improved by 11% when it was combined with offline handwriting recognizer. A novel dexterous classification technique to recognize online handwritten Kannada and Tamil characters using PCA with NN classifier at first stage and DTW classifier with different feature combinations at second stage was proposed by N. Murthy and Ramakrishnan (2011). In 2012, Prasad et al. (2012) recognized online handwritten isolated Kannada characters using PCA and DTW techniques, and attained better accuracy with statistical technique PCA. In 2013, Chowdhury et al. (2013) proposed a new technique (based on similarity measures between samples) to recognize online handwritten Bangla, Devanagari, Tamil and Telgu characters using Levenshtein distance metric, where shape and position information was used for representation of feature vector. Sundaram and Ramakrishnan (2013) introduced a two moduled script dependent lexicon free segmentation scheme for online handwritten Tamil words, and it solves the problem of under and over segmentation also. A segmentation study to deal with basic and compound characters in online handwritten Bangla words was presented by Bhattacharya et al. (2013) by extending their earlier work (Bhattacharya & Pal, 2012).

The work done for Indic script online handwriting recognition in recent five years is praiseworthy. In 2014, Samanta et al. (2014) used completely linked non-homogeneous HMM classifier with linear and circular features to recognize unconstrained online handwritten Bangla words, where a new technique was also proposed for determination of initial transition and emission probabilities for HMM to avoid over fitting of training data. In their work, the results of two HMMs were combined for recognition, where one HMM takes input in natural order and other in reverse order. Prasad and Kulkarni (2014) recognized online handwritten isolated Gujarati characters using ANFC and weighted k - NN classifiers, and attained better results with k - NN classifier. The first ever study for online learning of handwritten characters of Indic scripts in a semi-supervised environment was done by Kunwar et al. (2014), where they used naive bayes and RBN classifiers for online handwritten character recognition in Tamil script and attained better results with RBN. Frinken et al. (2014b) used BLSTM neural network for OHWR (containing compound characters) in Bangla script. They proposed an unsupervised technique to extract features. The technique was based on dissimilarity space embedding (DSE) of local neighborhoods around the points along the trajectory, to recognize online handwritten Bangla text was proposed by Frinken et al. (2014b).

In 2015, Ghosh and Roy (2015b) used two different feature extraction approaches to recognize online handwritten character recognition in Bengali and Devanagari scripts. In first approach, online handwritten strokes were partitioned into local zones, and structural and directional features were extracted in these zones. Although, the second scheme also used the local zone partitioning, but it detected the dominant points in every stroke and slope angles between successive dominant points were used for feature extraction. Chowdhury et al. (2015) used fuzzy linguistic rules to develop a framework for OHWR in Bangla script, where developed model comprises four key modules as: segmentation, fuzzy feature extraction, learning phase and recognition phase.

A novel script independent segmentation technique for OHWR was proposed by Samanta et al. (2015) in 2015, and used Bangla word samples for testing this segmentation scheme. This study also presented that the non-homogeneous HMM based recognition methodology explained in Samanta et al. (2014) is independent over different segmentation strategies and scripts. Gohel et al. (2015) proposed a low level stroke feature based technique for recognition of online handwritten numerals and characters in Gujarati script, where they used k -NN classifier for recognition and attained best results with combination of LLS and directional features. Sundaram and Ramakrishnan (2015) used bigram language models and re-evaluation strategy for OHWR in Tamil script. The SVM classifier was used for recognition of individual symbols after segmentation, and the recognition accuracy was improved by using (i) a bigram language model at the character level and (ii) expert classifiers for re-evaluating and disambiguating the various sets of confused symbols.

In 2016, Bhattacharya et al. (2016) proposed an end-to-end system to recognize online handwriting in Bangla script. The developed model combines the line segmentation, word segmentation, stroke segmentation, preprocessing, sub-stroke recognition, character recognition, word formation and post processing modules for continuous online Bangla handwriting recognition. Google's real world multi-language and multi-script recognition system for online handwriting was proposed by Keyser et al. (2016), it recognizes 22 scripts of 97 languages. The comparison of zone based features for OHWR in Bengali and Devanagari scripts was done by Ghosh and Roy (2016), where they employed HMM based recognition technique to study and compare three feature extraction methods. Kinjarapu et al. (2016) proposed an efficient SVM-based online handwriting recognition system for isolated handwritten Telugu characters, where they attained comparable recognition performances without preprocessing of data and using only 28 simple features in comparison to hundreds of features used in literature. A scheme to detect and clean noisy regions from online handwritten words in Bangla script was proposed by Bhattacharya et al. (2017). In 2018, a novel set of features for online handwritten Bangla character recognition was presented by Sen et al. (2018a), and the evaluation of this study was done using 15,000 isolated character samples. In the same year, Sen et al. (2018b) presented a stroke-based word segmentation approach for connected strokes in online Bangla handwriting. P. et al. (2018) proposed an RNN based OHWR system in Devanagari script. A writer independent unconstrained OHWR system composed of combined set of modified features in Bangla script was proposed by Bhattacharya et al. (2018), where Mod-NPen features were modified over Npen++ features for better recognition accuracy and HMM classifier was used for explicit-segmentation-free recognition. A computational analysis for online Kannada handwriting recognition with different pre-processing and testing methodology for SVM classifier was carried out by Ramya and Shama (2018), where the feature set consisted of normalized horizontal and vertical coordinates, writing direction and curvature at a point. Recently, an LSTM and BLSTM based cursive and non-cursive OHWR system in Devanagari and Bengali scripts was proposed by Ghosh et al. (2019) in 2019, where developed system carried out training of basic strokes using RNN after dividing the each word into three horizontal zones. The proposed system overcame the drawbacks of HMM based recognition systems and it provided new insights to develop similar systems for other Indic scripts as well.

2.2. Online handwriting recognition in Gurmukhi script

The above mentioned studies give a comprehensive review on the research achievements for Indic script online handwriting in recent years. However, compared with these Indic script studies, when it comes to online Gurmukhi handwriting recognition, the work done is very limited. The major available results for online handwritten Gurmukhi text recognition are on smaller units such as characters and strokes. In 2008, Sharma et al. (2008) used elastic matching technique to recognize online handwritten characters in Gurmukhi script,

where every character was recognized in two stages, the first stage recognizes unknown strokes and the characters based on identified strokes are recognized in second stage. In 2009, Sharma et al. (2009) proposed rearrangement of strokes technique, which is applied after post-processing phase for OHWR in Gurmukhi and it is also applicable to other Indic scripts such as Bangla and Devanagari. In 2013, Kumar and Sharma (2013b) developed a post processor to improve the recognition accuracy of online handwritten Gurmukhi characters, where an algorithm to form characters from strokes and the solution to deal with stroke sequencing and overwriting of strokes in Gurmukhi characters was proposed. The developed set theory based algorithm attained best results for one character sequencing as compared to multiple and unknown character sequencing. In 2014, Kumar et al. (2014) proposed a SVM based recognition algorithm for online handwritten Gurmukhi characters and recognized Gurmukhi characters based on multi strokes. A novel work for the development of benchmarked dataset for online handwritten Gurmukhi strokes based on minimal set of words was done by Singh et al. (2016) in 2016. In 2017, Singh et al. (2017) proposed a dominant points based script independent feature extraction technique to recognize online handwritten strokes and validated their results with benchmarked Gurmukhi dataset (Singh et al., 2016) also. Because of the inadequate available literature for OHWR in Gurmukhi script, it is very difficult to find existing work on online handwritten Gurmukhi words recognition. The only available work for an end-to-end OHWR in Gurmukhi script was done by Singh and Sharma (2019) in 2019, where proposed work dealt the unsolved problem of online handwritten Gurmukhi word recognition in real situations and proposed a four-step novel model based on a minimal set of words for OHWR of Gurmukhi script in four different data/writer dependent and independent modes of writing. Recently, in 2020, Singh et al. (2020) proposed a novel feature extraction technique to recognize online handwritten Gurmukhi strokes based on self controlled Ramer–Douglas–Peucker (RDP) algorithm, and achieved above 90% recognition accuracy on Gurmukhi dataset (Singh et al., 2016). This study overcome the shortcomings of the traditional chain code based feature extraction approach, and prepared a smaller length feature vector for handwritten strokes without preprocessing and without any control parameter to RDP.

3. Deep learning architecture

In the last few years, DL based networks has led to great performance on a variety of problems. The DL architectures have been successfully applied in different practical applications as computer vision, big data analysis, speech recognition, acoustic modeling for audio classification (Dong et al., 2021; Gu et al., 2018; Schmidhuber, 2015; Szegedy et al., 2014), image editing and video enhancement, statistical learning, recommendation systems (Liu et al., 2017), event detection, action recognition, intelligent surveillance, multimedia understanding (Yao et al., 2019), and cybersecurity (Dixit & Silakari, 2021). The DL architecture has demonstrated a powerful framework for background subtraction in video acquired by static cameras (Bouwman et al., 2018, 2019). Among different types DL networks used in these applications, CNNs have been most extensively studied and used (Gu et al., 2018). The research on CNN has emerged rapidly and attained the state-of-the-art results on various tasks too.

The DL architectures have also been widely used for pattern recognition tasks and achieved good performance for image recognition (Krizhevsky et al., 2012; Szegedy et al., 2015), face recognition (Taigman et al., 2014), text recognition (Ciregan et al., 2012; Ciresan et al., 2011; Goodfellow et al., 2013; Simard et al., 2003; Wang et al., 2012) and human pose estimation (Tompson et al., 2014). These architectures have also fundamentally altered the traditional form of pattern recognition and made an important development in several handwriting recognition tasks also. The major breakthrough of DL that the present study has also employed, the feature extraction and classification can be done automatically within DL models.

Different DL architectures as: CNN and RNN: LSTM and BLSTM specifically designed for sequential data, have been used in existing studies to perform text recognition in online and offline mode of writing in different scripts. The major problem with the RNN deep network is the vanishing gradient problem (Bengio et al., 1994) due to decay of output layer as it cycles through the recurrent connections of the network. This problem causes for access of incomplete range of contextual information by RNN, and the architecture of RNN cannot retain contextual information for a long period of time.

3.1. Convolutional Neural Network

Amongst different DL models, the CNN is the most widely used, especially in image recognition. The CNN is a particular type of multi-layer NN. Like other networks, CNNs are also trained by using back propagation algorithms, and the dissimilarity is in their architectures (Lecun et al., 1998; Simard et al., 2003). The CNN is a great fit for representation of an image structure, as image pixels have strong relation to their neighboring pixels and have little correlation with the far away pixels. Further the weight sharing strategy of CNN ensures that different parts of an image can share similar properties as texture and brightness. CNN can effectively abstract and extract 2D features. The CNN max-pooling layer can effectively absorb shape variations. Further, sparse connection with tied weights makes it possible to involve CNN with fewer parameters than a similar sized fully connected network. Most significantly, CNN can be trained with the gradient-based learning algorithm and it suffers less from the diminishing gradient problem. The handwritten text recognition using CNN is more challenging task than image recognition since the characters and words commonly have various appearances according to different writers, writing styles and writing surfaces. Simply by using CNN, promising results can be obtained for text recognition. Nevertheless, in order to get best results for handwritten text recognition using CNN, there are various other issues which need to be addressed for CNN as there is always need to choose the appropriate CNN based framework for handwriting recognition. For handwritten text recognition, CNN was initially applied for recognition of digits (LeCun et al., 1998). Then CNN and its variants are gradually adopted to various other handwriting applications.

3.1.1. CNN architecture

The CNN architecture consists of two key parts as feature extraction and classification. In feature extraction part, every layer of the CNN network obtains the output from its instant preceding layer as inputs and passes current output as an input to the immediate next layer, where classifier makes the predicted outputs associated with the input data. The CNN architecture has two basic layers as: convolution and pooling layers (LeCun et al., 1998). Every node of convolution layer performs the convolution operation on the input nodes and extracts the features from input. The max-pooling layer reduce the size of the network through maximum/average operation on input nodes. The outputs of $n - 1$ th layer are employed as input to n th layer, where the inputs pass through kernels set trailed by nonlinear function ReLU. The advanced CNN architectures consist of a stack of convolutional layers and max-pooling layers followed by fully connected and softmax layer at the end. Overall an efficient CNN architecture can be constructed using fundamental components as convolution layer, pooling layer, fully connected layer, and Softmax layer.

3.2. Recurrent neural network

An RNN is a class of DL networks where connections in nodes form a directed graph along a temporal sequence and it permits for exhibition of temporal dynamic behavior. An RNN is also derived from feed forward NN and it can process variable length sequences of inputs by using their internal state. This property of RNN makes it applicable

to unsegmented tasks as connected handwriting recognition (Graves et al., 2009) and speech recognition (Li & Wu, 2015; Sak et al., 2014). Recurrent neural network term refers to two broad network classes having alike general structure, where one RNN is a directed acyclic graph that can be unrolled and replaced with a strictly feed forward neural network, and other RNN is a directed cyclic graph that cannot be unrolled. Both types of RNN can have additional stored states, and the NN can directly control the storage. Other network or graph can also replace the storage, if it incorporates time delays or has feedback loops. These controlled states are referred as gated memory/state, and these are part of LSTMs and gated recurrent units.

3.2.1. RNN architecture

The RNNs are different as a variety of operations over a sequence of vectors over time are allowed by them. There are also different RNN architectures with respect to its applications. RNN architectures are classified into one to one, one to many, many to one and many to many architectures. One to one is a standard mode of classification without RNN and it is commonly used for image classification. One to many RNN takes a single input and produces a sequence of outputs, and it is commonly used for image captioning jobs where a set of caption words output is needed for a single input image. Many to one RNN architecture has a sequence of inputs and produces one output only. It is mostly used in those cases where inputs are given as sentences or set of words and output is given in form of a positive/negative expression. Many to many RNN architecture gives a sequence of outputs for a sequence of inputs and it is mostly used for video classification and machine translation jobs.

3.2.2. Long-Short Term Memory

An LSTM (Graves et al., 2009; Hochreiter & Schmidhuber, 1997) is a special type of RNN architecture, and it is particularly designed to overcome the vanishing gradient problem (Bengio et al., 1994). An LSTM hidden layer consists of recurrently connected memory blocks, where each block has one or more recurrently connected memory cells, for which three multiplicative gates: input gate, output gate, and forget gate are used to control the activation. These gates make it possible to access and store the contextual information over a long period of time. More specifically, the cell activation is not overwritten by new inputs until the input gate remains closed. Correspondingly, as long as the output gate stays open, the cell activation is accessed by the rest of the network and the activation cell's recurrent connection is controlled by forget gate. Similar to the CNN architecture, it has multiple forward and backward layers in each LSTM layer, and multiple feature maps at the output layer, and stacking of multiple LSTM layers with the use of max-pooling sub-sampling.

3.2.3. LSTM architecture

LSTM is an RNN architecture that takes values over arbitrary intervals. LSTM is well suitable to classify, process and predict time series given time lags of unknown duration. The weights in LSTM are updated using back propagation through time training algorithm. The cell state called gates is the main scheme of LSTM. The information can be added and removed to gates by LSTM and the input gate, output gate and forget gate are defined as following:

$$\begin{aligned} f_t &= \alpha(W_f.[h_{t-1}, x_t] + b_f), \\ i_t &= \alpha(W_i.[h_{t-1}, x_t] + b_i), \\ \tilde{C}_t &= \tanh(W_C.[h_{t-1}, x_t] + b_C), \\ C_t &= f_t * C_{t-1} + i_t * \tilde{C}_t, \\ O_t &= \alpha(W_O.[h_{t-1}, x_t] + b_O), \\ h_t &= O_t * \tanh(C_t). \end{aligned}$$

3.2.4. Bidirectional long-short term memory

In many pattern recognition problems, it needs to access the past and future contexts together. As an illustration, in handwriting recognition problems, knowing the characters that appear both to the left and right of it helps in character recognition. Bidirectional RNNs (BRNNs) (Schuster & Paliwal, 1997) can be used to obtain the contextual information in both the directions along the input sequence. BRNNs have two different hidden layers, where one layer is used to process the input sequence in the forward direction and the other in the backward direction. BRNN provides context to past and future of each point in the sequence because both hidden layers are linked to the same output layer. BRNNs were successfully applied for protein structure prediction and speech processing (Schuster & Paliwal, 1997), and BRNNs outperformed the standard RNNs in various sequence learning tasks. BLSTM are Combination of BRNNs and LSTM.

3.2.5. BLSTM architecture

BLSTMs process the input sequences in two directions having two sub layers to consider the full input context. Two sub layers of BLSTM can perform computations in both forward (\vec{h}) and backward (\overleftarrow{h}) hidden layers. Both \vec{h} and \overleftarrow{h} are combined to compute the output sequence (y) as following:

$$\begin{aligned} \vec{h}_t &= H(W_{x\vec{h}}x_t + W_{\vec{h}\vec{h}}\vec{h}_{t-1} + b_{\vec{h}}) \\ \overleftarrow{h}_t &= H(W_{x\overleftarrow{h}}x_t + W_{\overleftarrow{h}\overleftarrow{h}}\overleftarrow{h}_{t+1} + b_{\overleftarrow{h}}) \\ y_t &= (W_{hy}\vec{h}_t + W_{\overleftarrow{h}y}\overleftarrow{h}_t + b_h) \end{aligned}$$

Though the introduction of deep networks have seen a dramatic increase in the performance of recognition systems, but there are certain issues of these networks, like nonconvex training problem results to optimization algorithms may not return a global minima (Haeffele & Vidal, 2017; Yun et al., 2018). The robustness of successful convolutional and recurrent, deep neural networks is seldom appraised, especially after the attainment of high classification accuracy (Sengupta & Friston, 2018). The robustness of deep networks can also be improved using stability training (Giryes et al., 2015; Zheng et al., 2016). The numerical instabilities in derivative-based learning algorithms is one of critical issues of deep architectures, and it can be solved by using forward propagation techniques (Haber & Ruthotto, 2017). The deep networks based classifiers are also vulnerable to human-imperceptible adversarial perturbations, and it results DNN classifiers to output wrong predictions with high confidence. The robust detection of adversarial attacks can be done by modeling the intrinsic properties of deep networks (Zheng & Hong, 2018).

4. Architecture used

This section discusses the neural network architectures used for handwriting recognition from online and offline data. As CNN, also called ConvNet, are among the most popular deep learning techniques, and it concentrates on capturing spatial and temporal relationship of the input data. Usually, CNN refers to 2-dimensional CNN (Conv2D) but it further has two more variants that are 1-dimensional CNN (Conv1D) and 3-dimensional CNN (Conv3D). Conv2D is very well suitable in image classification, has also given good results in handwriting recognition (Li et al., 2018) but they are heavily parameterized which makes them computationally very expensive and require large amount of memory, whereas Conv1D uses lesser number of parameters, needs lesser memory and computational power (Gan et al., 2019). Unlike Conv2D, which can require hours and days to train the model (Ding et al., 2018; Gan et al., 2019), Conv1D can be trained in few minutes and a single machine without GPUs can be used to trained it. Thus Conv1D, which is very well suitable for problems as time series data, is also a very good choice in online handwriting recognition.

For recognition of online handwriting the present study used a network, for which Fig. 2 represents architecture for learning on online

Layer (type)	Output Shape	Param #
conv1d_21 (Conv1D)	(None, 3629, 64)	448
conv1d_22 (Conv1D)	(None, 3629, 64)	12352
max_pooling1d_6 (MaxPooling1D)	(None, 1814, 64)	0
conv1d_23 (Conv1D)	(None, 1814, 128)	24704
conv1d_24 (Conv1D)	(None, 1814, 128)	49280
max_pooling1d_7 (MaxPooling1D)	(None, 907, 128)	0
conv1d_25 (Conv1D)	(None, 907, 256)	98560
conv1d_26 (Conv1D)	(None, 907, 256)	196864
conv1d_27 (Conv1D)	(None, 907, 256)	196864
max_pooling1d_8 (MaxPooling1D)	(None, 453, 256)	0
flatten_3 (Flatten)	(None, 115968)	0
dense_9 (Dense)	(None, 512)	59376128
dropout_12 (Dropout)	(None, 512)	0
dense_10 (Dense)	(None, 512)	262656
dropout_13 (Dropout)	(None, 512)	0
dense_11 (Dense)	(None, 35)	17955
Total params: 60,235,811		
Trainable params: 60,235,811		
Non-trainable params: 0		

Fig. 2. CNN-DNN Architecture for online learning.

data. The network comprises of three blocks of CNN layers: first block has two Conv1D layers with 64 filters and one MaxPooling1D layer, second block has two Conv1D layers with 128 filters followed by one MaxPooling1D layer, and the third layer has three Conv1D layers with 256 filters followed by one MaxPooling1D layer. The output of CNN layers is flattened and passed through two fully connected Dense layers with 512 neurons and each followed by a dropout of 30% to avoid overfitting of the network. Finally, output of Dense layers is passed to another Dense layer with neurons equal to the number of classes which is called the output layer. The present study has used Conv1Ds with kernel size of 3, padding ‘same’ and rectified linear unit (ReLU) as an activation function and ReLU is the most commonly used activation function in deep neural networks, especially for CNNs. As these layers contain fewer parameters in comparison to 2-dimensional CNN (Conv2Ds), so these are computationally very efficient. The Maxpooling1D layer helps by reducing model size by half, where it sub-samples by receiving the maximum of two adjacent values in the CNN. The dropout layer helps in controlling the overfitting by dropping some of the inputs. The output layer uses softmax as the activation function for classification and provides class probabilities for each class. The network uses categorical entropy as loss function and RMSprop optimizer to solve the optimization problem. ReLU and softmax activation functions are defined below.

$$ReLU(x) = \max(0, x), \quad (1)$$

$$softmax(x)_i = \frac{\exp(x_i)}{\sum_j \exp(x_j)} \quad (2)$$

Fig. 3 represents architecture for learning on offline data. Deep learning needs large amount of data for learning so to make for that we use transfer learning from image datasets. Transfer learning helps to use deep learning techniques without requiring much data (Zhuang et al., 2020). Pre-trained deep learning models are trained on very large datasets for large number of classes, e.g., VGG16 is trained on ImageNet, which is a dataset of over 14 million images with 1000 classes. These pre-trained models are trained for complex image classification tasks but our dataset images consists of simple lines with classes. So

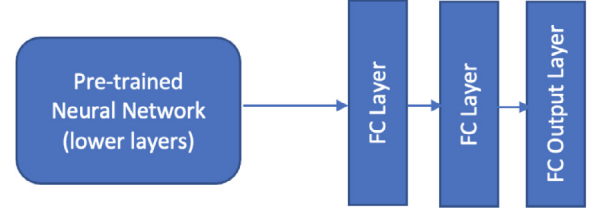


Fig. 3. Pre-trained-DNN Architecture for offline learning.

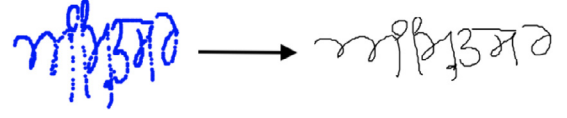


Fig. 4. Representation of a word in online(left side) and offline dataset (right side).

we use only lower level layers of these model for learning low level features from the images, which are then flattened and passed through two fully connected dense layers of 512 neurons before passing through the output layer. The output layer is also a fully connected layer with neurons equal to number of classes. We also use dropout layers with 40% value after two fully connected (FC) layers to avoid the over-fitting of the model. The fully connected dense layers use ReLU and output layer uses softmax activation functions. Since learning task is a multi-class classification problem so we use categorical cross-entropy as loss function, and RMSprop as an optimizer.

5. Experimentation

The present study provided the benchmarks results for online handwritten Gurmukhi word recognition using deep learning architecture with word based labeling approach in online mode of handwriting. Further, the overall results for each deep learning architectures are also evaluated over different interval of number of iterations.

5.1. Database and experimental setting

As a result of freely unavailable benchmarked datasets for OHWR in major Indic scripts, the development of datasets to train and assess the recognition model in Indic scripts is the first challenge to be met. For OHWR in Gurmukhi script, the only available benchmarked dataset was created by Singh et al. (2016) and Singh and Sharma (2019). Thus the present work used the benchmarked dataset of online handwritten Gurmukhi words developed by Singh et al. (2016) and Singh and Sharma (2019). The minimal set of words were used to train and test the recognizer (Singh et al., 2016). The Gurmukhi dataset (Singh & Sharma, 2019; Singh et al., 2016) was collected using pen tablet and digital writing board devices, where a sensor picks up the pen-up, pen-down and pen-tip movements (x, y). The information such as current position, movement direction, starting and stopping points, and (x,y) coordinate information of plotted points was obtained in data capturing process. The x and y coordinate values of collected data points of strokes change as pen moves to left/right and upward/downward, respectively.

The minimal set of words dataset “Online Handwritten Gurmukhi Strokes Dataset Based on Minimal Set of Words” (Singh et al., 2016) for OHWR was formed by collecting handwriting samples of hundred writers of different genders, handedness, expertise level and geographical locations. Singh et al. (2016) dataset contains a total number 39411 strokes contributed by hundred writers for 3.5K words, and the present OHWR study used original word data instead of stroke data for holistic approach based training of deep learning architectures of Gurmukhi word recognition. The (Singh & Sharma, 2019) dataset for OHWR study

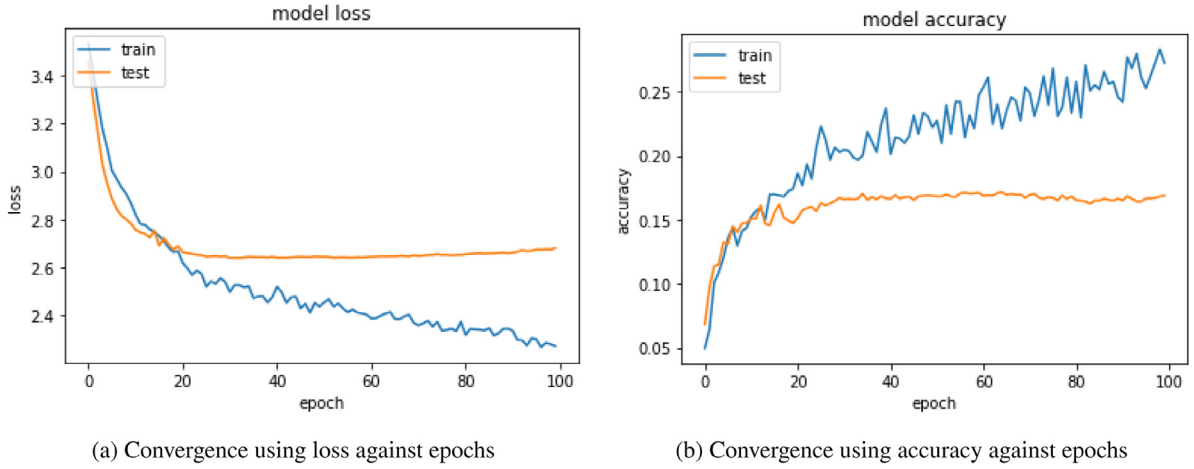


Fig. 5. Converge of CNN-DNN on online dataset.

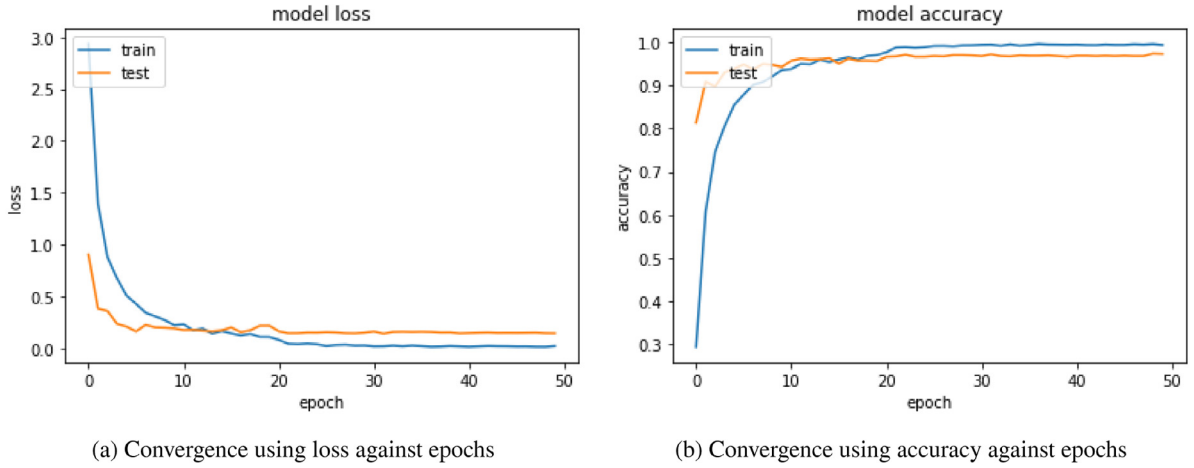


Fig. 6. Converge of VGG16-DNN on offline dataset.

of Gurmukhi script, developed by Singh and Sharma was collected for four different dependent and independent writing environments. This dataset is split into four parts as: data dependent and writer dependent environment, data dependent and writer independent, data independent and writer dependent environment, and data independent and writer independent. This dataset comprises 7.5K words comprising 74330 strokes (Singh & Sharma, 2019). The validation of present study has been done using both writer dependent and writer independent data in data dependent mode of Singh et al. (2016) and Singh and Sharma (2019) dataset collectively, and a total of 6K words dataset used for validation of this study.

Fig. 4 represents one sample of online data (left) and corresponding offline sample (right). We have converted online samples by joining the consecutive points of the online samples. The continuity in the offline samples helps in improving the learning. The two words appears of different sizes, although they represent the same sample, because online samples are plotted as they are written by the writers but offline samples are plotted on to a fixed image size which is calculated by average length and width of the whole online dataset. In this way, the online data is converted from series of (x, y) pairs to images which enable us to utilize transfer learning techniques.

The deep learning models are implemented using Keras library, and all the experiments are executed on a Linux server with 64 GB RAM and 56 CPUs and one GPU. We have split the dataset into 80:20 ratio for training and testing, respectively. Moreover, while splitting the dataset into train and test datasets, we have ensured that samples are divided in right proportion for writer efficiency of the language. We have selected,

using trial-and-error, a mini-batch of 32 samples, a custom learning rate which varies from 0.00005 to 0.000005. We have provided experiments to justify the choice of optimizer, mini-batch size, and also provided experiments to study train-test data splitting of 90:10, 70:30 and 60:40. Each experiment is run for 50 epochs because the proposed model converges very fast to the final solution. The performance metrics used for the experiments is only accuracy because the current recognition task is class-balanced and other metrics like recall, precision and F1-score gives results similar to accuracy. Moreover, we have reported the accuracy obtained at the last epoch.

5.2. Learning on online data

SVM and logistic regression are used as a baseline to compare the online recognition results with the deep learning techniques. Table 3 represents results with the online dataset. As it is clear from the table, deep learning cannot even beat the baseline techniques, in fact, all the technique perform poorly. SVM performs best followed by LR and CNN-DNN, respectively. The poor performance of models is because we are passing the complete word to the models which has several characters/strokes, and due to the small dataset as compared with the number of features.

Fig. 5 represents the convergence of CNN-DNN model on the online dataset. 5(a) shows convergence using loss or error against number of epochs and 5(b) shows convergence using accuracy against number of epochs. From both the figures, it is clear that, model stops improving on the test dataset but keeps on improving on the training dataset,

Table 3

Learning from online data.

Model	Results
SVM	27.7%
Logistic Regression	17.00%
CNN-DNN	16.87%

Table 4

Learning from offline data.

Model	Results
SVM	86.9%
Logistic Regression	84.5%
VGG16-DNN	97.23%
InceptionV3-DNN	97.06%

i.e., model overfits on the online dataset due to the limited size of the datasets. Techniques to avoid over-fitting like dropout is used but that does not improve the performance of the model. That is why the online dataset is converted to offline dataset where transfer learning can be used to deal with limited size of the dataset.

5.3. Learning on offline data

We have used two pre-trained neural network, viz., VGG16 (Simonyan & Zisserman, 2014) and InceptionV3 (Szegedy et al., 2016), based deep learning architectures, called as VGG16-DNN and InceptionV3-DNN, for handwriting recognition on the offline dataset. This is because our handwriting recognition task is simpler as compared with ImageNet dataset on which VGG16 and InceptionV3 are trained and we need only low level features from the lower layers of these pre-trained models. The convergence of VGG16-DNN model is shown in Fig. 6 using loss and accuracy against number of epochs over training and test datasets. The figure clearly shows very nice convergence, in few epochs, due to transfer learning and use of dropout to handle the over-fitting. We notice a huge improvement in the handwriting recognition due to two main factors. First, transfer learning helps to deal with limited data issue and helps in improving the performance, and second, the quality of dataset is improved by converting from online to offline. This is because offline data samples have more continuity due to joining the points on the online samples.

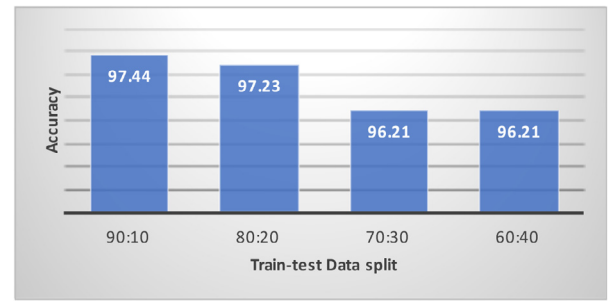
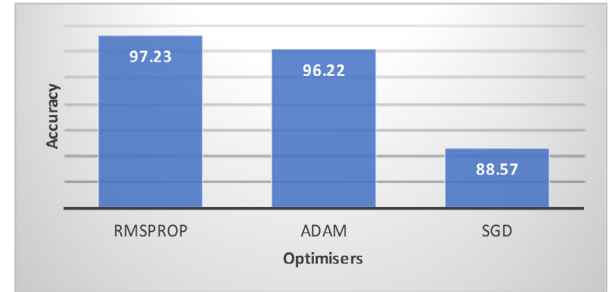
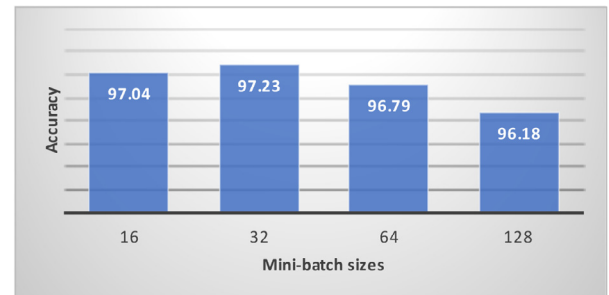
In terms of training time, the proposed model is quite fast as it converges in few epochs, where each epoch takes around 110 s. So the handwriting recognition task is executed in few minutes.

Table 4 compares the recognition rate of deep learning techniques with baseline methods, i.e., SVM and LR. It is observed that performance of all the methods have greatly improved by converting online data to offline. Moreover, deep learning techniques are outperforming than SVM and LR due to transfer learning, and SVM performing better than LR. VGG16-DNN gives slightly better results than InceptionV3-DNN because InceptionV3 is more complex for the handwriting recognition than VGG16.

We analyze the effect of train-test split with the best model, i.e., VGG16-DNN, and the results are represented in Fig. 7. We observe two points from the analysis: first, performance of the model increases with increase in the size of train data. This is because now model has more samples to learn from. Second, by reducing the train set from 90% to 60%, the accuracy dropped slightly. This clearly shows the power of transfer learning which helps apply deep learning to tasks with limited datasets.

Fig. 8 studies the use of different optimizers to solve the optimization problem associated with network and justifies our choice of optimizer. From the figure, it is clear that RMSprop performs better than widely used Adam, and SGD (Stochastic Gradient Descent) performs the worst in the current experimental settings.

Fig. 9 studies the effect of mini-batch size on the recognition task in hand and justifies the choice of mini-batch in the experiments. It is

**Fig. 7.** Analysis of dataset splitting.**Fig. 8.** Analysis of optimizers.**Fig. 9.** Analysis of mini-batch sizes.

observed that mini-batch does not significantly affect the recognition rate, although smaller mini-batches take more time to learn.

The present study uses the writer dependent/independent data in data dependent mode of online Gurmukhi handwriting for validation and there is only one available study for online handwritten Gurmukhi word recognition is Singh and Sharma (2019) where 85.98% and 84.80% recognition results are achieved in data dependent environment. The present results as 97.44%, 97.23% and 96.21% for 90:10, 80:20, 70:30 train and test data, respectively, are achieved for data dependent mode of online Gurmukhi handwriting, which outperformed the existing results by more than 10%. As there are not much available results for online Gurmukhi handwriting in this direction, so it is hard to compare our recognition results to a benchmark except one study (Singh & Sharma, 2019). But, there is a great scope of applying similar word recognition approach in other Indic scripts as Devanagari and Tamil scripts also, especially for applications requiring city names online handwriting recognition.

6. Conclusion

The handwriting recognition is the place of birth for DL, but it is a bit not satisfactory that the recent wave of leaps in accomplishment of results using DL in computer vision has so far missed the handwritten

text recognition greatly. It is also not inherently challenging for handwriting recognition that prevents machines from achieving a level of performance in the same way that is accomplished for face recognition and object recognition metrics (He et al., 2015; Taigman et al., 2014). The achievement of DL studies for computer vision, suggest that a successful handwritten text recognition system can be made without spending much efforts in atomic tasks, and it follows an end-to-end trend of DL. Similarly, the present study is a major step in this direction that recognized handwritten text using a different DL architecture and provided benchmark results for online handwritten Gurmukhi words in data dependent writing mode where best results achieved above 97% recognition accuracy. The future Indic script handwriting recognition systems based on DL can also benefit from this study by utilizing multiple tactics depending on the nature of the script. Looking forward, the final goal of handwritten text recognition is the construction of a universal reading system, which is adaptable not only for different scribes, but also for different script types. In such systems, the present study will make the DL based handwritten text recognition of the next script much easier than training from scratch.

CRedit authorship contribution statement

Sukhdeep Singh: Conceptualization, Methodology, Writing original draft, Data curation. **Anuj Sharma:** Supervision, Quality check, Writing -review & editing. **Vinod Kumar Chauhan:** Conceptualization, Methodology, Software.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

References

- Abandah, G., & M. Malas, T. (2010). Feature selection for recognizing handwritten arabic letters. *Dirasat Engineering Sciences Journal*, 37.
- Al-Taani, D. A., & Al-Haj, S. (2010). Recognition of on-line arabic handwritten characters using structural features. *Journal of Pattern Recognition Research*, 5, 23–37. <http://dx.doi.org/10.13176/11.217>.
- Altman, N. S. (1992). An introduction to kernel and nearest-neighbor nonparametric regression. *The American Statistician*, 46, 175–185.
- Arica, N., & Yaman-Vural, F. T. (2001). An overview of character recognition focused on off-line handwriting. *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews)*, 31, 216–233. <http://dx.doi.org/10.1109/5326.941845>.
- Arora, A., & Nambodiri, A. M. (2010). A hybrid model for recognition of online handwriting in Indian scripts. In *2010 12th international conference on frontiers in handwriting recognition* (pp. 433–438). <http://dx.doi.org/10.1109/ICFHR.2010.74>.
- Artieres, T., Marukat, S., & Gallinari, P. (2007). Online handwritten shape recognition using segmental hidden markov models. *Pattern Analysis and Machine Intelligence*, 29, 205–217.
- Babu, V. J., Prasanth, L., Sharma, R. R., & Bharath, A. (2007). Hmm-based online handwriting recognition system for telugu symbols. In *Proceedings of 9th international conference on document analysis and recognition* (vol. 1) (pp. 63–67).
- Bengio, Y., Simard, P., & Frasconi, P. (1994). Learning long-term dependencies with gradient descent is difficult. *IEEE Transactions on Neural Networks*, 5, 157–166. <http://dx.doi.org/10.1109/72.279181>.
- Benouareth, A., Ennajib, A., & Sellamia, M. (2008). Semi-continuous hmms with explicit state duration for unconstrained arabic word modeling and recognition. *Pattern Recognition Letters*, 29, 1742–1752.
- Bharath, A., & Madhvanath, S. (2007). Hidden markov models for online handwritten tamil word recognition. In *Proceedings of 9th international conference on document analysis and recognition*.
- Bharath, A., & Madhvanath, S. (2010). Online handwriting recognition for indic scripts (pp. 209–234). http://dx.doi.org/10.1007/978-1-84800-330-9_11.
- Bharath, A., & Madhvanath, S. (2012). Hmm-based lexicon-driven and lexicon-free word recognition for online handwritten indic scripts. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 34, 670–682.
- Bhaskarabhatla, A. S., & Madhvanath, S. (2004). Experiences in collection of handwriting data for online handwriting recognition in indic scripts. In *Proceedings of 4th international conference on language resources and evaluation*.
- Bhattacharya, U., Gupta, B. K., & Parui, S. K. (2007). Direction code based features for recognition of online handwritten characters of bangla. In *Proceedings of the 9th international conference on document analysis and recognition* (pp. 58–62).
- Bhattacharya, S., Maitra, D. S., Bhattacharya, U., & Parui, S. K. (2016). An end-to-end system for bangla online handwriting recognition. In *Proceedings of 15th international conference on frontiers in handwriting recognition* (pp. 373–378). <http://dx.doi.org/10.1109/ICFHR.2016.0076>.
- Bhattacharya, U., Nigam, A., Rawat, Y. S., & Parui, S. K. (2008). An analytic scheme for online handwritten bangla cursive word recognition. In *Proceedings of 11th international conference on frontiers in handwriting recognition* (pp. 320–325).
- Bhattacharya, N., & Pal, U. (2012). Stroke segmentation and recognition from bangla online handwritten text. In *International conference on frontiers in handwriting recognition* (pp. 740–745). <http://dx.doi.org/10.1109/ICFHR.2012.275>.
- Bhattacharya, N., Pal, U., & Kimura, F. (2013). A system for bangla online handwritten text. In *Proceedings of 12th international conference on document analysis and recognition* (pp. 1335–1339). <http://dx.doi.org/10.1109/ICDAR.2013.270>.
- Bhattacharya, N., Pal, U., & Roy, P. P. (2017). Cleaning of online bangla free-form handwritten text. *ACM Transactions on Asian and Low-Resource Language Information Processing*, 17, 8:1–8:25. <http://dx.doi.org/10.1145/3145538>, URL <http://doi.acm.org/10.1145/3145538>.
- Bhattacharya, N., Roy, P. P., Pal, U., & Setua, S. K. (2018). Online bangla handwritten word recognition. *Malaysian Journal of Computational Science*, 31, 300–310. <http://dx.doi.org/10.22452/mjcs.vol31no4.4>, URL <https://ejournal.um.edu.my/index.php/MJCS/article/view/14243>.
- Bilgin-Tasdemir, E., & Yanikoglu, B. (2018). A comparative study of delayed stroke handling approaches in online handwriting. *International Journal on Document Analysis and Recognition (IJ DAR)*, 22, 15–28. <http://dx.doi.org/10.1007/s10032-018-0313-2>.
- Bluche, T., Ney, H., & Kermorvant, C. (2013). Feature extraction with convolutional neural networks for handwritten word recognition. In *Proceedings of the 12th international conference on document analysis and recognition* (pp. 285–289).
- Bouwman, T., Javed, S., Sultana, M., & Jung, S. (2018). Deep neural network concepts for background subtraction: A systematic review and comparative evaluation.
- Bouwman, T., Javed, S., Sultana, M., & Jung, S. K. (2019). Deep neural network concepts for background subtraction: a systematic review and comparative evaluation. *Neural Networks*, 117, 8–66. <http://dx.doi.org/10.1016/j.neunet.2019.04.024>, URL <https://www.sciencedirect.com/science/article/pii/S0893360819301303>.
- Carbune, V., Gonnet, P., Deselaers, T., Rowley, H., Daryin, A., Calvo, M., Wang, L. L., Keysers, D., Feuz, S., & Gervais, P. (2019). Fast multi-language lstm-based online handwriting recognition.
- Chakraborty, B., & Chakraborty, G. (2002). A new feature extraction technique for on-line recognition of handwritten alphanumeric characters. *Information Sciences*, 148, 55–70. [http://dx.doi.org/10.1016/S0020-0255\(02\)00276-1](http://dx.doi.org/10.1016/S0020-0255(02)00276-1), URL <http://www.sciencedirect.com/science/article/pii/S0020025502002761>.
- Chauhan, V. K., Dahiya, K., & Sharma, A. (2019). Problem formulations and solvers in linear svm: a review. *Artificial Intelligence Review*, 52, 803–855.
- Chowdhury, K., Alam, L., Sarmin, S., Arefin, S., & Hoque, M. M. (2015). A fuzzy features based online handwritten bangla word recognition framework. In *2015 18th international conference on computer and information technology* (pp. 484–489). <http://dx.doi.org/10.1109/ICCITech.2015.7488119>.
- Chowdhury, S. D., Bhattacharya, U., & Parui, S. K. (2013). Online handwriting recognition using levenshtein distance metric. In *Proceeding of 12th international conference on document analysis and recognition* (pp. 79–83).
- Ciregan, D., Meier, U., & Schmidhuber, J. (2012). Multi-column deep neural networks for image classification. In *2012 IEEE conference on computer vision and pattern recognition* (pp. 3642–3649). IEEE.
- Ciresan, D. C., Meier, U., Gambardella, L. M., & Schmidhuber, J. (2011). Convolutional neural network committees for handwritten character classification. In *Proceedings of the international conference on document analysis and recognition* (pp. 1135–1139).
- Cortes, C., & Vapnik, V. (1995). Support-vector networks. *Mach. Learn.*, 20, 273–297. <http://dx.doi.org/10.1023/A:1022627411411>.
- Dan, C., Ueli, M., & Jurgen, S. (2012). Multi-column deep neural networks for image classification. *IEEE conference on computer vision and pattern recognition* (pp. 3642–3649).
- Deepu, V., Madhvanath, S., & Ramakrishnan, A. G. (2004). Principal component analysis for online handwritten character recognition. In *Proceedings of 17th international conference on pattern recognition* (pp. 327–330).
- Ding, H., Chen, K., Hu, W., Cai, M., & Huo, Q. (2018). Building compact cnn-dblstm based character models for handwriting recognition and ocr by teacher-student learning. (pp. 139–144). <http://dx.doi.org/10.1109/ICFHR.2018.2018.00033>.
- Dixit, P., & Silakari, S. (2021). Deep learning algorithms for cybersecurity applications: A technological and status review. *Computer Science Review*, 39, Article 100317. <http://dx.doi.org/10.1016/j.cosrev.2020.100317>, URL <https://www.sciencedirect.com/science/article/pii/S1574013720304172>.
- Dong, S., Wang, P., & Abbas, K. (2021). A survey on deep learning and its applications. *Computer Science Review*, 40, Article 100379. <http://dx.doi.org/10.1016/j.cosrev.2021.100379>, URL <https://www.sciencedirect.com/science/article/pii/S1574013721000198>.
- Duda, R. O., Hart, P. E., & Stork, D. G. (2000). *Pattern classification*. NewYork: John Wiley and Sons INC..
- Dutta, K., Krishnan, P., Mathew, M., & Jawahar, C. V. (2018). Improving cnn-rnn hybrid networks for handwriting recognition. In *2018 16th international conference on frontiers in handwriting recognition* (pp. 80–85). <http://dx.doi.org/10.1109/ICFHR-2018.2018.00023>.

- Fink, G. A., Vajda, S., Bhattacharya, U., Parui, S. K., & Chaudhuri, B. B. (2010). Online bangla word recognition using sub-stroke level features and hidden markov models. In *2010 12th international conference on frontiers in handwriting recognition* (pp. 393–398). <http://dx.doi.org/10.1109/ICFHR.2010.68>.
- Fink, G. A., Vajda, S., Bhattacharya, U., Parui, S. K., & Chaudhuri, B. B. (2010). Online bangla word recognition using sub-stroke level features and hidden Markov models. (pp. 393–398).
- Frinken, V., Bhattacharya, N., & Pal, U. (2014a). Design of unsupervised feature extraction system for on-line bangla handwriting recognition. In *International workshop on document analysis systems* (pp. 355–359).
- Frinken, V., Bhattacharya, N., Uchida, S., & Pal, U. (2014b). Improved blstm neural networks for recognition of on-line bangla complex words. Structural, syntactic, and statistical. *Pattern Recognition*, 404–413.
- Gan, J., Wang, W., & Lu, K. (2019). A new perspective: Recognizing online handwritten Chinese characters via 1-dimensional cnn. *Information Sciences*, 478, 375–390.
- Ghosh, V., & Kabir, E. (2010). Feature extraction for online farsi characters. In *2010 12th international conference on frontiers in handwriting recognition* (pp. 477–482). <http://dx.doi.org/10.1109/ICFHR.2010.81>.
- Ghosh, D., Dube, T., & Shivaprasad, A. (2010). Script recognition—a review. *IEEE Transactions on pattern analysis and machine intelligence*, 32, 2142–2161.
- Ghosh, R., & Roy, P. P. (2015). A novel feature extraction approach for online bengali and devanagari character recognition. In *2015 2nd international conference on signal processing and integrated networks* (pp. 483–488). <http://dx.doi.org/10.1109/SPIN.2015.7095313>.
- Ghosh, R., & Roy, P. P. (2015). Study of two zone-based features for online bengali and devanagari character recognition. In *Proceedings of 13th international conference on document analysis and recognition* (pp. 401–405). <http://dx.doi.org/10.1109/ICDAR.2015.7333792>.
- Ghosh, R., & Roy, P. P. (2016). Comparison of zone-features for online bengali and devanagari word recognition using hmm. In *Proceedings of 15th international conference on frontiers in handwriting recognition* (pp. 435–440). <http://dx.doi.org/10.1109/ICFHR.2016.0087>.
- Ghosh, R., Vamshi, C., & Kumar, P. (2019). RNN based online handwritten word recognition in Devanagari and Bengali scripts using horizontal zoning. *Pattern Recognition*, 92, 203–218. <http://dx.doi.org/10.1016/j.patcog.2019.03.030>, URL <http://www.sciencedirect.com/science/article/pii/S0031320319301384>.
- Giryes, R., Sapiro, G., & Bronstein, A. M. (2015). On the stability of deep networks.
- Gohel, C. C., Goswami, M. M., & Prajapati, V. K. (2015). On-line handwritten gujarati character recognition using low level stroke. In *2015 third international conference on image information processing* (pp. 130–134). <http://dx.doi.org/10.1109/ICIIP.2015.7414753>.
- Goodfellow, I. J., Bulatov, Y., Ibarz, J., Arnoud, S., & Shet, V. (2013). Multi-digit number recognition from street view imagery using deep convolutional neural networks. arXiv preprint [arXiv:1312.6082](https://arxiv.org/abs/1312.6082).
- Graves, A., Liwicki, M., Fernández, S., Bertolami, R., Bunke, H., & Schmidhuber, J. (2009). A novel connectionist system for unconstrained handwriting recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 31, 855–868. <http://dx.doi.org/10.1109/TPAMI.2008.137>.
- Gu, J., Wang, Z., Kuen, J., Ma, L., Shahroudy, A., Shuai, B., Liu, T., Wang, X., Wang, G., Cai, J., & Chen, T. (2018). Recent advances in convolutional neural networks. *Pattern Recognition*, 77, 354–377. <http://dx.doi.org/10.1016/j.patcog.2017.10.013>, URL <https://www.sciencedirect.com/science/article/pii/S0031320317304120>.
- Haber, E., & Ruthotto, L. (2017). Stable architectures for deep neural networks. *Inverse Problems*, 34, Article 014004.
- Haefele, B. D., & Vidal, R. (2017). Global optimality in neural network training. In *2017 IEEE conference on computer vision and pattern recognition* (pp. 4390–4398). <http://dx.doi.org/10.1109/CVPR.2017.467>.
- Hanmandlu, M., Mohan, K. M., Chakraborty, S., Goyal, S., & Choudhury, D. (2003). Unconstrained handwritten character recognition based on fuzzy logic. *Pattern Recognition*, 36, 603–623. [http://dx.doi.org/10.1016/S0031-3203\(02\)00069-9](http://dx.doi.org/10.1016/S0031-3203(02)00069-9), URL <http://www.sciencedirect.com/science/article/pii/S0031320302000699>.
- He, K., Zhang, X., Ren, S., & Sun, J. (2015). Delving deep into rectifiers: Surpassing human-level performance on imagenet classification. In *Proceedings of the IEEE international conference on computer vision* (pp. 1026–1034).
- Hochreiter, S., & Schmidhuber, J. (1997). Long short-term memory. *Neural Computation*, 9, 1735–1780. <http://dx.doi.org/10.1162/neco.1997.9.8.1735>.
- Jaeger, S., Manke, S., Reichert, J., & Waibel, A. (2001a). Online handwriting recognition: the npen++ recognizer. *International Journal on Document Analysis and Recognition*, 3, 169–180.
- Jaeger, S., Manke, S., Reichert, J., & Waibel, A. (2001b). Online handwriting recognition: the npen++ recognizer. *International Journal on Document Analysis and Recognition*, 3, 169–180. <http://dx.doi.org/10.1007/PL00013559>.
- Kang, L., Kumar, J., Ye, P., Li, Y., & Doermann, D. (2014). Convolutional neural networks for document image classification. In *2014 22nd international conference on pattern recognition* (pp. 3168–3172). <http://dx.doi.org/10.1109/ICPR.2014.546>.
- Keshari, B., & Watt, S. M. (2008). Online mathematical symbol recognition using svms with features from functional approximation. *Mathematical user-interfaces workshop*.
- Keyers, D., Deselaers, T., Rowley, H. A., Wang, L. L., & Carbune, V. (2016). Multi-language online handwriting recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 39, 1180–1194. <http://dx.doi.org/10.1109/TPAMI.2016.2572693>.
- Keyers, D., Deselaers, T., Rowley, H. A., Wang, L., & Carbune, V. (2017). Multi-language online handwriting recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 39, 1180–1194. <http://dx.doi.org/10.1109/TPAMI.2016.2572693>.
- Kimura, F., Takashina, K., Tsuruoka, S., & Miyake, Y. (1987). Modified quadratic discriminant functions and the application to chinese character recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 9, 149–153.
- Kinjarapu, A. K., Yelavarti, K. C., & Valurouthu, K. P. (2016). Online recognition of handwritten telugu script characters. In *2016 International conference on signal processing, communication, power and embedded system* (pp. 426–432). <http://dx.doi.org/10.1109/SCOPES.2016.7955866>.
- Krizhevsky, A., Sutskever, I., & Hinton, G. E. (2012). Imagenet classification with deep convolutional neural networks. In *Advances in neural information processing systems* (pp. 1097–1105).
- Kumar, M., Jindal, M. K., & Sharma, R. K. (2011). k-nearest neighbor based off-line handwritten gurmukhi character recognition. In *2011 international conference on image information processing* (pp. 1–4). <http://dx.doi.org/10.1109/ICIIP.2011.6108863>.
- Kumar, R., & Sharma, R. (2013a). An efficient post processing algorithm for online handwriting Gurmukhi character recognition using set theory. *International Journal of Pattern Recognition and Artificial Intelligence*, 27, Article 1353002. <http://dx.doi.org/10.1142/S0218001413530029>.
- Kumar, R., & Sharma, R. K. (2013b). An efficient post processing algorithm for online handwriting gurmukhi character recognition using set theory. *International Journal On Pattern Recognition and Artificial Intelligence*, 27.
- Kumar, R., Sharma, R. K., & Sharma, A. (2014). Recognition of multi-stroke based online handwritten Gurmukhi aksharas. *Proceedings of the National Academy of Sciences, India Section A: Physical Sciences*.
- Kunihiko, F. (1980). Neocognitron: A self-organizing neural network model for a mechanism of pattern recognition unaffected by shift in position. *Biological Cybernetics*, 36, 193–202.
- Kunwar, R., Pal, U., & Blumenstein, M. (2014). Semi-supervised online bayesian network learner for handwritten characters recognition. In *Proceedings of 22nd international conference on pattern recognition* (pp. 3104–3109).
- Lajish, V. L., & Kopparapu, S. K. (2010). Fuzzy directional features for unconstrained on-line devanagari handwriting recognition. In *2010 national conference on communications* (pp. 1–5). <http://dx.doi.org/10.1109/NCC.2010.5430176>.
- Lecun, Y., Bottou, L., Bengio, Y., & Haffner, P. (1998). Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86, 2278–2324. <http://dx.doi.org/10.1109/5.726791>.
- LeCun, Y., Bottou, L., Bengio, Y., & Haffner, P. (1998). Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86, 2278–2324.
- Li, Z., Jin, L., & Lai, S. (2018). Rotation-free online handwritten Chinese character recognition using two-stage convolutional neural network. (pp. 205–210). <http://dx.doi.org/10.1109/ICFHR-2018.2018.00044>.
- Li, X., & Wu, X. (2015). Constructing long short-term memory based deep recurrent neural networks for large vocabulary speech recognition. In *2015 IEEE international conference on acoustics, speech and signal processing* (pp. 4520–4524). <http://dx.doi.org/10.1109/ICASSP.2015.7178826>.
- Liu, W., Wang, Z., Liu, X., Zeng, N., Liu, Y., & Alsaadi, F. E. (2017). A survey of deep neural network architectures and their applications. *Neurocomputing*, 234, 11–26. <http://dx.doi.org/10.1016/j.neucom.2016.12.038>, URL <https://www.sciencedirect.com/science/article/pii/S0925231216315533>.
- M., Parizeau, Lemieux, A., & Gagne, C. (2001). Character recognition experiments using unipen data. In *Proceedings of sixth international conference on document analysis and recognition* (pp. 481–485). <http://dx.doi.org/10.1109/ICDAR.2001.953836>.
- Madhvanath, S., & Govindaraju, V. (2001). The role of holistic paradigms in handwritten word recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 23, 149–164. <http://dx.doi.org/10.1109/34.908966>.
- Mandal, S., R. Mahadev. Prasanna, S., & Sundaram, S. (2019). An improved discriminative region selection methodology for online handwriting recognition. *International Journal on Document Analysis and Recognition (IJDAR)*, 22, 1–14. <http://dx.doi.org/10.1007/s10032-018-0314-1>.
- McCulloch, W. S., & Pitts, W. (1943). A logical calculus of ideas immanent in nervous activity. *Bulletin of Mathematical Biophysics*, 5, 115–133.
- Mondal, T., Bhattacharya, U., Parui, S. K., & Das, K. (2010). On-line handwriting recognition of indian scripts - the first benchmark. In *Proceeding of the 12th international conference on frontiers in handwriting recognition* (pp. 200–205).
- N. Murthy, V., & Ramakrishnan, A. (2011). Choice of classifiers in hierarchical recognition of online handwritten kannada and tamil aksharas. *Journal of Universal Computer Science*, 17, 94–106. <http://dx.doi.org/10.3217/jucs-017-01-0094>.
- Nethravathi, B., Archana, C. P., Shashikiran, K., Ramakrishnan, A. G., & Kumar, V. (2010). Creation of a huge annotated database for tamil and kannada ohr. In *Proceedings of international conference on frontiers in handwriting recognition* (pp. 415–420).
- P., Keshri, Kumar, P., & Ghosh, R. (2018). Rnn based online handwritten word recognition in devanagari script. In *16th International conference on frontiers in handwriting recognition* (pp. 517–522). <http://dx.doi.org/10.1109/ICFHR-2018.2018.00096>.
- Parui, S. K., Guin, K., Bhattacharya, U., & Chaudhuri, B. B. (2008). Online handwritten bangla character recognition using hmm. In *Proceedings of 19th international conference on pattern recognition* (pp. 1–4).

- Plamondon, R., & Srihari, S. N. (2000). Online and off-line handwriting recognition: a comprehensive survey. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 22, 63–84. <http://dx.doi.org/10.1109/34.824821>.
- Prasad, G. K., Khan, I., Chanukotimath, N. R., & Khan, F. (2012). On-line handwritten character recognition system for kannada using principal component analysis approach: For handheld devices. In *Proceedings of world congress on information and communication technologies* (pp. 675–678).
- Prasad, J. R., & Kulkarni, U. V. (2014). Gujarati character recognition using adaptive neuro fuzzy classifier. In *Proceedings of international conference on electron system signal process computer technology* (pp. 402–407).
- Prasad, M. M., Sukumar, M., & Ramakrishnan, A. G. (2009). Divide and conquer technique in online handwritten kannada character recognition. In *Proceedings of the international workshop on multilingual OCR* (pp. 11:1–11:7). <http://dx.doi.org/10.1145/1577802.1577815>, URL <http://doi.acm.org/10.1145/1577802.1577815>.
- Prasanth, L., Babu, V., Sharma, R., Rao, G. V., & M, D. (2007). Elastic matching of online handwritten tamil and telugu scripts using local features. In *Proceedings of the ninth international conference on document analysis and recognition* (vol. 02) (pp. 1028–1032).
- Rabiner, L. R. (1989). A tutorial on hidden markov models and selected applications in speech recognition. *Proceedings of IEEE*, 77, 257–286.
- Rampalli, R., & Ramakrishnan, A. (2011). Fusion of complementary online and offline strategies for recognition of handwritten kannada characters. *Journal of Universal Computer Science*, 17, 81–93. <http://dx.doi.org/10.3217/jucs-017-01-0081>.
- Ramya, S., & Shama, K. (2018). The effect of pre-processing and testing methods on online kannada handwriting recognition: Studies using signal processing and statistical techniques. *Pertanika Journal of Science and Technology*, 26, 671–690.
- Rani, M., & Meena, Y. (2011). An efficient feature extraction method for handwritten character recognition. (pp. 302–309). http://dx.doi.org/10.1007/978-3-642-27242-4_35.
- Ren, G., & Ganapathy, V. (2019). Recognition of online handwriting with variability on smart devices. In *2019 IEEE international conference on acoustics, speech and signal processing (ICASSP)* (pp. 7605–7609). <http://dx.doi.org/10.1109/ICASSP.2019.8682706>.
- Roy, K., Sharma, N., Pal, T., & Pal, U. (2007). Online bangla handwriting recognition system. In *Proceedings of international conference on advances in pattern recognition* (pp. 117–122).
- Sak, H., Senior, A., & Beaufays, F. (2014). Long short-term memory recurrent neural network architectures for large scale acoustic modeling. In *Proceedings of the annual conference of the international speech communication association* (pp. 338–342).
- Samanta, O., Bhattacharya, U., & Parui, S. K. (2014). Smoothing of HMM parameters for efficient recognition of online handwriting. *Pattern Recognition*, 47, 3614–3629. <http://dx.doi.org/10.1016/j.patrec.2014.04.019>, URL <http://www.sciencedirect.com/science/article/pii/S0031320314001654>.
- Samanta, O., Roy, A., Bhattacharya, U., & Parui, S. K. (2015). Script independent online handwriting recognition. In *Proceedings of 13th international conference on document analysis and recognition* (pp. 1251–1255). <http://dx.doi.org/10.1109/ICDAR.2015.7333964>.
- Schmidhuber, J. (2015). Deep learning in neural networks: An overview. *Neural Networks*, 61, 85–117. <http://dx.doi.org/10.1016/j.neunet.2014.09.003>, URL <https://www.sciencedirect.com/science/article/pii/S0893608014002135>.
- Schuster, M., & Paliwal, K. K. (1997). Bidirectional recurrent neural networks. *IEEE Transactions on Signal Processing*, 45, 2673–2681. <http://dx.doi.org/10.1109/78.650093>.
- Sen, S., Bhattacharya, A., Singh, P. K., Sarkar, R., Roy, K., & Doermann, D. (2018a). Application of structural and topological features to recognize online handwritten bangla characters. *ACM Transactions on Asian and Low-Resource Language Information Processing*, 17, 1–16.
- Sen, S., Chowdhury, S., Mitra, M., Schwenker, F., Sarkar, R., & Roy, K. (2018b). A novel segmentation technique for online handwritten bangla words. *Pattern Recognition Letters*, <http://dx.doi.org/10.1016/j.patrec.2018.02.008>, URL <http://www.sciencedirect.com/science/article/pii/S0167865518300436>.
- Sengupta, B., & Friston, K. J. (2018). How robust are deep neural networks? *arXiv:1804.11313*.
- Sharma, A. (2009). *Online handwritten Gurmukhi character recognition* (Ph.D. thesis), Patiala: Thapar University.
- Sharma, A., Kumar, R., & Sharma, R. K. (2008). Online handwritten Gurmukhi character recognition using elastic matching. In *Proceedings of congress on image and signal processing* (pp. 391–396).
- Sharma, A., Kumar, R., & Sharma, R. K. (2009). Rearrangement of recognized strokes in online handwritten Gurmukhi words recognition. In *Proceedings of 10th international conference on document analysis and recognition* (pp. 1241–1245).
- Simard, P. Y., Steinkraus, D., & Platt, J. C. (2003). Best practices for convolutional neural networks applied to visual document analysis. In *ICDAR*.
- Simistira, F., Katsouros, V., & Carayannis, G. (2015). Recognition of online handwritten mathematical formulas using probabilistic svms and stochastic context free grammars. *Pattern Recognition Letters*, 53, 85–92. <http://dx.doi.org/10.1016/j.patrec.2014.11.015>, URL <http://www.sciencedirect.com/science/article/pii/S0167865514003651>.
- Simonyan, K., & Zisserman, A. (2014). Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*.
- Singh, S., Chauhan, V. K., & Smith, E. H. B. (2020). A self controlled rdp approach for feature extraction in online handwriting recognition using deep learning. *Applied Intelligence: The International Journal of Artificial Intelligence, Neural Networks, and Complex Problem-Solving Technologies*, 1–12.
- Singh, S., & Sharma, A. (2019). Online handwritten Gurmukhi words recognition: An inclusive study. *ACM Transactions on Asian and Low-Resource Language Information Processing*, 18, 21:1–21:55. <http://dx.doi.org/10.1145/3282441>, URL <http://doi.acm.org/10.1145/3282441>.
- Singh, S., Sharma, A., & Chhabra, I. (2016). Online handwritten gurmukhi strokes dataset based on minimal set of words. *ACM Transactions on Asian and Low-Resource Language Information Processing*, 16, 1–20.
- Singh, S., Sharma, A., & Chhabra, I. (2017). A dominant points-based feature extraction approach to recognize online handwritten strokes. *International Journal on Document Analysis and Recognition*, 20, 37–58. <http://dx.doi.org/10.1007/s10032-016-0279-x>.
- Srimany, A., Chowdhuri, S. D., Bhattacharya, U., & Parui, S. K. (2014). Holistic recognition of online handwritten words based on an ensemble of svm classifiers. (pp. 86–90).
- Sternby, J., Morwing, J., Andersson, J., & Friberg, C. (2009). Online Arabic handwriting recognition with template matching. *Pattern Recognition*, 42, 3278–3286.
- Sundaram, S., & Ramakrishnan, A. G. (2011). Lexicon-free, novel segmentation of online handwritten indic words. In *Proceedings of 11th international conference on document analysis and recognition* (pp. 1175–1179).
- Sundaram, S., & Ramakrishnan, A. G. (2013). Attention-feedback based robust segmentation of online handwritten isolated tamil words. *ACM Transactions on Asian Language Information Processing*, 12.
- Sundaram, S., & Ramakrishnan, A. G. (2015). Bigram language models and reevaluation strategy for improved recognition of online handwritten tamil words. *ACM Transactions on Asian and Low-Resource Language Information Processing*, 14, 8:1–8:28. <http://dx.doi.org/10.1145/2671014>, URL <http://doi.acm.org/10.1145/2671014>.
- Surinta, O., Schomaker, L., & Wiering, M. (2013). A comparison of feature and pixel-based methods for recognizing handwritten bangla digits. In *Proceedings of international conference on document analysis and recognition* (pp. 165–169).
- Swethalakshmi, H., Jayaraman, A., Chakravarthy, V. S., & Sekhar, C. C. (2006). Online handwritten character recognition of devanagari and telugu characters using support vector machines. In *Proceedings of 10th international workshop on frontiers in handwriting recognition*.
- Swethalakshmi, H., Jayaraman, A., Chakravarthy, V. S., & Sekhar, C. C. (2007). *Online handwritten character recognition for devanagari and tamil scripts using support vector machines* (Master's thesis), Indian Institute of Technology.
- Szegedy, C., Liu, W., Jia, Y., Sermanet, P., Reed, S., Anguelov, D., Erhan, D., Vanhoucke, V., & Rabinovich, A. (2015). Going deeper with convolutions. In *The IEEE conference on computer vision and pattern recognition*.
- Szegedy, C., Vanhoucke, V., Ioffe, S., Shlens, J., & Wojna, Z. (2016). Rethinking the inception architecture for computer vision. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 2818–2826).
- Szegedy, C., Zaremba, W., Sutskever, I., Bruna, J., Erhan, D., Goodfellow, I., & Fergus, R. (2014). Intriguing properties of neural networks. In *International conference on learning representations*. URL <http://arxiv.org/abs/1312.6199>.
- Taigman, Y., Yang, M., Ranzato, M., & Wolf, L. (2014). Deepface: Closing the gap to human-level performance in face verification. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 1701–1708).
- Tompson, J. J., Jain, A., LeCun, Y., & Bregler, C. (2014). Joint training of a convolutional network and a graphical model for human pose estimation. In *Advances in neural information processing systems* (pp. 1799–1807).
- Urala, K. B., Ramakrishnan, A. G., & Mohamed, S. (2014). Recognition of open vocabulary, online handwritten pages in tamil script. (pp. 1–6). <http://dx.doi.org/10.1109/SPCOM.2014.6984002>.
- Verma, B., Lu, J., Ghosh, M., & Ghosh, R. (2004). A feature extraction technique for online handwriting recognition (vol. 2) (pp. 1337–1341). <http://dx.doi.org/10.1109/IJCNN.2004.1380140>.
- Wang, T., Wu, D. J., Coates, A., & Ng, A. Y. (2012). End-to-end text recognition with convolutional neural networks. In *Proceedings of the 21st international conference on pattern recognition* (pp. 3304–3308). IEEE.
- Yao, G., Lei, T., & Zhong, J. (2019). A review of convolutional-neural-network-based action recognition. *Pattern Recognition Letters*, 118, 14–22. <http://dx.doi.org/10.1016/j.patrec.2018.05.018>, URL <https://www.sciencedirect.com/science/article/pii/S0167865518302058>, cooperative and Social Robots: Understanding Human Activities and Intentions.
- Yun, C., Sra, S., & Jadbabaie, A. (2018). A critical view of global optimality in deep learning. In *International conference on machine learning representations*.
- Zanchettin, C., Bezerra, B. L. D., & Azevedo, W. W. (2012). A knn-svm hybrid model for cursive handwriting recognition. In *Proceedings of the international joint conference on neural networks* (pp. 1–8).
- Zhang, X., Yin, F., Zhang, Y., Liu, C., & Bengio, Y. (2018). Drawing and recognizing Chinese characters with recurrent neural network. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 40, 849–862. <http://dx.doi.org/10.1109/TPAMI.2017.2695539>.
- Zheng, Z., & Hong, P. (2018). Robust detection of adversarial attacks by modeling the intrinsic properties of deep neural networks. In *Proceedings of the 32nd international conference on neural information processing systems* (pp. 7924–7933). Red Hook, NY, USA: Curran Associates Inc..

Zheng, S., Song, Y., Leung, T., & Goodfellow, I. (2016). Improving the robustness of deep neural networks via stability training. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 4480–4488).

Zhuang, F., Qi, Z., Duan, K., Xi, D., Zhu, Y., Zhu, H., Xiong, H., & He, Q. (2020). A comprehensive survey on transfer learning. *Proceedings of the IEEE*, 109, 43–76.

Zouari, R., Boubaker, H., & Kherallah, M. (2019). Multi-language online handwriting recognition based on beta-elliptic model and hybrid tdnn-svm classifier. *Multimedia Tools and Applications*, 78, 12103–12123. <http://dx.doi.org/10.1007/s11042-018-6764-0>.



Dr. Sukhdeep Singh is working as an Assistant Professor in Computer Science in D.M. College (affiliated to Panjab University, Chnadigarh) Moga. He did his Postdoctoral Research with Boise State University, USA. Sukhdeep Singh has completed his PhD in Computer Science (Pattern Recognition) from the Department of Computer Science and Applications, Panjab University Chandigarh (India) from November, 2013 to January, 2018. He has received his MCA degree from the same Institution. His research interests include Pattern Recognition and Machine Learning in the area of Handwriting Recognition. He is also interested to work in other areas of Pattern Recognition, Machine Learning and Image Processing.



Dr. Anuj Sharma is working in the Department of Computer Science and Applications, Panjab University, Chandigarh. His research interests include Pattern Recognition and Machine Learning in the areas of Document Analysis, Handwriting and Images Recognition, Classification and Machine learning algorithms. He has done his Post-Doc from RWTH Aachen, Germany and received PhD in Computer Science from Thapar Institute of Engineering and Technology, India.



Dr. Vinod Kumar Chauhan is a Research Associate in Industrial Machine Learning in the Institute for Manufacturing, Department of Engineering at University of Cambridge UK. He has a PhD in Machine Learning from Panjab University Chandigarh India. His research interests are in Machine Learning, Optimization and Network Science. He specializes in solving large-scale optimization problems in Machine Learning, handwriting recognition, flight delay propagation in airlines, robustness and nestedness in complex networks and supply chain design using mathematical programming, genetic algorithms and reinforcement learning.