Detection and Localisation
of Struck-Out-Strokes
in Handwritten Manuscripts

Arnab Poddar[1]([✉]), Akash Chakraborty[2], Jayanta Mukhopadhyay[3],
and Prabir Kumar Biswas[2]

[1] Advanced Technology Development Centre, Indian Institute of Technology
Kharagpur, Kharagpur, India
`arnabpoddar@iitkgp.ac.in`

[2] Department of Electronics and Electrical Communication Engineering,
Indian Institute of Technology Kharagpur, Kharagpur, India
`pkb@ece.iitkgp.ac.in`

[3] Department of Computer Science and Engineering, Indian Institute of Technology
Kharagpur, Kharagpur, India
`jay@cse.iitkgp.ac.in`

Abstract. The presence of struck-out texts in handwritten manuscripts
adversely affects the performance of state-of-the-art automatic handwritten document processing systems. The information of struck-out words
(STW) are often important for real-time applications like handwritten
character recognition, writer identification, digital transcription, forensic applications, historical document analysis etc. Hence, the detection
of STW and localisation of struck-out strokes (SS) are crucial tasks. In
this paper, we introduce a system for simultaneous detection of STWs
and localisation of the SS using a single network architecture based on
Generative Adversarial Network (GAN). The system requires no prior
information about the type of SS stroke and it is also able to robustly
handle variant of strokes like straight, slanted, cris-cross, multiple-lines,
underlines and partial STW as well. However, we also present a methodology to generate STW with high variability of SS for network learning. We have evaluated the proposed pipeline on publicly available IAM
dataset and also on struck-out words collected from real-world writers
with high variability factors like age, gender, stroke-width, stroke-type
etc. The evaluation metrics show robustness and applicability in real-world scenario.

Keywords: Handwritten document image · Struck-out words ·
Handwritten manuscripts · Generative Adversarial Networks (GAN)

1 Introduction

Automatic handwritten text recognition is a prime topic of research in digital document image analysis [20,23,24]. A prime limitation in most published reports on handwriting recognition is that they consider the documents containing no writing error or struck-out texts. However, a free-form handwritten manuscript often contains struck-out words. Some typical examples of struck-out words (STW) are shown in Fig. 1. Such word produces nonsense output in optical character recognition (OCR) or writer verification framework [12]. To tackle the problem, we need an automatic module to identify STW and analyse further if required.

The meta-information of STWs are often important for real-time applications like writer identification, digital transcriptions, handwritten character recognition, etc. The recent digital transcriptions of famous writers like G. Washington, J. Austen, H. Balzac, G. Flaubert etc. use annotations of the STWs [1,3,8,10]. In forensic applications also, a quick and automatic detection of struck-out texts and examining their patterns may provide important clues like behavioral and psychological pattern of the suspect [13] and mentally challenged patients [14]. Automatic detection and localisation of STWs and strokes may be helpful in such cases as well.

(a) Straight   (b) Slanted   (c) Partial struck

(d) Multiple   (e) Crossed   (f) Underline
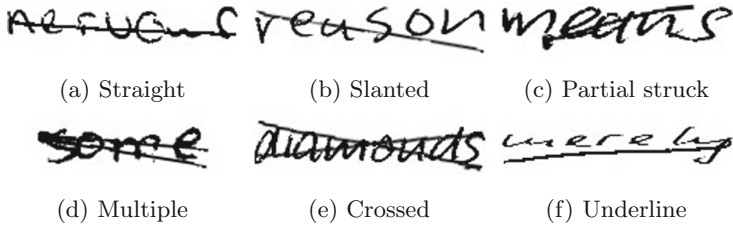
Fig. 1. Some examples of striked-through words with various kinds of strokes.

A few previous approaches dealt with the detection of struck-out texts. Tuganbaev and Deriaguine [26] registered a US patent for their crossed-out character recognizer using a feature-based classifier. The work in [19] presented a HMM (Hidden Markov Model) based crossed-out word recognition. A graph-based solution for detection of STW and localisation of SS from handwritten manuscripts was reported in [11]. More recently, a modified version of [11] was presented in [15], where morphological and graph based features were extracted from STW which was followed by an SVM classifier. In [12], a CNN-SVM based approach is proposed to detect STW in context of writer identification.

The approach presented in literature use separate modules for detection of SS and localisation of SS which allows error propagation from detection module to localisation modules [11,12,15]. The works are also dependent on prior information about the type of the SS and prescribe rule-based solutions for each

type of SS like straight, slanted, crossed, etc. which attracts manual intervention [11,12,15]. In this paper, we present an earliest attempt to tackle both the problems i.e., detection of STWs and localising the SS simultaneously without prior knowledge of the type of stroke. We use a single network architecture based on Generative Adversarial Network (GAN) [17,22] for localising the struck-out region. Further the system extracts concatenated features from the input image and the image localising the struck-out region. Finally, an SVM based classifier is used to classify between clean and struck-out word-images. The system uses no additional computation for localisation after the detection of SS. The system robustly handles different variant of strokes like straight, slanted, cris-cross, multiple-lines etc. and is also able to detect partially STWs. The system takes the handwritten word image I_{HW} and generates a mask image I_{SS} localising the potential SS. The I_{SS} is further used for detecting whether I_{HW} is a STW or a clean word. A simple block diagram of the proposed system is shown in Fig. 2. Since we require training samples of STWs to train deep learning architectures, we also present a technique to generate struck out words of various kind. We use the proposed pipeline on publicly available IAM dataset [21] and achieved encouraging results. The system is also tested on struck-out words from real world scenario which indicates the robustness and applicability of the system in challenging variability conditions.

The contribution of the paper are listed as follows

1. A single network architecture solution for simultaneous detection of struck-out words and localisation of strike-through stroke in handwritten manuscripts.
2. The proposed system requires no prior information about the type of strokes and is also able to handle partially struck-out handwritten words.
3. We introduce methods to generate STWs and corresponding ground truths from clean handwritten words and hand-drawn stroke templates.

The rest of the paper is organized as follows. The details of the proposed method are described in Sect. 2. The experimental set-up and datasets are described in Sect. 3. Then the experimental results are presented in Sect. 4. Finally, Sect. 5 summarizes the pros and cons of this approach and suggests the scope of future work.

2   Proposed Methodology

We propose the localisation of struck-out regions of the input word image and the detection of strike-through or clean word-image in two steps. Initially, the input image is passed through a localisation network, where the struck-out regions (if any) of the input word-image is localised. Further few simple features are calculated from input image and the corresponding struck-out region localised image. Further an SVM classifier is used to discriminate between clean words and words with struck-out strokes. Figure 2 shows an overall workflow of the proposed system.
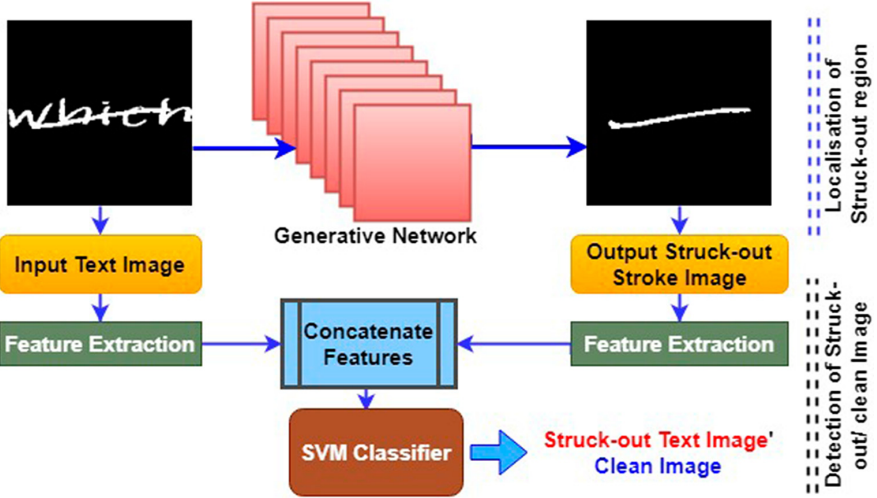
Fig. 2. Block diagram for the workflow of proposed system

We conceptualise the problem of localising the struck-out-stroke in handwritten word as an image to image translation problem, where we generate the SS image from a given image of a handwritten word. We directly feed the segmented word image to the generator network irrespective of the presence of SS (Fig. 2). The network is supposed to generate a mask image I_{SS} localising SS. Our generator model learns a mapping function from handwritten word images to their corresponding SS. The proposed network is expected to generate a uniform black image ideally for clean handwritten word input. The loss function used in the network simultaneously uses data loss, measured from mean square error and the structural loss (SSIM) of I_{SS} and expected ground-truth outcome I_{GT}. The detector of the adversarial model is implemented such that it differentiates between the real struck-out-strokes and its fake counterparts.

The generated I_{SS} is used for detection of STW. For this, the foreground pixels of the mask are considered which also appear as foreground in input image. We then consider the largest axis parallel bounding box of the connected components in the mask and the features of the contour is considered for detection of struck-out and clean words. The extracted features are finally fed into an SVM for the final decision of the clean and struck-out word (Fig. 2).

2.1   GAN Preliminary

In our proposed system, we use conditional GANs to learn a mapping from observed condition image x and random noise vector z, to $y, G : x, z \longrightarrow y$ [17,22]. The generator network G is trained to deliver outputs that are not differentiable from "real" images by its adversary discriminator, D.

2.2   Objective

In this work, we design the network to generate the SS image I_{SS} from a handwritten word image I_{HW}. First, we crop the word image from IAM dataset. Then, we resize binarised handwritten word and pad zeros to fit into size 128×128 as an input to the generator network. The discriminator network D acts as adversary to detect fake samples. Both the ground truth of real SS images (I_{GT}) and the generated ones (I_{SS}) are fed into discriminator D. During training, the discriminator D enforces the generator to produce realistic images SS. The objective function of the GAN network, conditioned on the input handwritten word images, can be expressed as:

$$\mathcal{L}_{cGAN}(G, D) = \mathbb{E}_{I_{HW} \sim p_{train}(I_{HW}), I_{GT} \sim p_{train}(I_{GT})}[\log D(I_{HW}, I_{GT})]$$
$$+ \mathbb{E}_{I_{HW} \sim p_{train}(I_{HW}), z}[\log(1 - D(I_{HW}, G(I_{HW}, z)))] \quad (1)$$

where G tries to minimize this objective against an adversarial D that tries to maximize it, i.e.

$$G* = \arg \min_{G} \max_{D} \mathcal{L}_{cGAN}(G, D) \quad (2)$$

We mix the GAN objective with a more traditional ℓ_1 loss, which encourages less blurring:

$$\mathcal{L}_{\ell 1}(G) = \mathbb{E}_{I_{HW} \sim p_{train}(I_{HW}), I_{GT} \sim p_{train}(I_{GT})}[||| I_{GT} - G(I_{HW}) ||_1] \quad (3)$$

We incorporate noise as dropout, applied on several layers of our generator in both training and evaluation phase. Here, we are interested to generate the struck-out strokes having structural similarity with ground-truth SS. Contrary to the $\ell 1$ loss, the structural similarity (SSIM) index provides a measure of the similarity by comparing two images based on luminance, contrast and structural similarity information. In our case, the handwritten word images are supposed to contain the SS inside the word. Hence structural similarity may play a vital role. Hence we also introduce the SSIM loss in the objective of the network.

For two images I^{HW} and I^{GT}, the luminescence similarity $l(I^{HW}, I^{GT})$, contrast similarity $c(I^{HW}, I^{GT})$ and structural similarity $s(I^{HW}, I^{GT})$ are calculated and the overall structural loss for the generator can be defined as

$$\mathcal{L}_{SSIM}(G) = 1 - \mathbb{E}_{I_{HW} p_{train}(I_{HW}), I_{GT} p_{train}(I_{GT}), z}[l(I^{HW}, I^{GT})$$
$$\times c(I^{HW}, I^{GT}) \times s(I^{HW}, I^{GT})] \quad (4)$$

Our final objective is

$$G* = \arg \min_{G} \max_{D} \mathcal{L}_{cGAN}(G, D) + \lambda \mathcal{L}_{\ell 1}(G) + \alpha \mathcal{L}_{SSIM}(G) \quad (5)$$

where λ and α are ratio control parameter for data-loss and structure-loss respectively.
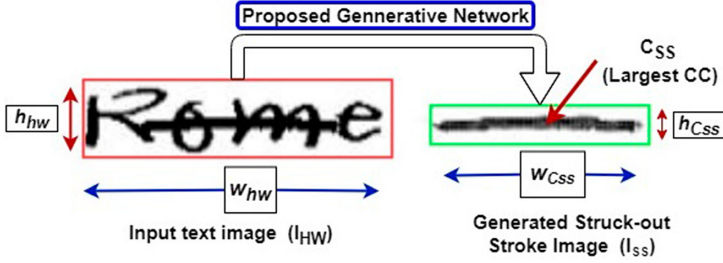
Fig. 3. An Example of localisation of struck-out stroke using proposed generative architecture. It shows various parameters for extraction of features from struck-out stroke image and input image for detection of struck-out and clean words.

2.3 Generator Architecture

The work adopts network architectures from Johnson et al. [18]. Here, 9 residual blocks are used in the generator network to train 128×128 size images. Let, `c7s1-k` denote a 7×7 Convolution-InstanceNorm-ReLU layer with `k` filters and stride 1. The parameter `dk` denotes a 3×3 Convolution-InstanceNorm-ReLU layer with `k` filters and stride 2. Reflection padding was used to reduce artifacts. Whereas `Rk` denotes a residual block that contains two 3×3 convolutional layers with the same number of filters on both layers. `uk` denotes a 3×3 fractional-strided-Convolution-InstanceNorm-ReLU layer with `k` filters and stride $\frac{1}{2}$. The network with 9 residual blocks consists of:

    c7s1-64, d128, d256, R256, R256, R256, R256, R256, R256, R256,
    R256, R256, u128, u64, c7s1-3

2.4 Discriminator Architecture

For discriminator networks, 70×70 PatchGAN is used [16]. Let `Ck` denote a 4×4 Convolution-InstanceNorm-LeakyReLU layer with k filters and stride 2. After the last layer, a convolution layer is applied to produce a 1-dimensional output. The proposed system does not use InstanceNorm for the first `c64` layer. It uses leaky ReLUs with a slope of 0.2. The discriminator architecture is: `C64, C128, C256, C512`.

2.5 Detection of Struck-Out Word

We use the generated mask images (I_{SS}) and input image I_{HW} to train an SVM for detection of STWs or clean words. We compute features from both I_{HW} and (I_{SS}) an d concatenate them to train an SVM classifier for detection of STws and clean words. We consider the foreground pixels which are also present in I_{HW}. The residual pixels in I_{SS} are discarded as noise and a new image I'_{SS} is generated. We compute simple feature-vector from I'_{SS} to train the SVM for detection of STW. Furthermore, the largest connected component

$C_{I'_{SS}}$ is selected from I'_{SS}. The pixel-width w_{ss} from leftmost foreground pixel to rightmost foreground pixel of I'_{SS} is calculated. The horizontal and vertical pixel-span $w_{c_{ss}}$ and $h_{c_{ss}}$ of contour $C_{I'_{SS}}$ is calculated to extract features. Furthermore, from image I_{HW} horizontal and vertical pixel-span of foreground pixels w_{hw} and h_{hw} are also calculated. An illustrative diagram is shown in Fig. 3 describing the procedure to extract features from I'_{SS} images for detection of struck-out words. We consider the following features from $C_{I'_{SS}}$, I_{SS} and I_{HW} to formulate SVM feature vector for detection:

1. The axis parallel bounding box and contour area of $C_{I'_{SS}}$
2. The ratio $w_{c_{ss}}/w_{ss}$ and $w_{c_{ss}}/w_{hw}$
3. The values of $h_{c_{ss}}$, h_{hw}, h_{ss}
4. set all values to 0 if $C_{I'_{SS}}$ is $NULL$.

Thus we train and test the SVM for detection of struck-out images from the features mentioned in above list.

    (a) straight   (b) partial Straight   (c) slanted

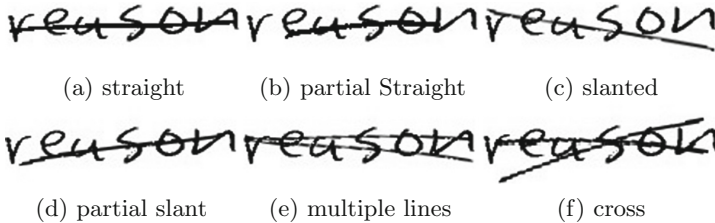   (d) partial slant   (e) multiple lines   (f) cross

Fig. 4. Various types of STWs generated from single clean word.

3   Database and Experimental Set-Up

The previous approaches mostly used classical image processing and machine learning techniques and used private databases with limited examples [11,12,15]. However, deep learning based methods require large number of training samples with high variability of writer, writing style, word-length, word-content, stroke-width etc. However there is no publicly available database to deal with STW in handwritten manuscripts. Here, we use original clean words from IAM database [21], and simultaneously collect separate stroke images various types like straight, slanted, cross, etc. A total of 2400 hand-drawn strokes are collected to generate the STWs for training and separate 1230 hand-drawn strokes are collected for testing. We have used 81412 handwritten word-images for training and 22489 word-images for testing the network for the task of SS localisation.

3.1   Generation of Struck-Out Words

The STWs are generated using the words from IAM dataset and collected hand-drawn strokes. Both the inputs, i.e., the word-images and stroke-images are taken from separate pool for training and testing. We describe the struck-out word generation procedure in Algorithm 1.

Algorithm 1: Generation of struck-out words

Result: Struck-out word

1. Compute height (h) and width (w) of input word;
2. Select a pool P of hand-drawn strokes having width within the range $w \pm r\%$;
3. Select randomly a stroke (s) from the pool P;
4. Give a rotation to s of degree $\pm\theta$;
5. Perturb the vertical center of s randomly within the range $\pm hg\%$ of the word-height h.;
6. Superimpose the stroke s on the word.;

In word generation, we introduce randomised rotation ($\pm\theta$), shift of vertical centre of stroke ($\pm hg\%$), and select random strokes from P in Algorithm 1 to ensure different variability like slant of strokes, vertical and horizontal position, type variability etc. We have varied these parameters $r\%, \pm\theta, \pm hg\%$ to generate various types of strokes like straight, slanted, cris-cross, underline etc. The strokes are used in a way to generate both full and partial STWs. The generation technique allows to generate various types of STWs using single clean word as shown in Fig. 4.

4   Experimental Results and Discussion

The models are trained in mixed setting with equal proportion of straight, slanted, partial straight, partial slanted, Multiple strike and crossed strokes with a total of 81412 STWs. The single trained model is used for testing with various types of strokes separately and also with mixed type accumulating all types of STWs. The control parameters λ, α used in Eq. 5 are given the values 50 for both and the learning rate is used 0.0002 for training.

4.1   Localisation of Struck Out Stroke

Localisation result of STWs using our proposed system is shown in Fig. 8, 9, 10, 11 and 12. Figure 13 shows the effectiveness of our proposed method also on underline strokes. We consider the generated I'_{SS} and use foreground pixels count in SS for performance evaluation. We compute precision (P), recall (R) and F-measure (FM) for each I_{SS} from input I_{HW}, and finally take the average (harmonic mean) of them. The performance metrics are measured in pixel-to-pixel setting. For a strike-through component true positive (TP), false positive (FP) and false negative (FN) are measured as follows:

Table 1. Performance comparison of objective functions: $\ell1, SSIM\,and\,\ell1+SSIM$ on STW with straight strokes only

Word length	Precision (%)			Recall (%)			F1-Score		
	$\ell1$	SSIM	$\ell1$ + SSIM	$\ell1$	SSIM	$\ell1$ + SSIM	$\ell1$	SSIM	$\ell1$ + SSIM
1	73.11	74.97	83.47	75.76	77.59	75.39	74.41	76.26	**79.22**
2	83.58	80.10	87.51	86.12	84.56	85.25	84.83	82.27	**86.37**
3	88.07	84.71	91.64	83.96	79.21	82.31	85.97	81.87	**86.73**
4	91.00	87.80	93.40	85.09	79.07	82.11	87.94	83.21	**87.39**
5	92.41	88.98	94.26	87.20	80.46	83.86	89.73	84.50	**88.76**
$\geq$6	90.81	86.88	93.68	89.94	84.24	88.51	90.37	85.54	**91.02**
All	89.91	86.38	92.86	87.19	81.78	85.17	88.53	84.02	**88.85**

Table 2. Results of localisation of struck-out region by the proposed system on IAM dataset. The network localising the struck-out region is trained with mixed type of struck-out words such as straight, partial straight, slanted, partial slanted, crossed, and multiple in equal proportion. Separate performance evaluation on different types of struck-out words are reported below.

Stroke type	Precision (%)	Recall (%)	F1 score
Straight	92.86	85.17	88.85
Partial straight	89.95	89.18	89.52
Slanted	96.54	84.60	90.17
Partial slanted	92.01	87.65	89.78
Crossed	95.51	82.72	88.65
Multiple	95.33	81.21	87.70
Mixed	93.97	83.92	88.66

– **TP**: number of black (object) pixels in SS (correctly classified),
– **FP**: number of black pixels those are incorrectly labeled as SS (unexpected),
– **FN**: number of black pixels of SS those are not labeled (missing result).

Table 1 presents the comparison of performance of localisation of SS with data loss $\mathcal{L}_{\ell1}(G)$, structural loss $\mathcal{L}_{SSIM}(G)$ and fusion of both. However, Table 1 also depicts the performance measures in varying character length (1, 2, 3, 4, 5, greater than 5), which show more robust and reliable performance to spot the stroke for larger character length. Fusing SSIM loss information with $\ell1$ loss improves performance of localisation of SS. The overall performance of the network is presented for various types of STWs in Table 2. We obtain consistent performance for various types of struck-out like straight, slanted, multiple strokes, crossed strokes, mixed etc. The system is also found to perform consistently well on partial STWs.

4.2   Detection of Strike-Out Textual Component

For detection of STWs we compute features from input image and I_{HW} generated I'_{SS} and concatenate them for classification. We compute the features

Table 3. Results of detection performance of struck-out and clean words on IAM dataset.

Stroke type	Precision (%)	Recall (%)	F1 score
Straight	98.60	99.37	98.98
Partial straight	96.40	98.03	97.21
Slanted	86.20	98.31	91.86
Partial slanted	95.56	97.85	96.74
Crossed	97.12	98.02	97.57
Multiple	95.81	97.24	96.52
Mixed	95.78	97.11	96.44

Table 4. Performance comparison of detection of struck-out words between proposed system and Lenet-SVM network [12] on IAM dataset

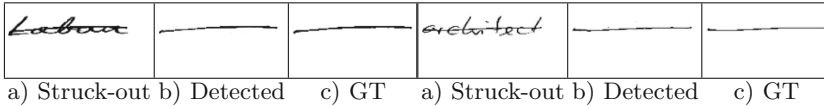| System type	Precision (%)	Recall (%)	F1 score
Lenet-SVM [12]	88.60	76.98	82.65
Proposed	95.78	97.11	**96.44**

for training and testing of SVM as presented in Sect. 2.5. The region localised images, i.e., I_{SS} are used to evaluate the detection performance. We subdivide the test set of localisation task, i.e., 22489 images into 70:30 ratio for training and testing the SVM for detection of struck-out and clean words. The performance metrics for detection performance for various strokes are presented in Table 3. We obtained very high values of accuracy for detection upto 98.93% for straight strokes and 97.31% for mixed strokes. We consistently obtained significant performance for other variant of strokes as depicted in Table 3 which indicates robustness of the system in different types of STWs (Figs. 5, 6 and 7).
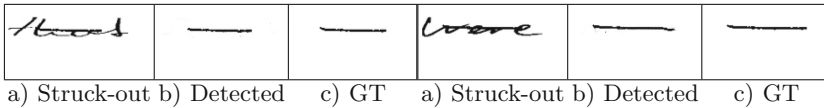

a) Struck-out b) Detected   c) GT   a) Struck-out b) Detected   c) GT
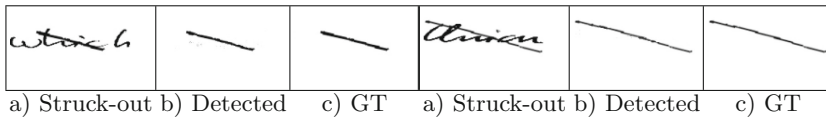
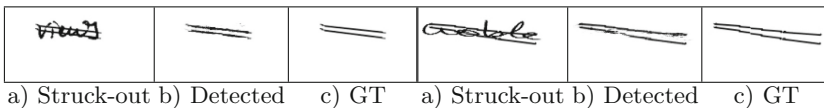**Fig. 5.** Generated output images with partially-slanted (above left) and fully slanted (above right) strokes.

a) Struck-out b) Detected   c) GT   a) Struck-out b) Detected   c) GT
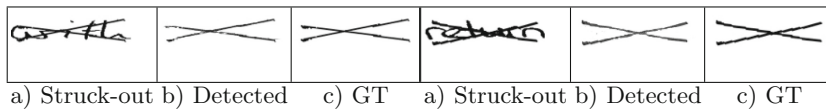
Fig. 6. Generated output images cross struck-out-strokes

a) Struck-out b) Detected   c) GT   a) Struck-out b) Detected   c) GT

Fig. 7. Generated output images straight struck-out strokes

a) Struck-out b) Detected   c) GT   a) Struck-out b) Detected   c) GT

Fig. 8. Generated output images partially-straight struck-out strokes

a) Struck-out b) Detected   c) GT   a) Struck-out b) Detected   c) GT

Fig. 9. Generated output images multiple struck-out strokes

4.3 Performance Comparison

In our proposed framework, we have encountered two tasks i.e., localisation of struck-out region of a word image and detection of struck-out and clean words. We have uniquely proposed a struck-out region localisation network and reported the performance metrics pixel-wise. However, the work in [12] and [15] reported struck-out word detection performance in a privately prepared data-set. The work in [12] uses a Lenet-SVM based deep-network architecture for detection performance. Here we compare the detection performance of the proposed system with that of [12] in publicly available IAM dataset [21]. The struck-out words are generated as described in Sect. 3.1. We have used word images from IAM dataset in 70:30 ratio for training and testing respectively. The struck-out words and clean words are both present in 50:50 ratio for evaluation. We have compared the performance of proposed system with that of the Lenet-SVM framework as in [12] with mixed type of struck-out words. The mixed type of struck-out words include struck-out stroke types like straight, partial straight, slanted, partial slanted, crossed, multiple as described in previous section. The Table 4 shows that the proposed system performs significantly better than the state-of-the-art system in terms of all precision, recall and F-1Score. As the performance is measured on widely used and publicly available IAM database [21], the system shows significant performance in presence of challenges like writer variability, age variability, cursiveness, etc.

4.4 Performance on Real Word Images

The results in Table 3 depict the performance of the proposed system on IAM database. Here we intent to show the performance of the system on the image

a) Struck-out b) Detected   c) GT   a) Struck-out b) Detected   c) GT
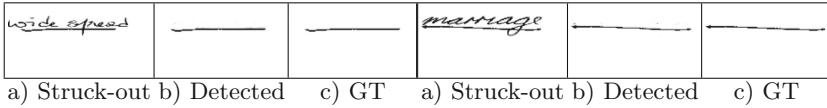
Fig. 10. Generated output images underline strokes

Table 5. Results for detection performance of struck-out and clean words on real unconstrained handwritten data-set.

Training data	Test data	Precision (%)	Recall (%)	F1 score
IAM	Real struck-out	87.70	99.39	93.18

collected from real world to evaluate the robustness of the proposed system. The IAM database inherently contains significant variability in terms of writing style, age of the writer, gender, texture of the page, ink of writing, stroke-width etc. The proposed struck-out region localisation network is trained on struck-out words from IAM data-set. However it would be informative to evaluate the performance of the proposed system on other data, collected from real-world writers. We have collected English handwriting from 45 individuals. The writers were of both the genders in the age group of 19–56 years with various regional and spoken-language background. Each writer provided 1 full page of handwriting sample. The content of handwriting is selected independently by the writers and written in running hand. The writers were requested to strike-through some of the words in their running handwriting style. The writers were given an A4 sized 75 GSM (g/m^2) white paper and instructed to use any pen of their choice with black or blue colored ink. Thus we collected 443 unconstrained various types of struck-out words. We also collected 1661 clean words to measure the detection performance. The Table 5 presents the performance metrics on the collected real data.
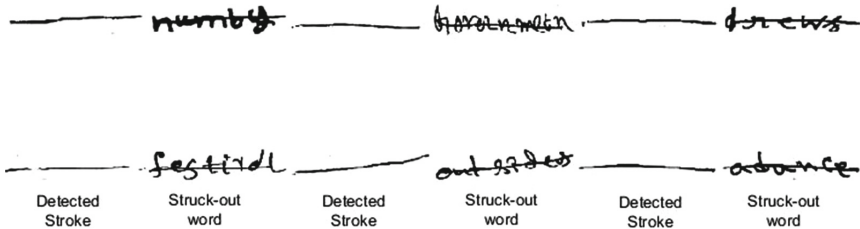
Fig. 11. Performance on real struck-out word images

The proposed struck-out region localisation network is trained on mixed simulated struck-out images as mentioned in previous section. The test results are obtained from collected real manuscripts of the writers. Figure 11 depicts few examples on real handwritten struck-out word images. The collected test dataset
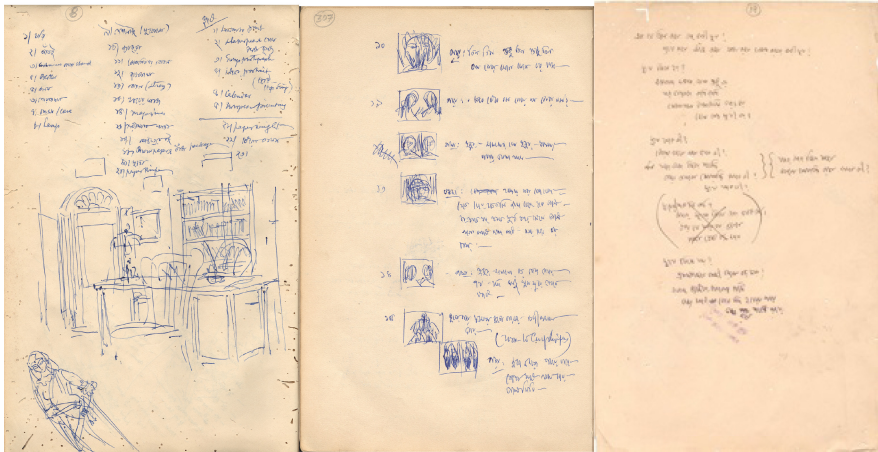
Fig. 12. Unconstrained offline document images from Satyajit Ray's movie-scripts of Goopi Gyne Bagha Byne and Apu Triology.

contains various types of struck-out words in running hand. We have obtained a high precision and f1-score on real world scenario. The proposed system seem to work robustly on the real world scenario.

4.5 Performance on Satyajit Ray's Manuscript

Here we present the result-images of struck-out strokes on Film-maker/writer Satyajit Ray's [2,7] manuscripts of his movie-scripts. We collected the scanned images of Satyajit Ray's movie-manuscripts with consent from National Digital Library of India [6,9]. The struck out words are collected from the script of the movies like 'Goopi Gayen Bagha Bayen' [4] and 'Apu Triology' [5]. Few pages of Satyajit Ray's manuscript from aforementioned movie-scripts are shown in the Fig. 12. The struck-out word-images are collected from annotated pages using MultiDIAS Annotation Tool [25].
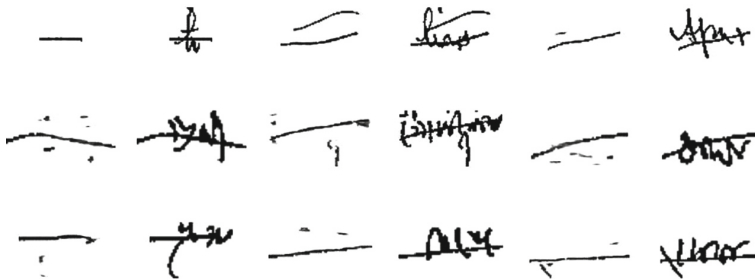

Fig. 13. Result images from Satyajit Ray's movie scripts of Goopi Gyne Bagha Byyne and Apu Triology on Both English (top row) and Bengali (middle and bottom row) language.

Satyajit Ray's manuscript images are written in nearly 1960–1980. The manuscript contains highly cursive words. The texture of the page and pen ink is quite different from that of images from IAM data-set. Our system is trained on IAM data-set with mixed types of struck-out words. Here we have shown the localisation of struck-out regions by the proposed localisation network. However, we have presented few result images on various environment to evaluate the robustness of the system. The resultant struck-out strokes are obtained from manuscript of Satyajit Ray as shown in Fig. 13. The performance is displayed on both cursive English and Bengali script. This shows that our system is useful in various scripts.

5   Conclusion

We present a single network architecture solution for simultaneous detection of STW and localisation of SS in handwritten manuscripts. It is the earliest attempt where the proposed system requires no prior information about the type of strokes and is also able to handle partially struck-out handwritten words. The experimentation is done on wide variability of SS types like straight, slanted, multiple strikes, underlines, crossed etc. We observed very high performance metrics for both detection of STWs and localisation of SS. The robustness and applicability of the proposed system is evaluated with real-time free form handwritten manuscripts. In future, the work can be extended with improvements in architectures and with other challenging and complex strike through texts. Further the localisation of struck-out regions can be used to reconstruct the clean word from struck-out word.

References

1. Bibliothèque de Rouen (Rouen Library), Rouen Cedex-76043, France. http://www.bovary.fr. Accessed 19 July 2008
2. Ray, S.: https://en.wikipedia.org/wiki/Satyajit_Ray. Accessed 22 Feb 2021
3. The Morgan Library & Museum, New York, USA-10016. http://www.themorgan.org. Accessed 19 July 2008
4. A brief about the movie Goopi Gyne Bagha Byne. https://en.wikipedia.org/wiki/Goopy_Gyne_Bagha_Byne. Accessed 22 Feb 2021
5. A brief about the movie series Apu Triology. https://en.wikipedia.org/wiki/The_Apu_Trilogy. Accessed 22 Feb 2021
6. A brief description about 'National Digital Library of India' on Wikipedia. https://en.wikipedia.org/wiki/National_Digital_Library_of_India. Accessed 22 Feb 2021
7. A brief description of 'Satyajit Ray' by Satyajit Ray Organisation. https://satyajitray.org/. Accessed 22 Feb 2021
8. George Washington Papers, The Library of Congress, USA. http://memory.loc.gov/ammem/gwhtml/gwhome.html. Accessed 19 July 2008
9. National Digital Library of India. https://www.ndl.gov.in/. Accessed 22 Feb 2021
10. Queensland State Archive, Australia-4113. http://www.archivessearch.qld.gov.au. Accessed 19 July 2008

11. Adak, C., Chaudhuri, B.B.: An approach of strike-through text identification from handwritten documents. In: 2014 14th International Conference on Frontiers in Handwriting Recognition, pp. 643–648. IEEE (2014)

12. Adak, C., Chaudhuri, B.B., Blumenstein, M.: Impact of struck-out text on writer identification. In: 2017 International Joint Conference on Neural Networks (IJCNN), pp. 1465–1471. IEEE (2017)

13. Brink, A., Schomaker, L., Bulacu, M.: Towards explainable writer verification and identification using vantage writers. In: Ninth International Conference on Document Analysis and Recognition (ICDAR 2007), vol. 2, pp. 824–828. IEEE (2007)

14. Caligiuri, M.P., Mohammed, L.A.: The Neuroscience of Handwriting: Applications for Forensic Document Examination. CRC Press, Boca Raton (2012)

15. Chaudhuri, B.B., Adak, C.: An approach for detecting and cleaning of struck-out handwritten text. Pattern Recogn. **61**, 282–294 (2017)

16. Gatys, L.A., Ecker, A.S., Bethge, M.: Image style transfer using convolutional neural networks. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 2414–2423 (2016)

17. Isola, P., Zhu, J.Y., Zhou, T., Efros, A.A.: Image-to-image translation with conditional adversarial networks. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 1125–1134 (2017)

18. Johnson, J., Alahi, A., Fei-Fei, L.: Perceptual losses for real-time style transfer and super-resolution. In: Leibe, B., Matas, J., Sebe, N., Welling, M. (eds.) ECCV 2016. LNCS, vol. 9906, pp. 694–711. Springer, Cham (2016). https://doi.org/10.1007/978-3-319-46475-6_43

19. Likforman-Sulem, L., Vinciarelli, A.: Hmm-based offline recognition of handwritten words crossed out with different kinds of strokes (2008)

20. Liu, C.L., Yin, F., Wang, D.H., Wang, Q.F.: Online and offline handwritten Chinese character recognition: benchmarking on new databases. Pattern Recogn. **46**(1), 155–162 (2013)

21. Marti, U.V., Bunke, H.: The IAM-database: an English sentence database for offline handwriting recognition. Int. J. Doc. Anal. Recogn. **5**(1), 39–46 (2002)

22. Mirza, M., Osindero, S.: Conditional generative adversarial nets. arXiv preprint arXiv:1411.1784 (2014)

23. Pal, U., Jayadevan, R., Sharma, N.: Handwriting recognition in Indian regional scripts: a survey of offline techniques. ACM Trans. Asian Lang. Inf. Process. (TALIP) **11**(1), 1 (2012)

24. Plamondon, R., Srihari, S.N.: Online and off-line handwriting recognition: a comprehensive survey. IEEE Trans. Pattern Anal. Mach. Intell. **22**(1), 63–84 (2000)

25. Poddar, A., Mukherjee, R., Mukhopadhyay, J., Biswas, P.K.: MultiDIAS: a hierarchical multi-layered document image annotation system. In: Sundaram, S., Harit, G. (eds.) DAR 2018. CCIS, vol. 1020, pp. 3–14. Springer, Singapore (2019). https://doi.org/10.1007/978-981-13-9361-7_1

26. Tuganbaev, D., Deriaguine, D.: Method of stricken-out character recognition in handwritten text, 25 June 2013. uS Patent 8,472,719