Abstract—We demonstrate the utility of an end-to-end trainable CNN-RNN hybrid architecture in recognizing Arabic text in videos and natural scenes. Our solution is based upon Convolutional Recurrent Neural Network (CRNN) model, which is proven quite effective in English scene text recognition. The model follows a segmentation-free, sequence to sequence transcription approach. The network transcribes a sequence of convolitional features from the input image to a sequence of target labels. This does away with the need for segmenting input image into constituent characters/glyphs, which is often difficult in case of Arabic. Further, the ability of RNNs to model contextual dependencies yields superior recognition results. The network is trained on synthetic images rendered from a large vocabulary of Arabic words and phrases. Our solution is evaluated on two publicly available video text datasets - ALIF and ACTIV. For the scene text recognition task a new dataset - IIIT Arabic scene text dataset is created. We report state of the art results on both of the video text datasets. We also benchmark our results on the IIIT Arabic scene text dataset.

Keywords—Arabic, Scene Text, Video Text, Synthetic Data, Deep Learning, Text Recognition.

I. INTRODUCTION

For many years, the focus of research on text recognition in Arabic has been on printed and handwritten documents [7], [10], [11], [12]. Majority of the works in this space were on individual character recognition [11], [12]. Since segmenting a Arabic word or line image into its sub units is challenging, such models did not scale well. In recent years there has been a shift towards segmentation-free text recognition models, mostly based on Hidden Markov Models (HMM) or Recurrent Neural Networks(RNN). Such models generally follow a sequence-to-sequence approach wherein the input line/word image is directly transcribed to a sequence of labels. [8], [9] use RNN based models for recognizing printed/handwritten Arabic script. Our approach to recognizing Arabic text on videos (video text recognition) and natural scenes (scene text recognition) follows the same paradigm. Video text recognition in Arabic has gained interest recently [21], [22], [34] and there are two benchmarking datasets available for the task [21], [22]. To the best of our knowledge, there has been no work so far on scene text recognition (recognizing text in natural scene images) in Arabic.

The vision community experienced a strong revival of neural networks based solutions in the form of Deep Learning. This process was stimulated by the great success of models like Deep Convolutional Neural Networks (DCNNs) in various feature extraction, object detection and classification tasks as seen in [1], [3]. These tasks however pertain to a set of problems where the subjects appear in isolation, rather than appearing as a sequence. Recognising such sequence-based objects often requires the model to predict a series of description labels instead of a single label. DCNNs are not well suited for such tasks as they generally work well in scenarios where the inputs and outputs are bounded by fixed dimensions. Also, the lengths of these sequence-like objects might vary drastically and that escalates the problem difficulty. Recurrent neural networks (RNNs) tackle the problems faced by DCNNs for sequence-based learning by performing a forward pass for each segment of the sequence-like-input. Such models often involve a pre-processing/feature extraction step, where the input is first converted into a sequence of feature vectors [4], [5]. These feature extraction stage would be independent of RNN-pipeline and hence they are not end-to-end trainable.

The sudden boom in video sharing on social networking websites and the increasing number of TV channels in today's world reveals videos to be a fundamental source of information. Effectively managing, storing and retrieving such huge amounts of data is not a trivial task. Video text recognition can greatly aid video content analysis and understanding, with the recognized text giving a direct and concise description of the stories being depicted in the videos. In case of news videos, the superimposed tickers running on the edges of the video frame are generally highly correlated to the people involved or the story being portrayed and hence provide a brief summary of the news event. These text transcriptions can greatly benefit the indexing mechanism of an automated video retrieval system by providing semantic clues about the events being described in the videos.

Reading text in natural scenes is relatively harder task
Automatic recognition of Arabic is a pretty challenging task due to various intricate properties of the script. There are only 28 letters in the script and it is written in a semi-cursive fashion from right to left. The letters of the script are generally connected by a line at its bottom to the letters preceding and following it, except for 6 letters which can be linked only to their preceding letters. Such an instance in a word creates a paw (part-of Arabic word) in the word. Arabic script has another exceptional characteristic where the shape of a letter changes depending on whether the letter appears in the word in isolation, at the beginning, middle or at the end. So in general, each letter can have one to four possible shapes which might have little or no similarity in shape whatsoever. Another frequent occurrence in Arabic is that of dots. The addition of a single dot to a letter can change the meaning of the word completely. Arabic also follows ligature, where two letters when combined form a third different letter in a way that they cannot be separated by a simple baseline (i.e., a complex shape represents this combined letter). These distinctive characteristics make automated Arabic script recognition more challenging than most other scripts.

B. Related Work

Though there has been a lot of work done in the field of text transcription in natural scenes and videos for the English script [4], [5], [13], [27], it is still in a nascent state as far as Arabic script is considered. Previous attempts made to address similar problems in English [27], [28] first detect individual characters and then character-specific DCNN models are used to recognize these detected characters. The shortcomings of such methods are that they require training a strong character detector for accurately detecting and cropping each character out from the original word. Also, for Arabic, this task becomes even more difficult due to the connectedness intricacies of the script, as discussed earlier. Another approach by Jaderberg et al. [13], was to treat scene text recognition as an image classification problem instead. To each image, they assign a class label from a lexicon of words spanning the English language (90k most frequent words were chosen to put a bound on the span-set). This approach however is limited to the size of lexicon used for its possible unique transcriptions and the large number of output classes add to training complexity. Hence the model is not scalable to inflectional languages like Arabic where number of unique words is much higher compared to English.

Another category of solutions typically embed image and its text label in a common subspace and retrieval/recognition is performed on the learnt common representations. For example, Almazan et al. [29] embed word images and text strings in a common vector-subspace, and thus convert the task of word recognition into a retrieval problem. Yao et al. [31] and Gordo et al. [32] used mid-level features for scene text recognition.

The recent works in this field [33], [5] generally follow a transcription approach to transcribe the input image into a sequence of labels. When the earlier works following the transcription model used handcrafted features, [33] put forward a novel approach where in the recurrent layers are fed by convolutional features extracted from the input image. Our approach is borrowed from this particular architecture called CRNN where the hybrid CNN-RNN network with a Connectionist Temporal Classification (CTC) loss is trained end-to-end.

Works on text extraction from videos has generally been in four broad categories, edge detection methods [35], [36], [37], extraction using connected components [38], [39], texture classification methods [40] and correlation based methods [41], [42]. The previous attempts at solving video text for Arabic used a two separate routines; one for extracting relevant image features and another for classifying features to script labels for obtaining the target transcription. [34] uses text-segmentation and statistical feature extraction followed by fuzzy k-nearest neighbour techniques to obtain transcriptions. Similarly, [20] experiment with two feature extraction routines; CNN based feature extraction and Deep Belief Network (DBN) based feature extraction, followed by a Bi-directional Long Short Term Memory (LSTM) layer [15], [16].

The rest of the paper is organized as follows; section II describes in detail the CRNN architecture, section III describes the experiments we perform to test the CRNN with the implementation level details and verified results, section IV concludes with the findings of our work.

II. CRNN Architecture

The deep neural network architecture, CRNN, consists of three components. The initial convolutional layers, the middle recurrent layers and a final transcription layer.

The CRNN architecture is essentially a hybrid CNN-RNN model. The role of the convolutional layers is to extract robust feature representations from the input image and feed it into the recurrent layers which in turn will transcribe them to an output sequence - a sequence of labels or characters. The sequence to sequence transcription is achieved by a CTC layer at the output.

The convolutional layers follow a VGG [6] style architecture without the fully-connected layers. All the images are scaled to a fixed height before being fed to the convolutional layers. The convolutional components then create a sequence of feature vectors from the feature maps by splitting them column-wise, which then act as inputs to the recurrent layers.
The symbols ‘k’, ‘s’ and ‘p’ stand for kernel size, stride and padding size respectively.

On top of the convolutional layers, the recurrent layers take each frame from the feature sequence generated by the convolutional layers and make predictions. The recurrent layers consist of deep bidirectional LSTM nets. RNNs are an optimal choice for our unconstrained end-to-end system as they have a strong capability of capturing contextual information within a sequence. RNNs are capable of handling variable length sequences. Number of parameters in a RNN is independent of the length of the sequence. All we need to do is to unroll the network as many times as the number of time-steps in the input sequence. This in our case helps in unconstrained recognition. Your output could be any sequence of labels derived from your label set. Traditional RNN units (vanilla RNNs) face the problem of vanishing gradients [14] and hence we use (LSTM) units instead, a type of RNN unit specifically designed to tackle the vanishing-gradients problem [15], [16]. The special design of the LSTM cell allows it to learn long-term dependencies and hence it is a good fit for our image-based sequence learning task where such dependencies are frequent. In a text recognition problem, contexts from both directions (left-to-right and right-to-left) are useful and complementary to each other in outputting the right transcription. Therefore, we combine a forward and a backward oriented LSTM to create a bi-directional LSTM unit. Multiple such bi-directional LSTM layers can be stacked to make the network deeper and gain higher levels of abstractions over the image-sequences as shown in [17].

The transcription layer at the top of the CRNN is used to translate the predictions generated by the recurrent layers into label sequences for the target language. The (CTC) layer’s conditional probability is used in the objective function as shown in [18]. The objective is to minimize the negative log-likelihood of conditional probability of ground truth. This objective function calculates a cost value directly from an image and its ground truth label sequence, eliminating the need of manually label all the individual components in sequence. In other words the input sequence can be transcribed directly to the output sequence without the need for a target defined at each time-step. The CRNN, though composed of multiple modules of convolutional and recurrent layers, can hence be trained using a single loss function and is end-to-end trainable. The complete network configuration used for the experiments can be seen in (Fig. 2).

### III. Experiments

In this section we demonstrate the efficacy of above architecture in recognizing Arabic script appearing in video frames and natural scene images. In case of scene text, our experiments concern only with scene text recognition. It is assumed that input images are cropped word images from natural scenes, not full scene images. Since there were no previous works in Arabic scene text recognition we introduce a new dataset - IIT Arabic scene text dataset and benchmark our results on this new dataset. For video text recognition problem the results are reported on two existing video text datasets - ALIF [20] and ACTIV [22].

**Fig. 3:** Sample images from the rendered synthetic Arabic scene text dataset. The images closely resemble real world scene images.

**A. Datasets**

Models for both video text and scene text recognition problems are trained using synthetic images rendered from a large vocabulary using freely available Arabic Unicode fonts (Fig. 3). Details on the rendering process are detailed here [19]. Around 2 million video text line images and 4 million scene text word images were used for training the respective models.
The model for video text recognition task was initially trained on the Synthetic video text dataset and then fine-tuned on the train partitions of real-world datasets, ALIF and ACTIV.

ALIF dataset consists of 6,532 cropped text line images from 5 popular Arabic News channels. ACTIV dataset is larger than ALIF and contains 21,520 line images from popular Arabic News channels. The dataset contains video frames where in bounding boxes of text lines are annotated.

Fig. 4: Sample images from the IIIT Arabic scenetext dataset.

IIIT Arabic scene text dataset was curated by downloading freely available images containing Arabic script from Google Images. The dataset consists of 500 word images of Arabic script occurring in various scenarios like local markets & shops, billboards, navigation signs, graffiti, etc. and spans a large variety of naturally occurring image-noises and distortions (Fig. 4). The images were manually annotated by human experts of the script and the transcriptions as well as the image data is being made publicly available for future research groups to compare and improve performance in solving this task.

B. Implementation Details

The CRNN model’s convolutional layers follow the VGG architecture [6]. In the 3rd and 4th max-pooling layers the pooling windows used are rectangular instead of the usual square windows used in VGG. The advantage of doing this is that the feature maps obtained after the convolutional layers are wider and hence we obtain longer feature sequences as inputs for the recurrent layers that follow. To enable faster batch learning all input images are resized to a fixed width and height (32x100 for scene text and 32x504 for video text). We have observed that resizing all images to a fixed width is not affecting the performance much. The images are horizontally flipped before feeding them to the convolutional layers since Arabic is read from right to left.

To tackle the problems of training such deep convolutional and recurrent layers, we used the batch normalization [24] technique. Two batch-norm layers were inserted after the 5th and 6th convolutional layers respectively, which accelerated the training process greatly.

The network is trained with stochastic gradient descent (SGD). Gradients are calculated by the back-propagation algorithm. Precisely, the transcription layers’ error differentials are back-propagated with the forward-backward algorithm, as shown in [18]. While in the recurrent layers, the Back-Propagation Through Time (BPTT) [25] algorithm is applied to calculate the error differentials. The hassle of manually setting the learning-rate parameter is taken care of by using ADADELTA optimization [26].

C. Results

The accuracies obtained by CRNN and other previous works [20], [22] in the field of video text recognition are stated in Table I. Since there has been no work done in Arabic scene text recognition, we compare the results obtained on IIIT Arabic scene text dataset using with a popular English free OCR - Tesseract [43]. The performance has been evaluated using the following metrics; CRR - Character Recognition Rate, WRR - Word Recognition Rate, LRR - Line Recognition Rate. In the below equations, RT and GT stand for recognized text and ground truth respectively.

\[
CRR = \frac{n\text{Characters} - \sum \text{EditDistance}(RT, GT)}{n\text{Characters}}
\]

\[
WRR = \frac{n\text{WordsCorrectlyRecognized}}{n\text{Words}}
\]

\[
LRR = \frac{n\text{ImagesCorrectlyRecognized}}{n\text{Images}}
\]

It should be noted that even though the methods compared on for video text recognition use a separate convolutional architecture for feature extraction, unlike our end-to-end trainable CRNN architecture, we obtain better character and line-level accuracies for the Arabic video text recognition task and set the new state-of-the-art for the same.

\[
Tesseract = 94.16 \quad 55.03 \quad 90.71 \quad 44.90
\]

\[
\text{ConvNet-BLSTM} = 90.73 \quad 39.39 \quad 87.64 \quad 31.54
\]

\[
\text{DBN-BLSTM} = 83.26 \quad 26.91 \quad 81.51 \quad 27.03
\]

\[
\text{ABBYY} = 98.17 \quad 79.87 \quad 97.84 \quad 77.38
\]

\[
\text{CRNN} = 97.44 \quad 67.08
\]

Fig. 5: Qualitative results of the SceneText Recognition. On the left are images which were correctly recognized and on the right are examples which the CRNN failed to recognize.

TABLE II: Accuracy for Scene Text

<table>
<thead>
<tr>
<th>SceneText</th>
<th>CRR(%)</th>
<th>LRR(%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Tesseract [43]</td>
<td>97.44</td>
<td>67.08</td>
</tr>
<tr>
<td>CRNN</td>
<td>98.17</td>
<td>79.87</td>
</tr>
</tbody>
</table>

TABLE I: Accuracy for Video text

<table>
<thead>
<tr>
<th>VideoText</th>
<th>ALIF Test</th>
<th>ALIF Test2</th>
<th>ActIV</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>CRR(%)</td>
<td>LRR(%)</td>
<td>CRR(%)</td>
</tr>
<tr>
<td>ConvNet-BLSTM [20]</td>
<td>94.16</td>
<td>55.03</td>
<td>90.71</td>
</tr>
<tr>
<td>DBN-BLSTM [20]</td>
<td>90.73</td>
<td>39.39</td>
<td>87.64</td>
</tr>
<tr>
<td>ABBYY [23]</td>
<td>83.26</td>
<td>26.91</td>
<td>81.51</td>
</tr>
<tr>
<td>CRNN</td>
<td>98.17</td>
<td>79.87</td>
<td>97.84</td>
</tr>
<tr>
<td>IIIT-Arabic</td>
<td>97.44</td>
<td>67.08</td>
<td></td>
</tr>
</tbody>
</table>

Lower accuracies on scene text recognition problem testify the inherent difficulty associated with the problem compared to printed or video text recognition.

IV. CONCLUSION

We demonstrate that state of the deep learning techniques can be successfully adapted to rather challenging tasks like Arabic text recognition. The newer, script and language agnostic approaches are well suited for low resource languages like
Arabic where the traditional methods often involved language specific modules. The success of RNNs in sequence learning problems has been instrumental in the recent advances in speech recognition and image to text transcription problems. And this came as a boon for languages like Arabic where the segmentation of words into sub word units was often troublesome. The sequence learning approach could directly transcribe the images and also model the context in both forward and backward directions. And in the CRNN architecture this seq2seq framework is fed by a deep convolutional feature extractor which derives robust feature sequences from the input image.

On video text recognition we report the best results so far, on both the datasets. Compared to the previous best, where a CNN and RNN were trained separately and used as a feature-extractor and classifier respectively, the architecture we follow can be trained end-to-end.

Scene text recognition is a much harder problem compared to OCR or video text recognition. The variability in terms of lighting, distortions and typography make the learning pretty hard. With better feature representations and learning algorithms available, we believe the focus should now shift to harder problems like scene text recognition. We hope the introduction of IIIT Arabic scene text dataset and the initial results would instill an interest among the Arabic computer vision community.

ACKNOWLEDGMENT

The authors would like to thank Maaz Anwar for helping annotate the IIIT Arabic dataset and Dr. Girish Varma for his timely help and discussions while finalising the network architecture.

REFERENCES