Unconstrained Scene Text and Video Text Recognition for Arabic Script

Mohit Jain, Minesh Mathew and C. V. Jawahar
Center for Visual Information Technology, IIIT Hyderabad, India.
mohit.jain@research.iiit.ac.in, minesh.mathew@research.iiit.ac.in, jawahar@iiit.ac.in

***Abstract*—Building robust recognizers for Arabic has always been challenging. We demonstrate the effectiveness of an end-to-end trainable CNN-RNN hybrid architecture in recognizing Arabic text in videos and natural scenes. We outperform previous state-of-the-art on two publicly available video text datasets - ALIF and ACTIV. For the scene text recognition task, we introduce a new Arabic scene text dataset and establish baseline results. For scripts like Arabic, a major challenge in developing robust recognizers is the lack of large quantity of annotated data. We overcome this by synthesizing millions of Arabic text images from a large vocabulary of Arabic words and phrases. Our implementation is built on top of the model introduced here [37] which is proven quite effective for English scene text recognition. The model follows a segmentation-free, sequence to sequence transcription approach. The network transcribes a sequence of convolutional 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 for Arabic script. Further, the ability of RNNs to model contextual dependencies yields superior recognition results.**

Keywords*—*Arabic, Arabic Scene Text, Arabic 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], [11], [12], [13]. Majority of the works in this space were on individual character recognition [12], [13]. Since segmenting an 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. Methods such as [9] and [10] use RNN based models for recognizing printed/handwritten Arabic script. Our attempt to recognize 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 [24], [25], [38]. There are now two benchmarking datasets available for the task ALIF [24] and ACTIV [25]. In the scene text domain, [8] uses sparse coding of SIFT features for a bag-of-features model using spatial pyramid matching for the recognition task at the character level in Arabic. To the best of our knowledge, there has been no work so far on scene text recognition at word level in Arabic.

The computer vision community experienced a strong revival of neural networks based solutions in the form of Deep Learning in recent years. This process was stimulated by the success of models like Deep Convolutional Neural Networks (DCNNs) in 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 a sequence of 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.

Fig. 1: Examples of Arabic Scene Text (left) and Video Text (right) recognized by our model. Our work deals only with the recognition of cropped words/lines. The bounding boxes were provided manually.

The upsurge 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 video 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 news videos, 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.

Reading text in natural scenes is relatively harder task compared to printed text recognition. The problem has been drawing increasing research interest in recent years. This can partially be attributed to the rapid development of wearable and mobile devices such as smart phones, google glass and self-driving cars, where scene text is a key module to a wide range of practical and useful applications. While the recognition of text in printed documents has been studied extensively in the form of Optical Character Recognition (OCR) Systems, these methods generally do not generalize very well to a natural scene setting where factors like inconsistent lighting conditions, variable fonts, orientations, background noise and image distortions add to the problem complexity. Even though Arabic is the fourth most spoken language in the world after Chinese, English and Spanish, to the best of our knowledge, there has not been any work in the field of word-level scene text recognition, which makes this area of research ripe for exploration.

A. Intricacies of Arabic Script

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. 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 character 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. Scene Text and Video Text Recognition

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], [14], [30], it is still in a nascent state as far as Arabic script is considered. Previous attempts made to address similar problems in English [30], [31] first detect individual characters and then character-specific DCNN models are used to recognize these detected characters. The short-comings of such methods are that they require training a strong character detector for accurately detecting and cropping each character out from the original word. In case of Arabic, this task becomes even more difficult due to the intricacies of the script, as discussed earlier. Another approach by Jaderberg et al. [14], 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). However, this approach 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. [32] 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. [35] and Gordo et al. [36] used mid-level features for scene text recognition.

The segmentation-free, transcription approach was proved quite effective for Indian Scripts OCR [21], [22] where segmentation is often problematic. Similar approach was used in [5] for scene text recognition. Handcrafted features derived from image gradients were used with a RNN to map the sequence of features to a sequence of labels. Unlike the problem of OCR, scene text recognition required more robust features to yield results comparable to the transcription based solutions for OCR. A novel approach combining the robust convolutonal features and transcription abilities of RNN was introduced by [37]. We employ this particular architecture 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 [39], [40], [41], extraction using connected components [42], [43], texture classification methods [44] and correlation based methods [45], [46]. The previous attempts at solving video text for Arabic used two separate routines; one for extracting relevant image features and another for classifying features to script labels for obtaining the target transcription. In [38] text-segmentation and statistical feature extraction are followed by fuzzy k-nearest neighbour techniques to obtain transcriptions. Similarly, [23] 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 [16], [17].

The rest of the paper is organized as follows; Section II describes the hybrid CNN-RNN architecture we use. Section III focuses on the Arabic text recognition from a transcription perspective. Section IV presents the results on benchmarking datasets, followed by a discussion based on the qualitative results. Section V concludes with the findings of our work.

II. Hybrid CNN-RNN Architecture

The network consists of three components. The initial convolutional layers, the middle recurrent layers and a final transcription layer. This architecture we employ was introduced in [37].

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 of labels or characters. The sequence to sequence transcription is achieved by a CTC layer at the output.

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

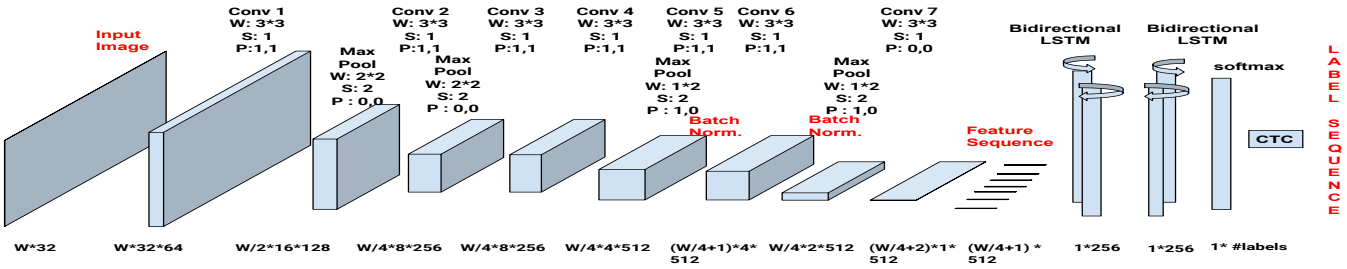

Fig. 2: The deep neural network architecture employed in our work, which was devised by [37]. The symbols 'k', 's' and 'p' stand for kernel size, stride and padding size respectively.

by splitting them column-wise, which then act as inputs to the recurrent layers (Fig. 2). This feature descriptor is highly robust and most importantly can be trained to be adopted to a wide variety of problems [1], [3], [14].

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. Number of parameters in a RNN is independent of the length of the sequence, hence we can simply unroll the network as many times as the number of time-steps in the input sequence. This in our case helps perform unconstrained recognition as the predicted output can be any sequence of labels derived from the entire label set. Traditional RNN units *(vanilla* RNNs*)* face the problem of vanishing gradients [15] and hence (LSTM) units are used which were specifically designed to tackle the vanishing-gradients problem [16], [17]. 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, a combined forward and backward oriented LSTM is used 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 [18].

The transcription layer at the top of the network 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 [19]. The complete network configuration used for the experiments can be seen in (Fig. 2).

III. Arabic Text Transcription

The efficacy of a seq2seq approach lies in the fact that the input sequence can be transcribed directly to the output sequence without the need for a target defined at each time-step. This was proven quite effective in recognizing complex scripts where sub-word segmentation is often difficult [22]. In addition, contextual modelling of the sequence in both the directions, forward and backward, is achieved by the bidirectional LSTM block. Contextual information is critical in making accurate predictions for a Script like Arabic where there are many similar looking characters/glyphs. The hybrid architecture replaces the need for any handcrafted features to be extracted from the image. Instead, more robust convolutional features are fed to the RNN layers.

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). For scene text, the images were rendered from a vocabulary of a quarter million most commonly used words in Arabic Wikipedia. A random word from the vocabulary is first rendered into the foreground layer of the image by varying the font, stroke color, stroke thickness, kerning, skew and rotation. Later a random perspective projective transformation is applied to the image, which is then blended with a random crop from a natural scene image. Finally the foreground layer is alpha composed with a background layer, which is again a random crop from a random natural image. The synthetic line images for video text recognition are rendered from random text lines crawled from Arabic news websites. The rendering process is much simpler since real video text usually have uniform color for the text and background and lacks any perspective distortion. Details on the rendering process are described here [20]. 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 wherein bounding boxes of text lines are annotated.

A new Arabic scene text dataset was curated by downloading freely available images containing Arabic script from Google Images. The dataset consists of 2000 word images of Arabic script occurring in various scenarios like local

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

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 upon.

B. Implementation Details

Convolutional block in the network follows 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 re-sized 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* [27] technique. 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 [19]. While in the recurrent layers, the Back-Propagation Through Time (BPTT) [28] 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 [29].

IV. Results and Discussion

In this section we demonstrate the efficacy of above architecture in recognizing Arabic script appearing in video frames and natural scene images. 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 at word level, the baseline results are reported on the new Arabic scene text we introduce. For video text recognition problem the results are reported on two existing video text datasets - ALIF [23] and ACTIV [25].

Results on video text recognition are presented in Table I. Since there has been no work done in word-level Arabic scene text recognition, we compare the results obtained on the Arabic scene text dataset using a popular free English OCR - Tesseract [47]. 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.

$$CRR = \frac{(nCharacters - \sum EditDistance(RT, GT))}{nCharacters}$$

$$WRR = \frac{nWordsCorrectlyRecognized}{nWords}$$

$$LRR = \frac{nTextImagesCorrectlyRecognized}{nImages}$$

It should be noted that even though the methods compared on for video text recognition use a separate convolutional architecture for feature extraction, unlike the end-to-end trainable 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.

TABLE I: Accuracy for Video Text Recognition.

VideoText	**ALIF Test1**		**ALIF Test2**		**AcTiV**	
	CRR(%)	*LRR(%)*	*CRR(%)*	*LRR(%)*	*CRR(%)*	*LRR(%)*
ConvNet-BLSTM [23]	94.36	55.03	90.71	44.90	–	–
DBN-BLSTM [23]	90.73	39.39	87.64	31.54	–	–
ABBYY [26]	83.26	26.91	81.51	27.03	–	–
CNN-RNN hybrid network	**98.17**	**79.67**	**97.84**	**77.98**	**97.44**	**67.08**

The hybrid CNN-RNN architecture we employ outperforms previous video text recognition benchmarks.

TABLE II: Accuracy for Scene Text Recognition.

SceneText	**Arabic SceneText Dataset**	
	CRR(%)	*WRR(%)*
Tesseract [47]	17.07	5.92
Baseline using the hybrid CNN-RNN	**75.05**	**39.43**

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

On video text recognition, we report the best results so far on both the datasets. 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 and we provide baseline results for the same. A qualitative analysis of the recognition tasks can be seen in Fig. 5.

V. Conclusion

We demonstrate that the state-of-the-art deep learning techniques can be successfully adapted to some 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. 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. 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 a new Arabic scene text dataset and the initial results would instill an interest among the Arabic computer vision community to pursue this field of research further.

Fig. 5: Qualitative results of Scene Text Recognition. For each image, the annotations on bottom-left and bottom-right are the label and model prediction, respectively.

ACKNOWLEDGMENT

The authors would like to thank Maaz Anwar, Anjali, Saumya, Vignesh and Rohan for helping annotate the Arabic scenetext dataset and Dr. Girish Varma for his timely help and discussions.

REFERENCES

[1] A. Krizhevsky, I. Sutskever and G.E. Hinton, *Imagenet classification with deep convolutional neural networks*. NIPS 2012.

[2] Y. LeCun, L. Bottou, Y. Bengio and P.Haffiner, *Gradient-based learning applied to document recognition*. Proceedings of the IEEE, 1998.

[3] R.B. Girshick, J. Donahue, T. Darrel and J. Malik, *Rich feature hierarchies for accurate object detection and semantic segmentation*. CVPR 2014.

[4] A. Graves, M. Liwicki, S. Fernandez, R. Bertolami, H. Bunke and J. Schidhuber, *A novel connectionist system for unconstrained handwriting recognition*. TPAMI, 2009.

[5] B. Su and S. Lu, *Accuracte scene text recognition based on recurrent neural network*. ACCV 2014.

[6] K. Simonyan and A. Zisserman. *Very deep convolutional networks for large-scale image recognition*. CoRR 2014.

[7] Prasad, Rohit, et al. *Improvements in hidden Markov model based Arabic OCR*, ICPR 2008.

[8] M. Tounsi, I. Moalla, A.M. Alimi, F. Lebouregois, *Arabic characters recognition in natural scenes using sparse coding for feature representations*, ICDAR 2015.

[9] Yousefi, Mohammad Reza, et al. *A comparison of 1D and 2D LSTM architectures for the recognition of handwritten Arabic*, SPIE 2015.

[10] Ul-Hasan, Adnan, et al, *Offline printed Urdu Nastaleeq script recognition with bidirectional LSTM networks*, ICDAR 2013.

[11] El-Hajj, Ramy, Laurence Likforman-Sulem, and Chafic Mokbel, *Arabic handwriting recognition using baseline dependant features and hidden Markov modeling*. ICDAR 2005.

[12] Amin, Adnan, Humoud Al-Sadoun, and Stephen Fischer, *Hand-printed Arabic character recognition system using an artificial network*. Pattern recognition 29.4 (1996): 663-675.

[13] Amin, Adnan. *Off-line Arabic character recognition: the state of the art*. Pattern recognition 31.5 (1998): 517-530.

[14] M. Jaderberg, K. Simoyan, A. Vedaldi and A. Zisserman, *Reading text in the wild with convolutional neural networks*. IJCV 2015.

[15] Y. Bengio, P.Y. Simard and P. Frasconi, *Learning long-term dependencies with gradient descent is difficult*. NN 1994.

[16] F.A. Gers, N.N. Schraudolph and J. Schmidhuber, *Learning precise timing with LSTM recurrent networks*. JMLR 2002.

[17] S. Hochreiter and J. Schmidhuber, *Long short-term memory*. Neural Computation 1997.

[18] A. Graves, A. Mohamed and G.E. Hinton, *Speech recognition with deep recurrent neural networks*. ICASSP 2013.

[19] A. Graves, S. Fernandez, F.J. Gomez and J. Schmidhuber, *Connectionist temporal classification: labelling unsegmented sequence data with recurrent neural networks*. ICML 2006.

[20] Praveen Krishnan and C.V. Jawahar, *Generating Synthetic Data for Text Recognition*. arXiv 2016.

[21] Naveen Sankaran and C.V. Jawahar, *Recognition of printed Devanagari text using BLSTM Neural Network*. ICPR 2012.

[22] Minesh Mathew, Ajeet Kumar Singh and C.V. Jawahar, *Multilingual OCR for Indic Scripts*. DAS 2016.

[23] S. Yousfi, B. Sid-Ahmed and G. Christophe, *ALIF: A dataset for Arabic embedded text recognition in TV broadcast*. ICDAR 2015.

[24] Yousfi, Sonia, Sid-Ahmed Berrani, and Christophe Garcia, *Arabic text detection in videos using neural and boosting-based approaches: Application to video indexing*. ICIP 2014.

[25] O. Zayene, J. Hennebert, S.M. Touj, R. Ingold, N.E. Amara, *A dataset for Arabic text detection, tracking and recognition in news videos-AcTiV*. ICDAR 2015.

[26] K. Wang, B. Babenko and S. Belongie, *End-to-end scene text recognition*. In ICCV, 2011.

[27] S. Ioffe and C. Szegedy, *Batch normalization: Accelerating deep network training by reducing internal covariate shift*. ICML 2015.

[28] P.J. Werbos, *Backpropagation through time: what it does and how to do it*. Proceedings of the IEEE 1990.

[29] M.D. Zeiler, *ADADELTA: an adaptive learning rate method*. CoRR 2012.

[30] T. Wang, D.J. Wu, A. Coates and A.Y. Ng, *End-to-end text recognition with convolutional neural networks*. CVPR 2014.

[31] A. Bissacco, M. Cummins, Y. Netzer and H. Neven, *Photoocr: Reading text in uncontrolled conditions*. ICCV 2013.

[32] Anand Mishra, Karteek Alahari and C.V. Jawahar, *Word spotting and recognition with embedded attributes*. TPAMI 2014.

[33] J. Almazan, A. Gordo, A. Fornes and E. Valveny, *Scene Text Recognition using Higher Order Langauge Priors*. BMVC 2012.

[34] J.A. Rodriguez-Serrano, A. Gordo and F. Perronnin, *Label embedding: A frugal baseline for text recognition*. IJCV 2015.

[35] C. Yao, X. Bai, B. Shi and W. Liu, *Strokelets : A learned multi-scale representation for scene text recognition*. ICPR 2012.

[36] A. Gordo, *Supervised mid-level features for word image representation*. CVPR 2015.

[37] B. Shi, X. Bai and C. Yao, *An end-to-end trainable neural network for image-based sequence recognition and its application to scene text recognition*. arXiv 2015.

[38] Halima, M. Ben, Hichem Karray, and Adel M. Alimi, *Arabic text recognition in video sequences*. arXiv 2013.

[39] L. Agnihotri and N. Dimitrova, *Text detection for video analysis*. CVPR workshop, 1999.

[40] L. Agnihotri., N.Dimitrova, M.Soletic, *Multi-layered Videotext Extraction Method*. ICME 2002.

[41] S. Hua,X., X.-R. Chen., *Automatic Location of Text in Video Frames*. MIR 2001.

[42] A. K. Jain and B. Yu, *Automatic text location in images and video frames*. PR 1998.

[43] R. Lienhart and F. Stuber, *Automatic text recognition indigital videos*. Proceedings of SPIE Image and Video 1996.

[44] H. Karray, M. Ellouze and M.A. Alimi, *Indexing Video Summaries for Quick Video Browsing*. Computer Communications and Networks 2009.

[45] H. Karray , A.M. Alimi, *Detection and Extraction of the Text in a video sequence*. ICES 2005.

[46] M. Kherallah, H. Karray, M. Ellouze, A.M. Alimi, *Toward an Interactive Device for Quick News Story Browsing*. ICPR 2008,

[47] "Tesseract OCR", *https://github.com/tesseract-ocr/tesseract*.