Localizing and Recognizing Text in Lecture Videos

Kartik Dutta, Minesh Mathew, Praveen Krishnan and C.V. Jawahar
CVIT, IIIT Hyderabad, India
{kartik.dutta, minesh.mathew, praveen.krishnan}@research.iiit.ac.in and jawahar@iiit.ac.in

***Abstract*—Lecture videos are rich with textual information and to be able to understand the text is quite useful for larger video understanding/analysis applications. Though text recognition from images have been an active research area in computer vision, text in lecture videos has mostly been overlooked. In this work, we investigate the efficacy of state-of-the art handwritten and scene text recognition methods on text in lecture videos. To this end, a new dataset - LectureVideoDB[1] compiled from frames from multiple lecture videos is introduced. Our experiments show that the existing methods do not fare well on the new dataset. The results necessitate the need to improvise the existing methods for robust performance on lecture videos.**

***Index Terms*—Word recognition, Lecture video, MOOCs, Video text, Word spotting**

I. Introduction

With increasing interest in e-learning in the form of OpenCourseWare (OCW) lectures and Massive Open Online Courses (MOOCs), freely available lecture videos are abundant. Understanding lecture videos is critical for educational research, particularly in the context of MOOCs which has become synonymous with distance learning. For example a lecture video can be analyzed to understand a teacher's engagement with the learners, on which frames does the viewers pay more attention [1] etc. The figures, images and text in lecture videos are vital cues for understanding any lecture video. Text is present almost everywhere in a lecture video; particularly in lectures on Science, Mathematics and Engineering. Text alone could be used for a variety of of tasks like keyword generation, video indexing and enabling search and extracting class notes [2]–[5].

Text in lecture videos comprise of handwritten text written on a blackboard or a paper, text written using a stylus on a tablet and displayed on a screen or font rendered text appearing in presentation slides (digital text). Lectures are recorded using one or more cameras, and the camera(s) are typically positioned to directly face the blackboard or the presentation slides. Usually text recognition from presentation slides is less challenging as the text is more legible, there is little variation in style and there is more contrast. At the same time text on blackboard is handwritten and not very legible due to poor lighting, smaller size or poor contrast. On blackboard or on paper the lecturer may write over figures and equations, and this makes the scene cluttered, making it harder to detect the text. Figure 2 shows few samples from the new, LectureVideoDB dataset, illustrating the different types of text present in lecture videos and the complexities involved in detection and recognition.

[1]https://cvit.iiit.ac.in/research/projects/cvit-projects/lecturevideodb

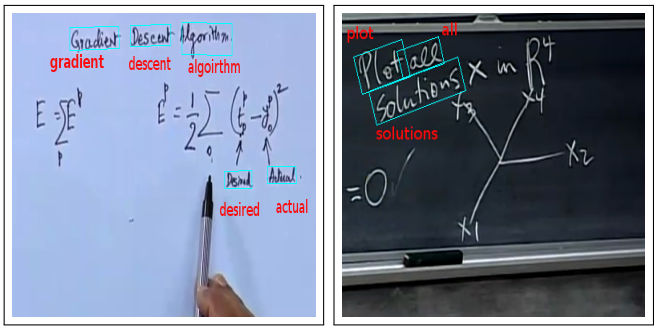

Fig. 1. Visualization of text localization and recognition results on frames from the LectureVideoDB dataset.The left frame is taken from a video where the instructor uses paper to explain the concept while in the right picture, the instructor uses a blackboard. Please note that equations and symbols are not annotated in our dataset.

Understanding the text appearing in images/videos has been an active research area in computer vision, under three general categories - printed, handwritten and scene text. Traditionally text recognition had been centered around recognizing printed text in documents or Optical Character Recognition (OCR). An overview of the development of the popular OCR system - Tesseract [6] is a good primer to the research in this space. Handwriting Recognition (HWR) focuses on handwritten text in documents. The major challenges in HWR stems from the inherent complexity in recognizing human handwriting where, practically each person has a different style of writing. Scene text recognition unlike OCR or HWR, deals with recognizing text in natural scene images. Scene text is harder to recognize owing to variations in background, lighting, texture, and orientation. Understanding text in videos, specifically broadcast videos has also been an area of interest within the document community. Text overlaid over the broadcast videos is useful to understand the context of the part of the video and to enable searching [7]. The detection in this case is relatively easier since the overlaid text in broadcast videos appear at fixed positions of the frame, is mostly horizontally oriented, little variation and good contrast.

In the area of document analysis, word spotting has evolved as a complementary approach to address the problem of searching and indexing textual content in images. Word spotting has typically been used in cases where the recognition based approaches are yet to mature. Word spotting [8] basically refers to locating a keyword (e.g word image) given in

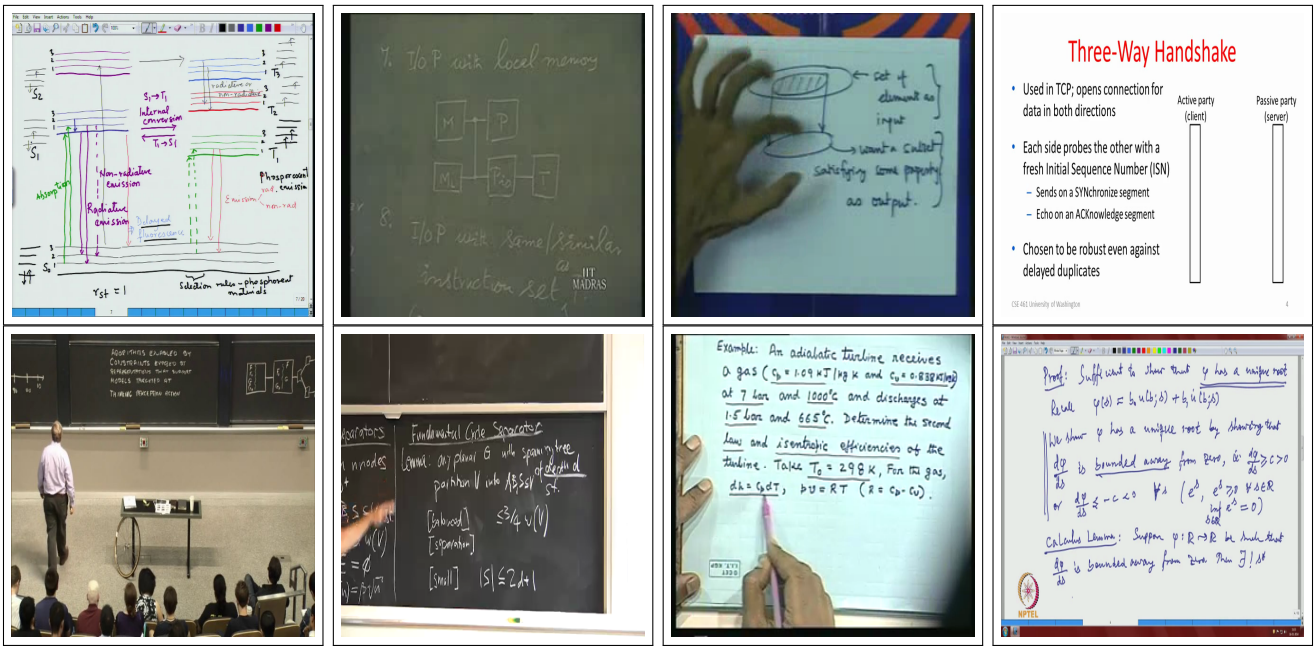

Fig. 2. Sample frames from the LectureVideoDB datatset. Scenes cluttered with text and figures, green boards where text background contrast is poor, low resolution images and less legible handwritings make the text recognition harder in lecture videos.

the form of a query from an underlying corpus of word images. Here the query could be either an exemplar word image or the corresponding text itself.

What drew our interest to text recognition in lecture videos is the unique case where the detection and recognition methods need to deal with text in different modalities. Despite multitude of works on lecture videos and MOOCs, there are very few which specifically look into this problem. And we need to investigate how would the existing methods which are developed for printed, handwritten or scene text would fare on text in lecture videos. This motivated us to explore it further in the following directions:-

- Considering the fact that text in lecture videos is such a vital cue in understanding the videos, we introduce a new dataset - LectureVideoDB where textual words from lecture videos are annotated. The dataset comprises of over 5000 frames annotated with word bounding boxes and the corresponding text ground truths.
- We demonstrate that existing state-of-the-art detection, recognition and word spotting methods in scene text and HWR do not yield similar performance on the new dataset. This necessitates the need for further research in this domain.

II. Related Work

Scene Text Recognition: Earlier approaches [9], [10] to scene text understanding generally worked at the character or individual glyph level (bottom-up approaches). Characters from the detected text regions are segmented out and are fed to a classifier which classifies the character patches into the characters in the language. More recent models follows segmentation-free approaches where the words could be recognized without the need for sub-word segmentation. Such models generally use a seq2seq framework built on Recurrent Neural Networks (RNN). Segmentation-free approaches to word transcription along with use of deep features derived from Convolutioanl Neural Networks (CNN) helped to achieve state-of-the art results for scene text recognition [11], [12].

Handwritten Recognition: Similar to scene text recognition, newer methods in HWR also uses seq2seq formulation [13], [14] which typically uses an underlying CNN-RNN hybrid network for feature extraction and prediction. There are also methods which uses multi-dimensional RNNs (MDRNNs) [15] instead of regular, uni-dimensional RNNs. Given the limited performance of unconstrained prediction, most of the methods in this space, use either a lexicon [14] or language model [15] for arriving at the final output string.

Word Spotting: In the domain of word spotting, the key challenge lies in finding an efficient holistic representation for word images. Most of the recent works use deep neural networks for learning the features. In [16], the author uses the features from the penultimate layer of a deep CNN network, while [17]–[20] learns the features by embedding a word image into different attributes spaces such as PHOC, semantic attributes (ngrams, word2vec) etc. These embedding gives a unified representation of both text and its corresponding images and are invariant to different styles and degradations.

Lecture Videos: Text in lecture videos has largely been unexplored, except for few isolated works. One of the early works in this space detect and recognize text in presentation slides to synchronize the slides with the lecture videos [3]. The

Fig. 3. Some sample cropped word images from the LectureVideoDB dataset. The first row images are taken from slides, 2nd from whiteboard, 3rd from paper and 4th from blackboard.

TABLE I
DETAILS OF THE LECTUREVIDEODB DATASET. HERE TYPE REFERS TO THE PRESENTATION/WRITING MEDIUM USED BY THE INSTRUCTOR.

Type	#Frames	#Words	#Writers
Slides	1145	52225	5
Whiteboard	945	21160	7
Paper	1281	27900	9
Blackboard	2103	36460	14

text detection is based on edge detection and geometry based algorithms and a commercial OCR is used for recognition. Video indexing and keyword search is made possible by text recongition in [4]. Off-the-shelf OCR systems are used for the same. In another work both Automatic Speech Recognition (ASR) and OCR are used to generate keywords for lecture videos [21].

III. LECTURE VIDEO (LECTUREVIDEODB) DATASET

Text recognition in lecture videos have largely been unattended, and there are no publicly available benchmarking datasets for the same. This motivated us to compile a new dataset, for text detection and recognition in lecture videos. The dataset is created from course videos of 24 different courses across science, management and engineering. The camera angle and distance to the blackboard varies in these videos, but the text being presented is always in focus in the videos. These courses are offered by e-learning initiatives such as MIT OCW [22], Khan Academy [23] and NPTEL [24]. Out of the 24 courses whose lectures videos were used in the making of this dataset, the video quality varies widely, with 6 courses having videos in resolution of 1280×720 pixels, 7 courses having a resolution of less than 640×360 pixels and the rest in between. The four styles/modalities of text present in the dataset are the following:

1) Slides: This set includes the frames where a presentation slide is shown. The text in this case is mostly born digital text, which is relatively more legible and free from distortions.
2) Whiteboard: This category generally encompasses frames where either the instructor is using a physical white board along with markers to explain a concept or is using a digital pad to write on a personal computer.

TABLE II
PARTITION DETAILS OF THE LECTUREVIDEODB DATASET

Partition	#Frames	#Words	#Writers
Train	3170	82263	17
Val	549	15379	5
Test	1755	40103	13
Total	5474	137745	35

TABLE III
TYPE OF IMAGES IN THE 3 PARTITIONS OF THE LECTUREVIDEODB DATASET. HERE WE ARE REFERRING TO THE NUMBER OF SEGMENTED WORD IMAGES.

Partition	Slides	Whiteboard	Paper	Blackboard
Train	27371	15214	15867	23811
Val	7757	0	1452	6170
Test	17097	5946	10581	6479
Total	52225	21160	27900	36460

3) Paper: The instructor explains the lecture content on a paper using a pen.
4) Blackboard: Frames where the instructor writes on a blackboard using a chalk.

Table I describes the breakdown of the LectureVideoDB across the four modalities mentioned above. While extracting the frames from the course videos, we save only those frames where there is a considerable change to the scene visually. This helps us to avoid frames where the content is repeating. Also we retain few frames with little text content in the dataset. These frames act as distractors for the detection and recognition modules.

After saving the frames, we used the pre-trained model of TextSpotter [25], to predict word bounding boxes for all the saved frames. We do not annotate equations and symbols. The results of TextSpotter are used as the seed boxes for the next round of manual annotation where human annotators annotate each word by marking the bounding boxes and entering the corresponding ground truth word. LabelImg [26] is used for the annotation process. Fig. 3 shows a few cropped word images that are part of the LectureVideoDB dataset in each modality. As one can notice, the word images posses different styles and also contains blurriness artifacts. Table II, III give the partition details of the frames and the number of extracted words into the train, val and test set. The partitioning was done in such a way that all the frames and extracted word images from all the videos in a course belong to only one of the 3 sets. This also makes the set of writers disjoint between all the 3 partitions.

IV. METHOD

As mentioned before, the methods used in this work are adopted from state-of-the art scene text detection, handwriting recognition and word spotting methods. This section presents a brief summary of these methods which we use for experiments in Section V.

A. Word Localization

For detecting textual words in the lecture video frames, we use two state-of-the-art scene text detection methods - EAST [27] and Textboxes++ [28]. Both the methods, unlike traditional multi-staged detection models, employ a deep fully convolutional neural network (FCN) to directly output the four coordinates of the localized text regions. The only post-processing involved in both cases is a Non Maximal Suppression (NMS) applied on the boxes outputted by the neural network. Both the methods are capable of detecting arbitrarily oriented text, and is suitable for lecture videos where text, particularly the ones on blackboards usually have arbitrary orientations.

B. Word Recognition

Once a word is localized in the image, the job of the word recognizer is to transcribe the word to its corresponding text. To this end we use CRNN [11] and CRNN-STN [13] architectures. The original CRNN architecture in [11] comprise of convolutional layers followed by a bi-directional RNN and a CTC transcription layer [29]. This hybrid CNN-RNN architecture combines the feature learning abilities of a CNN and the sequence learning abilities of an RNN into a single end to end trainable network. At the end of the convolutional layers a sequence of features are passed on to the bidirectional RNN. The RNN layers models the sequential structure of the input sequence. On top of the RNN block is a fully connected classification layer with SoftMax activation. At each timestep (an instance in the input sequence), the classification layer outputs the probability distribution over the output classes. At the test time various decoding methods such as naive decoding, lexicon based decoding or beam search can be used to arrive at an output string of characters, from the classification scores across the timesteps.

The two major differences between the vanilla CRNN and the CRNN-STN lies in the convolutional block. CRNN-STN accommodates more number of layers by adding residual convolutional layers. The deeper CNN architecture helps to learn better features from the word images. Another difference is that CRNN-STN uses a Spatial Transformer Network (STN) [30]. The STN block learns to correct geometric distortions in the word images and this improves the recognition results. The STN block is a part of the larger CRNN-STN and the entire network is trained end-to-end using the CTC transcription loss like the original CRNN.

C. Word Spotting

Given that one of the important use cases in lecture videos is spotting keywords for retrieving relevant videos, in this work, we adopt [31] which is one of the recent end-to-end word spotting method. The proposed architecture [31] contain two parallel streams of network for feature extraction, one for feeding the real handwritten word image while the other stream is the concatenation of label information using a synthetic image and its corresponding text represented using PHOC [32] features. The features from individual streams are given to an embedding layer which projects both of them into a common subspace where the image and its corresponding text lie close to each other. The architecture uses a multi-task loss function where the cross entropy based classification loss is applied to the features coming from individual streams and a cosine embedding loss is applied after the embedding layer. Given a trained network, the holistic representation for word images are computed as the L_2 normalized features activation at the penultimate layer of the network, which is found suitable for word spotting.

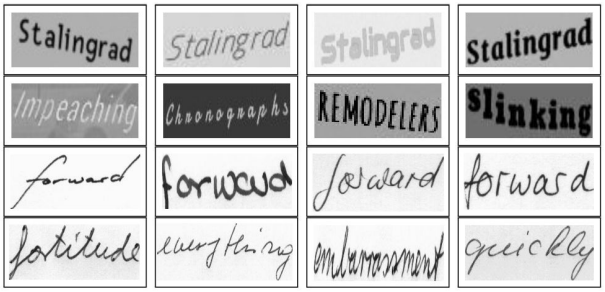

Fig. 4. Examples of synthetic images in MJSynth [33] scene text dataset (top 2 rows) and images in the IAM [34] dataset(bottom 2 rows).

V. Experiments

A. Datasets

In addition to the LectureVideoDB, we use the following two public datasets in our work. Both the datasets are used for pre-training the word recognition models.

- **IAM Handwriting Database [34]:** It includes contributions from over 600 writers and comprises of 115,320 words in English.
- **MJSynth [33]:** This is a synthetically generated dataset for scene text recognition. It contains 8 million training images and their corresponding ground truth words. Fig. 4 shows a few sample images from the IAM and MJSynth datasets.

B. Text Localization

For detection we use the same evaluation method which is typically been followed for scene text detection [9]. A bounding box is counted as a match if it overlaps a ground truth bounding box (intersection over union) by more than 50%. Table IV shows the performance of state of the art text detectors Textboxes++ [28] and EAST [27] on the four different modalities of the LectureVideoDB and the complete test set. The models used here are off-the-shelf models provided by the authors. Quite evidently both the methods perform well only on the text on presentation slides. In most of the failure cases, the detector split a single word into multiple bounding boxes (Fig. 5) and hence increasing the number of false positives. This occurs more in case of handwritten text where breaks in the cursive writing are confused with spaces between words.

TABLE IV
WORD LOCALIZATION PERFORMANCE OF VARIOUS ARCHITECTURES ON THE DIFFERENT SPLITS OF LECTUREVIDEODB TEST SET. THE PERFORMANCE IS EVALUATED IN THE SAME FASHION AS [9]

Architecture	Data-Split	Recall	Precision	F-score
Textboxes++ [28]	Slides	0.69	0.96	0.80
	Whiteboard	0.37	0.47	0.41
	Paper	0.51	0.67	0.58
	Blackboard	0.62	0.69	0.66
	Full-TestSet	0.58	0.76	0.66
EAST [27]	Slides	0.83	0.90	0.86
	Whiteboard	0.42	0.42	0.47
	Paper	0.56	0.69	0.62
	Blackboard	0.61	0.73	0.66
	Full-TestSet	0.68	0.73	0.70

C. Word Recognition

Table V shows the recognition results of various variants of the CNN-RNN hybrid architecture on the test set of LectureVideoDB dataset. The evaluation metrics used in this case are Word Error Rate (WER) and Character Error Rate (CER), which are commonly used for word recognition in HWR and scene text recognition. The various models and their training strategies are mentioned below:

- CRNN uses the architecture mentioned in [11]. It is trained only on the MJSynth dataset.
- CRNN-Finetune uses the architecture mentioned in [11]. It is first pre-trained on the MJSynth dataset and then fine-tuned on the train set of LectureVideoDB dataset.
- CRNN-STN$_{synth}$ uses the architecture mentioned in [13]. It is trained only on the MJSynth dataset.
- CRNN-STN$_{IAM}$ uses the same architecture as above. It is first pre-trained on the train set of the IAM dataset and then pre-trained on the MJSynth dataset.
- CRNN-STN-Finetune$_{IAM}$ uses the same architecture as above. It is first pre-trained on both the IAM and MJSynth datasets and then fine-tuned on the train set of LectureVideoDB.

From Table V we can see that, even the models fine tuned on LectureVideoDB do not yield results comparable to the performance of these methods on scene text datasets [11] or IAM. Though the lecture videos comprise of plentiful of handwritten text, CRNN variants trained purely on IAM training data, performs poorly on LectureVideoDB (we do not report these numbers since the error rates are very high).

In order to better understand the reason for this poor performance, we separately report results for the four different kinds of modalities present in the LectureVideoDB in Table VI. As expected, the performance on word images extracted from slides is quite good, compared to the other three modalities. This is in line with our earlier observation that text on the slides are pretty legible and easier to recognize among the four modalities. Fig. 6 shows the recognized outputs for a few sample images from the LectureVideoDB dataset using the CRNN-STN-Finetune$_{IAM}$ model in an unconstrained setting. Some of the errors shown in the figure can be attributed to the ambiguity in the original handwritten image.

TABLE V
WORD RECOGNITION PERFORMANCE OF VARIOUS ARCHITECTURE ON THE LECTUREVIDEODB DATASET. HERE THE LEXICON USED WAS THE SET OF ALL UNIQUE WORDS PRESENT IN THE DATASET.

Architecture	WER	CER	Lexicon
CRNN [11]	62.96	27.98	Free
CRNN-STN$_{synth}$	60.68	27.06	
CRNN-STN$_{IAM}$	**58.92**	**26.42**	
CRNN-Finetune [11]	41.66	16.83	
CRNN-STN-Finetune$_{IAM}$	**35.52**	**13.92**	
CRNN-Finetune [11]	22.85	9.78	Based
CRNN-STN-Finetune$_{IAM}$	**20.00**	**8.53**	

TABLE VI
WORD RECOGNITION PERFORMANCE OF VARIOUS ARCHITECTURES ON THE DIFFERENT SPLITS OF LECTUREVIDEODB TEST SET.

Architecture	Data-Split	WER	CER
CRNN-Finetune [11]	Slides	9.12	3.28
	Whiteboard	53.18	30.41
	Paper	42.88	18.29
	Blackboard	41.56	17.79
	Full-Testset	41.66	16.83
CRNN-STN-Finetune$_{IAM}$	Slides	**6.86**	**2.63**
	Whiteboard	**48.92**	**28.47**
	Paper	**34.62**	**13.58**
	Blackboard	**36.82**	**15.71**
	Full-Testset	**35.52**	**13.92**

D. Word Spotting

We follow the evaluation protocol for word spotting as presented in [32] using the train/val/test splits created for the LectureVideoDB dataset. We conduct both query-by-string (QBS) and query-by-example (QBE) on the test corpus. For QBE setting, the queries are the subset of words taken from the test corpus having a frequency more than 1. However all the words were kept in the retrieval set. For QBS scenario, we take the unique set of strings in the test set as queries. In both cases, we report the mean average precision value (mAP) which is standard measure for a retrieval task such as word spotting. We also removed stopwords from the query set and the performance is evaluated in a case-insensitive manner.

Table VII, presents the quantitative results of word spotting on LectureVideoDB dataset under both QBE and QBS setting. Here we first evaluated the performance of the pre-trained End2End embedding network [31] on IAM train set. As one can observe the performance is quite inferior where we report the QBE and QBS mAP of 0.4311 and 0.4531. In comparison with IAM test set performance [31], this is quite low. Further the network is fine tuned using the training samples from the LectureVideoDB. Here we observe a clear improvement of performance with QBE being reported at 0.7909 and QBS of 0.7404, however this still does not reach the level of performance on handwritten words in IAM. This inferior performance can be attributed to the complexity of the underlying problem for spotting text in instructional videos due to the presence of multiple modalities along with the challenges posed by image capture and low resolution videos of some courses.

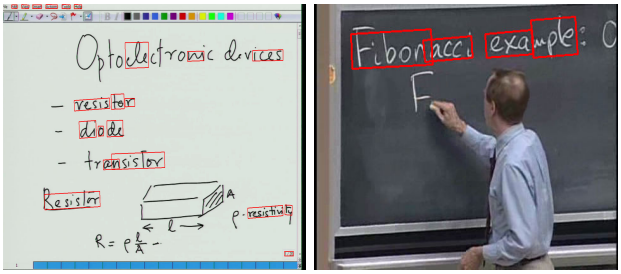

Fig. 5. Sample failure cases for text localization from the LectureVideoDB dataset. In both cases the predicted bounding boxes intersect a single word multiple times. Also in the left image the detector is not able to detect all the words present in the image.

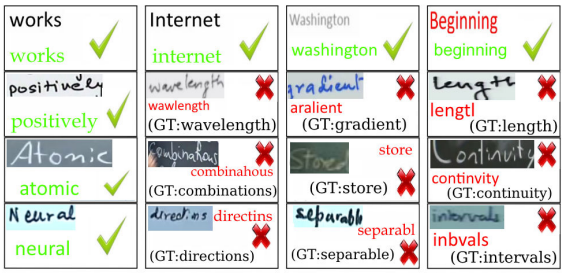

Fig. 6. Qualitative recognition results for word recognition on the LectureVideoDB dataset using the CRNN-STN-Finetune$_{IAM}$ architecture. Examples from slides, whiteboard, blackboard and paper are shown in the first, second, third and fourth row respectively.

TABLE VII
WORD SPOTTING PERFORMANCE USING END2END EMBEDDING ARCHITECTURE ON THE VARIOUS SPLITS OF LECTUREVIDEODB DATASET.

Train Dataset	mAP		Data-Split	mAP	
	QBE	QBS		QBE	QBS
IAM	0.4311	0.4531	Slides	0.7272	0.7157
			Whiteboard	0.3628	0.3234
			Paper	0.2165	0.3882
			Blackboard	0.0721	0.2156
LectureVideoDB	**0.7909**	**0.7404**	Slides	0.8726	0.7977
			Whiteboard	0.6205	0.4799
			Paper	0.8035	0.8037
			Blackboard	0.7151	0.7028

VI. CONCLUSION

In this paper, we present the results of text detection and recognition on the new LectureVideoDB dataset, using existing state-of-the art methods for scene text and handwritten text. In future, we plan to work towards developing methods which can work well on settings where text of multiple modalities appear together in complex and low resolution images. Another problem that we are interested is in making use of the recognized text for larger video understanding problems.

REFERENCES

[1] L. Breslow, D. E. Pritchard, J. DeBoer, G. S. Stump, A. D. Ho, and D. T. Seaton, "Studying learning in the worldwide classroom: Research into edx's first MOOC," *Research & Practice in Assessment*, 2013.

[2] H. Yang, M. Siebert, P. Luhne, H. Sack, and C. Meinel, "Lecture video indexing and analysis using video OCR technology," in *SITIS*, 2011.

[3] F. Wang, C.-W. Ngo, and T.-C. Pong, "Synchronization of lecture videos and electronic slides by video text analysis," in *ACM MM*, 2003.

[4] T. Tuna, J. Subhlok, and S. Shah, "Indexing and keyword search to ease navigation in lecture videos," in *AIPR*, 2011.

[5] G. C. Lee, F.-H. Yeh, Y.-J. Chen, and T.-K. Chang, "Robust handwriting extraction and lecture video summarization," *Multimedia Tools and Applications*, 2017.

[6] R. W. Smith, "History of the Tesseract OCR engine: what worked and what didn't," in *DRR*, 2013.

[7] T. Sato, T. Kanade, E. K. Hughes, M. A. Smith, and S. Satoh, "Video OCR: indexing digital news libraries by recognition of superimposed captions," *Multimedia Systems*, 1999.

[8] T. M. Rath and R. Manmatha, "Word spotting for historical documents," *IJDAR*, 2007.

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

[10] A. Mishra, K. Alahari, and C. V. Jawahar, "Scene text recognition using higher order language priors," in *BMVC*, 2012.

[11] 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," *PAMI*, 2016.

[12] C.-Y. Lee and S. Osindero, "Recursive recurrent nets with attention modeling for OCR in the wild," in *CVPR*, 2016.

[13] K. Dutta, P. Krishnan, M. Mathew, and C. V. Jawahar, "Unconstrained handwriting recognition on devanagari script using a new benchmark dataset," in *DAS*, 2018.

[14] C. Wigington, S. Stewart, B. Davis, B. Barrett, B. Price, and S. Cohen, "Data augmentation for recognition of handwritten words and lines using a CNN-LSTM network," in *ICDAR*, 2017.

[15] P. Voigtlaender, P. Doetsch, and H. Ney, "Handwriting recognition with large multidimensional long short-term memory recurrent neural networks," in *ICFHR*, 2016.

[16] P. Krishnan and C. V. Jawahar, "Matching handwritten document images," in *ECCV*, 2016.

[17] S. Sudholt and G. A. Fink, "PHOCNet: A deep convolutional neural network for word spotting in handwritten documents," in *ICFHR*, 2016.

[18] ——, "Evaluating word string embeddings and loss functions for CNN-based word spotting," in *ICDAR*, 2017.

[19] P. Krishnan, K. Dutta, and C. V. Jawahar, "Deep feature embedding for accurate recognition and retrieval of handwritten text," in *ICFHR*, 2016.

[20] T. Wilkinson and A. Brun, "Semantic and verbatim word spotting using deep neural networks," in *ICFHR*, 2016.

[21] H. Yang and C. Meinel, "Content based lecture video retrieval using speech and video text information," *IEEE Transactions on Learning Technologies*, 2014.

[22] MIT opencourseware. [Online]. Available: https://ocw.mit.edu/index.htm

[23] Khan Academy. [Online]. Available: https://www.khanacademy.org/

[24] NPTEL. [Online]. Available: http://nptel.ac.in/

[25] A. Gupta, A. Vedaldi, and A. Zisserman, "Synthetic data for text localisation in natural images," in *CVPR*, 2016.

[26] LabelImg. Github. [Online]. Available: https://github.com/tzutalin/labelImg

[27] X. Zhou, C. Yao, H. Wen, Y. Wang, S. Zhou, W. He, and J. Liang, "East: An efficient and accurate scene text detector," in *CVPR*, 2017.

[28] M. Liao, B. Shi, and X. Bai, "Textboxes++: A single-shot oriented scene text detector," *arXiv preprint arXiv:1801.02765*, 2018.

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

[30] M. Jaderberg, K. Simonyan, A. Zisserman *et al.*, "Spatial transformer networks," in *NIPS*, 2015.

[31] P. Krishnan, K. Dutta, and C. V. Jawahar, "Word spotting and recognition using deep embedding," in *DAS*, 2018.

[32] J. Almazán, A. Gordo, A. Fornés, and E. Valveny, "Word spotting and recognition with embedded attributes," *PAMI*, 2014.

[33] M. Jaderberg, K. Simonyan, A. Vedaldi, and A. Zisserman, "Synthetic data and artificial neural networks for natural scene text recognition," *arXiv preprint arXiv:1406.2227*, 2014.

[34] U.-V. Marti and H. Bunke, "The IAM-database: an english sentence database for offline handwriting recognition," *IJDAR*, 2002.