Evaluation Framework for Video OCR

Padmanabhan Soundararajan[1], Matthew Boonstra[1], Vasant Manohar[1], Valentina Korzhova[1], Dmitry Goldgof[1], Rangachar Kasturi[1], Shubha Prasad[2], Harish Raju[2], Rachel Bowers[3], and John Garofolo[3]

[1] Computer Science and Engineering, University of South Florida, Tampa, FL, USA
{psoundar, boonstra, vmanohar, korzhova, r1k, goldgof}@cse.usf.edu
[2] VideoMining Corporation, State College, PA, USA
{hraju, sprasad}@videomining.com
[3] National Institute of Standards and Technology (NIST), Information Technology Lab - Information Access Division, Speech Group
{john.garofolo, rachel.bowers}@nist.gov

Abstract. In this work, we present a recently developed evaluation framework for video OCR specifically for English Text but could well be generalized for other languages as well. Earlier works include the development of an evaluation strategy for text detection and tracking in video, this work is a natural extension. We sucessfully port and use the ASR metrics used in the speech community here in the video domain. Further, we also show results on a small pilot corpus which involves 25 clips. Results obtained are promising and we believe that this is a good baseline and will encourage future participation in such evaluations.

1 Introduction

The importance of indexing and retrieval technologies in video is poised for a big leap. There is an ever growing need to do search, based on the text appearing in video. There are more systems coming out with algorithms specifically recognizing text in video content. Evaluating this is equally important to check the progress and also give developers feedback on what scenarios they have difficulties in the transcription.

In this work, we present an evaluation framework specifically designed for evaluating English text recognition in video. While detection and tracking are necessary, they are not evaluated here. Please refer to [1] for a similar evaluation scenario involving detection and tracking text in video. The contributors of the system output were only scored on the recognition performance. For the system to recognize text mean that they are also able to detect the words spatially in the video frame and potentially track them across frames.

2 Text Recognition Task

The goal of the text recognition task is to recognize text objects in a video sequence. This task does not require the system to track these text blocks in a

P. Kalra and S. Peleg (Eds.): ICVGIP 2006, LNCS 4338, pp. 829–836, 2006.

video frame; that part of the task is relegated to the text detection and tracking task. The text will be annotated at the word level according to the annotation guidelines.

The performance of the task is scored at the frame level and be based on how accurate the system recognizes the characters in each word in the frame. The system input and output tags are pre-determined earlier. The text is transcribed at the word level. Text which is annotated as unevaluable by the evaluators and annotators will not be evaluated. To keep things simple in this initial phase, only alpha-numeric characters will be considered, capitalization and word-external punctuation will be filtered from both the system output and reference transcripts. Word-internal punctuations such as hyphens and apostrophes are not filtered. Also, line breaks constitute word boundaries, so wrapped words are treated as separate text tokens. At a higher level, special cases which are not evaluated are:

1. Scrolling text.
2. Dynamic Text
3. Reference Text with Readability Levels Greater Than 1. (See Section 3

For this particular task, annotation tags will include:

1. Video Filename.
2. Object id (unique for the frame).
3. BBox location parameters upper left corner, height, width and rotation attributes for each word.
4. The transcription of each word (each BBox contents).

3 Ground Truth Annotations

For any evaluation, it is important and highly critical to have good quality annotations. There are many ways to annotate a text object and one of the standard method to do so in the OCR community: each text word is bounded by a rectangular box. If the words are occluded then the boxes are approximated and also the specific attributes are marked as occluded so that they can be removed from evaluations if necessary. Additionally as required each individual word box is transcribed so that the error rates can be computed.

There are many free and commercially available tools which can be used for ground truthing videos such as Anvil, VideoAnnex, ViPER [2] and many others. In our case, we used ViPER [1] (Video Performance Evaluation Resource), a ground truth authoring tool developed by the University of Maryland.

Fig 1 shows a sample annotation using ViPER for text in a broadcast news segment. Observe that each word is enclosed in a Bounding box and further, the actual annotations have a unique ID for each box along with their transcriptions.

[1] http://viper-toolkit.sourceforge.net

Fig. 1. Sample Annotation Frame showing Word boundaries in a Broadcast News Clip

3.1 Annotation Guidelines

To ensure quality in-terms of these annotations, a well defined set of guidelines are established which are strictly enforced and adhered by each annotator. Further, some of the clips are doubly annotated (two different annotators annotate the same clip) and their performances compared visually as well as being subjected to rigorous software checks. The software checks are too detailed to list here but essentially the philosophy is that all attributes are compared (each object ID has many attributes) and any inconsistencies are ironed out by fine-tuning the annotation guidelines. This process by itself undergoes numerous iterations.

Every new text area is marked with a box when it appears in the video. Moving and scaling the selection box tracks the text as it moves in succeeding frames. This process is done at the line level (with offsets specified for word boundaries) until the text disappears from the frame.

There are two types of text:

- Graphic text is anything overlaid onto the picture. Example, the "CNN" logo in Fig 1.
- Scene text is anything in the background/foreground of what is actually being filmed.

Text readability consists of three levels. Completely unreadable text is signified by READABILITY = 0 and is defined as text in which no character is identifiable. Partially readable text is given READABILITY = 1 and contains characters that are both identifiable and non-identifiable. Clearly readable text is assigned READABILITY = 2 and is used for text in which all letters are identifiable.

The OCCLUSION attribute is set to TRUE when the text is cut off by the bounds of the frame or by another object. The LOGO attribute is set to TRUE when the text region being marked is a company logo imprinted in stylish fonts. Example, the text "CNN" in Fig 1.

Of all the objects of interest in video, text is particularly difficult to be uniformly bound. For this reason, text regions are marked meticulously based on a comprehensive set of rules, namely,

- All text within a selected block (word) must contain the same readability level and type.
- Blocks of text (word) must contain the same size and font. Two allowances are given to this rule. A different font or size may be included in the case of a unique single character and the font color may vary among text in a group.
- The bounding box should be tight to the extent that there is no space between the box and text. The maximum distance from the box to the edge of bounded text may not exceed half the height of the characters when Readability = 2 (clearly readable). When Readability = 0 or 1 the box should be kept tight but does not require separate blocks for partial lines in a paragraph.
- Text boxes may not overlap other text boxes unless the characters themselves are specifically transposed atop one another.

The additional set of attributes described above is used in deciding whether a particular text region should be evaluated. The specific settings for evaluating a text region used in this evaluation are - TEXT-TYPE = Graphic, READABILITY = 2, OCCLUSION = FALSE and LOGO = FALSE.

All other regions are treated as "Don't Care" where the system output is neither penalized for missing nor given credit for detecting. It has to be noted that each of these attributes can be selectively specified to be included in evaluation through the scoring tool that we have developed.

4 Performance Measures

The performance measure for the recognition task is based on insertion, deletion and substitutions errors at the word level. The measure requires a unique one-to-one mapping of ground truth and detected text object using some optimization (see Section 4.1). The mapping will be performed using spatial information and also WER (Word Error Rate) score obtained. Both of these have equal weighting in the internal matching algorithm. By this strategy, we make sure that the system generated and the reference words are closest to each other both in the spatial sense and also in the language sense. The Word Error Rate is defined as:

$$WER(t) = \frac{(Insertion + Substitution + Deletion)}{(\text{Total Reference Words})} \tag{1}$$

where t indicates the particular frame. The WER(t) is then averaged for the full clip and on the whole dataset to obtain the Word Error Rate (WER).

On each mapped word, we also compute the Character Error Rate (CER). The true CER is then averaged out for the entire set of words in the whole dataset. The WER and CER are both standard error metrics in the Speech Recognition Evaluations [3]. Fig 2 shows an example explaining the impact on WER measure resulting from Insertion, Substitution and Deletion errors.

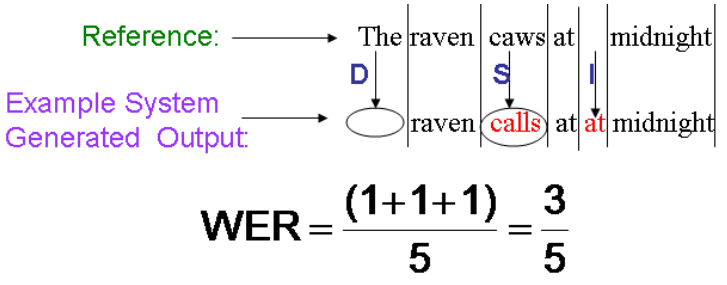

Fig. 2. Example WER Computation on different system generated errors

4.1 Matching Strategies

The maximal scoring is obtained for the *optimal* ground-truth and system output pairs. Potential strategies to solve this assignment problem are the weighted bipartite graph matching [4] and the Hungarian algorithm [5].

	DT_1	DT_2	...	DT_M
GT_1	x			
GT_2				x
⋮				
GT_N		x		

Assume that there are N ground truth (GT) objects and M detected (DT) objects. A brute force algorithm would have an exponential complexity, a result of having to try out all possible combination of matches (n!). However, this is a standard optimization problem and there are standard techniques to get the optimal match. The matching is generated with the constraint that the sum of the chosen function of the matched pairs is minimized or maximized as the case may be. In usual assignment problems, the number of objects in both cases are equal, i.e, when $N = M$. However, this is not a requirement and unequal number of objects can also be matched.

There are many variations of the basic Hungarian strategy most of which exploit constraints from specific problem domains they deal with. The algorithm has a series of steps which is followed iteratively and has a polynomial time complexity, specifically some implementations have $O(N^3)$. Faster implementations have been known to exist and have the current best bound to be at

$O(N^2 logN + NM)$ [6]. In our case, we take advantage of the fact that the matrix is mostly sparse by implementing a hash function for mapping sub-inputs from the whole set of inputs.

5 Results and Conclusions

The results are obtained on 25 clips in the Broadcast News domain. These clips contain both CNN and ABC newsfeeds. The total time of video evaluated is about 62 minutes. The total number of word objects that occurred in this entire dataset is 4178. The total number of word frame instances is 68,738. Since this is a pilot study, we had only one participant (anonymized here). This is helpful in setting a baseline for this task before beginning a formal evaluation.

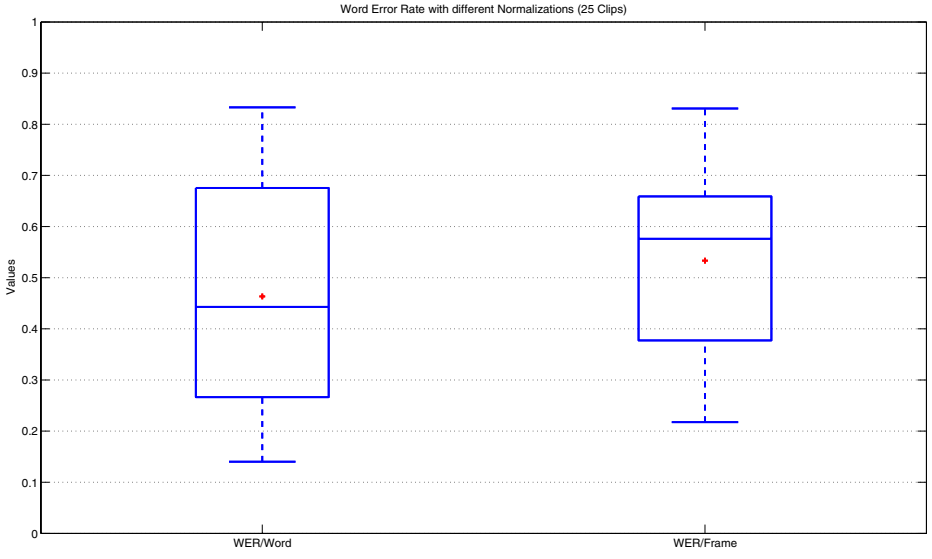

Fig. 3. WER Score Distribution on all 25 Clips (with different normalizations) '+' indicates the mean value

The WER obtained over the entire dataset is 0.423 and the CER is 0.282. Fig 3 shows the boxplots of the WER obtained using different normalizations. We can infact compute the error rates with respect to the total number of words occuring in a particular clip: the distribution of which is shown in the first boxplot. The second boxplot shows the scores obtained after normalizing with respect to the total number of frames in the entire clip.

We should also note that some of these errors could potentially occur due to the system locating the word at a wrong location (since detection is inherently assumed). We could re-evaluate the performance by giving prior knowledge of

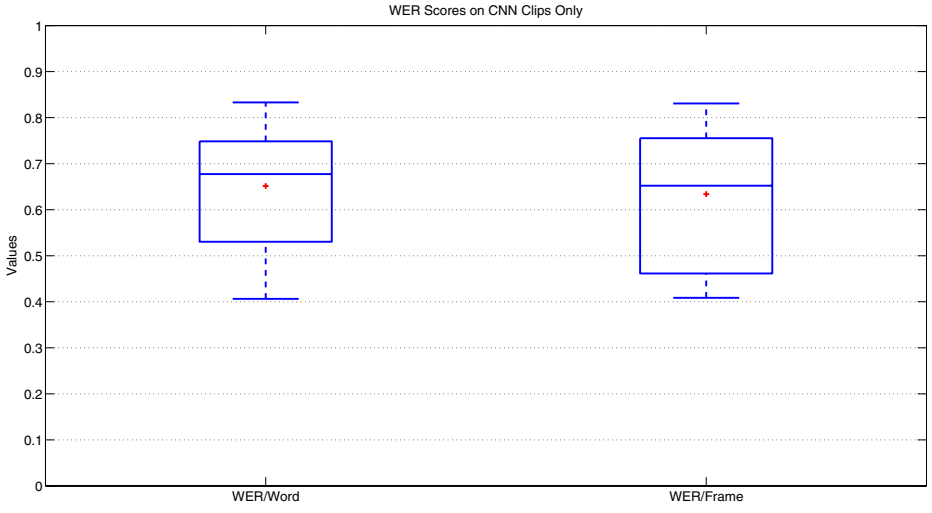

Fig. 4. WER Score Distribution on only the CNN Clips (12 clips) '+' indicates the mean value

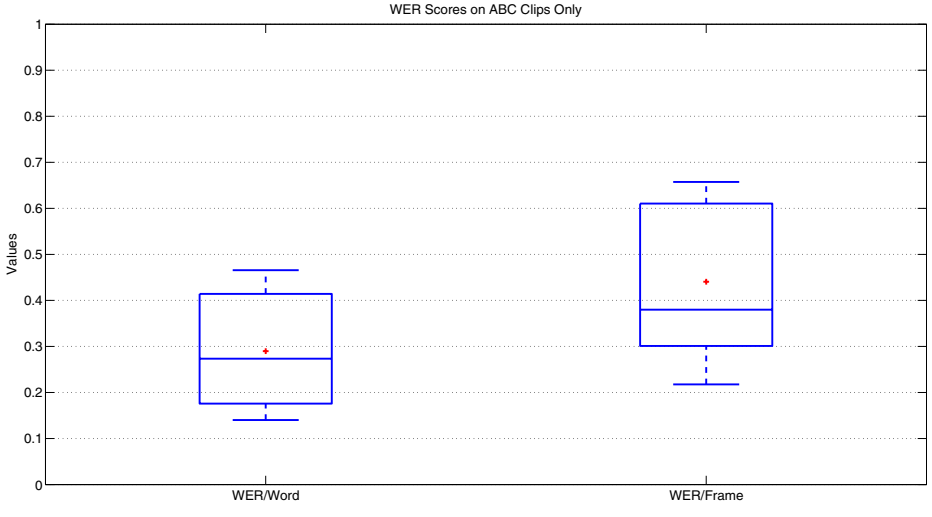

Fig. 5. WER Score Distribution on only the ABC Clips (13 clips) '+' indicates the mean value

the word locations and get the recognition error rates. Nevertheless, we again re-iterate that the scores obtained here are good baselines that can be improved.

Figs 4 and 5 shows the performance based on the CNN and ABC newsfeeds. As can be seen, the performance on CNN clips is worse than the performance

on ABC clips. This can be attributed to the fact that there are less captions in the ABC newsfeeds compared to the CNN for this dataset.

We have shown a practical OCR evaluation framework in video. Useful annotations and metrics have resulted in making this evaluation framework possible. In future, more challenging forms of text including other languages can also be evaluated. We could also include harder to read as well as dynamic and scrolling text. Further, we could also extend the evaluation to include semantic knowledge where a system has to include a knowledge model for better performance. Challenges arise in the form of defining newer metrics, refine the annotations and also the interpretations of the systems for scoring.

References

1. Manohar, V., Soundararajan, P., Boonstra, M., Raju, H., Goldgof, D., Kasturi, R., Garofolo, J.: Performance Evaluation of Text Detection and Tracking in Video. In: 7_{th} IAPR Workshop on Document Analysis Systems (DAS). Volume 3872. (2006) 576–587
2. Doermann, D., Mihalcik, D.: Tools and Techniques for Video Performance Evaluation. In: ICPR. Volume 4. (2000) 167–170
3. McCowan, I., Moore, D., Dines, J., Gatica-Perez, D., Flynn, M., Wellner, P., Bourlard, H.: "on the use of information retrieval measures for speech recognition evaluation". Technical report, IAIDP (2005)
4. Papadimitriou, C.H., Steiglitz, K.: Combinatorial optimization: algorithms and complexity. Prentice-Hall, Inc., Upper Saddle River, NJ, USA (1982)
5. Munkres, J.R.: Algorithms for the Assignment and Transportation Problems. J. SIAM **5** (1957) 32–38
6. Fredman, M.L., Tarjan, R.E.: Fibonacci Heaps and their uses in Improved Network Optimization Algorithms. Journal of ACM **34** (1987) 596–615