Using Audio Based Disambiguation for Improving Handwritten Mathematical Content Recognition in Classroom Videos

Smita Vemulapalli Monson H. Hayes III
Center for Signal and Image Processing (CSIP), School of Electrical and Computer Engineering
Georgia Institute of Technology, Atlanta, GA 30332, USA
{smita,mhh3}@gatech.edu

ABSTRACT

We consider the problem of recognizing handwritten mathematical content in classroom videos that capture the content written on the whiteboard and the content spoken by the instructor. While the problem of recognizing handwritten textual content from videos has been studied before, recognition of handwritten mathematical content and the use of audio content from classroom videos to assist in recognition, however, presents us with a new set of challenges. In this paper, we outline such challenges, and present the description of an end to end system that makes use of both video and audio based recognition components to improve the accuracy of handwritten mathematical content recognition. We have implemented the system using an open source implementation of a text recognizer and a commercially available phonetic word spotter. Preliminary results reported in this paper demonstrate the viability of our approach.

1 Introduction

Extraction, identification and summarization of classroom video content has received considerable attention recently [8, 9]. This increased activity, in part, can be attributed to the numerous e-learning and advanced learning initiatives that either use classroom videos as the primary medium of instruction or make them available online for reference by the students. As the volume of such recorded video content increases, it is amply clear that in order to efficiently browse and/or search through the available video content there is a need for tools that can extract, identify and summarize the content of such videos. In this paper, we focus on the problem of recognizing handwritten mathematical content in classroom videos that capture the handwritten content on the whiteboard and the content spoken by the instructor.

The choice of the whiteboard as a preferred and effective medium for explaining and teaching complex mathematical and scientific concepts is well established [8]. While there has been some research in recognizing handwritten textual content from classroom videos [9], our work, to the best of our knowledge, is the first to focus on recognition of mathematical content from classroom videos. Although searches are still likely to be performed on the textual data, for completeness and easy retrieval it is necessary to recognize and convert the mathematical content written on the whiteboard into the same digital format as the recognized textual content. To improve the recognition accuracy, in the face of occlusions, image noise, camera movement and the inherent complex structure of the mathematical content, our approach (as shown in Figure 1) relies on using audio recognition to assist in character and structure disambiguation. Given the noisy nature of the classroom videos, there is an overabundance of challenges, even when dealing with only the video or the audio component for content recognition. In this paper, however, we focus more on the challenges that arise due to the use of both video and audio text recognition components and due to the mathematical nature of the content. These challenges are described next:

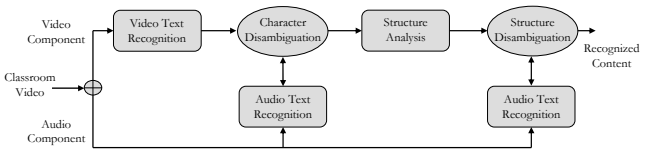

Figure 1: Proposed end-to-end system

- *Mathematical Content* - Most of the existing speech recognition engines perform very poorly for mathematical content due to mismatch in mathematical grammar and English grammar which such engines are trained upon. Also, there are several non-trivial issues that relate to structure analysis for handwritten mathematical content.
- *Corresponding Audio Segment Identification* - Unlike many other recognizer (classifier) combination techniques, our approach is based on recognizers that operate on different (video and audio) streams. In particular, given a set of recognized candidate characters generated by the video text recognizer it is non-trivial to find the appropriate vicinity in the audio stream.
- *Weighted Combination Approach* - There are several considerations based on which the set of recognized candidate characters or candidate structures generated by the video and the audio text recognizer can be combined. These include weighted combination techniques where the weights assigned to each recognizer can vary for each character and/or mathematical structure, or could be dependent on the content and clarity of individual video and audio streams contained in the video, or in a more general implementation be dependent on the instructor.

1.1 Roadmap

The remainder of this paper is organized as follows. In Section 2, we present an overview of the proposed end-to-end system. Section 3 contains a more detailed description of the system, focusing mainly on the video text recognition, the audio text recognition and the character disambiguation stages. Section 4 contains the implementation details followed by the related work in Section 5. Finally we conclude in Section 6 with a brief discussion of the future work.

2 System Overview

In this section, we present an overview of the proposed end-to-end system that can be used to recognize handwritten mathematical content from classroom videos that capture the handwritten content on the whiteboard as well as the content spoken by the instructor. As shown in Figure 1, there are two main stages in the mathematical content recognition system: the *character recognition* stage and then the *structure analysis* stage. Our approach involves making use of the audio component to disambiguate the purely video based recognition results. Therefore, we make use of the audio in both stages of our system. The *character disambiguation* stage makes use of the audio component to reduce the ambiguity in the character recognition results of a purely video-based text recognizer. Similarly, the *structure disambiguation* stage makes use of the audio to reduce ambiguity in the purely video-based structure analysis results. The *video text recognition* stage includes all preprocessing required to extract and segment handwritten text (in this case, mathematical content) regions from the video followed by the recognition of segmented characters. The *audio text recognition* stage may be a speech recognition engine or an audio word spotting tool that can recognize mathematical audio content. A detailed description of these stages is provided in the following section.

3 Detailed Description

The following sub-sections provide more details about the various recognition and disambiguation stages involved in the proposed end-to-end system.

3.1 Video Text Recognition

The video text recognizer, as shown in Figure 2, takes the video component of the classroom video as input and outputs the recognized handwritten mathematical characters. Since the input is a video stream, there are multiple preprocessing steps that must completed on the video stream before the characters can be recognized. To perform character segmentation, we employ a component-labeling algorithm that uses the contour tracing technique [2] to detect all blobs (connected components) in an image. Additionally, thresholds are used to remove foreground objects and background noise. Whenever a new blob is detected, it is placed into a virtual frame into which all subsequently detected blobs will be placed. This virtual frame is called the *Frame of Interest (FOI)*. The *Duration of Interest (DOI)* corresponds to the duration between two complete erasures of the whiteboard by the instructor and may be used to determine the audio search window. A sample graph depicting the variation of the number of blobs detected with time in a controlled environment with minimal occlusions is shown in Figure 3. In this case, the *FOI* is comparable to the peaks in graph and the *DOI* refer to the segments between two consecutive *FOI*. Automatic timestamp generation is performed by using the time of the frame when the segmented character (blob) first appeared in the video. The segmented

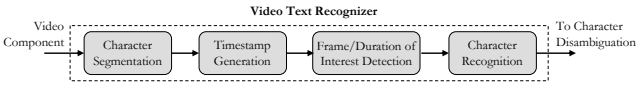

Figure 2: Video text recognizer

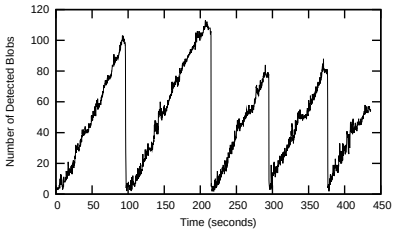

Figure 3: Variation of the number of detected blobs with time

characters with timestamps are then passed to the character recognizer, which may return multiple recognition results (called *video options*) along with the corresponding confidence attributes, for every segmented character. Section 4 provides more details about the character recognizer used in our implementation.

3.2 Audio Text Recognition

Most of the existing speech recognizers rely heavily on context and grammar information to perform recognition of English text. However, due to fact that mathematical audio input does not follow the same grammar, such recognizers perform very poorly for mathematical content. To avoid such issues, we make use of a commercial phonetic word spotting tool as part of our audio text recognizer. The character disambiguation stage supplies the audio text recognizer with a list of possible *video options* along with the generated timestamp for each segmented character. The word spotting tool searches the audio component for all possible occurrence of audio segments that may correspond to each *video option*. For instance, *video option* '+', in our system, corresponds to the audio equivalents 'plus', 'add', 'addition', 'sum' and 'summation' that need to searched in the audio component. As seen in Figure 5, for every *video option* that is searched, several occurrences of the audio equivalents may be detected in the audio component. These are the *audio options* corresponding to the *video option* being searched. When the word spotter outputs several *audio options* for a single *video option*, we compute a set of audio features for each *audio option* and use these features for initial pruning of the unlikely *audio options*. These computed features are used once again in the character disambiguation stage. Audio features may include the confidence attribute associated with each recognized *audio option*, the difference between the time of occurrence of the character in the video component and in the audio component and also the presence of the textual neighbors in the neighborhood of the audio occurrence. Both the audio text recognizers in Figure 1 perform in a similar manner although the nature of the input search strings given to each of these recognizers may not be the same.

3.3 Character Disambiguation

The character disambiguation stage receives, as input, the *video options* that are generated by the video text recognizer for each segmented character. The confidence attribute associated with each *video option* is used to spot ambiguity in the recognition of a given segmented character. The audio text recognizer is then called for each ambiguous character by passing the corresponding *video options* as input. As mentioned earlier, the audio text recognizer returns a set of

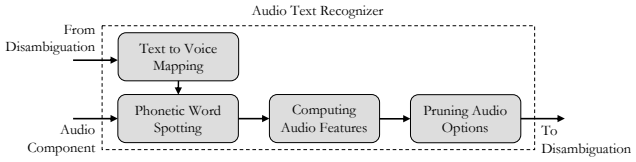

Figure 4: Audio text recognizer

Video Options (time = 52.40)	
Text	*Probability*
g	0.75
8	0.67
S	0.66

Audio Options					
Time	*Prob*	*Time*	*Prob*	*Time*	*Prob*
60.57	0.55	77.84	0.51	62.95	0.41
25.11	0.66	57.15	0.66	102.22	0.62
65.40	0.64	59.38	0.62	109.66	0.53

Figure 5: Example showing the video and audio options for a single blob

possible *audio options* and the corresponding confidence attributes, for each *video option*. The disambiguation proceeds in two steps. In the first step, the audio features are used to select the most likely *audio option* corresponding to every *video option*. In the second step, a final decision is made for selecting the best *video option* based on the confidence attributes generated by both the recognizers.

An example showing the disambiguation step is shown in Figure 5. The figure shows three *video options* corresponding to a single segmented character and also the corresponding *audio options*. Some of the these *audio options* may be eliminated at the pruning step of the audio text recognizer. In the first step of the disambiguation, at most one *audio option* is selected for each *video option* and in the second step, the best *video option* is selected for the given segmented character. In our experiment, we were able to correctly recognize this segmented blob as the character '8' in spite of the error in video text recognition due to the use of the audio component for character disambiguation.

3.3.1 Specifics & Simulation Results

To detect ambiguity in the video text recognizer output, we experimented with several types of thresholding conditions that need to be satisfied for a character to be considered ambiguous. These conditions listed in Equations 1, 2 and 3 use τ_1, τ_2 and τ_3 as threshold values. Let (m_i, c_i) represent a single *video option* with m_i being the recognized mathematical character and c_i being the corresponding confidence attribute. The *video options* $\mathcal{V}$ as generated by our recognizer can then be represented as $\mathcal{V} = \{(m_i, c_i) | 1 \leq i \leq n \wedge c_i > c_{i+1} \forall i < n\}$. When using Equations 2 and 3, those m_j whose c_j do not satisfy the thresholding condition may be pruned away. In our experiments we have used a combination of thresholding functions from Equations 1 and 2.

$$c_1 < \tau_1 \quad (1)$$

$$\exists j, 1 < j \leq n \mid c_j/c_1 > \tau_2 \quad (2)$$

$$\exists j, 1 < j \leq n \mid c_1 - c_j < \tau_3 \quad (3)$$

For the first step of the disambiguation, we have used the rank-based decision making technique Borda Count [7] on the audio probabilities and the audio features to select the best *audio option* for every *video option*. For tie breaking, we have used the value of the audio feature corresponding to the time difference between occurrence of the segmented character in the video component and occurrence of the corresponding voice in the audio component. The second step implements the Borda Count technique on the video and audio confidence attributes to generate the final character recognition result for the segmented character. In case of a tie, we employ the weighted sum rule on the video and audio confidence attributes.

Figures 6, 7 and 8 shows the variation in video text recognition accuracy after disambiguation when different weights are assigned to each of the recognizers *i.e.* the audio and video components. The output of the purely video-based text recognizer was modified to simulate the range of recognition accuracies. However, the regions corresponding to significantly high values of video recognition accuracies are not of interest as we may not consider the characters to be ambiguous and chose not to use the audio component. We are more interested in the mid range of video recognition accuracies where audio disambiguation improves the video character recognition results and we notice that the GOCR engine tends to work close to this range.

We can see in Figure 5 that most of the audio probabilities generated by the word spotting engine are in the range of 0.4 to 0.6. The reason for this is that the characters 'g', '8' and 'S' have very few phonemes in their search string. This may increase the chances of false positives and in this case leads to lesser values of the confidence attribute. Such problems do not arise when searching for larger search strings such as 'addition' and 'integral'. We have conducted simple experiments which confirm that by using the neighbors as part of the search string, we may get more confident word spotting results . For example, when searching for the characters '7', '4' and '6' in the audio component, each of these characters were found correctly by the word spotting tool but the audio probabilities associated with them were 0.45, 0.56 and 0.61 respectively whereas the audio search of the string '746' resulted in a voice probability of 0.90. We are currently working on implementing an algorithm that generates several possible search strings for the word spotting tool based on the video neighbors.

3.4 Structure Analysis

The structure analysis stage takes the recognized characters along with their timestamps and the location on the whiteboard and converts it into a one-dimensional representation of the mathematical equation. This stage often requires a suitable data structure and mathematical grammar for parsing the equations. We are currently in the process of incorporating a suitable structure analysis implementation into our system. The errors introduced during the character recognition stage may make the task of structure analysis even more challenging.

3.5 Structure Disambiguation

The structure of equations is expressed and understood better from the video component. However, there are scenarios where the audio component may prove to be useful in disambiguation. For example, when writing 'a^2' in an equation, if the '2' is not significantly above the base 'a', it may be considered to be '$a2$' or if the '2' is written well above the 'a' and closer to another equation, the '2' may be assigned to the other equation. In such cases, we may be able to spot the expression "*a squared*" in the audio component and this may be used for structure disambiguation.

4 Implementation

The end-to-end system described above is currently being implemented in C++. While our system and disambiguation algorithms are not tied to a specific implementation of

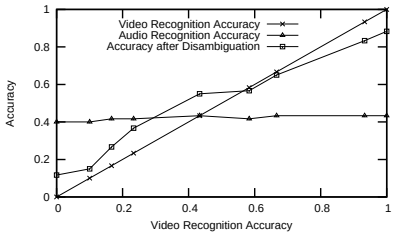

Figure 6: Disambiguation using weighted sum-rule (w_v=0.5,w_a=0.5)

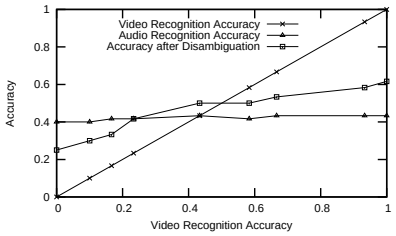

Figure 7: Disambiguation using weighted sum-rule (w_v=0.25,w_a=0.75)

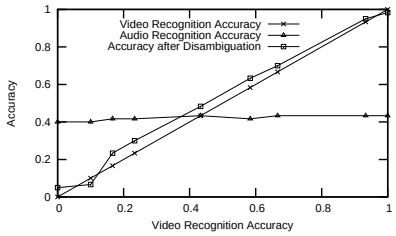

Figure 8: Disambiguation using weighted sum-rule (w_v=0.75,w_a=0.25)

a character recognizer or a speech recognizer, our current implementation, however, makes use of the GOCR [4] character recognizer and Nexidia's audio word spotter [6] for the preliminary evaluations and findings reported in this paper. These software components are described next.

GOCR. GNU Optical Character Recognition (GOCR) [4] is an open source OCR program that converts text images into machine-readable text. It is a rule-based recognizer that was designed for printed characters. Therefore, it is works best when the handwritten characters looks very similar to the printed equivalent, such as in characters *'m'* and *'n'* and may lead to more recognition errors when dealing with characters such as *'r'* and *'q'* which may often be handwritten in the cursive style instead of the printed style. The GOCR engine is designed to return a single recognized character for every segmented character and in cases when the match calculated internally is below a threshold, the character is not recognized. Our modified version of GOCR returns several *video options* for each segmented character, even in cases when the match is not very high. The modified engine also returns a corresponding confidence attribute for every *video option* which is proportional to the number of recognition rules that have been satisfied. The current system recognizes alphabets, numbers and basic arithmetic operators and we are in the process of extending the character set of the recognizer.

Nexidia. Our implementation of the audio text recognizer makes use of a commercially available phonetic word spotting software called Nexidia [6]. Nexidia converts the audio files into a time-aligned phonetic sequence, performs searches using a patented phonetic search technology and returns the time and the confidence attribute associated with every "hit" of the search string. High search speeds and the ability to work with non-standard grammar patterns and different languages without prior training are some of the advantages of this software.

5 Related Work

There exists a vast collection of literature related to the recognition of both printed and handwritten text. However, due to the structure analysis stage, the problem of recognizing mathematics is quite challenging and makes use of a different set of techniques [1]. Most of these techniques work on equations that are input by scanning a printed document [3, 5] or by using special tools such as a tablet PC and other pen-based systems [10, 11]. This paper outlines a system to recognize handwritten mathematical content from a classroom video. Although several video based text recognition systems [9] have been implemented, they mostly work on the text and not mathematics. Also, we make use of the audio content to disambiguate the video text recognition results and again in the structure analysis stage. This audio-based disambiguation makes use of simple rank-based and value-based classifier combination techniques [7] after addressing the fact that there is no one-to-one correspondence between the characters written and the words spoken in the classroom video and also finding the relevant audio segment for a written character.

6 Conclusions and Future Work

In this paper we proposed an end-to-end system for improving the accuracy of handwritten mathematical content recognition from classroom videos. The proposed system relies on the audio component of the classroom video to resolve the ambiguities in recognition of characters and the analysis of the structure of mathematical content. Our preliminary experiments suggest that, if utilized intelligently, audio based disambiguation can help in improving the accuracy of handwritten mathematical character recognition. The problem domain presents one with a wide array of future challenges that range from sophisticated audio-based character disambiguation and structure disambiguation techniques to adaptive weight assignment (for different recognizers, different characters or different users) for combining the confidences reported by video and audio based text recognizers. The creation of a suitable data set of classroom videos (with both video and audio components) for training and testing purposes is another avenue for future work.

7 Acknowledgments

The authors would like to thank Nexidia Inc. for providing the phonetic search software used in our implementation.

8 References

[1] K.-F. Chan and D.-Y. Yeung. Mathematical expression recognition: a survey. *IJDAR*, 3(1):3–15, 2000.

[2] F. Chang, C.-J. Chen, and C.-J. Lu. A linear-time component-labeling algorithm using contour tracing technique. *Comput. Vis. Image Underst.*, 93(2):206–220, 2004.

[3] R. J. Fateman, T. Tokuyasu, B. P. Berman, and N. Mitchell. Optical character recognition and parsing of typeset mathematics. *Journal of Visual Communication and Image Representation*, 7:2–15, 1996.

[4] Gocr. *http://jocr.sourceforge.net/*.

[5] H.-J. Lee and M.-C. Lee. Understanding mathematical expressions using procedure-oriented transformation. *Pattern Recognition*, 27(3):447–457, 1994.

[6] Nexidia. *http://www.nexidia.com/*.

[7] S. Tulyakov, S. Jaeger, V. Govindaraju, and D. Doermann. Review of classifier combination methods. In *Studies in Computational Intelligence: Machine Learning in Document Analysis and Recognition*, pages 361–386. 2008.

[8] L. wei He and Z. Zhang. Real-time whiteboard capture and processing using a video camera for remote collaboration. *IEEE Transactions on Multimedia*, 9(1):198–206, 2007.

[9] M. Wienecke, G. A. Fink, and G. Sagerer. Toward automatic video-based whiteboard reading. *IJDAR*, 7(2–3):188–200, 2005.

[10] R. Zanibbi, D. Blostein, and J. R. Cordy. Recognizing mathematical expressions using tree transformation. *IEEE Trans. Pattern Anal. Mach. Intell.*, 24(11):1455–1467, 2002.

[11] R. C. Zeleznik, T. Miller, C. Li, and J. J. L. Jr. Mathpaper: Mathematical sketching with fluid support for interactive computation. In *Smart Graphics*, pages 20–32, 2008.