AUTOMATIC DETECTION AND EXTRACTION OF ARTIFICIAL TEXT IN VIDEO

Jovanka Malobabić, Noel O'Connor, Noel Murphy, Sean Marlow

Adaptive Information Cluster, Centre for Digital Video Processing

Dublin City University, Dublin, Ireland

Jovanka@eeng.dcu.ie

ABSTRACT

A significant challenge in large multimedia databases is the provision of efficient means for semantic indexing and retrieval of visual information. Artificial text in video is normally generated in order to supplement or summarise the visual content and thus is an important carrier of information that is highly relevant to the content of the video. As such, it is a potential ready-to-use source of semantic information. In this paper we present an algorithm for detection and localisation of artificial text in video using a horizontal difference magnitude measure and morphological processing. The result of character segmentation, based on a modified version of the Wolf-Jolion algorithm [1][2] is enhanced using smoothing and multiple binarisation. The output text is input to an "off-the-shelf" non-commercial OCR. Detection, localisation and recognition results for a 20min long MPEG-1 encoded television programme are presented.

1. INTRODUCTION

The need to handle large volumes of digital video data highlights the importance of the provision of efficient means for automated content-based indexing. The real value of the information stored in a large digital video archive is dependent on its accessibility. Artificial (i.e. open caption or non-scene) text appearing in video is usually closely related to the visual content and is a strong candidate for high-level semantic indexing for retrieval, offering an alternative or complementary approach to indexing based on low-level features extracted from the video or audio signal. An index built by detecting, extracting and recognising the artificial text contained in a video sequence enables keyword-based queries in a manner similar to text-based retrieval.

Our approach for detecting and extracting artificial text regions in uncompressed video frames is based on localisation of regions featuring a high concentration of short vertical edges that are horizontally aligned. Text regions are enhanced by smoothing and bi-linear interpolation and are subsequently binarised by local thresholding in order to retain only pixels that exhibit high local contrast relative to the maximum contrast of the image.

The paper is organised as follows. In section 2, approaches upon which our work is based are summarised. Section 3 presents the algorithmic details of our approach for detection, localisation, enhancement and character segmentation. Section 4 presents the evaluation procedure and results obtained. Finally, section 5 provides a conclusion.

2. RELATED WORK

The vast majority of algorithms for text detection and extraction make use of typical characteristics of artificial text appearing in video, such as high contrast to the background, high density of short edges of varying orientation, horizontal alignment, various geometrical properties and temporal stability [1][2]. The first algorithms for detection/extraction of text from images were developed for still images. The methods used for still images had to be adapted for use with video given factors such as the considerable difference in quality, the low resolution of video frames, the presence of noise, the possibility of characters touching and complex backgrounds. Additional challenges to be addressed are the diversity of fonts, styles, colours, size and orientations that text occurring in video can exhibit.

Lienhart and Effelsberg [3] used colour segmentation in the RGB space combined with edge analysis and empirically determined geometrical restrictions without making any assumptions about the text alignment. Temporal redundancy of text in video was exploited to eliminate non-text regions. The disadvantage of their approach is that it appears to work for large fonts only.

The approach of Lienhart and Wernicke [4], which used the properties of high contrast and high frequency to detect and localise the occurrences of artificial text, was capable of handling text sizes ranging from 8 pixels to half the frame, as well as estimating the text colour by colour quantization and comparison of colour histograms. Temporal redundancy was exploited to determine the colour, size and position of a particular text occurrence through comparison of colour, size and position of text located in adjacent frames.

Miene, Hermes and Ioannidis [5] adopted an approach based on region-growing methods in a colour-segmented image, followed by segmentation of characters from the background based on size and alignment constraints. Character candidates are clustered into word candidates by clustering regions of similar colour and height whose length does not exceed a certain maximum value.

Wolf and Jolion [1][2] applied a detection algorithm to each frame of the video sequence. All processing was performed on grey-scale images. Their approach makes use of following properties of text in video: (i) grey level properties (high contrast in given directions), (ii) morphological properties (spatial distribution and shape), (iii) geometrical properties (height, width, height-to-width ratio) and (iv) temporal properties (stability). Temporal redundancy is exploited to determine the final text bounding boxes and to obtain an enhanced image that is then binarised and passed to OCR for recognition. A combination of morphological processing and imposition of geometrical constraints was used to remove non-text regions. Segmentation was performed using a modified Niblack's algorithm [1][2], which uses local thresholding.

The principal differences between our approach and that of [1] [2] are that our detection method is applied to every I-frame only, that we use the magnitude of the symmetrical horizontal difference as a measure of probability that a pixel belongs to a text region, that all text regions in frame are bounded by a single box, and that text segmentation is performed twice.

3. THE PROPOSED APPROACH

The functional diagram of the system is presented in Figure 1. The detection algorithm, which operates in the uncompressed domain, is applied to every I-frame of the MPEG-1 video sequence only, thus exploiting the temporal stability of artificial text. A single rectangular box that bounds all candidate text regions is defined for each I-frame. Following image enhancement and morphological processing these rectangular boxes are cropped and binarised, and subsequently passed to the OCR module. These steps are described in more detail in the following.

3.1. Detection and localisation

Our detection method is based on texture analysis, relying on the property of Latin script to form a texture characterised by a high density of vertical edges aligned horizontally. The method operates on an uncompressed video frame in a YUV colour space. Temporal stability of text in video is taken into account through an assumption that a particular text appearance has to remain visible for a certain minimum duration (i.e. approx. 1 second) in order to be readable. It is therefore sufficient that only I-frames be examined and analysed for the presence of text that appears over a number of consecutive frames [4].

3.1.1. Edge detection and processing

The magnitude of the symmetrical horizontal difference is calculated for each pixel in the luminance component of the frame. Each pixel value in the resulting image is a measure of the probability that it belongs to a text region. Pre-processing prepares the edge map for binarisation by joining the vertical edges horizontally into clusters corresponding to words and text lines. The edge map is first smoothed using a 3x3 binomial filter, which is followed by blurring horizontal using a 3x1 mask. A small blurring mask is used in order to avoid connecting noisy areas to areas containing text. Erosion by a cross mask is then carried out to clear the top intensity layer in the greyscale image [6]. As a result, bright text areas slightly shrink in size, but so do the noisy edges in the background. Subsequent smoothing increases the size of text regions as does a final 3x3 dilation. Figure 2 illustrates the effects of some of these processing steps.

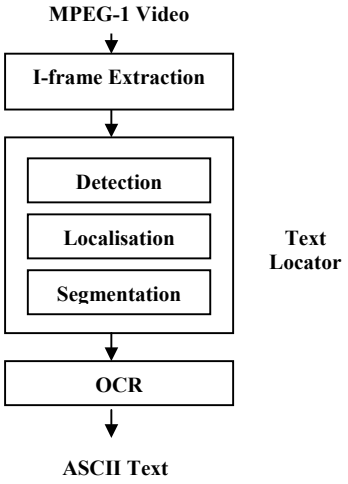

Fig.1. System block diagram

3.1.2. Binarisation

Binarisation of the edge map is performed in order to separate text-containing regions from the rest of the frame using Otsu's global thresholding method as described in [1][2]. An optimal threshold is calculated based on the grey level histogram by assuming Gaussian distributions of text pixels and non-text pixels. The method aims to maximise the interclass variance. The optimal threshold is calculated using the formula [1][2]:

$$t = \arg\max_t (\omega_0 \omega_1 (\mu_1 - \mu_0)^2)$$

where ω_0 is the normalised mass of class 0 (i.e. the number of pixels in the class divided by the total number of pixels in the image), ω_1 is the normalised mass of class 1, and μ_1 and μ_0 are mean grey levels for each of the classes. Unlike [1][2], in this system thresholding is implemented based on a 64-bin histogram using a single threshold. Ideally, this step results in an image featuring clusters of white pixels in areas corresponding to the text regions. In practice, small clusters of white pixels may appear elsewhere in the frame. Binarisation is followed by post processing to remove these noisy areas. Figure 2 shows the result of the edge map binarisation before and after morphological processing.

3.1.3. Fitting bounding boxes

The aim of this step is to fit a single bounding box that encloses all text areas in the frame. This requires that as much noise as possible be removed beforehand, otherwise there is a risk that the bounding box may potentially grow to reach the size of the

Fig. 2. (a) Input image, (b) horizontal difference magnitude, (c) (d)binarised edge map before and after morphological processing

frame. As can be seem from Figure 2, the binarised edge map contains some noise pixels. In order to remove these, several steps are taken. The first step is to use a 3x3 median filter that deals well with the noise spikes whilst preserving the edges. In the sample frame the benefit of median filtering in removing noise is not obvious as the noise spot is not a single pixel. However, its averaging effect is clear as the tiny black spots were removed from the white regions. The next step is a 3x1 dilation followed by a 5x5 opening. As can be seen in Figure 2, a 5x5 erosion succeeds in removing the noisy spot while the subsequent dilation with the same size structuring element restores the desired white clusters to their initial size.

Finally, a dilation in the horizontal direction using a 7x1 structuring element connects text pixels into text lines. In order to compensate for any damage to text regions during the previous processing, the text box size is adjusted by growing it by 5 pixels in all four directions. Geometrical constraints are imposed on bounding boxes and those failing to satisfy minimum area and width criteria are discarded. In Figure 3, the cropped text region identified from Figure 2 is presented.

Fig. 3. Cropped text image

3.2. Segmentation

The purpose of the segmentation stage is to separate the character pixels from the background pixels and to form an image that contains only black character pixels on a white background, which is a suitable input for the recognition stage. Segmentation by local thresholding is performed based on the assumptions that (i) characters have high contrast to the background and (ii) characters are monochromatic regions. Some segmentation results are shown in Figure 4.

3.2.1. Pre-processing of cropped image

In order to meet the high-resolution requirement imposed by OCR, the cropped image is bi-linearly interpolated by a factor of 4. This ratio is chosen so as to ensure that the smallest size font that occurs in video, such as movie subtitles, is enlarged sufficiently to constitute a suitable input to the OCR stage. This decision is based on a comparison of the movie subtitles font size in a test video and the suggested character size supplied with the OCR package we use. A last pre-binarisation step involves filtering using a 3x3 median filter in order to remove noise spikes.

3.2.2. Binarisation of cropped image

Separation of character pixels from non-character or background pixels is based on local thresholding using a modified Niblack's algorithm as presented in [1][2]. The binarisation decision is made using a rectangular 5x5 window that is shifted across the image using the mean and standard deviation of grey levels in a window. Only those pixels that exhibit a high local contrast relative to the maximum contrast and the contrast of the window are retained. The following equation is used for the calculation of the threshold value [1][2]:

$$T = (1 - a)m + aM + a\frac{s}{R}(m - M)$$

where m is the mean grey level value in a window, s is the standard deviation of grey values in the window, M is the minimum grey level value for the whole image, R is maximum standard deviation for all windows. It is suggested in [1][2] that the parameter a be set to 0.5. However, as character strokes of segmented text appeared to be too thin and fragmented with parameter a set to 0.5, different values were investigated and 0.1 was determined to be the most suitable for our purposes. Following the smoothing step after the first segmentation, the second segmentation is performed as we noticed that it improves the quality of the segmented characters.

3.3. Recognition

Recognition is performed using freely available optical character recognition software known as Clara OCR [7]. This OCR package does not have in-built fonts and thus requires training. A considerable amount of effort has to be put into training so as to build a sufficiently large database of character patterns in order to enable Clara OCR to deal successfully with a variety of sizes, fonts and the varying degree of character fragmentation that occurs in the segmentation process. It is clear from our experiments that overall recognition results critically depend on the quality of the input provided by the segmentation stage. Any fragmentation or damage to the characters due to the presence of noise in video is likely to considerably disrupt the character recognition process.

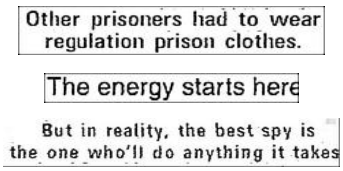

Fig. 4. Segmentation results

4. EXPERIMENTAL RESULTS

4.1. Test Corpus

In order to evaluate the performance of the system, testing was carried out on two MPEG-1 encoded CIF video sequences from our own video database. The first video sequence contained 1000 frames with 26 frames containing the appearance of artificial text. The second sequence contained 30000 frames, 795 of which contained artificial text. The ground truth used for evaluation was created by manually transcribing the sequences. Figure 5 shows examples of images used.

Fig. 5. Examples of images from our database

4.2. Results of text detection

Text detection performance was evaluated manually against the ground truth by determining the percentage of characters in a frame that have been successfully located and enclosed by a bounding box. Detection recall was defined as the ratio between the number of characters enclosed and the total number of characters that appear in a frame. For each new appearance of text on screen, the best detection result was manually chosen. Automating this process using temporal information will be the basis for our future work in this area. Analysis of the accuracy of detection within a frame over the entire test corpus showed the following. The best-candidate detection recall for the first sequence was 95%, and 83 % for the second sequence, giving an average overall detection recall of 83.2%.

	# frames	Recall (best candidate)
Seq 1	1000	95%
Seq 2	30000	83%
Overall	31000	83.2%

4.3. Results of recognition

Since the main focus of our work is on segmentation and not OCR, and given the significant effort required to train the OCR package using segmentation results, we have only evaluated recognition performance for sequence 1. Recognition performance was evaluated through comparison of the OCR recognition results with the manually generated ground-truth. Character-based recognition recall ranged from 81-84% while the recognition precision was within the range 66-74%.

5. CONCLUSION

In this paper we present a method to detect, localise and segment artificial text from video. The evaluation of the method showed moderately good detection recall. However, currently the evaluation is based upon manual selection of the best detection results for the appearance of a given piece of text. Further research is required in order to utilise temporal information in order to automate this process, e.g. by accumulating segmentation results over a number of frames. Further work on training the OCR package for recognition using segmentation results across different sequences is also required.

6. ACKNOWLEDGEMENTS

This material is based on works supported by Science Foundation Ireland under Grant No.03/IN.3/I361 and Enterprise Ireland under Grant No. CFTD/03/216. The support of IST-2000-32795 SCHEMA, RINCE and HEA National Development Plan is gratefully acknowledged.

7. REFERENCES

[1] C. Wolf , J.M. Jolion and F. Chassaing, "Text Localization, Enhancement and Binarization in Multimedia Documents", *Proceedings of the Int. Conference on Pattern Recognition (ICPR) 2002,* vol.4, IEEE Computer Society,Quebec City, Canada, pp.1037-1040, August 2002.

[2] C. Wolf and J.M. Jolion, "Extraction and Recognition of Artificial Text in Multimedia Documents", Technical Report RVF-RR-2002.01, Available: http://rvf.insa-lyon.fr/~wolf/papers/tr-rfv-2002, February 2002

[3] R. Lienhart and W. Effelsberg, "Automatic text segmentation and text recognition for video indexing", *ACM/ Springer Multimedia Systems*, vol.8, pp.69-81, January 2000.

[4] A. Wernicke and R. Lienhart, "On the Segmentation of Text in Videos", *IEEE Int. Conference on Multimedia and Expo (ICME2000),* Vol.3, pp. 1511-1514, July 2000.

[5] A. Miene, Th. Hermes and G. Ioannidis, "Extracting Textual Inserts from Digital Videos", *Proc. of the 6th Int. Conference on Document Analysis and Recognition (IDCAR'01)*, Seattle, USA, pp.1079-1083, September 2001.

[6] H. Bässmann and Ph.W. Besslich, *Ad Oculus, Digital Image Processing, Student Version 2.0*, ITP, London, 1995.

[7] Clara OCR Advanced User's Manual and Tutorial, Available at: http://www.claraocr.org

[8] R. Lienhart, "Automatic Text Recognition for Video Indexing", *Proc. ACM Multimedia 96*, Boston, USA, pp.11-20, November 1996

[9] Victor Wu, R. Manmatha, Edward M. Riseman, "Finding Text In Images", *Proc. Of the 2nd ACM Int. Conference. on Digital Libraries*, 1997.