From Text Detection in Videos to Person Identification

Johann Poignant, Laurent Besacier, Georges Quénot, Franck Thollard

FROM TEXT DETECTION IN VIDEOS TO PERSON IDENTIFICATION

Johann Poignant, Laurent Besacier, Georges Quénot, Franck Thollard

UJF-Grenoble 1 / UPMF-Grenoble 2 / Grenoble INP / CNRS, LIG UMR 5217, Grenoble, F-38041, France

first.lastname@imag.fr

ABSTRACT

We present in this article a video OCR system that detects and recognizes overlaid texts in video as well as its application to person identification in video documents. We proceed in several steps. First, text detection and temporal tracking are performed. After adaptation of images to a standard OCR system, a final post-processing combines multiple transcriptions of the same text box. The semi-supervised adaptation of this system to a particular video type (video broadcast from a French TV) is proposed and evaluated.

The system is efficient as it runs 3 times faster than real time (including the OCR step) on a desktop Linux box. Both text detection and recognition are evaluated individually and through a person recognition task where it is shown that the combination of OCR and audio (speaker) information can greatly improve the performances of a state of the art audio based person identification system.

***Index Terms*—** Video OCR, text detection, text recognition, semi-supervised parametrization, person identification.

1. INTRODUCTION

Automatic video indexing is a domain in expansion due to the increasing number of television channels, together with the growing number of audio-visual resources available on the web. Identification of a person that appears in videos raises a lot of interest in the information research community. It was initially a mono-modal challenge, with either face recognition or speaker recognition, but became a multi-modal challenge, with the fusion of the aforementioned informations [1, 2]. Other sources of information available in the video can also be used to identify people in video (overlaid texts [3, 4], spoken words [5], anchor information [4], ...).

The overlaid texts contained in video represent rich and reliable information, when the text can be accurately transcribed. Although Optical Character Recognition (OCR) is now accurate, it requires a good detection of the text zones in the image (from now called text box) together with a well binarized image. The text detection step must find a maximum amount of text (we look for a high recall), but also find the exact coordinates of the boxes that contains text. Without an accurate surrounding box, the quality of text recognition is degraded, leading to poor performances. Thus, our main focus here is the text detection as we rely on a third party program for OCR. Our text detection is evaluated through a person recognition task. The currently available person recognition systems are strongly linked to the corpus they were developed on, as they mainly use supervised methods. This induces an important human cost for annotating these corpora. Although some of the modalities can be used without re-training (*e.g* face detection, speaker segmentation), some other modalities can be greatly affected by a corpus change (*e.g.* recognizing the voice can be affected by the quality of the recording, and the surrounding noise, hence needing to build a corpus-based model [6], video quality can affect the face recognition). As far as text detection is concerned, changing the corpus can induce many changes: first, depending on the corpus, text can be found in all frames, on some frame, or never; secondly the localization of text boxes is also corpus dependent (*e.g.* logo, scrolling text, sport results, names, ...); third, the text can be written in different forms (font, size, font color and background color, orientation, alphabet,...). One of the contribution of this work is a semi-supervised tuning of the system that allows an efficient adaptation to new corpora.

The first section presents related works. The second section gives a global overview of the system. Then we detail our contribution[1]. Next section shows the results in term of human cost with a semi-supervised approach for text detection. The penultimate part presents the evaluation of the full recognition system and finally the last part describes the contribution of overlaid text to person detection in video.

2. RELATED WORKS

Broadly speaking, all the methods share three steps: text detection, text recognition and post-processing. Two kinds of texts have to be considered: scene text (*e.g.* a text written on a road sign), and overlaid text. Lienhart *et al.* [7] clearly set what an overlaid text is. Scene text processing is out of the scope of this paper. We refer the interested reader to [8] as a reference to a recent work in this field.

This work was partly realized as part of the Quaero Program and the QCompere project, respectively funded by OSEO (French State agency for innovation) and ANR (French national research agency).

[1]The binary of our system is available at *mrim.imag.fr/johann.poignant*.

2.1. Overlaid text detection

As far as text boxes detection is concerned, techniques in [7, 9] are all based on texture, color, contrast and geometry of the text. For texts that appear in any orientation, corner detection can be used as in [10] for example.
Cai *et al.* [11] present one of the first two steps technique for spatial detection: i) horizontal dilatation on the whole image followed by local vertical dilatation; ii) local application of horizontal and vertical dilatations refines the coordinates of the found boxes. Many works in the literature *e.g.* [12, 13] share the following strategy: first, find a maximum of candidate boxes, and second, remove non relevant texts. Similarly [14] applies both a vertical and horizontal Sobel filter in order to detect the characters edges. A dilatation with few iterations allows to connect characters together. An opening operation is then applied in order to isolate connected components; then, they detect text lines with a method based on horizontal and vertical projections. This generates a lot of false positives boxes. A second local detection with machine learning refinement is used to filter these false alarms.
All these techniques are purely image based. A temporal filtering can be applied in a video context to filter false alarms [9] and recover locally missed boxes.

2.2. Text recognition

Concerning text recognition, two strategies are possible: either set up a dedicated OCR (this case is out of the scope of this paper and we refer the interest reader to [15]) or use any classical OCR. Specific treatments are performed before sending the image to the OCR: increasing the image quality by averaging the images extracted from consecutive video frames [9, 16], increasing the image resolution using a linear interpolation [9]. One of the difficulties, regarding the overlaid text transcription in videos, is to provide very good binarized image. Several techniques exist in the literature, we can mention [17] and [9] among other studies.

3. GLOBAL OVERVIEW OF THE SYSTEM

We are interested in compressed videos with low resolution (MPEG1, 288x352), and consider horizontal, non scrolling text, written in Latin alphabet. Following [14], we perform a two steps detection (see fig 1), in which the coarse detection is closely related to the one of [9], namely obtained through a Sobel filter followed by dilatation/erosion. This limits the number of false alarms after the coarse detection, which improves the processing time of the refine detection (section 4). Although our temporal processing includes the one of [9], it also includes a recovery step that allows to correct the start/end of text boxes. A semi-supervised parametrization is used to adapt the system to a particular corpus (section 5).
To overcome the problem of transcription errors due to binarization, we extract several binarized images of the same text

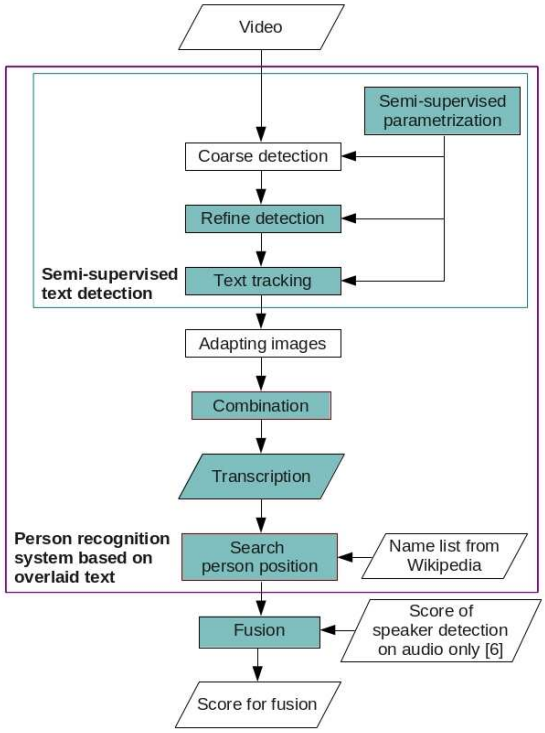


Fig. 1. System overview.

temporally shifted and we combine the transcriptions in order to improve the quality of the final transcription. To the best of our knowledge, this is the first time this kind of combining is used for OCR in video (section 4.3). Regarding the text recognition system, we rely on a third party OCR, namely the publicly available OCR from Google: `Tesseract`: http://code.google.com/p/tesseract-ocr/.
Last but not least, a fusion between our system and an audio based system [6] is used to improve person recognition (section 7).

4. TEXT DETECTION AND RECOGNITION

We detail here the three steps of our text video OCR system: i) text detection, ii) text tracking and iii) image adaptation and combination.

4.1. Text detection

As mentioned earlier, the spatial detection holds two parts. First, a coarse detection selects connected components with a high recall, the second filters boxes and refines coordinates (see fig 2.a, original image of our running example).

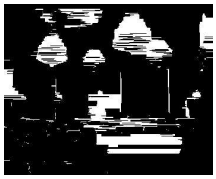
4.1.1. Coarse detection

The text detection is performed on all the frames of a video. It begins with a horizontal Sobel filter that highlights one of the

main features of the Latin alphabet: a texture of vertical bars connected by horizontal bars. We can see the result in fig 2.b. After binarization, an operation of horizontal dilation and erosion connects characters of a same string (fig 2.c). A filtering step cleans the image with vertical and horizontal erosion and dilation (fig 2.d). In the resulting image, we select the areas that satisfy geometric constrains. We thus obtain a coarse detection of the texts boxes coordinates (fig 2.e).

(a) Original image of our running example.

(b) Sobel filter.



(c) Dilatation and erosion step.

(d) Filtering step.

(e) Coarse detection.

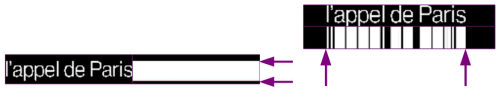
(f) Refined detection.

(g) Final result of the spatio-temporal detection.

Fig. 2. Different steps of spatial-temporal detection.

4.1.2. Refine detection

A second detection is performed on each individual text box. After binarization using the Sauvola algorithm [17], the number and the area variance of connected components allows us to detect if the text is written in black on white or vice versa in the binarized image. This fine detection follows two steps



(a) find Y coordinates (b) find X coordinates

Fig. 3. Local refined detection.

in order to find the horizontal and vertical coordinates. An operation of horizontal (resp. vertical) morphological dilatation finds the vertical (resp. horizontal) coordinates of texts (fig 3.b). The *white* blocks (arrow in fig 3 a and b) allow to estimate both horizontal and vertical coordinates.

This fine detection allows us to filter boxes that do not respect the text geometrical constraints. In our example (fig 2.f), the coarse detection found eleven boxes, whereas five of them were removed by the fine detection. Although the background is rather complex, only four false positive remains, and the true positives have not been filtered out. Fortunately, this kind of false positive can be filtered out as they do not satisfy a required temporal consistency.

Note that, at this step of the processing, other algorithms like the one in [14] would have raised many candidate boxes. This is not an issue in term of the final quality (as these boxes will probably be filtered out with machine learning refinement), but this can be an issue from a computational point of view.

4.2. Text tracking on successive frames

The next step takes advantage of the fact that a given text appears identically on many successive frames. The temporal information is used to filter out false positive boxes but also to recover boxes for which the detection locally failed. We can see in our example (fig 2.g) that four boxes do not have temporal stability.

4.3. Adapting images boxes for the OCR software and combination of multiple transcriptions

After the text detection step, we need to adapt text box images to the OCR software. We artificially increase the resolution of these images with a bicubic interpolation. Next we apply a binarization on images using a threshold calculated with the Sauvola algorithm [17]. To enhance the quality of the text transcription, we apply the OCR on several images, for a same box, temporally shifted (fig 4).

Let us note $\bar{I}_{i,j}$ the image built by averaging the images in the range $[i, j]$. Preliminary experiments have shown that 10 frames is a good value. Two kinds of average images are computed: the global average image $\bar{I}_{1,M}$ and a set of local average images computed on a sub-range of size n: $\bar{I}_{k,l}$, where $l - k = n - 1$. The transcripts are computed for both the global average image and the local ones.

In fig 4 the transcriptions for our running example are presented: the first transcription corresponds to the global average image ($\bar{I}_{1,M}$), the others correspond to the transcriptions temporally shifted ($\bar{I}_{1,n}, \bar{I}_{n+1,2n}, ...$).

The transcription obtained from the global average image (namely *l'aopel de Paris*), leads to one character error. The frame presented in our running example is included in the third range and the transcript of the image $\bar{I}_{2n+1,3n}$ is *l'ap"pel de Paris*. In order to combine all these hypotheses, we build a mesh graph, with, as a backbone, the transcript

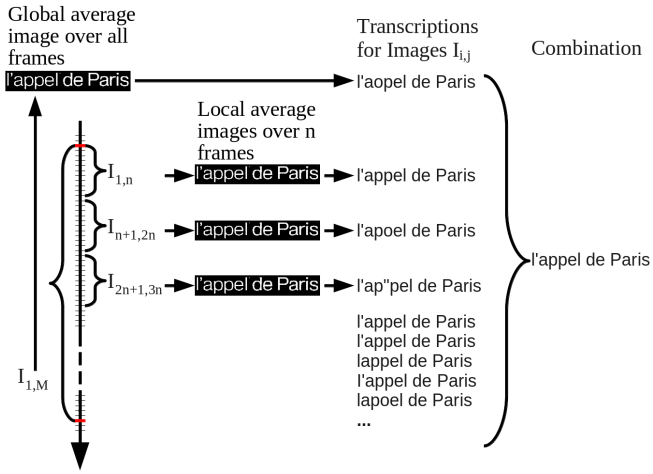


Fig. 4. Multiple transcriptions for one text box.

from the global average image. The mesh graph is a confusion network obtained using dynamic alignment. The method comes from automatic speech recognition and is implemented in the SRILM toolkit (www.speech.sri.com/projects/srilm/); the fig 5 shows the beginning of the mesh graph on our example. The Viterbi algorithm is used to select the most probable path. This method corrects the mistake on the characters mentioned above.

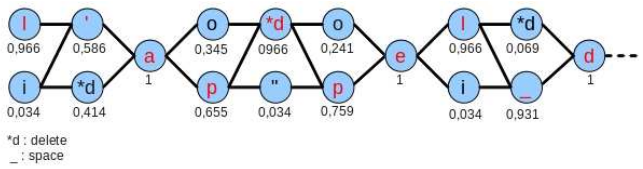


Fig. 5. Partial mesh graph.

In order to make it flexible w.r.t. corpus changes, all the possible parameters are available for tuning. Next section addresses the problem of adapting the system to a new corpus while minimizing the human cost.

5. SEMI-SUPERVISED PARAMETERS SETTING

From a research viewpoint broadcast news is an interesting type of video for person recognition. In such videos there is a large number of different persons with few appearance (interviewed person, reporter) and a very small number of anchors. Audio sources are variable (voice recorded in studio or not, spontaneous *vs* prepared speech, different background noises, ...); video sources are also variable (external *vs* internal, TV studio, shooting with different quality). There are many texts on the screen but with very different characteristics depending on the channel.
For our system, the default settings can be used, but adapting to a particular corpus can improve the final quality. So we estimated the amount of annotation required (*i.e.* the human cost) to parametrize our text detection system.

Some characteristics can be set *a priori*: orientation, alphabet, scrolling,.... But the system needs to be tuned in order to adapt the other features. The goal here is to limit the amount of manually annotated data. The parametrization can be divided into several steps. For each step the parameters vary in a predefined range. The parameters defined for a given step are fixed while estimating the parameters values of the next steps.
To assess the amount of annotations useful for the parametrization, we have annotated 2 hours of broadcast news (three JT of France 2, French TV). 512 text boxes have been temporally and spatially annotated, the annotation took 4 hours. We wanted to know how the quality of the tuning is sensible with respect to the number of annotated boxes. We consequently tuned the system on one video (corresponding to 212 boxes). We randomly selected 10 sets with respectively 1, 25, 50, 75, 100, 150, 200 annotated boxes. The performance of the system is then evaluated on the two other videos.

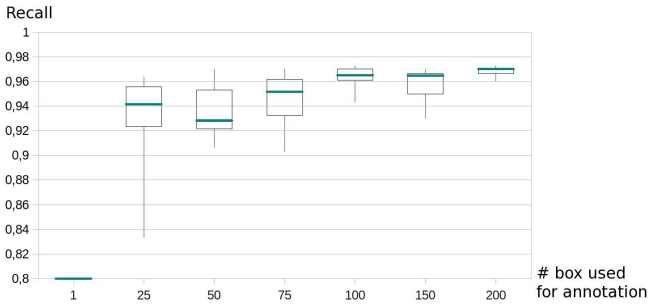


Fig. 6. Evolution of the recall with the number of boxes annotated. The boxplots are computed on ten different sets.

Fig 6 shows a boxplot graph with the recall for text boxes correctly detected[2] as a function of the number of annotated boxes used for parametrization.
The annotation of a single box gives a very low detection rate (percentiles: min, 1/4, 1/2, 3/4, max: 0.144 0.533 0.572 0.622 0.763), but we can see that when 25 boxes are used, the detection rate reaches 94% in median. The detection rate stabilized between 100 and 200 annotations over 96%. We can consider that 100 to 150 annotations is enough to tune the system for this corpus, thus reducing the annotation requirement to roughly one hour which is not much in the field of person recognition. Note that the average of the pixel-wise F-measure of the detected boxes is 0.89.

[2]A box is considered as correctly recognized if the F-measure computed on pixels between the reference box and the hypothesis box is larger than 0.5.

6. EVALUATION OF THE FULL TEXT RECOGNITION SYSTEM

Once the system has been parametrized for detecting the text boxes, the corresponding part of the image can be sent to an OCR system. We now evaluate the whole system on the basis of its performance at the character recognition level. Consistently with the previous section, we used a parametrization for text detection obtained using 200 annotated boxes.

The Corpus used for evaluation consists of 59 videos of France 2 TV News from February 1 to March 31 2007. The average length of these videos is about 36 minutes, which represents just over 35 hours of video. 29,166 key frames were automatically extracted [18]. The texts extracted from these key frames have been manually annotated.
We conducted our evaluation on 29k frames, with focus on person name and person function written on the screen (excluding journalists credited at the end of the story). From the 29k images assessed, 4,414 frames contain text that appears in 9,256 text boxes. As the OCR has not been trained on these data, we apply a simple ad-hoc post-processing to correct some recurrent errors (*e.g.*, systematically changing "ii" into "m" in the `Tesseract` output, ...).

The Evaluation measure used as a metric for assessing the recognition of texts is the Levenshtein distance, computed either on words or characters. The reference does not contain spatial position for text boxes, so we have aligned reference and hypothesis if the similarity (based on Levenshtein distance) is higher than 50%.

Results are presented in Table 1. We use as baseline text recognized while using images binarized by the Sauvola algorithm [17]. This algorithm is still part of the state of the art as it would have been ranked 6^{th} at the DIBCO 2009 challenge [19]. For sake of completeness we provide the word error rate even though the words were not yet post-processed. The error rates mentioned are thus over-estimated. Consequently, we focus our analysis on the character error rate. We also provide in parenthesis the error rate on the detected text box only. This error gives the true performance of the OCR only independently of text detection.

Using the combination allows a significant improvement of performances as the character error rate (CER) drops from 8.6% to 4.6% on full text. As can be seen, the names of people tend to be more readable by our system (Full Text, *vs* Names). Most of the errors on Names are due either to boxes that are not correctly detected or to references and hypothesis that could not be aligned (2.6% to 0.9%): It is worth noting that in our corpus some names are written in 5 pixels high (*e.g* player names of a team), whereas person function is written in 7 pixels high with a uniform background. In the latter case, the combination decreases the character error rate from 2.8% to 1.7%. Therefore, even on mid-sized text on a uniform background, the combination improves the transcription.

	Number of		Sauvola [17]	
Type	words	characters	err rate words (%)	err rate character (%)
Full Text	30905	154904	19.2 (16.5)	8.6 (6.2)
Names*	3230	19248	9.6 (6.4)	3.4 (1.7)
Functions	5794	32472	9.3 (9.1)	2.8 (2.6)
			Sauvola [17]+combination	
Full Text	30905	154904	**11.6** (9.3)	**4.6** (2.7)
Names*	3230	19248	**7.1** (4.0)	**2.6** (0.9)
Functions	5794	32472	**6.3** (6.0)	**1.7** (1.5)

* Name that appears alone and without credit.

Table 1. Character and word error rate with and without combination described. Performance in parenthesis give score for the OCR only on detected text box.

The response time (on a Intel Xeon Core 2 Duo, 3 GHz, 4 Gb of RAM) of the system is 728s for a 2184s video (MPEG1, 352x288, 25 frames/sec): 441s for the detection step and 287s for the transcription step. The efficiency however depends upon the number/duration of the text boxes found.

7. PERSON DETECTION IN VIDEOS

We are interested in measuring to which extend overlaid text can improve person recognition system in video. The idea is to recognize people that speaks in a shot. We postulate that when a name is written alone in a box text, the person is currently speaking. Person names can be divided in two categories: "famous names" *vs* "unknown names". We consider a text as famous name if it belongs to a list of names extracted from Wikipedia. The spatial position of "famous names" gives insight of where names tends to be written in the corpus. Among the 1908 text boxes we found at the "famous name" positions, 98.75% of the time the person corresponding to the text is present on the shot and 96.8% he/she is currently speaking. Note that experiments on other corpora (3 channels, 8 shows) exhibits the same behavior.
As an audio-only baseline, we use the system described in [6] for 20 targets (politicians and athletes). This system is based on person recognition from audio only. We compared it with our system for the same person recognition task. We also evaluated different fusions (fig 7) of both modalities: OCR extended to speaker segment, OCR extended to speaker segment and AUDIO if we don't have any information from OCR for the segment. The sampling rate used here is 1 second.

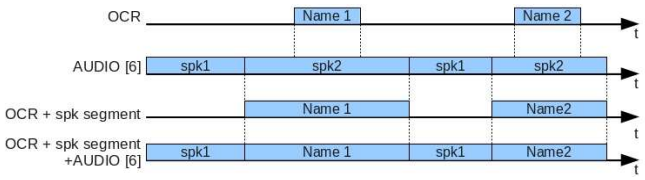


Fig. 7. Fusion.

We can see in table 2 that the OCR alone has a very high precision but a low recall. In order to increase the F-measure we extend it to speaker segment (OCR + spk segment in the table). This keeps a recall higher than the AUDIO [6]. As a final system (OCR+spk segment+AUDIO) we apply the following rule: if the OCR has information on the person, we keep the output of OCR otherwise we use the output of the AUDIO system. This simple fusion scheme improves the F-measure from 66.9% to 77.3%.

Type	Precision	Recall	F-Measure
OCR	**0.968**	0.105	0.19
AUDIO [6]	0.688	0.652	0.669
OCR+spk segment	0.894	0.343	0.496
OCR+spk segment+AUDIO [6]	0.736	**0.814**	**0.773**

Table 2. Enhanced speaker recognition system.

8. CONCLUSION

We present in this article an OCR system that detects and recognizes overlaid text in video. We proceed in two steps for text detection. The first one, recall oriented, makes a coarse detection. Temporal and spacial processing are then used to filter out false positives and to recover some missed text boxes. We pay attention to make the system efficient, both in term of human annotation requirements and time response: the system only requires 100 annotated boxes (roughly 1 hour of annotation) in order to be correctly tuned and found 96% of text boxes. It runs 3 times faster than real time (including the OCR step) on a current desktop Linux box.
The system has been evaluated on 37 hours of broadcast news, the post processing step combines multiple transcriptions of the same text box and significantly decreases character error rate (8.6% to 4.6%) as compared to transcriptions based on images binarized using the Sauvola algorithm [17]. We also show that overlaid text in video improves audio person detection (66.9% to 77.3%) using basic fusion schemes.

Further work for person recognition will include fusion with other modalities (face recognition, external resource, *etc.*), and use of other information in overlaid text (place, date, *etc.*). We also plan to work on binarization as OCR systems are quite sensible to the quality of the binarization.

9. REFERENCES

[1] T. Choudhury, B. Clarkson, T. Jebara, and A. Pentl, "Multimodal person recognition using unconstrained audio and video," in *AVBPA*, 1999, pp. 291–300.

[2] A. Albiol, L. Torrest, and E.J. Delpt, "The indexing of persons in news sequences using audio-visual data," in *IEEE ICASSP*, 2003, vol. 3, pp. 6–10.

[3] R. Houghton, "Named faces: Putting names to faces," *IEEE Intelligent Systems*, vol. 14, pp. 45–50, 1999.

[4] M.-Y. Chen and A. Hauptmann, "Searching for a specific person in broadcast news video," in *ICASSP*, 2004, pp. 1036–1039.

[5] S. Satoh and T. Kanade, "Name-it: Association of face and name in video," in *CVPR*, 1996.

[6] V.B. Le, C. Barras, and M. Ferras, "On the use of GSV-SVM for speaker diarization and tracking," in *Odyssey*, 2010, pp. 146–15.

[7] R. Lienhart and W. Effelsberg, "Automatic text segmentation and text recognition for video indexing," *ACM/Springer Multimedia Systems*, pp. 69–81, 1998.

[8] R. Minetto, N. Thome, M. Cord, J. Fabrizio, and B. Marcotegui, "Snoopertext: A multiresolution system for text detection in complex visual scenes," in *IEEE ICIP*, 2010, pp. 3861–3864.

[9] C. Wolf, J.-M. Jolion, and F. Chassaing, "Text localization, enhancement and binarization in multimedia documents," in *ICPR*, 2002, pp. 1037–1040.

[10] X.-S. Hua, X. r. Chen, L. Wenyin, and H.-J. Zhang, "Automatic location of text in video frames," in *ACM Multimedia Workshops - MIR*, 2001, pp. 24–27.

[11] M. Cai, J. Song, and M.R. Lyu, "A new approach for video text detection," in *Image Proc.*, 2002, pp. 117–120.

[12] Q. Ye and Q. Huang, "A new text detection algorithm in images/video frames," in *Advances in Multimedia Information Proc. - PCM*, 2005, pp. 858–865.

[13] C. Jung, Q. Liu, and J. Kim, "A stroke filter and its application to text localization," *Pattern Recogn. Lett.*, vol. 30, pp. 114–122, 2009.

[14] M. Anthimopoulos, B. Gatos, and I. Pratikakis, "A two-stage scheme for text detection in video images," *Image Vision Comput.*, vol. 28, pp. 1413–1426, 2010.

[15] F. Einsele, R. Ingold, and J. Hennebert, "A HMM-based approach to recognize ultra low resolution anti-aliased words," in *PReMI'07*, 2007, pp. 511–518.

[16] J.Xi, X.-S. Hua, X.-R. Chen, L.Wenyin, and H.-J. Zhang, "A video text detection and recognition system," *ICME*, p. 222, 2001.

[17] J. Sauvola and M. Pietikinen, "Adaptive document image binarization," *Pattern Recognition*, pp. 225–236, 2000.

[18] G. Quénot, D. Moraru, and L. Besacier, "CLIPS at TRECvid: Shot boundary detection and feature detection," in *Proceedings of TRECVID*, 2003, pp. 35–40.

[19] B. Gatos, K. Ntirogiannis, and I. Pratikakis, "DIBCO 2009: document image binarization contest," *IJDAR*, pp. 35–44, 2011.