HAL
open science

New Gamma Correction Method for real time image text extraction

Mohamed Amin Ben Atitallah, Rostom Kachouri, Ahmed Ben Atitallah, Hassene Mnif

To cite this version:

Mohamed Amin Ben Atitallah, Rostom Kachouri, Ahmed Ben Atitallah, Hassene Mnif. New Gamma Correction Method for real time image text extraction. 9th International Conference on Image Processing Theory, Tools and Applications (IPTA 2019), Nov 2019, Istanbul, Turkey. pp.1-6, 10.1109/IPTA.2019.8936103 . hal-02293895

HAL Id: hal-02293895

https://hal.science/hal-02293895

Submitted on 22 Sep 2019

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

New Gamma Correction Method for real time image text extraction

Mohamed Amin Ben Atitallah
Gaspard Monge Computer Science Laboratory, ESIEE Paris
National Engineering School of Gabes
University of Gabes
LETI (E.N.I.S.), University of Sfax
Sfax, Tunisia
benatitallahmohamedamin@yahoo.fr

Rostom Kachouri
Gaspard Monge Computer Science Laboratory, ESIEE Paris
University Paris-Est Marne-la-Vallée
Paris, France
rostom.kachouri@esiee.fr

Ahmed Ben Atitallah
Department of Electrical Engineering, Jouf University,
2014 Aljouf, Saudi Arabia
LETI (E.N.I.S.), University of Sfax
Sfax, Tunisia
abenatitallah@ju.edu.sa

Hassene Mnif
LETI (E.N.I.S.), University of Sfax
ENET'com Sfax
Sfax, Tunisia
hassene.mnif@enetcom.usf.tn

***Abstract*—Text extraction and character recognition in manuscripts are a very important part of image processing and pattern recognition. The recently proposed Gamma Correction Method (GCM) is a popular method used in the processing of a color image and in particular in the extraction of text from a complex image. However, The GCM consumes a lot of time to extract the text and this is because of the complex calculations of different operations in the different blocks of GCM which is not suitable to real-time applications. In this paper, we propose an efficient Gamma Correction Method acceleration. Our proposal allows to minimize the GCM execution time while ensuring the same reliability of the initial version. The experimental results show an important optimization compared to the literature.**

***Index Terms*—GCM, Software acceleration, OCR, Text extraction, Optimization, Real time.**

I. Introduction

Text extraction is divided into several sub-problems: Detection, localization, location, extraction, improvement and recognition. Most of the time the first three steps can be confused, but in reality they are perfectly distinct. The detection determines the presence of texts or not in a given image. Localization is a process that determines the location of text in an image and generates frames around the text. The registration is performed to reduce the processing time of the location and keeps in memory the original location of the text in the image. The extraction transforms the text into a binary image, then extracts it from the background of the original medium and improves it to give a better resolution of the text. Recognition, using the optical character recognition (OCR) [1] technique, transforms a printed (analog) text into a digital file composed of ASCII characters instead of pixels (analog file)[2].

978-1-7281-3975-3/19/$31.00 ©2019 IEEE

The performance measurement depends on the recognition capacity of the different existing algorithms. In our case, we will extract the text and see the result directly on the images. Text embedded in the image is more difficult to extract. The difficulty of localization depends on several factors such as: Length, height, complexity of the background: texture and shape of the background, contrast and multiplication of long texts. Indeed, these different (rather random) variations make the problem of extracting text extremely difficult especially in complex images [3].

Existing methods have been proposed to resolve this problems. Zhong et al. [4] propose an algorithm based on the RGB histogram which will be evaluated in a limited number of classes and then their distinguished peaks are extracted repetitively. All extracted components are then classified into "text" or "non-text". Zhou et al. [5, 6] propose a graduated clustering algorithm combining an RGB color distance and the spatial proximity of colors in the image space. The algorithm gives partially interesting results on the basis of images resulting from more or less complex internet streamers. Sin et al. [7] propose a progressive approach. In summary, each region is transformed into a one-dimensional sequence, and the region is validated as a text field if the auto-correlation function of its Fourier spectrum contains a significant number of peaks. This algorithm is used in the case of the detection of long texts on road signs.

Among the most recent text extraction methods for text images, Sumathi and al. (2012) describe in detail the different schemas existing on the extraction of text from an image and in 2014 they proposed a new algorithm for the extraction of text in an image. This proposed method allows to delete the background of the image based on the Gamma Correction

Method (GCM) [8]. However, this method consumes a lot of time to extract the text and this is because of the complex calculations of different operations in the different blocks of GCM and that's not good for the real-time applications. In this paper, we propose an efficient Gamma Correction Method acceleration. Our proposal allows to minimize the GCM execution time while ensuring the same reliability of the initial version. The experimental results show an important optimization compared to the literature.

The next of this paper is organized as follows, section 2 presents the GCM algorithm. In section 3, we propose our new gamma correction method. Experimental result are given in section 4. Finally, conclusion and perspective are drawn in section 5.

II. Extraction text using GCM and related work

The gamma correction method is a popular method used in the processing of a color image and in particular in the extraction of text from a complex image. The importance of this method is to remove the background of the image and locate the text region. The correction is given by $\gamma = x^{\gamma}$ which controls the overall brightness of the image. When the quantity of γ is less than one, the transformed image becomes lighter than the original image; and when the amount of γ is greater than one, the transformed image becomes darker than the original image as shown in Figure 1. The correct estimate of gamma value improves the contrast of the image. This correction must be applied to pixel intensities.

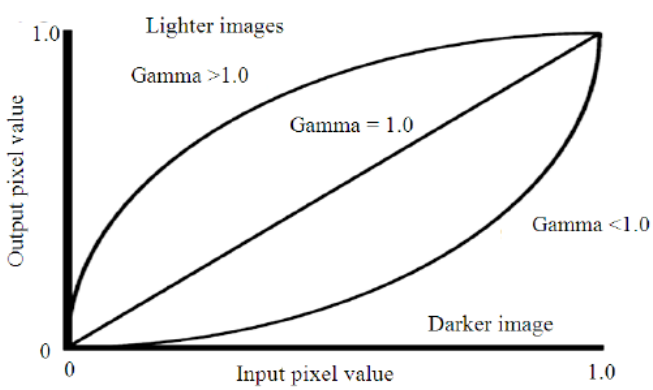

Fig. 1. The effect of gamma correction on image [9]

In fact, for text extraction, the GCM [8] generates 100 modified images for each gamma value (the value of gamma can vary from 0.1 to 10.0 with increments of 0.1) from the original image. Then for each modified image, it generates the image threshold (1) using Otsu's method [10] and the Gray Level Co-occurence Matrix (GLCMs) [11] to calculate two texture features which are contrast (2) and energy (3). Then, it estimates the best gamma value from the contrast, the energy and the threshold of the image to generate the binarized and the corrected image.

(a) (b) (c)

Fig. 2. GCM Outputs: (a) original image: "Contrast = 490.934, Energy = 0.000560853", (b) corrected image: "γ= 5.5", (c) binarized image: "T = 0.246094"

$$\sigma_w^2(T) = w_1(T)\sigma_1^2(T) + w_2(T)\sigma_2^2(T) \tag{1}$$

Where σ represents the inter-class variance and ω represents the probability of being in class one or two.

$$Contrast = \sum_{n=0}^{N_g-1} n^2 \sum_{i=1}^{N_g} \sum_{j=1,|i-j|=n}^{N_g} p(i,j) \tag{2}$$

$$Energy = \sum_{i=1}^{N_g}\sum_{j=1}^{N_g} p(i,j)^2 \tag{3}$$

With Ng is the number of gray level and p(i, j) represent the value of the GLCM, (i, j $\in$ [0, 255]).

Table 1 gives an overview of the set of values generated for the illustrated image in Figure 2 (a). From these values the optimal gamma estimation is made based on rules defined in [8].

Gamma	Contrast	Energy	Threshold (Otsu)
0.1	20.2623	0.00550906	0.902344
0.2	67.5417	0.00202073	0.820313
...			
5.3	479.48	0.0890308	0.246094
5.4	475.225	0.0955083	0.246094
5.5	471.715	0.102772	0.246094
5.6	69.0865	0.0766992	0.117188
...			
9.9	350.64	0.0767035	0.328125
10	348.745	0.485401	0.324219

TABLE I
An overview of the set of values generated for the illustrated image in Figure 2

For example in the case of the Figure 2 (a), the rule 2 [8] is applied since for γ = 1 (original image), the value of the energy $\leq$ 0.05 and the value of contrast $\leq$ 1000. This made it possible to determine the optimal gamma value (γ = 5.5) enabling to delete the background of the image (Figure 2 (b)) and consequently the selection of the threshold value (T = 0.246094). This threshold allows the conversion of the grayscale image to a binary image (Figure 2 (c)).

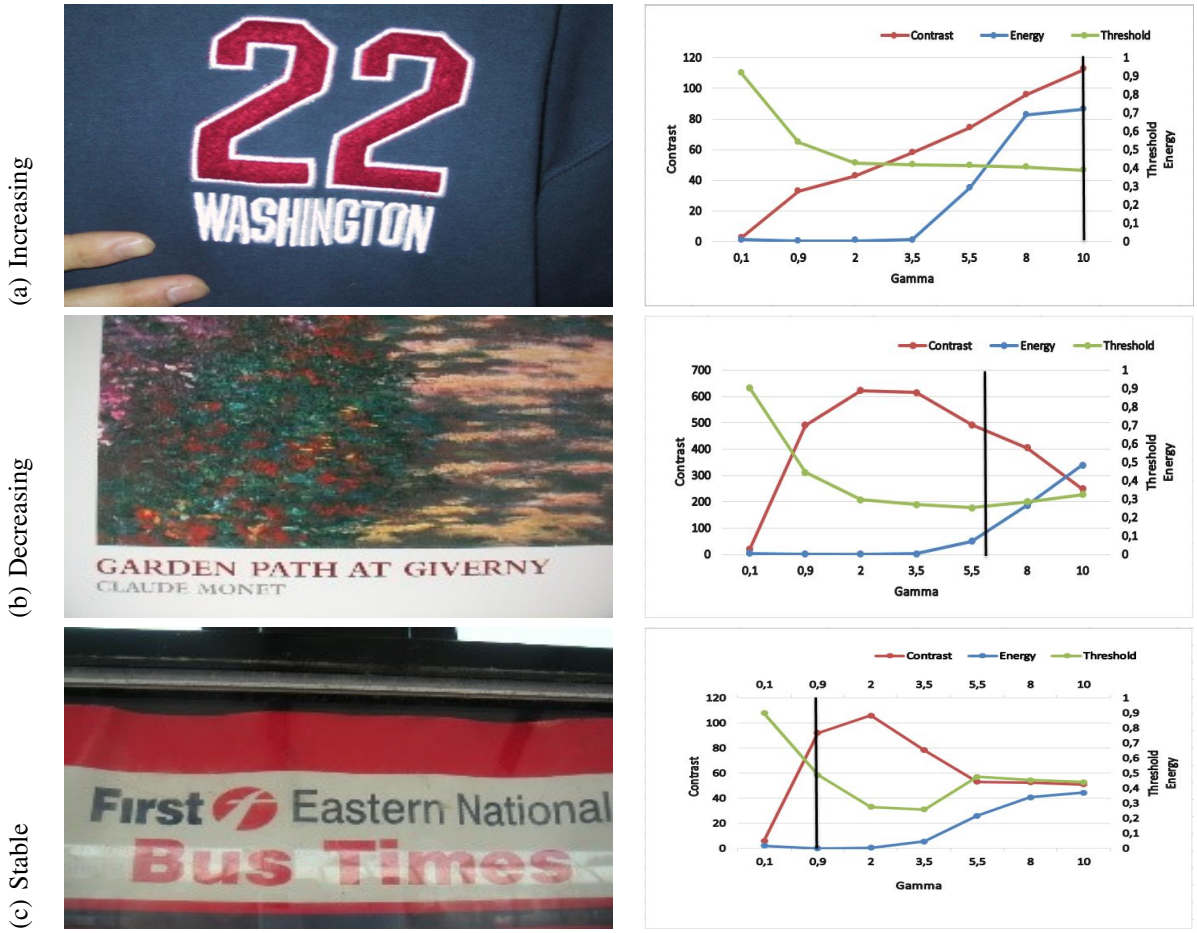

Fig. 3. Three main image behaviors: (a) increasing contrast curve, (b) decreasing contrast curve, (c) stable contrast curve.

The execution time to apply the GCM on the image in Figure 2 (a) is 4.2649 seconds. This value has a negative influence for real-time applications. In this context, some existing methods have been proposed a GCM acceleration to reduce its execution time.

Mahmood Farshbaj Doustar and Hamid Hassanpour [12] proposed a new adaptive method of gamma correction. This method estimates the gamma values in different regions of an image and transforms these individual pixels of the image according to the estimated gamma values.

Rostom and al [13] proposed an acceleration of the Gamma Correction Method (AGCM). In fact, the AGCM method predicts a sub-range of values where the selected gamma is expected to be found by the GCM. This method is more efficient compared to the GCM but the execution time is still higher.

III. Proposed method to accelerate the GCM

In this section we will present our proposed new gamma correction method, that we will call NGCM. In fact, our proposed method centred on minimizing the number of modified images. In fact, our goal is to propose to generate n modified images instead of generating 100 modified images (where n $\lll$ 100).

A. Study of the GCM parameters

In this part we will apply the GCM method on 3 images of different complexity. In fact, this application allowed us to analyze the relationship between the parameters calculated by the GCM (contrast, energy and Threshold) and the Selected Gamma Value (SGV).

As shown in Figure 3, for each image we have drawn the energy, contrast and threshold curves for 7 modified images, for Gamma = $\{0.1, 0.9, 2.5, 4, 5.5, 6.5, 10.0\}$.

The 7 values have been selected in areas containing gamma values that determine the behavior of the parameter curves, so that the entire curve can be reliably categorized.

We note that the best value of gamma depends essentially on contrast values. In the next part we will propose our rules to accelerate the GCM method.

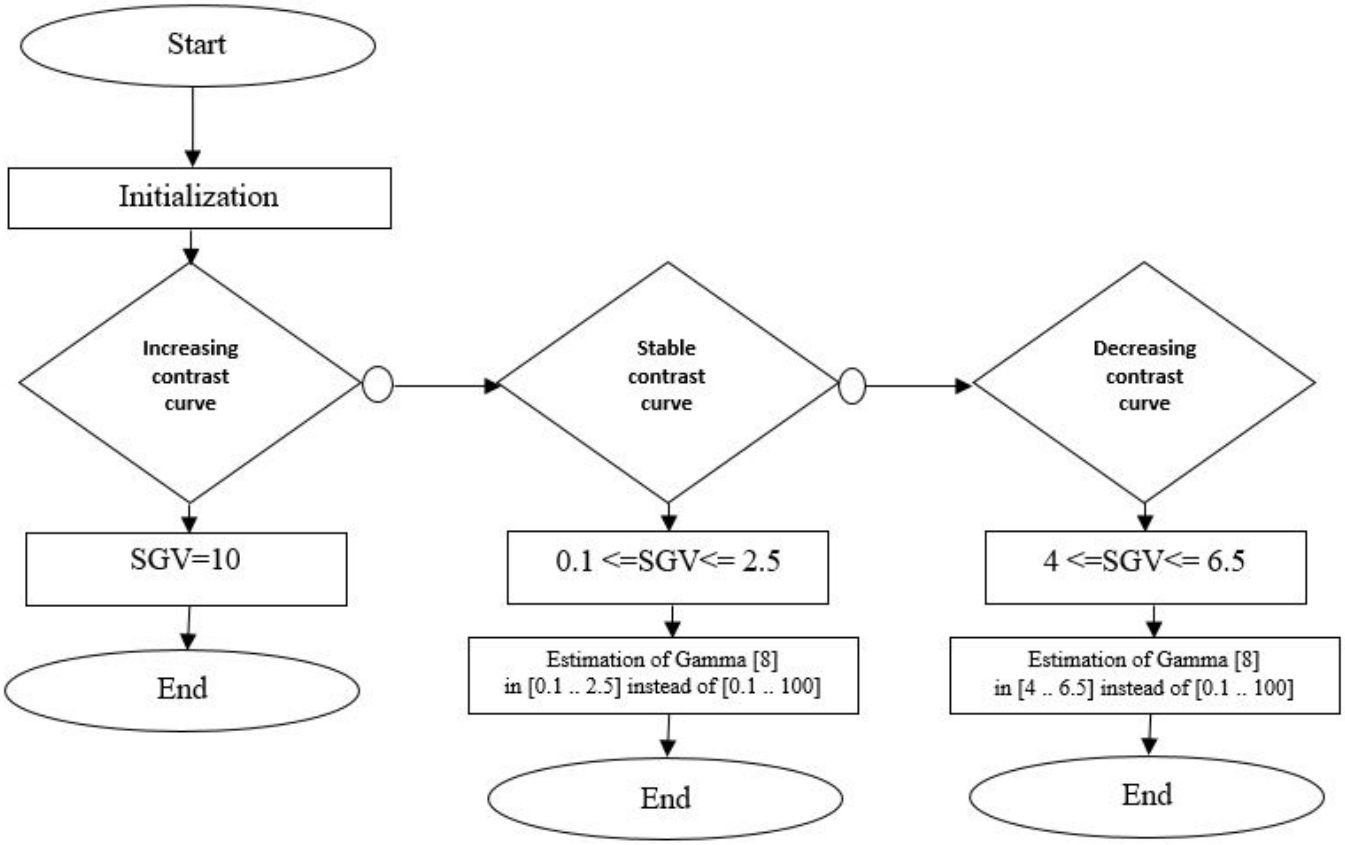

Fig. 4. Our proposed NGCM flow chart.

B. Proposed rules

As previously stated, the contrast curve is the main factor (first-order characteristic) to influence the gamma value area. The other factors (second-order characteristics) that should be evaluated in the second phase are the high energy values and the Otsu threshold. We describe the main behaviors and their relation to the selected gamma value.

Our study leads to mention that the contrast curve has three main behaviors: Always increasing, stable or decreasing. An increasing curve (see Figure 3 (a)) has an optimal gamma value of 10.0 and are generally images without many variations or features. An stable curve (see Figure 3 (b)), the selected gamma value is always located in the range of [0.1 – 2.5]. An decreasing curve (see Figure 3 (c)), the selected gamma value is always located in the range of [4 - 6.5].

C. Proposed algorithm of the gamma estimation

The Figure 4 represents our proposed organizational flow chart. This flow chart makes it possible to determine the sub range of the gamma value.

The organizational flow chart has 3 main branches: increasing contrast curve, decreasing contrast curve and the stable contrast curve. After the examination of the different outputs of GCM, we found that some type of images always take the same value of gamma. Generally, we find this case in images that have a dark background with a little light feature (increasing curve), in this case, the selected gamma value is usually equal to 10.0, in cases where the image has a complex background with a clear text (descending curve) usually have a selected gamma value that belongs to the middle section, in this case, the selected gamma value is generally located in [4 - 6.5], and while the other case (stable curve) the selected gamma value is generally located in [0.1 - 2.5]. In the cases of stable and decreasing curves the gamma value will be estimated using the three rules of Sumathi [5], but only for 25 gamma values instead of 100.

IV. Experimental results

For the purpose of evaluating, we applied the NGCM, AGCM and GCM algorithms on multiple images taken from ICDAR [14] database. We developed the three methods in C++ and we executed them on a PC intel i7 clocked at 3.5 GHz. The partition of the range of gamma in sub range proposed by the NGCM, makes it possible to minimize the execution time with keeping the same quality of AGCM and GCM in most cases (see Figure 5), see even improve it in some cases as Figure 5, image 6 for example.

The Figure 6 presents the superposed execution-times of the GCM, AGCM and NGCM over the ICDAR image set. The

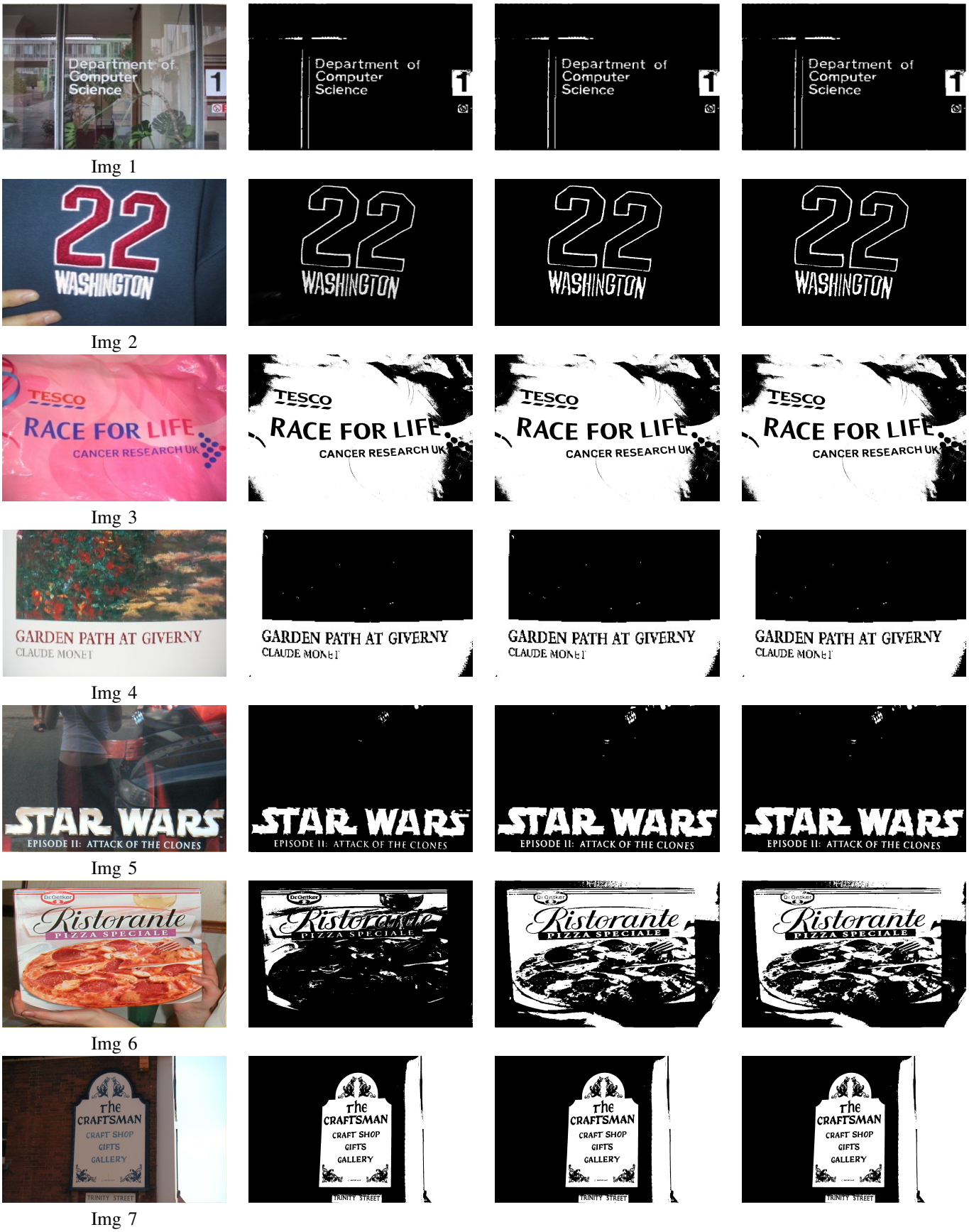

Fig. 5. Efficiency comparison between the gamma correction and binarization of GCM, AGCM and NGCM.

execution-time of the NGCM is dramatically reduced. It can not be generalized in a fraction of the GCM's execution-time because the gamma sub-range prediction size is variable from 0.1 to 2.5 in stable case, from 4 to 6.5 in decreasing case and simply 10 in increasing case. This makes that the number of gamma-modified images that have to be calculated and features analyzed is variable, while the GCM always analyses 100 images and the AGCM analyses 20 to 60 images. For example in the case of the image 2 of Figure 5, the execution-time of the NGCM is equal to 0.04 second while the execution-time of the GCM is equal to 4 second and 1.4 second of AGCM. In the case of the image 7 of Figure 5, the execution-time of the NGCM is equal to 0.15 second while the execution-time of the GCM is equal to 3.1 second and 0.9 second of AGCM.

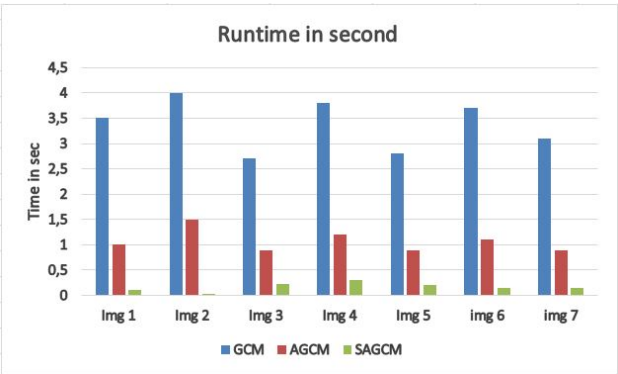

Fig. 6. Comparaison in execution time between the GCM, AGCM and NGCM over the ICDAR image set.

To verify the good functionality of our proposed method in real time applications. We applied the NGCM on video of a television newscast and webcam video stream. Figure 7 illustrates our NGCM real time execution on television newscast video. We notice that the NGCM extract 23 frames / second for a (416 * 240) frame size.

V. Conclusion

This paper introduced the GCM as a robust method for the segmentation and binarization of text features in complex color images. We presented a behavioral study of textural features curves and an effective sample method that retains the behavior of these curves. Our proposed method reduces dramatically the execution-time of the entire process while ensuring the same high level of efficiency and reliability of the original method. This improvement allows the NGCM to be integrated in real-time applications. Further improvement of the execution time can be done by a parallelized version of the algorithm.

Acknowledgment

We thank our colleagues from ESIEE Paris, who provided insight and expertise that greatly assisted the research and improved the manuscript.

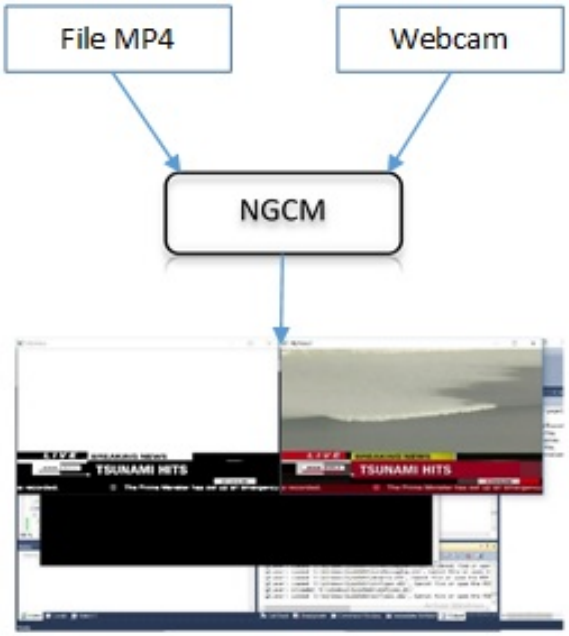

Fig. 7. NGCM real time execution on television newscast video (23 f/sec for a 416*240 frame size).

References

[1] S.V. Rice, F.R. Jenkins, and T.A. Nartker. The 5th annual test of OCR accuracy. Technical Report 96-01, Information Science Research Institute, University of Nevada, Las Vegas, April 1996.
[2] Y. Zhong, K. Karu, and A.K. Jain. Locating text in complex color images. Pattern Recognition, 28(10) :1523–1535, 1995.
[3] B.K. Sin, S.K. Kim, and B.J. Cho. Locating characters in scene images using frequency features. In 16th International Conference on Pattern Recognition (ICPR'02) - Volume 3, pages 489–492, Quebec, Canada, 2002.
[4] Y. Zhong, K. Karu, and A.K. Jain. "Locating text in complex color images. Pattern Recognition", 28(10) :1523–1535, 1995.
[5] J. Zhou and D.P. Lopresti. Locating and recognizing text in WWW images. Information Retrieval, 2(2/3) :177–206, 2000.
[6] J. Zhou, D.P. Lopresti, and T. Tasdizen. Finding text in color images. In Document Recognition V SPIE , San Jose, pages Vol 3305 : 130–140, 1998.
[7] B.K. Sin, S.K. Kim, and B.J. Cho. Locating characters in scene images using frequency features. In 16th International Conference on Pattern Recognition (ICPR'02) - Volume 3, pages 489–492, Quebec, Canada, 2002.
[8] Sumathi, C.P. and G. Gayathri Devi, "AUTOMATIC TEXT EXTRACTION FROM COMPLEX COLORED IMAGES USING GAMMA CORRECTION METHOD" Journal of Computer Science, 2012.
[9] https://wikipedia.org/wiki/Correction_gamma.
[10] Mahmood FarshbajDoustar and Hamid Hassanpour, A LOCALLY-ADAPTIVE APPROACH FOR IMAGE GAMMA CORRECTION. Shahrood University of Technology, Shahrood, Iran, 2010.
[11] Rostom Kachouri, Christian Armas, Mohamed Akil. Gamma correction acceleration for real-time text extraction from complex colored images. Image Processing (ICIP) 2015, Sep 2015, Quebec City, QC, Canada.
[12] N. Otsu, "A threshold selection method from gray-level histograms," IEEE Transactions on Systems, Man and Cybernetics, vol. SMC-9, pp. 62–66, January 1979.
[13] R.M. Haralick, K. Shanmugam, and T. Dinstein, "Textural features for image classification," IEEE Transactions on Systems, Man and Cybernetics, vol. SMC-3, pp. 610– 621, November 1973.
[14] ICDAR Dataset, "Text in scene images," http://robustreading.opendfki.de/wiki/SceneText.