

Digital Images Preprocessing for Optical Character Recognition in Video Frames Reconstructed from Compromising Electromagnetic Emanations from Video Cables

Santiago Morales-Aguilar⁽¹⁾, Chaouki Kasmi⁽¹⁾⁽³⁾, Milosch Meriac⁽¹⁾, Felix Vega⁽¹⁾⁽²⁾, Fahad Alyafei⁽¹⁾

(1) Directed Energy Research Centre, Technology and Innovation Institute, Abu Dhabi, United Arab Emirates, <https://tii.ae>

(2) Universidad Nacional de Colombia - Sede Bogotá, Bogotá, Colombia

(3) Faculty of Electrical Engineering, Helmut Schmidt University, Hamburg, Germany

Abstract

This work presents an analysis of the performance of two different preprocessing techniques for improving the quality of video frames previously reconstructed from compromising electromagnetic emanations produced by video interfaces connecting external monitors. Each technique depends on parameters such as white and black thresholds, kernel size of blur filter, contrast gain, brightness level, among others, and for this reason, an optimization process is applied by implementing an evolution strategy.

1 Introduction

Since the sixties, it has been known that unintended electromagnetic emanations would cause information leakage in electronic devices such as computers [1, 2]. These vulnerabilities were initially exploited only through specialized laboratory equipment not available to most of the people. However, now this is feasible with moderate budgets; for example, with software-defined radios (SDRs) and open-source software, video frames from external monitors can be reconstructed as shown in [3].

However, in a real scenario, someone should spend hundreds of hours watching the reconstructed video to gather relevant information, and this is a time-consuming task. Besides, environmental interference decreases the quality of images, which are noisy and difficult to interpret. For these reasons, the dependence on human intervention hinders the chances to recover any meaningful information.

On the other hand, classification and recognition of objects in digital images have improved recently, thanks (in part) to machine learning algorithms such as artificial neural networks (ANN) [4]. Machine learning has also been applied to the development of OCR tools, a technology that aims to extract machine-coded text from digital images. For example, Tesseract [5] is a well know OCR platform that, in its last version (version 4.0.0 at the time of writing of this paper), makes use of ANN to accomplish its task. However, in preliminary tests, this platform could not recognize any text from the reconstructed video frames.

An application of OCR for noisy reconstructed frames has been studied in [6]. The analysis has shown promising results to extract specific text characters automatically. Fol-

lowing that study, this paper proposes the evaluation of two preprocessing techniques applied to improve the noisy images prior to text recognition. While the mentioned work makes use of complex tools, here, the focus is to demonstrate the capability of adjusting the noisy frames with simpler techniques, so Tesseract is capable of recognizing the text on them.

The structure of this document is as follows: Section 2 describes the strategy implemented to evaluate the preprocessing of reconstructed video frames; later, in Section 3, the results are shown and discussed and finally in Section 4 the respective conclusions are presented.

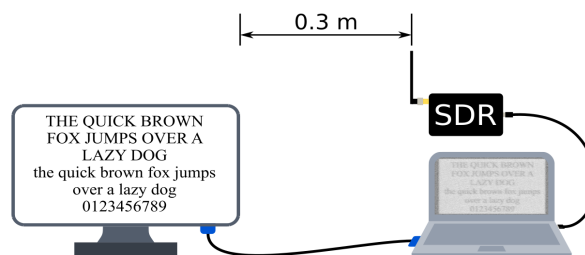


Figure 1. Physical setup to create the data set of reconstructed video frames through compromising electromagnetic emanations. The position and distance (0.3 m) between the target monitor and the SDR should remain fixed. The monitor displayed a pangram in uppercase and lowercase letters and numerical characters. The text was black with white background in full screen mode.

2 Description of the Strategy

2.1 Initial Setup

The first step was to create a data set of the reconstructed video frames. A physical arrangement (Fig. 1) was set up, and TempestSDR [3] was installed and configured; the position of the monitor and the SDR, and the distance between them remained fixed during data capture. Table 1 summarizes the characteristics of the video interface. The target monitor statically displayed a black and white text image in full screen. The text contains a pangram in both uppercase and lowercase as well as the ten numeric digits for a total of 94 characters (including spaces): "THE QUICK BROWN FOX JUMPS OVER A LAZY DOG the quick brown fox

Table 1. Characteristics of the video interface used in video frames reconstruction.

Condition	Value
Video Interface	VGA
Screen Resolution	1980 x 1080
Refresh Rate	60 Hz
Screen Size	24 inches

jumps over a lazy dog 0123456789".

The text was presented in two different types of fonts (Arial and Times New Roman), each font with ten different sizes: 30, 35, 40, 45, 50, 55, 60, 70, 80, and 90 pts (under the stated circumstances these are equivalent to approximately 45, 54, 63, 73, 83, 92, 102, 110, 130 and 148 horizontal pixels respectively). In each case, 20 frames were taken for a data set of 400 images. Fig. 2 depicts some samples. Finally, it is worth mentioning that in addition to the SDR connected to a commercial monopole antenna, no additional radio-frequency equipment was used.



Figure 2. Some examples of the original images and the reconstructed versions. Two fonts, and ten different sizes for each font were used. In each case there were 20 frames for a data set composed of 400 images.

2.2 Image Preprocessing Stage

As can be seen in Fig. 2, the captured frames are noisy, making it impossible for the OCR to recognize any text, although it has its image adjustment tools. For this reason, it is necessary to modify the image, so that noise is reduced and text identification is possible. Image processing tools can be used in a previous stage, such as the increase or the decrease in contrast and brightness, blur filters, threshold level evaluation, binary erosion/dilation, among others. It is proposed to evaluate the two different conditioning techniques described below.

a. Manipulation of the binary image: First, a blur filter is applied, then a threshold level determines black and white pixels, thus obtaining a binary image. The image is then eroded and dilated back to reduce noise [7]. With this technique, four parameters must be optimized, the kernel size of the blur filter, black and white threshold, erosion kernel size, and dilation kernel size. Fig. 3 illustrates the process.

b. Contrast and brightness variation: First, a blur filter is employed, then the image is modified by applying a linear transformation, depicted in equation (1), where F is the matrix representation of a grayscale image, and α and β are gain and level parameters respectively, which modify the level of contrast and brightness of the image. With this technique, three parameters must be optimized, α gain, β level, and blur filter kernel size. Fig. 3 illustrates the process.

$$F = \begin{bmatrix} f_{00} & f_{01} & \dots & f_{0w-1} \\ f_{10} & f_{11} & \dots & f_{1w-1} \\ \vdots & \vdots & \ddots & \vdots \\ f_{h-10} & f_{h-11} & \dots & f_{h-1w-1} \end{bmatrix}; (f_{ij}) \in [0,255]$$

$$F' = \alpha F + \beta; (f'_{ij}) \in [0,255] \quad (1)$$

Table 2 summarizes parameters and ranges for each one of the mentioned techniques.

2.3 Text Recognition and Evaluation

The preprocessed image (by one of the two techniques described above) continues to the text recognition stage with Tesseract. It is worth mentioning that this tool was employed due to its open-source nature and positive reviews. The Levenshtein distance [8] is used to compare the detected text with the reference string. This index measures the similarity between two strings counting how many modifications (exchanges, additions, or deletions of characters) must be made in one string to match the other one. An index of zero indicates that both strings are equal.

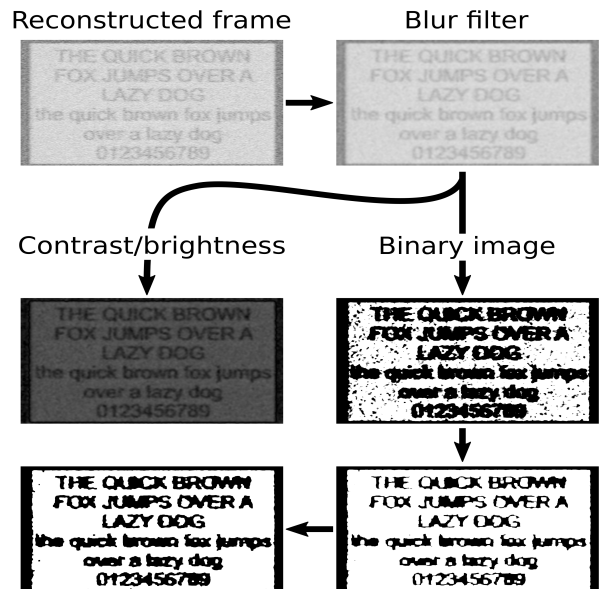


Figure 3. The steps of the two methods of preprocessing are shown. The blur filter is common to both. After that, contrast and brightness modification is applied on one method, and binary threshold, erosion, and dilation on the other.

Table 2. Parameters and their ranges according to the technique used.

Parameter	Min Value	Max Value	Type
Manipulation of the Binary Image			
Blur Filter Kernel Size	2	30	Integer
Black/White Threshold	2	253	Integer
Erosion Kernel Size	2	30	Integer
Dilation Kernel Size	2	30	Integer
Contrast and brightness variation			
Blur Filter Kernel Size	2	30	Integer
Alpha contrast gain	-255.0	255.0	Floating point
Beta brightness level	-64770.0	65025.0	Floating point

Algorithm 1: Evolution strategy used to optimize pre-processing parameters.

```

1 Create random population  $P$  of 100 individuals;
2 Set number of generations  $G$  to 0;
3 while  $G < 60$  do
4    $P' = \emptyset$ ;
5   Set  $n$  to 0;
6   Get population elite  $E$  from  $P$  (the best 20%);
7   while  $n < 200$  do
8     do
9       Randomly choose  $p1$  and  $p2$  from  $P$  giving
10      preference to individuals in  $E$  ;
11      Obtain  $c$  from pairing  $p1$  and  $p2$ ;
12      Mutate parameters in  $c$  (60% of mutation
13      probability);
14      while  $c$  is repeated in  $P$  and  $P'$ ;
15      Add  $c$  to  $P'$  ;
16       $n = n + 1$  ;
17    end
18    Evaluate all the individuals in  $P'$  ;
19    Delete 120 worst individuals from  $P'$  ;
20     $P = E + P'$  ;
21    Sort  $P$  by the fitness value of the individuals;
22     $G = G + 1$  ;
23  end

```

2.4 Optimization with Evolution Strategy

In order to obtain the optimal values of the parameters in Table 2, an evolution strategy [9] is applied to each one of the mentioned preprocessing techniques. The algorithm is briefly described next (Algorithm 1). An initial population of 100 individuals is randomly chosen (line 1); it is worth clarifying that in evolutionary computation, an individual refers to a set of parameters to be optimized. Each individual is evaluated employing the Levenshtein distance as the fitness value. Then, an iterative process begins (line 3), where new individuals are obtained from pairing, and parameter mutation (from line 4 to line 15); elitism [10] is used (20% of the population, line 6). A total of 200 new individuals are created (no repetitions allowed, line 12), evaluated (line 16), and the best 80 chosen (line 17) for the next population (line 18). The new population (again 100 hundred) is sorted according to their fitness value (line 19). Regarding the pairing and mutation processes, each new individual is obtained from two members of the previous population with a chance of 50% in inheriting each

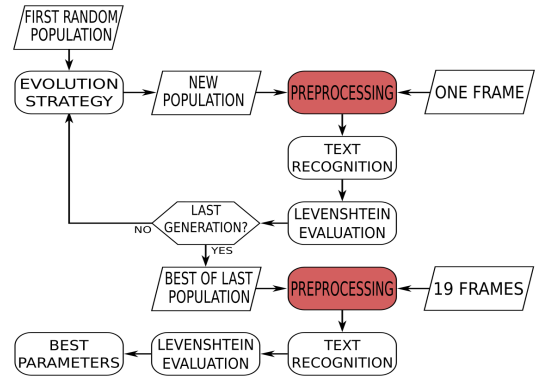


Figure 4. Flowchart of the whole process to evaluate the proposed preprocessing methods for OCR. The red boxes indicate where the this preprocessing is done.

parameter from the first or the second "parent" (line 10). Subsequently, there is a 60% probability of mutating the inherited parameter (line 11). There were 60 generations (equal to 12000 evaluations). Preliminary tests showed no improvements in the optimization process after this number of generations.

One frame was used in this process, and the remaining 19 frames evaluated the optimized parameters. With this strategy, over-fitting is reduced. This evolution strategy was used in the 20 different fonts (font type and font size). Finally, Fig. 4 illustrates the process of Subsections 2.2, 2.3, and 2.4.

3 Results and Discussion

The curves in Fig. 5 illustrate the evolution of the average fitness value of the elite. The results indicate that both preprocessing methods had a positive effect on sizes of 80 and 90 pts, with a score lower than 30 in the evaluation index. For the 70 pts size fonts, the results were variable with scores below and above 30 (Fig. 5c and Fig. 5d, respectively). In the case of fonts of 60 pts or smaller, no significant results were achieved, possibly because there is not enough information to recover the text.

Due to the previous results, the text is recognized on the remaining 19 frames with the best individuals for the cases of Arial 80 and 90 pts, and Times New Roman 80 and 90 pts. Table 3 shows a summary of the obtained scores. It can be seen that the binary image method offers better results than the contrast/brightness method. It is also worth noting that better results were obtained recognizing Times New Roman font. Possibly the "serifs" of this type of font help to conserve information of the core of the letter (for example, when erosion is applied). It can be seen that some parameters seem to converge to a range of values; a good example is the threshold value in the binary image method or the α in contrast/brightness. Other parameters vary in different scenarios. This behavior could be evidence that some steps in the mentioned techniques are more relevant for the improvement of the image.

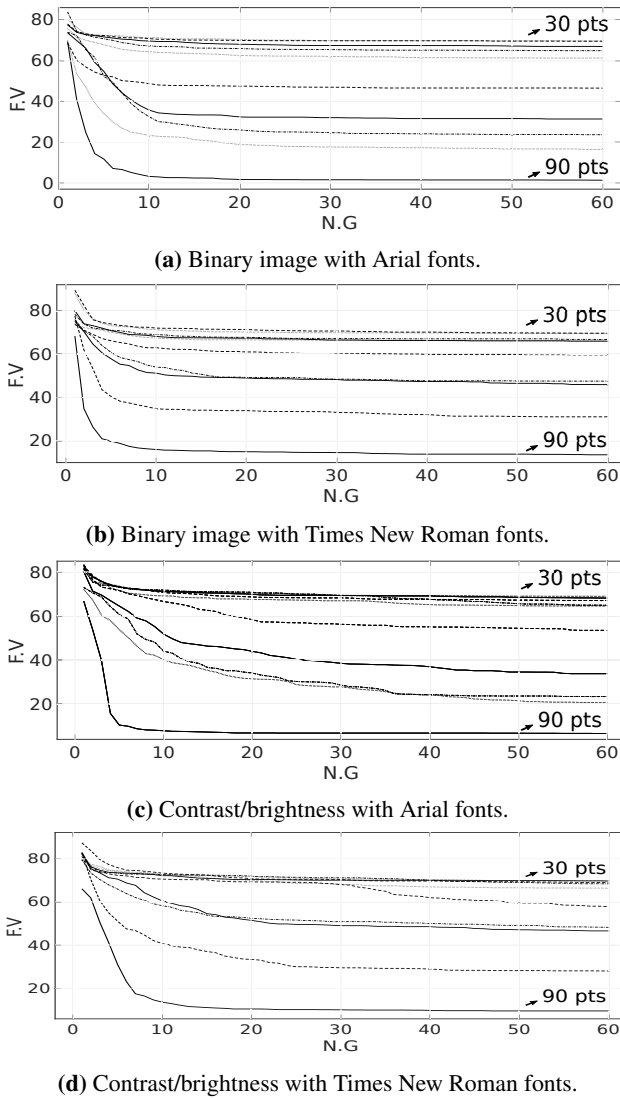


Figure 5. Evolution of the fitness value of the elite population for the 40 different fonts tested. Horizontal axis is the number of generations (N.G) and vertical axis is the fitness value (F.V).

4 Conclusions

The present work concludes that the application of image preprocessing methods, applied to reconstructed video frames from compromising emanations of video interfaces connecting external monitors, can be used within a set of tools that allow the recovery of information without human intervention. Although positive preliminary results were obtained with large font sizes, it would still be necessary to include several improvements to achieve a system capable of retrieving information in real scenarios where the diversity of fonts and sizes is enormous. For example, the RF interface could be improved in such a way that the captured frames are more detailed and less noisy, the OCR could be trained (fine-tuning) to recognize distorted versions of the fonts, and a post-processing stage consisting of a natural language engine could be added, so it corrects and clarifies text recognition errors that may prevail up to that point.

Table 3. Best average fitness values after evaluating 19 frames with the best parameters after optimization.

Contrast/Brightness α, β , Blur size Average F.V									
A80		A90		T80		T90			
-0.68, 108.40, 24	62	-1.68, 264.36, 21	37	0.89, -142.42, 18	44	1.55, -254.25, 22	35		
-1.08, 172.40, 24	63	-1.68, 262.32, 21	42	-0.69, 110.15, 18	45	1.59, -261.45, 22	35		
-1.33, 212.48, 24	63	-2.20, 347.00, 30	42	-0.59, 94.55, 18	45	1.56, -255.80, 21	37		
-1.59, 253.04, 24	64	-1.15, 180.50, 20	44	-0.60, 96.00, 18	45	0.76, -125.28, 19	38		
Binary Image Erode size, Dilate Size, Threshold, Blur size Average F.V									
A80		A90		T80		T90			
7, 11, 201, 20	49	5, 13, 198, 16	36	0, 6, 202, 13	43	5, 6, 205, 23	36		
9, 10, 202, 18	55	10, 17, 200, 19	39	4, 13, 201, 15	45	2, 7, 204, 24	38		
10, 16, 202, 18	56	8, 17, 199, 18	39	3, 10, 202, 13	46	4, 5, 204, 24	39		
12, 14, 203, 20	56	8, 16, 199, 18	39	0, 10, 202, 13	46	8, 5, 206, 21	41		

References

- [1] W. Van Eck, "Electromagnetic radiation from video display units: An eavesdropping risk?" *Computers & Security*, vol. 4, no. 4, pp. 269–286, 1985.
- [2] M. G. Kuhn and R. J. Anderson, "Soft tempest: Hidden data transmission using electromagnetic emanations," in *International Workshop on Information Hiding*. Springer, 1998, pp. 124–142.
- [3] M. Marinov, "Remote video eavesdropping using a software-defined radio platform," M.S. thesis, University of Cambridge, jun 2014. [Online]. Available: <https://github.com/martinmarinov/TempestSDR>
- [4] P. Winston, *Artificial Intelligence*, ser. A-W Series in Computerscience. Addison-Wesley Publishing Company, 1992. [Online]. Available: <https://books.google.ae/books?id=b4owngEACAAJ>
- [5] R. Smith, "An overview of the tesseract ocr engine," in *Ninth International Conference on Document Analysis and Recognition (ICDAR 2007)*, vol. 2. IEEE, 2007, pp. 629–633.
- [6] F. Lemarchand, C. Marlin, F. Montreuil, E. Nogues, and M. Pelcat, "Toxicia: Apprentissage profond appliqué à l'analyse des signaux parasites compromettants," 2019.
- [7] J. Liang, J. Piper, and J.-Y. Tang, "Erosion and dilation of binary images by arbitrary structuring elements using interval coding," *Pattern Recognition Letters*, vol. 9, no. 3, pp. 201–209, 1989.
- [8] V. I. Levenshtein, "Binary codes capable of correcting deletions, insertions, and reversals," in *Soviet physics doklady*, vol. 10, no. 8, 1966, pp. 707–710.
- [9] A. Auger, "Convergence results for the $(1, \lambda)$ -s-ases using the theory of ϕ -irreducible markov chains," *Theoretical Computer Science*, vol. 334, no. 1-3, pp. 35–69, 2005.
- [10] S. Baluja and R. Caruana, "Removing the genetics from the standard genetic algorithm," in *Machine Learning Proceedings 1995*. Elsevier, 1995, pp. 38–46.