Evaluating Performance of an Adult Pornography Classifier for Child Sexual Abuse Detection

Mhd Wesam Al-Nabki  
Dept. IESA.  
Universidad de León  
Researcher at INCIBE  
mnab@unileon.es

Eduardo Fidalgo  
Dept. IESA.  
Universidad de León  
Researcher at INCIBE  
eduardo.fidalgo@unileon.es

Roberto A. Vasco-Carofilis  
Dept. IESA.  
Universidad de León  
Researcher at INCIBE  
rvasc@unileon.es

Francisco Jañez-Martino  
Dept. IESA.  
Universidad de León  
Researcher at INCIBE  
fjamn@unileon.es

Javier Velasco-Mata  
Dept. IESA.  
Universidad de León  
Researcher at INCIBE  
jvelm@unileon.es

Abstract—The information technology revolution has facilitated reaching pornographic material for everyone, including minors who are the most vulnerable in case they were abused. Accuracy and time performance are features desired by forensic tools oriented to child sexual abuse detection, whose main components may rely on image or video classifiers. In this paper, we identify which are the hardware and software requirements that may affect the performance of a forensic tool. We evaluated the adult porn classifier proposed by Yahoo, based on Deep Learning, into two different OS and four Hardware configurations, with two and four different CPU and GPU, respectively. The classification speed on Ubuntu Operating System is 5 and 2 times faster than on Windows 10, when a CPU and GPU are used, respectively. We demonstrate the superiority of a GPU-based machine rather than a CPU-based one, being 7 to 8 times faster. Finally, we prove that the upward and downward interpolation function, using an upward or a downward interpolation function, affects the performance of the classifier in terms of accuracy and processing time.

Index Terms—Computer Vision, Adult Pornography Classification, Hardware Requirements

Type of contribution: Short original research

I. INTRODUCTION

Possession of Child Sexual Abuse Material (CSAM) is one of the most terrible crimes against children because it involves the sexual and violent abuse of innocent minors. Manual search for evidence in a seized hard drive can be a long and complex process due to the enormous number of files. Furthermore, when it comes to finding illegal material in the field of the police search, reliability and speed are essential. This is because the police forces have a limited time to search CSAM content on seized devices, and within this time slot, they can differentiate between taking the suspect in detention or not.

This paper is part of the European project Forensic Against Sexual Exploitation of Children (4NSEEK) [1], and its primary goal is to provide a forensic tool to detect CSAM via the combination of several modules: File Name Classifier (FNC) [2], Sexual Organ Detector (SOD) [3], Signature Camera Detection (SCD) [4], Adult Pornography Detector (APD) [5] and a Face detector, Age and Gender (FAG) estimator [6], [7], [8]. All these systems work simultaneously to identify CSAM (Fig. 1).

The speed and confidence of the prediction are critical when investigating a crime related to child sexual abuse. This paper focuses on finding the optimum hardware and software requirements to obtain the best performance of the APD system. Specifically, this paper attempts to answer three crucial questions: 1) what is the best Operating System (OS) to be used for deploying the software, Windows or Linux OS?, 2) what is the prediction speed using a Graphical Processing Unit (GPU) and a Central Processing Unit (CPU)? and 3) does the resizing of the input image, using an upward or a downward interpolation function, affects the performance of the classifier in terms of accuracy and processing time?

II. LITERATURE REVIEW

Several researchers have addressed the problem of identifying pornography images. A traditional strategy to identify nudity in images depends on detecting human skin in the image using color [9], [10] and/or texture [11]. When an input image contains a high percentage of pixels with colors close to the skin, it is considered as an indicator of nudity. However, this signal only is not reliable since a face and hands images...
have many skin pixels while being non-porn. Also, the color of the skin has a wide range that can match with other objects in the input image. To cope with this limitation, researchers have developed a bag of visual words (BOVW) model that attempts to extract the most frequent patches that exist on a set of training images and try to find it in the test images [12], [13].

The rise of Deep Learning (DL) techniques through the automatic feature extraction have revolutionized the state-of-the-art performance [14], [15], [16], [17], [18], [19], [20]. Yahoo Inc. proposed Not Suitable for Work (NSFW) convolutional neural network model [19] to identify adult pornography images. Moustafa et al. [14] used a combination of ConvNets, whereas they fused and fine-tuned AlexNet and GoogLeNet to adapt these models to pornographic data. Their model has shown a remarkable increase in the classification accuracy on the NPDIP Pornographic-800 and Pornographic-2k datasets [21]. Wang et al. presented a novel approach, called Strongly-supervised Deep Multiple Instance Learning (SDMIL) that models each input image as a bag of overlapped image patches, and they trained the model as a Multiple Instance Learning problems. Wehrmann et al. [17] used a Convolutional Neural Network and long short-term memory (LSTM) recurrent networks for detecting pornography content.

III. METHODOLOGY

To build the Adult Pornography Detector (APD), we adopted the Not Suitable for Work (NSFW) [22] model because it is dedicated to recognizing pornography images. A graphical representation of the model is shown in Figure 2. The NSFW model uses ResNet-50-thin architecture as a pretrained network [23], which was trained on 1,000 ImageNet dataset classes [24]. To adapt the ResNet-50-thin to a binary classifier, only the last layer was replaced with a two nodes fully-connected layer. After that, the weights of the model were finetuned on the NSFW dataset. Since the NSFW image classification model expects an input image size to be 256×256 pixels, a pre-processing function is called to resize the image to the desired size before predicting its category. Two popular techniques were proposed to change the size of the input image to fit with the input size of the model [25]; they are padding with zeros or interpolation. In this work, the APD module adopts the latter approach to resize the input image into the desired size.

IV. EXPERIMENTAL SETTINGS

To measure the performance of the APD module, we proposed a test set of 6,000 images, randomly selected from the Pornography Database [21]. The dataset is balanced whereas the non-pornographic and the pornographic classes have the same number of samples, i.e. 3,000 images. Fig. 3 shows samples of both categories of the dataset.

(a) Pornography class
(b) Non-pornography class

Fig. 3. Samples from the Pornography Database

It can be observed that the dataset contains challenging images that expose skin explicitly, while they are not pornographic, such as the samples illustrated in Fig. 4.

Fig. 4. Challenging samples from the non-pornography class

In contrast, other images do not involve skin exposure but they refer to the pornography class, like the samples shown in Fig. 5.

Table 1 presents the used computer machines to conduct the experiments of this paper. All the used machines are provided with Ubuntu 18.04 OS, except machine #4, which has a dual boot OS of Windows 10 and Ubuntu.

https://sites.google.com/site/pornographydatabase/
of the prediction. The APD module expects an image of 256x256 pixels. However, in the real case scenario, the size of the input image may vary significantly, as it might be smaller or larger than the desired size. Typically, a preprocessing function is called to resize them upward or downward. In this experiment, we downscale the input images by 25%, 50%, 75%, and 100% (the latter size refers to the original input size of the image, without resizing). Next, to feed the APD module with the input image, we call the preprocessing function to adjust the image size to the correct input size, i.e., 256x256 pixels.

Table III shows that resizing the input image does not affect the prediction time adversely. Instead, we observed faster performance when the images were downsampled before feeding it to the APD model. In our experiments, we realized that resizing the input images into 25% of their original size obtained the fastest prediction time.

Additionally, we estimated the accuracy of the APD model after resizing the images, as shown in Table IV. Interestingly, we did not record significant changes in the prediction performance of the model using the other resize values did not influence the prediction accuracy, except when resizing the image to 25% of its original size. In this case, the F1 score of the model increased from 0.73 to 0.74. Therefore, we can conclude that this upward and downward interpolation process to adjust the input image size does not affect the performance negatively, and it may lead to a positive impact.

### Table II

| Machine ID | GPU Processing Time (seconds) | CPU Processing Time (seconds) |
|------------|------------------------------|------------------------------|
| M. 1       | 57.38                        | 589.13                       |
| M. 2       | 80.61                        | 493.25                       |
| M. 3       | 86.61                        | 442.61                       |
| M. 4       | 89.19                        | 453.43                       |

### Table III

| Resize (%) | Nvidia RTX 2060 (seconds) | Intel Core i9 (seconds) |
|------------|---------------------------|-------------------------|
| 100%       | 57.88                     | 442.64                  |
| 75%        | 50.17                     | 435.95                  |
| 50%        | 48.32                     | 439.68                  |
| 25%        | 47.38                     | 428.42                  |

### Table IV

| Resize (%) | Precision | Recall | F1 Score | Accuracy |
|------------|-----------|--------|----------|----------|
| 100%       | 0.78      | 0.74   | 0.73     | 0.74     |
| 75%        | 0.78      | 0.74   | 0.73     | 0.74     |
| 50%        | 0.78      | 0.74   | 0.73     | 0.74     |
| 25%        | 0.81      | 0.75   | 0.74     | 0.75     |
VI. CONCLUSION AND FUTURE WORK

This paper analyzed the performance of Adult Pornography Detector (APD), which is a core component of the Forensic Against Sexual Exploitation of Children (4NSEEK) project to identify Child Sexual Abuse Material (CSAM). The APD adopted the Not Suitable for Work (NSFW) model to detect pornography images, and we established our experimentation on a balanced dataset of 6,000 images selected randomly from the Pornography Database.

Our analysis discovered that deploying the APD on an Ubuntu OS is faster than Windows 10 in terms of prediction time. Ubuntu OS was, at least, 5 and 2 times faster than Windows 10 in CPU and GPU machines, respectively. Furthermore, we found that using a GPU-based machine, i.e., Nvidia RTX 2060, is 7 to 8 times faster than a CPU-based machine, i.e., Intel Core i9, with a processing time of 57.88s and 442.61s, respectively. Finally, we realized that APD is robust against the upward and downward resizing of the input image on the classifier’s accuracy and speed. Also, we observed a slight improvement in the prediction accuracy and the processing time when the input images were downscaled to 25% of its original size.

In the future, we plan to enhance the performance of the base classification model. Concretely, we want to explore advanced pre-trained models, such as Inception Resnet and MobileNetV2.

ACKNOWLEDGEMENTS

This work was supported by the framework agreement between the University of León and INCIBE (Spanish National Cybersecurity Institute) under Addendum 01. This research has been funded with support from the European Commission under the 4NSEEK project with Grant Agreement 821966. This publication reflects the views only of the author, and the European Commission cannot be held responsible for any use which may be made of the information contained therein. We acknowledge NVIDIA Corporation with the donation of the TITAN Xp and Tesla K40 GPUs used for this research.

REFERENCES

[1] S. N. C. Institute, “Forensic against sexual exploitation of children,” https://www.incibe.es/en/european-projects/4nseek, 2020, accessed: 2020-03-09.
[2] M. Al-Nabki, E. Fidalgo., E. Alegre., and R. Aláiz-Rodríguez., “File name classification approach to identify child sexual abuse,” in Proceedings of the 9th International Conference on Pattern Recognition Applications and Methods - Volume 1: ICPRAM, INSTICC. SciTePress, 2020, pp. 228–234.
[3] A. Tabone, A. Bonnici, S. Cristina., R. Farrugia., and K. Camilleri., “Private body part detection using deep learning,” in Proceedings of the 9th International Conference on Pattern Recognition Applications and Methods - Volume 1: ICPRAM, INSTICC. SciTePress, 2020, pp. 205–211.
[4] G. S. Bennabhaktula., E. Alegre., D. Karataynovaya., and G. Azizapardiz., “Device-based image matching with similarity learning by convolutional neural networks that exploit the underlying camera sensor pattern noise,” in Proceedings of the 9th International Conference on Pattern Recognition Applications and Methods - Volume 1: ICPRAM, INSTICC. SciTePress, 2020, pp. 205–211.
[5] A. Gangwar, E. Fidalgo, E. Alegre, and V. González-Castro, “Pornography and child sexual abuse detection in image and video: A comparative evaluation,” in 8th International Conference on Imaging for Crime Detection and Prevention (ICDP 2017). IET, 2017.
[6] D. Chaves, E. Fidalgo, E. Alegre, and P. Blanco, “Improving speed-accuracy trade-off in face detectors for forensic tools by image resizing,” V Jornadas Nacionales de Investigación en Ciberseguridad, 2019.

D. Chaves, E. Fidalgo, E. Alegre, F. Jámez Martino, and J. Velasco-Mata, “Cpu vs gpu performance of deep learning based face detectors using resized images in forensic applications,” in 9th International Conference on Imaging for Crime Detection and Prevention (ICDP-2019), 2019, pp. 93–98.

D. Chaves., E. Fidalgo., E. Alegre., F. Jámez-Martino., and R. Biswas., “Improving age estimation in minors and young adults with occluded faces to fight against child sexual exploitation,” in Proceedings of the 15th International Joint Conference on Computer Vision, Imaging and Computer Graphics Theory and Applications - Volume 5: VISAPP, INSTICC. SciTePress, 2020, pp. 721–729.

Y.-C. Lin, H.-W. Tseng, and C.-S. Fuh, “Pornography detection using support vector machine,” in 16th IPPR Conference on Computer Vision, Graphics and Image Processing (CVGIP 2003), vol. 19, 2003, pp. 123–130.

H. Zuo, W. Hu, and O. Wu, “Patch-based skin color detection and its application to pornography image filtering,” in Proceedings of the 19th international conference on World Wide Web, 2010, pp. 1227–1228.

S. J. Sathisch and S. H. Sengamada, “Texture-based pornography detection,” Jul. 3 2008, uS Patent Appl. 11/715,051.

T. Deselaers, L. Pimentidis, and H. Ney, “Bag-of-visual-words models for adult image classification and filtering,” in 2008 18th International Conference on Pattern Recognition. IEEE, 2008, pp. 1–4.

C. X. Ries and R. Lienhart, “A survey on visual adult image recognition,” Multimedia tools and applications, vol. 69, no. 3, pp. 661–688, 2014.

M. Aoustaf, “Applying deep learning to classify pornographic images and videos,” Pacific Rim Symposium on Image and Video Technology, 2015.

Y. Wang, X. Jin, and X. Tan, “Pornographic image recognition by strongly-supervised deep instance learning,” in 2016 IEEE International Conference on Image Processing (ICIP). IEEE, 2016, pp. 4418–4422.

M. Perez, S. Avila, D. Moreira, D. Moraes, V. Testoni, E. Valle, S. Goldenstein, and A. Rocha, “Video pornography detection through deep learning techniques and motion information,” Neurocomputing, vol. 230, pp. 279–293, 2017.

J. Wehrmann, G. S. Simões, R. C. Barros, and V. F. Cavalcante, “Adult content detection in videos with convolutional and recurrent neural networks,” Neurocomputing, vol. 272, pp. 432–438, 2018.

M. Islam, P. Watters, A. N. Mahmood, and M. Alazab, Toward Detection of Child Exploitation Motivated by a Forensic Approach. Cham: Springer International Publishing, 2019, pp. 221–246.

Y. Inc., “Not suitable for work (nsfw) classification using deep neural network cellphone models,” https://github.com/yahoose/opensnsfw/, 2020, accessed: 2020-03-09.

D. Austin, A. Sanzgiri, K. Sankaran, R. Woodard, A. Lissack, and S. Seljan, “Classifying sensitive content in online advertisements with deep learning,” International Journal of Data Science and Analytics, pp. 1–12, 2020.

S. Avila, N. Thome, M. Cord, E. Valle, and A. D. A. Araújo, “Pooling in image representation: The visual codeword point of view,” Computer Vision and Image Understanding, vol. 117, no. 5, pp. 453–465, 2013.

J. Mahadeokar and G. Pesavento, “Open sourcing a deep learning solution for detecting nsfw images,” https://yahooeng.tumblr.com/post/151148689421/open-sourcing-a-deep-learning-solution-for, 2016, accessed: 2020-03-10.

K. He, X. Zhang, S. Ren, and J. Sun, “Deep residual learning for image recognition,” in Proceedings of the IEEE conference on computer vision and pattern recognition, 2016, pp. 770–778.

J. Deng, W. Dong, R. Socher, L. J. Li., K. Li., and L. Fei-Fei, “Imagenet: A large-scale hierarchical image database,” in 2009 IEEE conference on computer vision and pattern recognition. IEEE, 2009, pp. 248–255.

M. Hashemi, “Enlarging smaller images before inputting into convolutional neural network: zero-padding vs. interpolation,” Journal of Big Data, vol. 6, no. 1, p. 98, 2019.

D. Misal, “Linux vs windows: Which is the best os for data scientists?” https://www.analyticsindiamag.com/linux-vs-windows-which-is-the-best-os-for-data-scientists/, 2020.

C. Szegedy, S. Ioffe, V. Vanhoucke, and A. A. Alemi, “Inception-v4, inception-resnet and the impact of residual connections on learning,” in Thirty-first AAAI conference on artificial intelligence, 2017.

M. Sandler, A. Howard, M. Zhu, A. Zhmoginov, and L.-C. Chen, “MobileNetv2: Inverted residuals and linear bottlenecks,” in Proceedings of the IEEE conference on computer vision and pattern recognition, 2018, pp. 4510–4520.