Privacy Protection in Digital Content: A Multi-Faceted Approach to Face, Text, and Logo Blurring Using Computer Vision Techniques.

AbdeAli (K20-0122)
Syed Sufyan Imran (K20-0386)
Aun Ali (K20-0286)

Department of Computer Science, FAST
Fundamentals of Computer Vision
Mr. Farrukh Shahid

May 14, 2023

Contents

1 Introduction

In an era characterized by an overwhelming prevalence of digital media, privacy has emerged as a pivotal concern. Images and videos, which form a significant part of this digital landscape, often carry sensitive information that individuals or organizations wish to protect. This information could be in various forms, such as human faces, textual data, or even recognizable brand logos. Automated tools capable of detecting and anonymizing these elements can play a crucial role in maintaining privacy and preventing unauthorized use of such information.

This report introduces a comprehensive computer vision project which has been designed to tackle three specific privacy issues. The first use case focuses on the blurring of specific faces in video content, an application with vast utility in maintaining personal privacy and compliance with regulations in publicly shared videos. Leveraging the capabilities of a deep learning model, specifically the res10_300x300_ssd_iter_140000.caffemodel, the project successfully identifies and blurs targeted faces in videos.

The second use case deals with the detection and blurring of sensitive textual information in documents, a requirement often encountered in legal, healthcare, and financial industries, among others. This aspect of the project employs pytesseract_OCR, a powerful optical character recognition (OCR) tool, to identify text and subsequently blur sensitive sections.

Lastly, the third use case involves the detection and blurring of brand logos in videos. This application can be particularly useful in cases where brand identity needs to be protected or unauthorized brand promotion is to be avoided. For this use case, the project employs OpenCV's template matching, an efficient method for logo detection.

With its three-pronged approach, the FCV Project addresses crucial privacy considerations in digital content. The following sections will delve into the methodologies used, discuss the results, and explore potential future enhancements to the system.

2 Literature Review

2.1 ”Real-Time Object Detection System using Caffe Model” (2019)[1]

Author: Vaishali, Shilpi Singh.

In the research paper titled ”Real-Time Object Detection System using Caffe Model,” the authors address the challenges involved in real-time object detection using deep learning techniques and the Caffe model. The paper focuses on the complexities and obstacles encountered in accurately and efficiently detecting and classifying objects in real-world scenarios.

The authors dives into the related technologies, discussing various models and frameworks used for object detection. These include R-CNN (Region-based Convolutional Neural Network), Single Shot MultiBox Detector (SSD), AlexNet, YOLO (You Only Look Once), VGG (Visual Geometry Group), MobileNets, and TensorFlow. Each model has its own characteristics and advantages in terms of accuracy, speed, and applicability.

The authors specifically focus on the Caffe model, which is a deep learning framework known for its speed, modularity, and openness. Caffe provides model definitions, optimization settings, and pre-trained weights, making it convenient for implementing object detection systems. The paper highlights the versatility and performance of Caffe in various applications, including image classification, semantic segmentation, facial recognition, and speech recognition.

Furthermore, the authors discuss the significance of OpenCV, an open-source computer vision and machine learning library, in building the real-time object detection system. OpenCV offers a wide range of optimized algorithms and tools for tasks such as face recognition, object tracking, 3D modeling, and augmented reality. Its integration with Caffe enables efficient access to webcam streams and the application of object detection to each frame.

2.2 "An Overview of the Tesseract OCR Engine" (2007) [2]

Author: Ray Smith

The paper provides an overview of The Tesseract OCR engine, developed between 1984 and 1994, aimed to improve OCR accuracy for low-quality print. It followed a step-by-step process, handling both fixed-pitch and proportional text. Tesseract had a two-pass recognition approach, using an adaptive classifier to improve accuracy in the second pass. The engine could recognize skewed pages without de-skewing, saving image quality. It employed techniques like chopping joined characters and associating broken characters to enhance word recognition. Tesseract also had a static character classifier that matched features against prototypes. Linguistic analysis was minimal, with word selection based on different categories. An adaptive classifier, trained based on the static classifier's output, improved discrimination within documents. Overall, Tesseract achieved significant advancements in OCR accuracy, introducing innovative approaches in line finding, word recognition, character classification, and adaptive training.

3 Methodology

3.1 Face Detection and Blurring in Videos

For this use case, the project employs a deep learning model - res10_300x300_ssd_iter_140000.caffemodel. The video is decomposed into frames, and each frame is passed through the model, which detects regions likely to contain faces. These regions are represented by bounding boxes, each associated with a confidence score.

Simultaneously, reference images are loaded and their face encodings are computed using the face_recognition library. The detected faces in the video frames are then matched against these reference face encodings. When a match is found, the corresponding region in the frame is blurred using the GaussianBlur function from the OpenCV library. The modified frames are then recomposed into a video with the targeted faces blurred.

3.2 Text Detection and Blurring in Documents

The second use case makes use of pytesseract_OCR, an Optical Character Recognition (OCR) tool. The document image is converted into grayscale and the pytesseract_OCR tool is then used to detect text.

Specific sensitive information, defined in a predefined list, is identified within the detected text. The image regions containing this sensitive information are blurred using OpenCV's GaussianBlur function. The kernel size for the blur operation is dynamically determined based on the size of the detected text box, ensuring optimal blurring regardless of the text size.

3.3 Logo Detection and Blurring in Videos

The third use case involves the detection and blurring of brand logos in videos, accomplished using OpenCV's template matching technique.

Template images of the logos to be detected are prepared and loaded. The video is decomposed into frames, and each frame is converted into grayscale. Template matching is performed on each grayscale frame against the prepared logo templates. Matches with a high correlation score, determined by a preset threshold, are considered successful detections. The regions of the frame containing the detected logos are blurred using the GaussianBlur function from OpenCV. The modified frames, with the detected logos blurred, are recomposed back into a video.

4 Results and Discussion

The FCV Project was developed to safeguard privacy in digital content, focusing on three use cases: face blurring in videos, text blurring in documents, and logo blurring in videos.

4.1 Face Detection and Blurring in Videos

The deep learning model effectively detected and blurred targeted faces in various conditions. However, video quality and computational efficiency remain areas for improvement.

4.2 Text Detection and Blurring in Documents

The text detection tool worked well for machine-printed text, successfully blurring predefined sensitive information. Challenges lie in handling handwritten text and complex document layouts, which may require advanced OCR tools or machine learning models.

4.3 Logo Detection and Blurring in Videos

The template matching technique accurately detected and blurred well-defined logos. Improvements can be made for handling logos that are partially hidden, distorted, or at unusual angles, possibly through feature-based matching algorithms or specialized machine learning models.

Overall, CV_Project shows significant promise in maintaining privacy in digital content, with room for improvements and potential expansion to other privacy areas such as license plate or voice data anonymization.

5 Conclusion

The CV_Project, designed to protect privacy in digital content, has effectively addressed three primary use cases: blurring specific faces in videos, obfuscating sensitive text in documents, and blurring recognizable brand logos in videos. Utilizing a deep learning model for face detection, an Optical Character Recognition tool for text detection, and a template matching technique for logo detection, the project has demonstrated its potential in preserving privacy across various digital media.
Despite the promising results, there are areas for enhancement in each use case, including handling video quality and computational efficiency for face detection, improving recognition of handwritten text and complex document layouts for text detection, and better detection of partially obscured or distorted logos. These provide opportunities for future research and development.
In conclusion, the CV_Project represents a significant step towards a comprehensive privacy protection solution in digital content. Its modular design allows for continual improvements and adaptations in line with advancements in computer vision and machine learning technologies. Furthermore, its potential for expansion to additional privacy areas such as license plate or voice

data anonymization underscores its potential as a versatile tool for privacy preservation in the digital age.

References

[1] Vaishali, Shilpi Singh. "Real-time object detection system using caffe model." Int Res J Eng Technol 6.5 (2019): 5727-5732./.

[2] Smith, R. (2007). An Overview of the Tesseract OCR Engine. Ninth International Conference on Document Analysis and Recognition (ICDAR 2007) Vol 2. DOI: https://doi.org/10.1109/icdar.2007.4376991.