Face Detection and Privacy Preservation with Optimized Centerface

Saketh Krishna Dumpuru (116061887)

Manjunaatt Girish Kumar (116847153)

May 18, 2025

CONTENTS

1 Introduction

Face anonymization has emerged as a critical imperative in our increasingly surveilled world, where privacy concerns intersect with ubiquitous monitoring systems and widespread digital content sharing. While the field has evolved from rudimentary pixelation to sophisticated GAN based approaches, delivering consistent anonymization performance in real-time environments remains an elusive challenge. Our project addresses this gap by implementing a robust privacy preservation framework that leverages the streamlined efficiency of the CenterFace detection model combined with our proprietary adaptive blurring algorithm. The delicate balance between comprehensive anonymization and maintaining visual data utility represents a core tension in this domain. Overly aggressive privacy measures can undermine legitimate analytical functions from behavior recognition to environmental assessment while insufficient protection compromises individual privacy rights. Our research navigates this complex terrain through a meticulously engineered pipeline that optimizes the CenterFace architecture's exceptional detection capabilities with custom-developed enhancements. These include advanced preprocessing techniques, rigorous detection validation protocols, sophisticated temporal coherence mechanisms for video streams, and contextually intelligent blurring strategies. The culmination of these efforts is a high performance system capable of delivering real-time face anonymization across diverse applications.

2 Related Works

Face detection techniques have evolved significantly, from classical methods like Viola-Jones and Haar cascades, which are known for their simplicity and computational efficiency, to modern deep learning-based methods like YOLO, SSD, RetinaNet, and CenterFace, offering higher accuracy and robustness in complex environments.

Face anonymization approaches typically include pixelation, Gaussian blurring, and GAN-based anonymization, each with varying levels of privacy preservation and visual quality. The main methodology of this project is inspired by the general concept of using efficient deep learning models for face detection in anonymization pipelines.

However, our implementation utilizes the **CenterFace model directly** and incorporates **custom-developed pre-processing, validation, temporal smoothing, and adaptive blurring techniques** to achieve a balance between anonymization effectiveness, accuracy, and computational efficiency tailored to our specific project goals.

3 Methodology

Our project's methodology involves a multi-stage pipeline for processing input images or video frames to achieve privacy-preserving face anonymization. The flow of data through the system is sequential, with each stage building upon the output of the previous one.

3.1 Pipeline Flow:

1. **Input Acquisition:** The system receives an input image or a frame from a video stream.
2. **Pre-processing:** The input image/frame undergoes contrast enhancement.
3. **Face Detection:** The pre-processed image is passed to the CenterFace model for initial face localization.
4. **Face Validation:** Detected regions are subjected to ORB keypoint and RANSAC-based geometry checks to filter out false positives.
5. **Temporal Smoothing (Video Only):** For video sequences, validated detections are tracked across frames using IoU overlap.
6. **Custom Adaptive Blurring:** Based on the final, validated, and tracked face locations, an adaptive blurring filter is applied.
7. **Output Generation:** The anonymized image or frame is generated. For videos, these frames are reassembled into an output video file, optionally including the original audio.

3.2 Technical Implementation Details:

Our face anonymization pipeline is implemented in Python, leveraging established computer vision and deep learning libraries. The process for each input image or video frame follows the sequential flow outlined above.

- **Input Acquisition:**

  Input is handled using OpenCV's cv2.VideoCapture for both video files and webcam streams, and cv2.imread for static images. This provides a standard interface for reading image data, returning frames as NumPy arrays in BGR format.

- **Pre-processing (CLAHE):**
  To enhance contrast and improve the robustness of subsequent face detection, each input frame is subjected to Contrast Limited Adaptive Histogram Equalization (CLAHE). The frame is first converted from BGR to the LAB color space using cv2.cvtColor(frame, cv2.COLOR_BGR2LAB). CLAHE is then applied exclusively to the L (lightness) channel using cv2.createCLAHE(clipLimit=2.0, tileGridSize=(8,8)). The clipLimit parameter (set to 2.0) controls the contrast limiting, preventing over-amplification of noise,

while tileGridSize (set to 8x8) defines the grid size over which histogram equalization is performed locally. The enhanced L-channel is then merged back with the original A and B channels using cv2.merge() and converted back to BGR format. This yields a contrast-enhanced image enhanced_frame.

- **Face Detection (CenterFace ONNX):**
  Face detection is performed on the enhanced_frame using the CenterFace model, loaded as an ONNX model via the onnxruntime library. The model is initialized with onnxruntime.InferenceSession, configured to utilize available hardware acceleration providers (e.g., CUDA) where possible. The enhanced_frame is preprocessed into a blob using cv2.dnn.blobFromImage, which handles resizing to the model's expected input dimensions (dynamically adjusted to be a multiple of 32), mean subtraction (implicitly 0), and scaling (implicitly 1.0). This blob is then passed to the ONNX Runtime session for inference. The CenterFace model outputs a heatmap indicating face centers, scale factors, offset predictions, and 5 facial landmark coordinates for potential detections. These raw outputs are decoded to produce bounding boxes ([x1, y1, x2, y2]) and landmark sets ([x1, y1, ..., x5, y5]) with associated confidence scores. A confidence threshold (e.g., 0.5) is applied to filter out low-confidence detections. Bounding box and landmark coordinates are then rescaled from the model's input resolution back to the original frame dimensions.

- **Face Validation (ORB & RANSAC):**
  This stage processes the raw detections from the CenterFace model to improve precision and reduce false positives.

  1. **ORB Keypoint Detection:** For each detected bounding box, the corresponding image patch is extracted from the enhanced_frame. The ORB (Oriented FAST and Rotated BRIEF) feature detector (cv2.ORB_create(nfeatures=500)) is applied to the grayscale version of this patch using orb.detect(). A detection is considered potentially valid only if the number of detected ORB keypoints within the bounding box is greater than or equal to a predefined minimum threshold (min_orb_kp = 15). This checks for the presence of sufficient local texture and features characteristic of a face.

  2. **RANSAC-based Landmark Geometry Validation:** For detections that pass the ORB check, we perform a RANSAC (Random Sample Consensus) validation of the facial landmark geometry. The detected 5-point landmarks for the current frame are compared against the temporally smoothed landmarks from the *previous* frame. We use cv2.estimateAffinePartial2D with method=cv2.RANSAC to estimate a partial affine transformation between the two sets of landmark points. This function iteratively samples point pairs to find the transformation that aligns the largest number of inliers within a specified ransacReprojThreshold (set to 5.0 pixels). A detection's landmarks are considered geometrically consistent if the RANSAC process successfully estimates a transformation with a

number of inliers greater than or equal to a minimum threshold (min_inliers = 4). This step robustly verifies that the detected face's structure is consistent with its appearance in the previous frame, helping to reject spurious detections caused by non-face objects or transient patterns. Only detections that pass both the ORB and RANSAC validation steps are considered final validated faces for the current frame.

- **Temporal Smoothing (IoU Tracking - Video Only):**

  For video input, validated face detections are tracked across frames to maintain consistent anonymization masks. This is achieved using the Intersection over Union (IoU) metric. For each validated bounding box in the current frame, its IoU is calculated with the bounding boxes of all currently tracked faces. A current detection is associated with a previous track if their IoU exceeds a predefined threshold (e.g., 0.1 in the code, with a preference for the highest IoU match). New tracks are started for unassociated detections, and old tracks are dropped if they are not matched in the current frame for a certain duration. The bounding box coordinates for tracked faces are then smoothed using an exponential moving average: smoothed_box = alpha * previous_box + (1 - alpha) * current_box, where alpha (set to 0.6 in the code) is the smoothing weight. This temporal smoothing reduces jitter and provides more stable anonymization regions in video sequences.

- **Custom Adaptive Blurring:**
  Based on the final set of validated (and temporally smoothed, for video) face bounding boxes, an anonymization filter is applied. Our custom adaptive strategy selects the blurring method based on the size (area) of the detected face:

  1. For each validated bounding box, its area is calculated.
  2. A size threshold (currently a fixed area of 10000 pixels in the code) determines whether Gaussian or Mosaic blurring is applied.
  3. If the face area is above the threshold, a Gaussian blur is applied to the face region using cv2.GaussianBlur. The kernel size (gaussian_ksize, default 25 in the code) is a configurable parameter, which is clamped to be odd and not exceed the dimensions of the face region to ensure valid kernel sizes.
  4. If the face area is below the threshold, a mosaic filter is applied. This is implemented by resizing the face region down to a small size determined by a fixed mosaic_size parameter (default 10 in the code), and then resizing it back to the original face region dimensions using cv2.INTER_NEAREST interpolation to create the pixelated effect.

  The chosen blurring filter is applied to the corresponding region in the processed_frame, overwriting the original face pixels. Note that while the choice of blurring method is adaptive based on face size, the specific kernel/block size applied for Gaussian or Mosaic blurring is determined by fixed parameters in the current implementation, not

dynamically scaled proportionally or inversely proportionally to the individual face dimensions.

- **Implementation Details:**
  The core pipeline logic is encapsulated within the process_frame function, which takes a frame and previous frame state (for temporal smoothing) as input and returns the anonymized frame. Helper functions apply_clahe and apply_blur handle specific steps. The CenterFace class wraps the ONNX model inference, including dynamic input shape handling and result decoding. The main script includes functions (blur_faces_in_image, blur_faces_in_video, blur_faces_in_webcam_and_record) to handle different input sources and manage the frame processing loop, temporal state (stored in prev_boxes_vid and prev_lms_vid), and output saving using cv2.VideoWriter. Parameters Face anonymization has become increasingly important with growing concerns about privacy in public and digital environments. With extensive surveillance systems and the sharing of multimedia content online, protecting individual privacy through effective anonymization techniques is crucial. Current anonymization methods vary from simple pixelation to so- phisticated generative adversarial networks (GANs). However, real-time anonymization with consistent accuracy remains a challenge. This project aims to implement a robust, privacy- preserving face anonymization method using the CenterFace model for detection, coupled with a custom-developed adaptive blurring technique, capable of operating effectively in real-time scenariosfor detection thresholds, validation criteria, smoothing weight, and blurring sizes are defined as global variables, allowing for easy adjustment. Error handling is included for file loading, video writing, and potential issues within processing steps.

4 Results

The results of our implemented face anonymization pipeline demonstrate its effectiveness in detecting and anonymizing faces in both static images and video sequences. Qualitative evaluation shows robust face detection performance across various lighting conditions and poses, supported by our validation steps which successfully reduce false positives observed with the raw detector output.

Our custom adaptive blurring technique provides visually distinct anonymization outputs. Gaussian blurring is applied to larger faces, resulting in a smooth obscuration, while mosaic blurring is used for smaller faces, providing a pixelated effect.

Video processing results highlight the benefit of our temporal smoothing, showing stable tracking and consistent application of the anonymization masks across frames, which contributes to a smoother and less distracting output compared to frame-by-frame processing without tracking.

Based on the confidence scores, the blurring takes place which can be seen in the below given examples.

The below google drive link contains the video files captured through a webcam in order to see the blurring take place in real time.

`https://drive.google.com/file/d/1X-X1i241TG4cA_WYbRgDKeXGKUCwCtAa/view?usp=sharing`

The images are majorly taken from the wider face dataset. Here are some of the examples,

Figure 1: Press conference

Figure 2: Cheerleader

Figure 3: Karachi Stock Exchange

Figure 4: Basketball Celebration

Figure 5: Basketball team

Figure 6: Classical Dancers

Figure 7: Picnic Art

5 Conclusion

This project successfully developed and implemented a privacy-preserving face anonymization pipeline leveraging the CenterFace model for efficient face detection, complemented by custom pre-processing, validation, temporal smoothing, and an adaptive blurring technique. The integration of CLAHE enhanced image contrast, improving detection robustness. The ORB and RANSAC validation steps effectively filtered out false positives, increasing the precision of the anonymization. For video inputs, the IoU-based temporal smoothing ensured consistent tracking and stable application of anonymization masks across frames. The custom adaptive blurring strategy provided a flexible approach, applying Gaussian blur to larger faces and mosaic blur to smaller ones, balancing visual quality with privacy requirements. While the current implementation demonstrates a functional and effective system for real-time face anonymization, it also serves as a strong foundation for further research and development in this critical area.

6 Discussion and Future Work

Advancing Face Detection: While CenterFace offers a good balance of speed and accuracy, exploring more recent state-of-the-art face detection models could yield further improvements, particularly in challenging scenarios involving extreme poses, heavy occlusions, or very small faces. Models like SCRFD, which demonstrate superior performance on benchmarks like WIDER FACE, could be evaluated. Integrating these models would involve updating the ONNX model loading and inference logic, potentially requiring adjustments to pre-processing and output decoding based on the specific model architecture. Investigating lightweight or quantized versions of advanced models could maintain real-time performance while boosting detection capabilities.

Refining Validation Techniques: The current ORB and RANSAC validation steps provide a robust filter for false positives. Future work could explore more sophisticated validation methods. For instance, incorporating a lightweight face attribute classifier (e.g., checking for eye/mouth presence) or utilizing a dedicated landmark detection model (if not provided by the primary detector) could offer alternative or complementary validation signals. Investigating machine learning-based approaches to classify detections as true positives or false positives based on a richer set of features could also be a direction.

Sophisticated Anonymization Methods: The adaptive blurring technique, while effective and computationally efficient, represents a basic form of anonymization. Future work could explore more advanced, potentially learned, anonymization methods.

GAN-based Anonymization: Replacing blurred faces with synthetic faces generated by a GAN could offer a higher degree of privacy while potentially preserving more visual context or allowing for controlled attributes (e.g., generating a synthetic face with a similar expression but different identity). This would involve training or fine-tuning a conditional GAN model and integrating its inference into the pipeline. Challenges include maintaining

real-time performance and ensuring the generated faces are sufficiently diverse to prevent re-identification.

Diffusion Models: Recent advancements in diffusion models for image generation could also be explored for generating synthetic faces. These models have shown impressive results in generating high-fidelity images and offer fine-grained control over generated attributes, which could be beneficial for anonymization.

Learned Anonymization Filters: Instead of predefined blurring or mosaic, a small neural network could be trained to predict and apply an optimal, perhaps spatially varying, anonymization filter based on the detected face region.

Exploring Stronger Privacy Guarantees: Beyond visual anonymization, future work could investigate integrating concepts from differential privacy or k-anonymity. Applying noise or transformations to facial features or embeddings in a differentially private manner could provide mathematical guarantees against re-identification, although this often comes at the cost of data utility. Implementing k-anonymity could involve replacing a face with an "average" face from a group of k similar individuals, ensuring that the individual is indistinguishable from at least k-1 others.

Integrating audio handling using libraries like moviepy to combine the processed video frames with the original audio track would also enhance the utility of the tool for video anonymization tasks.

By pursuing these technical directions, the project can evolve into a more sophisticated, robust, and privacy-preserving face anonymization system, contributing further to the development of ethical and effective computer vision applications.

7 References

P. Viola and M. Jones, "Rapid object detection using a boosted cascade of simple features," Proceedings of the 2001 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 2001, pp. I-511-I-518 vol. 1.

T. Ojala, M. Pietikäinen, and T. Mäenpää, "Face detection with multi-resolution gray-scale and rotation invariant texture measures," European Conference on Computer Vision, 2002, pp. 240-253.

H. Li, Z. Fan, B. Xiang, and S. Jiang, "Cascaded convolutional networks for face detection," 2015 IEEE International Conference on Multimedia Expo (ICME), 2015, pp. 1-6.

K. Zhang, Z. Zhang, Z. Li, and Y. Qiao, "Joint Face Detection and Alignment using Multitask Cascaded Convolutional Networks," IEEE Signal Processing Letters, vol. 23, no. 10, pp. 1499-1503, 2016.

W. Liu et al., "SSD: Single Shot MultiBox Detector," European Conference on Computer Vision, 2016, pp. 21-37.

J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, "You Only Look Once: Unified, Real-Time Object Detection," 2016 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), 2016, pp. 779-788.

J. Deng et al., "RetinaFace: Single-Shot Multi-Level Face Localisation in the Wild," 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), 2020, pp. 5203-5212.

Z. Zhou, D. Wang, and P. Krähenbühl, "Objects as Points," arXiv preprint arXiv:1904.07850, 2019.

Y. Xu, W. Yan, H. Sun, G. Yang, and J. Luo, "CenterFace: Joint Face Detection and Alignment Using Face as Point," arXiv preprint arXiv:1911.03599, 2019.

B. Cao et al., "GAN-based Face Anonymization," arXiv preprint arXiv:1807.05054, 2018.

H. Hukkelas, K. R. Jensen, and F. Lindseth, "DeepPrivacy: A Generative Adversarial Network for Face Anonymization," International Symposium on Visual Computing, 2019, pp. 418-429.

F. Hellmann et al., "GANonymization: A GAN-based Face Anonymization Framework for Preserving Emotional Expressions," arXiv preprint arXiv:2305.02143, 2023.

J. Guo et al., "SCRFD: Spatial Contextual Attention for Robust Face Detection," arXiv preprint arXiv:2105.04714, 2021.