Project Goals and Objectives

Our project aims to implement a facial detection algorithm capable of identifying faces, drawing bounding boxes around them, and assigning confidence scores to each detection. The overarching goal is to replicate the functionality of OpenCV's pretrained Haar cascade classifiers using our own implementation. Key milestones include feature extraction via integral images, training a face detection classifier with AdaBoost, and implementing multiscale detection to handle faces of varying sizes and orientations.

Summary of Progress

We have made significant progress toward our goal over the past three weeks, completing Milestone 2 and advancing toward Milestone 3.

Milestone 1: Preliminary Analysis and Research: We conducted an in-depth analysis of facial detection techniques, focusing on the Viola-Jones algorithm and OpenCV's pretrained Haar cascade classifiers. This phase helped establish a strong theoretical foundation for our project. We identified and curated relevant datasets, including the BioID Face Database and the Caltech 101 dataset, which serve as our primary sources of positive (faces) and negative (non-faces) samples.

Milestone 2: Feature Extraction with Integral Images: Using integral images, we implemented the extraction of Haar-like features, which are critical inputs for training the AdaBoost classifier. The extraction process was optimized for computational efficiency and validated against standard benchmarks. Figures in the appendix demonstrate true positive features detected from our testing datasets.

Progress Toward Milestone 3: Training the Classifier

While Milestone 3 is ongoing, we have:

- Preprocessed datasets to create balanced training data of positive and negative samples.
- Implemented the AdaBoost algorithm to combine weak classifiers trained on Haar features.
- Begun testing the classifier on a subset of images, observing initial detection performance.

Preliminary tests show the classifier can identify facial features but requires further fine-tuning to improve accuracy and reduce false positives.

Challenges and Solutions

Dataset Preparation. Creating balanced datasets was a significant challenge. Positive samples were collected from the BioID and Caltech face datasets, while negative samples were drawn from non-facial categories within the Caltech 101 dataset and generated random noise images. Balancing the number of samples and ensuring sufficient diversity was essential to avoid training bias.

AdaBoost Algorithm. Implementing AdaBoost was technically demanding. The algorithm requires a careful balance of weak classifier weights to create an effective strong classifier. Challenges included:

- Avoiding overfitting to training data.
- Handling false positives in non-facial images and false negatives in images with low-resolution faces.

Classifier Testing. Initial tests revealed that some Haar features identified by our algorithm correspond to edges and intensity changes unrelated to faces. While this is expected behavior for early-stage classifiers, it underscores the need for improved feature selection and parameter tuning.

Current Obstacles and Questions
While we have overcome many challenges, the following issues require attention:

1. Classifier Evaluation: We are uncertain about the best statistical metrics to evaluate classifier performance. Metrics like precision, recall, and F1-score are under consideration, but guidance on best practices for this task would be helpful.
2. Generalization: The classifier struggles to generalize well to unseen data. Features detected in non-facial images often resemble those in the training set, leading to false positives. Suggestions for mitigating overfitting and improving generalization would be valuable.
3. Parameter Optimization: AdaBoost's performance is highly sensitive to the number of weak classifiers and their weighting strategy. Guidance on optimizing these parameters to balance speed and accuracy would be appreciated.

Next Steps and Future Plans
Plans Moving Forward
The next steps for our project involve completing the final steps of Milestone 3 and preparing for Milestone 4. Based on the feedback and notes from the last report, we've adjusted our approach to ensure we are moving forward with a more focused and specific plan:

Finalize the AdaBoost Implementation (Milestone 3): We will continue refining the AdaBoost algorithm and finalize the classifier implementation. This involves:

- Testing on a diverse set of images to evaluate performance on both seen and unseen data.
- Fine-tuning parameters such as the number of weak classifiers and adjusting the weighting strategy for more accurate face detection.
- Evaluating performance using precision, recall, and F1-score metrics to better understand detection quality and error rates.

Begin Preparations for Milestone 4 (Multiscale Face Detection): Milestone 4 will focus on improving the scalability of the face detection algorithm by implementing a multiscale detection approach. This will allow the algorithm to detect faces at various sizes, improving its robustness for images with faces at different distances. We plan to:

- Rework the detection process to incorporate a sliding window approach at multiple scales, where the window size changes as the image is scanned at different zoom levels.
- Implement pyramid images (e.g., Gaussian pyramids) for multiscale testing, where the image is progressively downscaled to detect faces at different resolutions.
- Optimize the process to handle the computational load of multiscale detection efficiently without sacrificing detection accuracy.

Evaluate Multiscale Performance: Once Milestone 4 is implemented, we will thoroughly evaluate how well the classifier performs with faces at different scales. This evaluation will include:

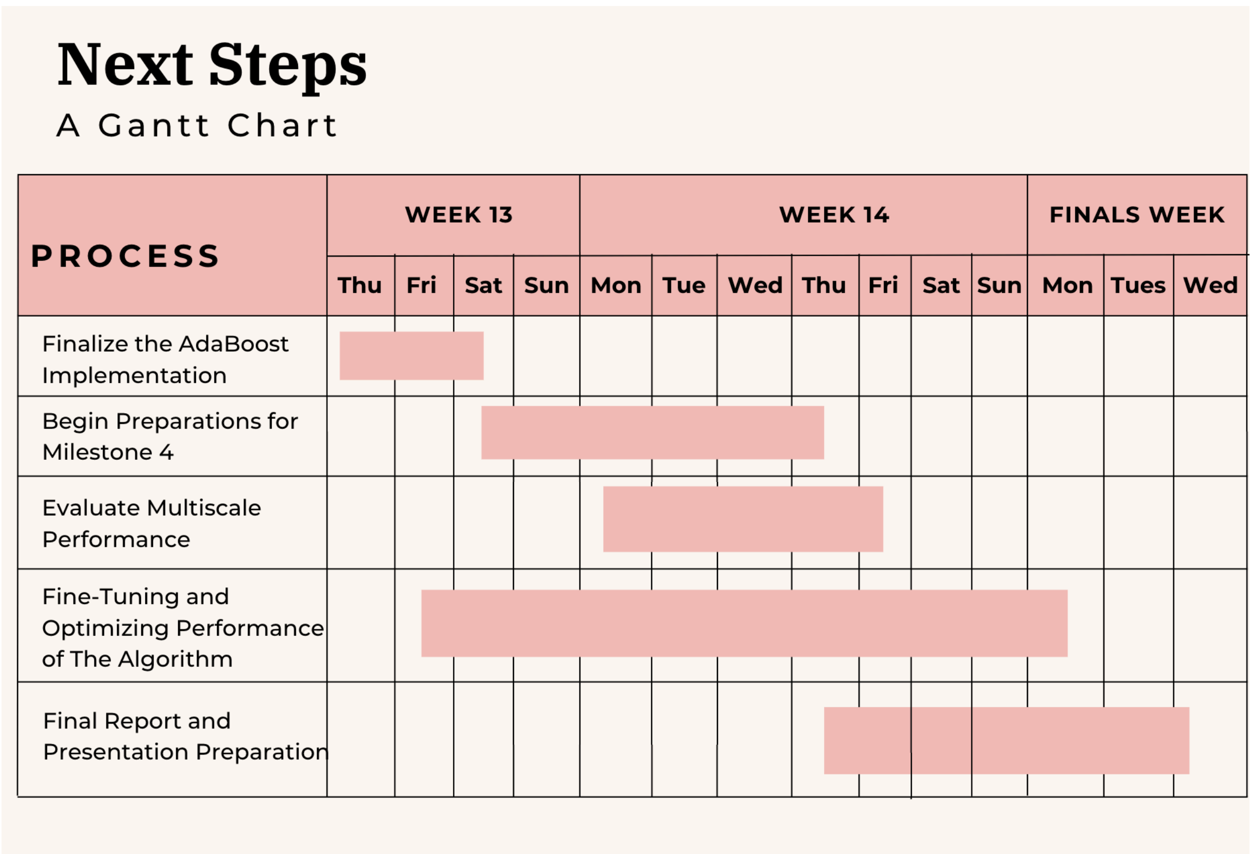
- Testing the classifier with images that have faces in varying sizes (close-up, far away, etc.) to ensure the algorithm detects faces at all scales.
- Optimizing the detection window to balance detection accuracy and processing time.
- Addressing potential challenges such as false positives when scaling, particularly with smaller faces or faces with unusual orientations.

Final Report and Presentation:
As we move through Milestones 3 and 4, we will begin preparing the final report. This will include a

detailed comparison of our implementation to OpenCV's pretrained classifier and the Viola & Jones algorithm. We will focus on:

- Methodology: Documenting each step of our process from feature extraction through classifier training and multiscale implementation.
- Results**:** Presenting performance metrics and evaluating the effectiveness of our classifier.
- Lessons Learned**:** Reflecting on challenges faced and how we overcame them, particularly around dataset balancing, classifier training, and multiscale face detection

Below is a timeline of how we wish to proceed with our next steps:



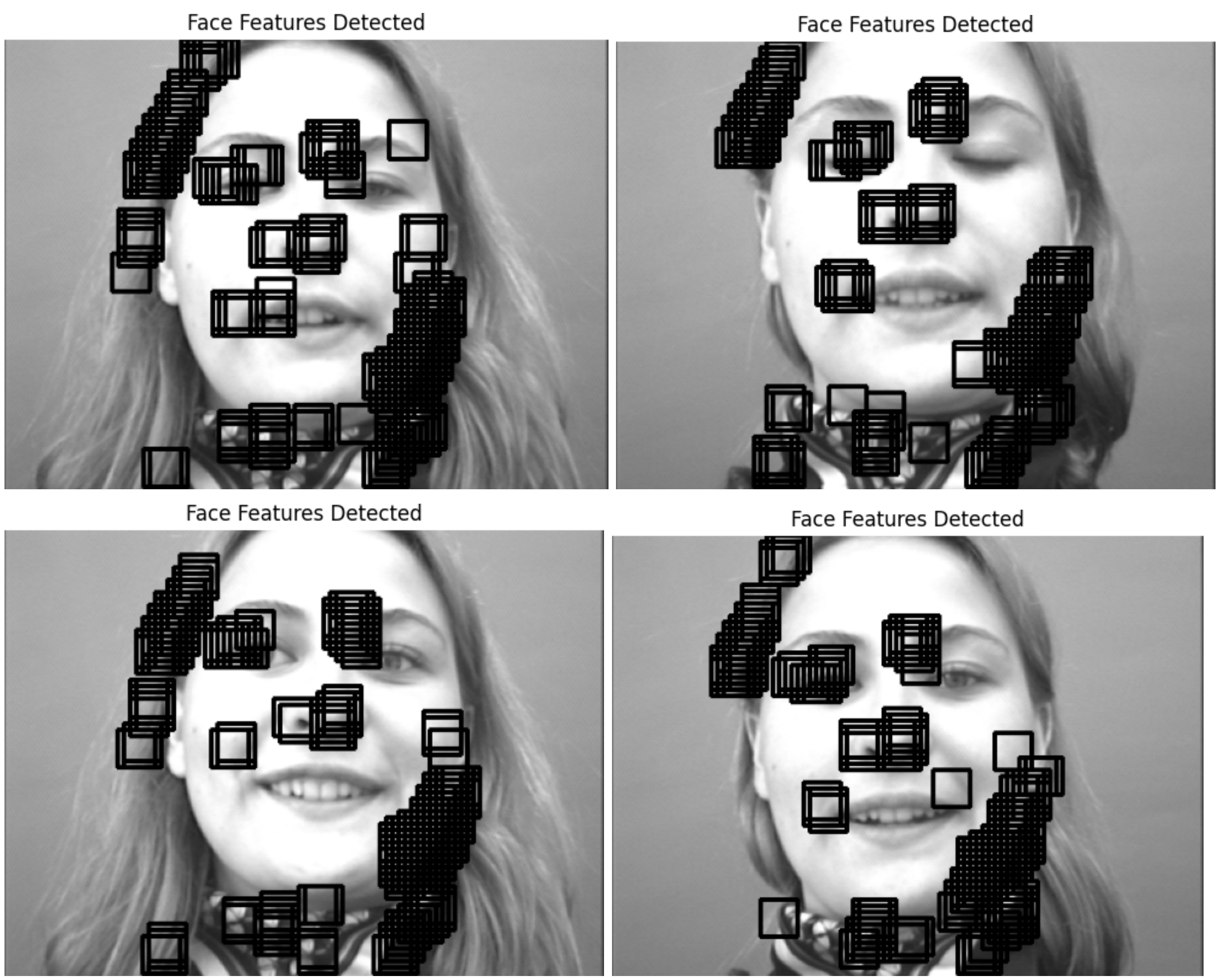
Conclusion

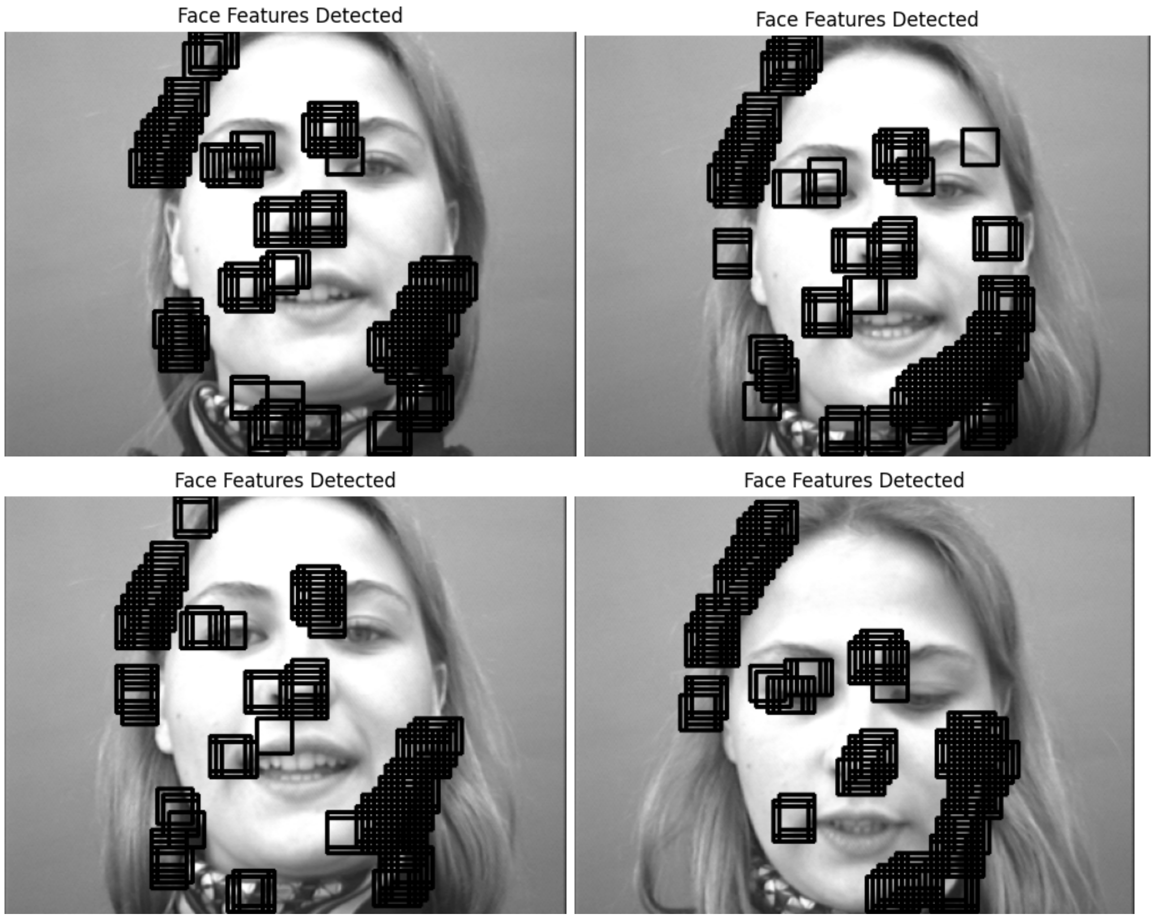
We have made substantial progress in implementing our facial detection algorithm. While we face challenges in optimizing and generalizing the AdaBoost classifier, we are confident in our ability to overcome these hurdles and achieve our goals. The next phase will focus on completing Milestone 3, starting multiscale detection, and preparing for final evaluation and reporting.

Appendix

Database	Description
BioID Face Database	Contains images of 23 individuals captured from multiple scales and perspectives.
Caltech 101 Dataset	Comprises diverse images categorized by their contents, spanning various object classes.

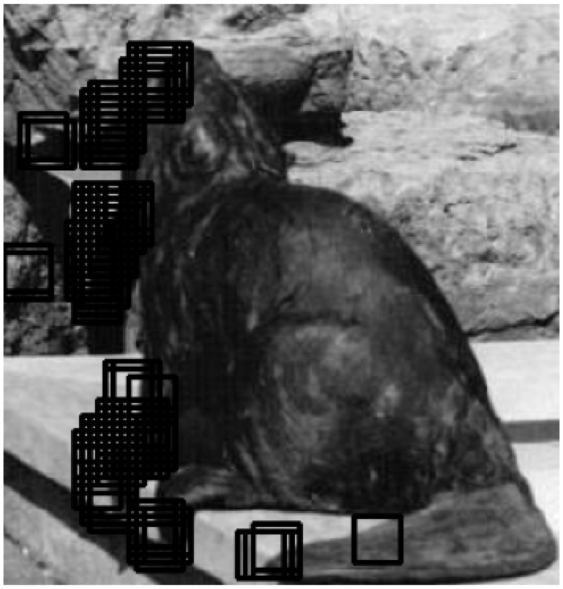
Datasets Utilized. This figure provides a list of the datasets used in our project, along with a brief description of their contents and how they were applied in our facial detection and recognition tasks.



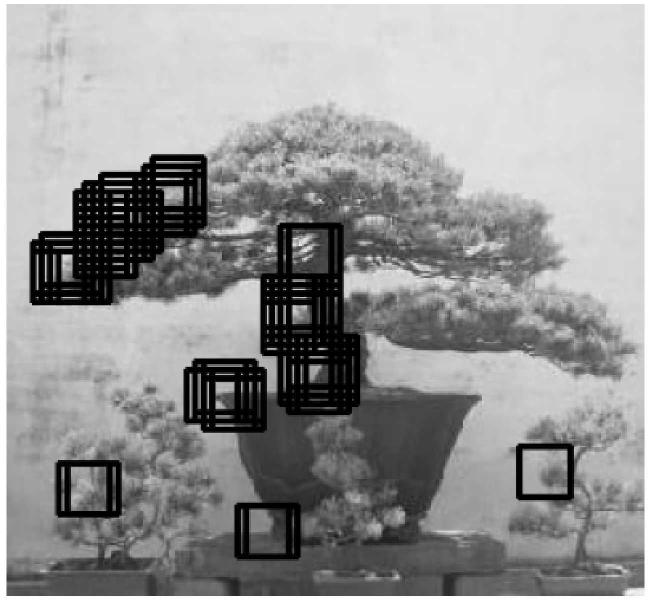


True Positive Haar Features Extracted from Testing Dataset. These figures highlight the Haar features that correspond to true positives in the testing dataset. These features represent the areas of the images that the algorithm has correctly identified as facial regions. The visualization demonstrates the algorithm's capacity to detect relevant facial features based on the Haar-like patterns extracted during the detection process.

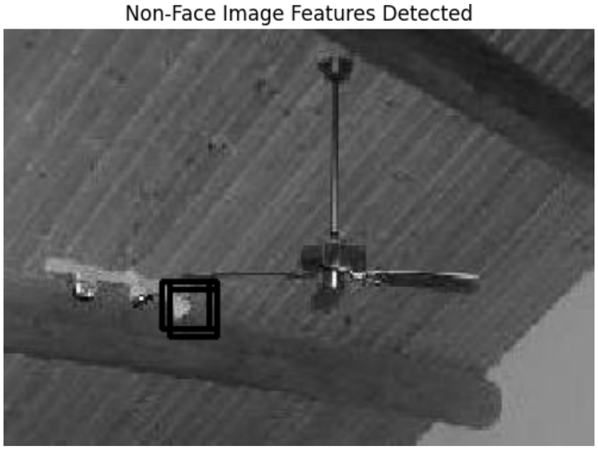
Non-Face Image Features Detected



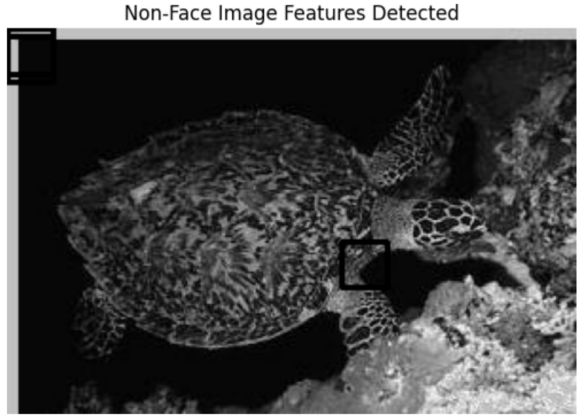
Non-Face Image Features Detected

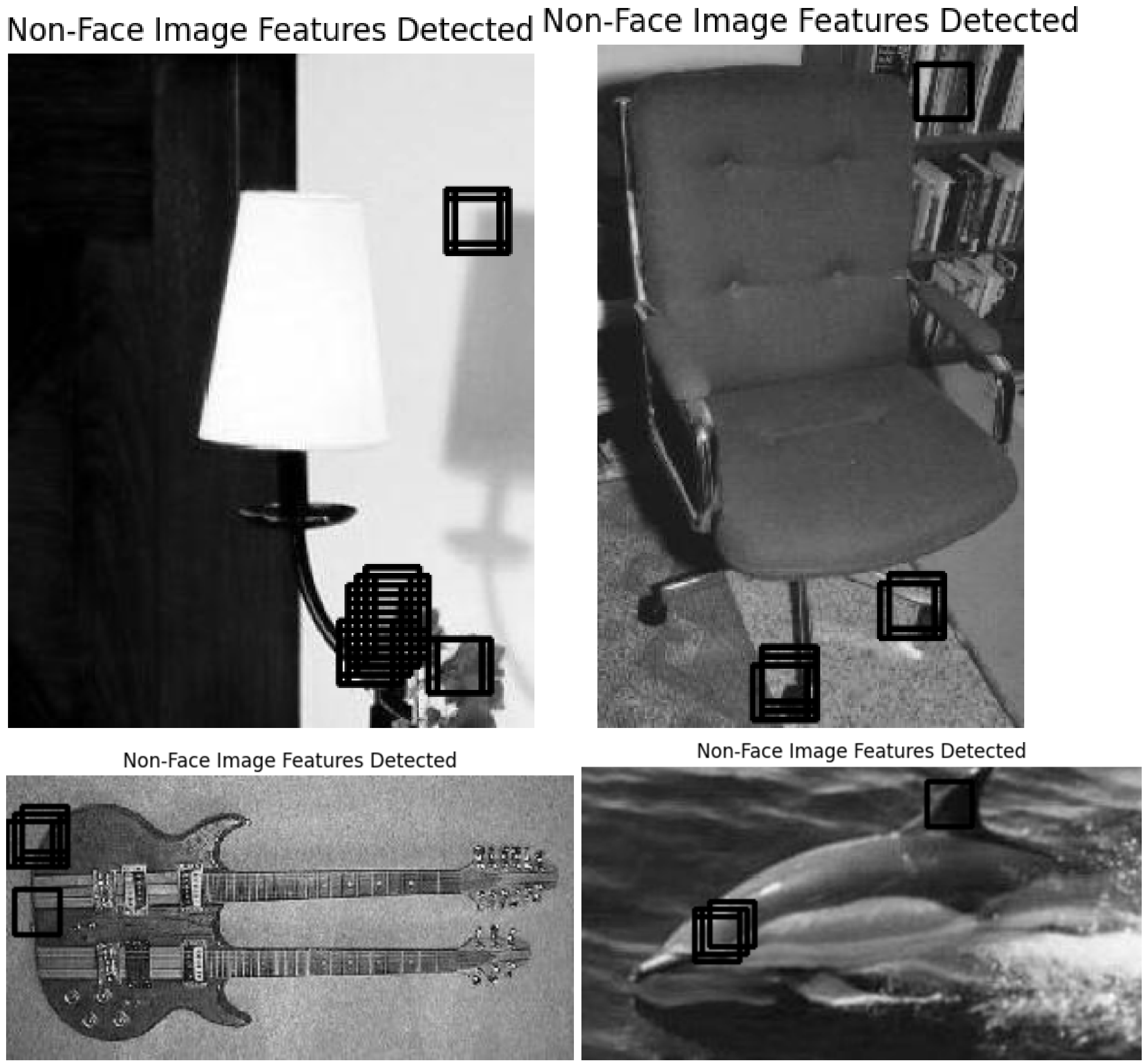


Non-Face Image Features Detected

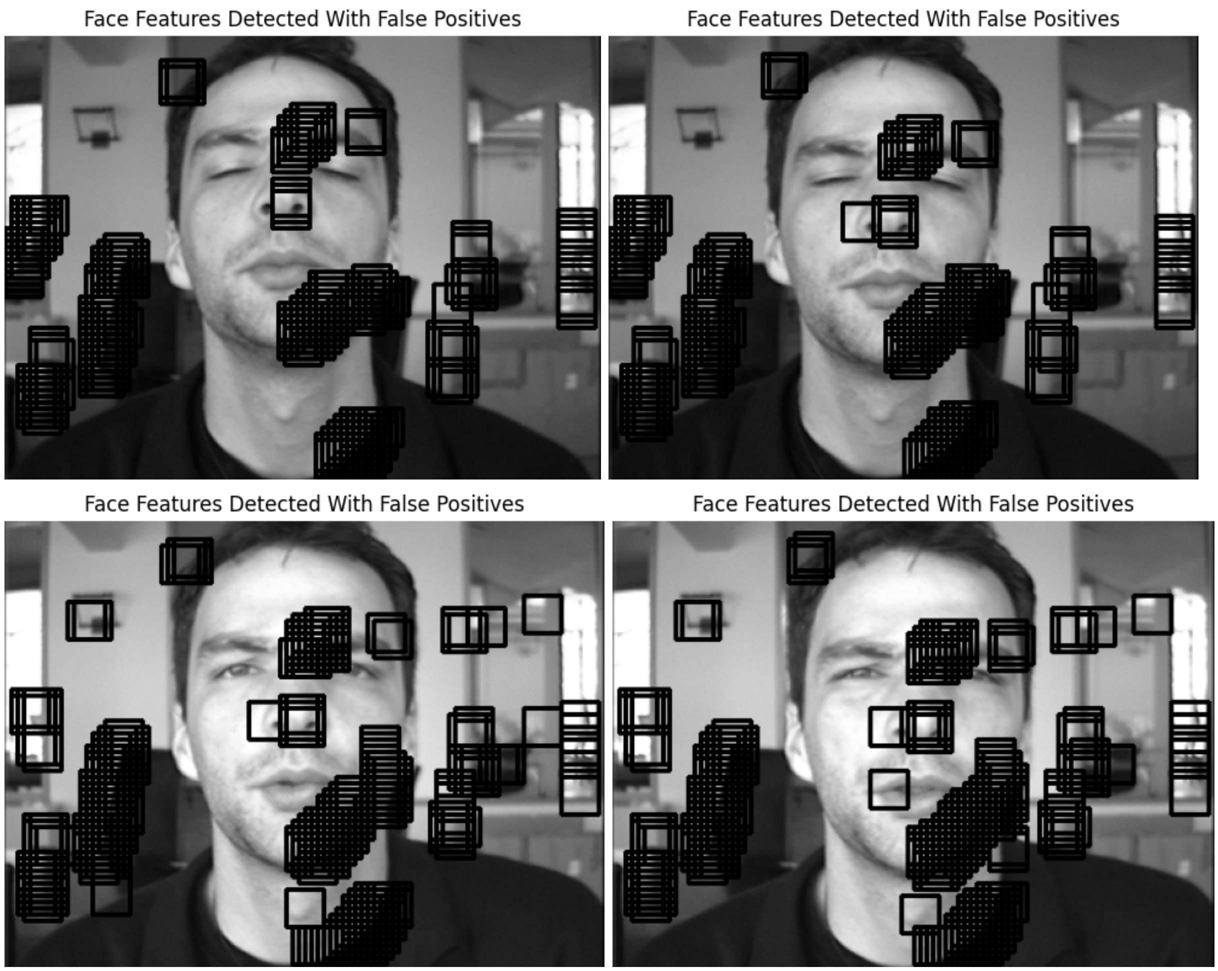


Non-Face Image Features Detected





Non-Face Image Features Detected. These figures illustrate the Haar features detected in non-face images from the dataset. While the algorithm identifies numerous features per image, these features exhibit low correspondence with facial patterns, resulting in low confidence scores. The visualization highlights areas where the algorithm's detection mechanism diverges from true positive results.



False Positives and True Positives in Testing Dataset. This figure showcases both false positive and true positive Haar features detected in the testing dataset. The visualization contrasts correctly identified facial features with erroneous detections in non-facial regions, providing insight into the algorithm's current performance and areas for improvement in classification accuracy.

Bibliography

BioID Face Database. Retrieved from https://www.bioid.com/face-database/.

Caltech 101 Database. Retrieved from https://www.kaggle.com/datasets/imbikramsaha/caltech-101/.

Freund, Y., & Schapire, R. E. (1997). *A Decision-Theoretic Generalization of On-Line Learning and an Application to Boosting.* Journal of Computer and System Sciences, 55(1), 119–139.

Integral Image Documentation. Retrieved from https://docs.opencv.org/master/d7/d8b/tutorial_py_face_detection.html.

OpenCV Library. Retrieved from https://opencv.org.

Viola, P., & Jones, M. (2001). Rapid Object Detection using a Boosted Cascade of Simple Features. *IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR)*.