Report on Real-Time Face Detection Using Viola-Jones Algorithm with Haar Features

Report File

Team 3

Team members:

1) Jambulov Javokhir 12204567 (Team leader)
2) Sairmurodov Elyor 12204556

Course name: Digital Image Processing Design

Professor: Kakani Vijay Date: 2023/10/31

Github link: https://github.com/JavokhirJambulov/DIP_class/tree/main/project1

Demo link: project1 team3 Javokhir and Elyor

Introduction

The objective of this project was to implement the Viola-Jones (VJ) face detection algorithm using OpenCV and train a Haar cascade classifier for real-time face detection. Additionally, we aimed to evaluate the performance of the face detector. This report summarizes the methodology, results, and observations from our work.

Methodology

We used the following approach for training our Haar cascade classifier:

- Utilized a graphical user interface (GUI) for training.

- Incorporated publicly available face-related datasets for training.

- Compiled a team-specific dataset, including all team members' faces in various orientations, scales, and illumination conditions.

- In total, we have used 600 positive and 3000 negative images, with a 1:5 ratio. And 20 stages.

- Used the trained Haar cascade classifier for testing and evaluation.

Team member contributions

Jambulov Javokhir (team leader)

Data Collection and Preparation: played a pivotal role in data collection and preparation for our face detection project. He was responsible for curating public face-related datasets and creating our team's self-made dataset, ensuring diversity in orientations, scales, and lighting conditions.

https://www.kaggle.com/datasets/arnaud58/landscape-pictures

https://www.kaggle.com/datasets/ashwingupta3012/human-faces

Training the Haar Cascade Classifier: took the lead in training the Haar cascade classifier. He fine-tuned the classifier using our dataset and the public datasets, ensuring the model's robustness in detecting faces under various conditions.

GitHub Management: managed the team's GitHub account efficiently. He uploaded all project-related files, including the source code, cascade classifier, report, self-made dataset, and demo video. His organizational skills ensured a smooth submission process.

Saidmurodov Elyor

Algorithm Implementation: was responsible for implementing the Viola-Jones algorithm using OpenCV. He wrote the code to load the trained classifier and perform real-time face detection. His coding skills were crucial to the project's success.

Testing and Evaluation: took the lead in testing and evaluating the face detection system. He devised test cases to determine the minimum and maximum detectable sizes of faces and measured the average detection time. His work provided valuable insights into the system's performance.

Documentation: was responsible for maintaining detailed documentation of the project. He created a comprehensive report that described our work, including the training process and the challenges faced during the project.

Testing & Evaluation

For testing and evaluation, we followed these parameters:

- Image size: 1024 x 768 pixels.

- Problem 1: Determined the minimum detectable size of faces.

- Problem 2: Identified the maximum detectable size of faces.

- Problem 3: Calculated the average detection time per image and per face.

For problems 1 and 2, we used the image to test the minimum and maximum size of the faces, but fr problem 3 we used OpenCV's real-time video capture. **All the codes and results can be found in the demo and in Github**

Training

Our training phase resulted in a well-constructed Haar cascade classifier, incorporating both public datasets and our team's self-made dataset. This comprehensive dataset allowed for robust face detection, even under challenging conditions.

Performance Evaluation

We observed the performance of our face detector in the presence of six common challenges in object detection:

1. Viewpoint Variation: Our detector showed robustness in detecting faces from various angles.

2. Deformation: The detector could handle facial deformations to some extent but struggled with extreme deformations.

3. Occlusion: The detector demonstrated good performance even in the presence of partial occlusions.

4. Illumination Conditions: Our detector was resilient to varying lighting conditions.

5. Cluttered or Textured Background: The presence of cluttered backgrounds did not significantly affect the detector's accuracy.

6. Intra-class Variation: The detector showed some limitations in distinguishing between individuals with very similar facial features.

Difficulties that we have faced during the project

1) Data Collection and Quality: we have faced challenges in collecting and curating face datasets, particularly the self-made dataset. Capturing team members' faces in various orientations, scales, and lighting conditions was time-consuming and led to quality variations in the data.
2) Training the Haar Cascade Classifier: we have encountered difficulties in fine-tuning the classifier parameters and managing the long training times. Balancing the trade-off between detection accuracy and training time can be a complex task.
3) When we first trained the cascade, there weren't enough positive images because we trained only 100 positives. But the results weren't accurate enough. Then we trained the model on 600 positives and 3000 negatives and got really accurate results.
4) One more thing to mention, increasing the buffer size from 1024 to 5120 rapidly improved our training time, which went down from 3 hours to 1 hour.

Conclusion

In conclusion, our project successfully implemented the Viola-Jones face detection algorithm using Haar features. We trained a robust Haar cascade classifier and evaluated its performance under various challenges. Our work represents a significant step towards real-time face detection and provides valuable insights into the Viola-Jones algorithm's capabilities and limitations.