2D/3D Facial Feature Verification System

Mentor: Murali Subbarao
Zyad Gomaa, Tyler Caffery, Brendan Vuraldor, Ammad Mehdi

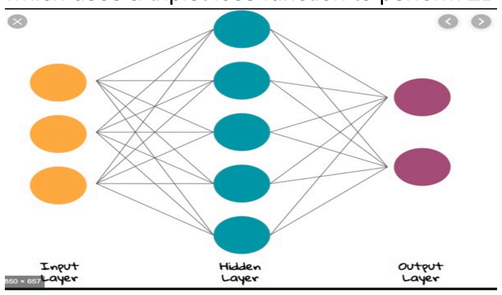
Stony Brook University
Electrical and Computer Engineering

Summary

Problem Statement:
In the past few years several home security cameras have hit the market and are loved by the consumer base. However, these cameras often simply alert a user of motion allowing them to then look through the camera via an app, and when they do more than this it is only using standard 2D image capture. This method does not provide very much security and can easily be spoofed.

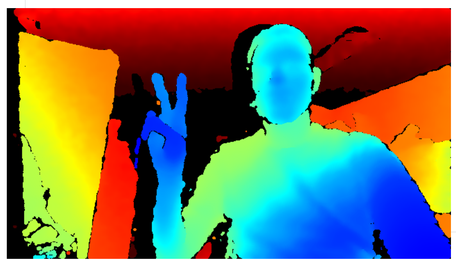
Solution Statement:
Through the integration of live video streaming, and facial recognition using both 2D and 3D a user will have real time activity of who exactly is outside their home in real time, and spoofing will be solved. The same technology could be used in the future for object detection outside the home.

Machine Learning:
Machine learning is the process of using data to drive solutions to problems in ways humans either cannot, or do not want to take the time to do. This data is pushed through statistical models and algorithms which use inference to find predictions, numerical outcomes, or classifications. The Convolutional Neural Network is the most used architecture for image processing, as it filters and prepares the data within the architecture. This architecture can be cascaded, or layered, to more efficiently break down images. This algorithm was used for facial detection, combined with facial embeddings which uses a triplet loss function to perform 2D recognition.

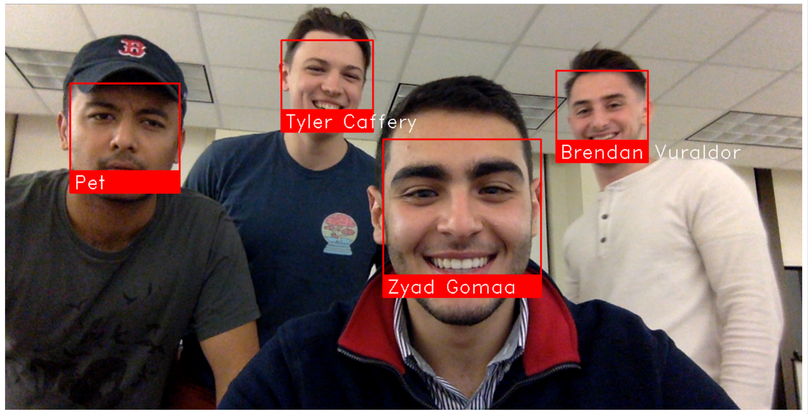


$$\|f(x_i^a) - f(x_i^p)\|_2^2 + \alpha < \|f(x_i^a) - f(x_i^n)\|_2^2$$

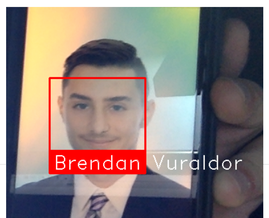
Intel Realsense D435i:
The Intel Realsense D435i acts as a standard webcam but has the addition of stereo image and an infrared projector, allowing for the capture of depth data in a similar fashion of human vision. Intel also released an SDK, a viewer application, and the Realsense API which uses Python, C++, or MATLAB.

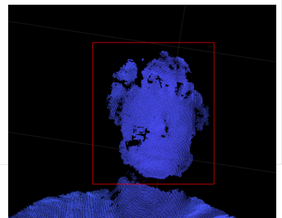


2D/3D Facial Recognition



- Implemented using pretrained cascaded CNNs to detect the facial features of nose, eyes, and the corners of the mouth. The coordinates are returned and used to outline with a rectangle where a face is within the image.
- Next, a database of peoples' faces who a user wants to recognize is composed of a unique vector of numbers called a face embedding. When an image or video is passed through the algorithm, it breaks down the detected face into a new embedding and uses Euclidean distances to compare the new face to any faces in the database, looking for a minimum distance to be a match.
- If a match is found, the face is recognized, and the name is displayed on the image. This math was implemented using pretrained models in an API.
- This method is prone to spoofing (pictured below), where someone could hold an image of a person in front of the camera, and it will be recognized as the person standing there when they are not, which could cause a security breach
- This is solved using 3D data capture of the Intel Realsense D435i. A point cloud is created of a user's face. This is then filtered and aligned to accommodate for different angles of capture, environments, and distance from the camera. All null points are then removed, and the image is cropped for unified formatting across the database.
- This is done on every user to create a database of registered users. When a new face is scanned, the Euclidean distances are compared to registered users, and if there is a match within a specified tolerance the output is that person's name. This process was used both to verify the 2D output, and to ensure a person was standing in real time in front of the camera.





Results and Discussion

In the end, integrating such new technology into an application was more difficult than initially considered. Most problems encountered were hard to find solutions to because the computer vision community was lacking experience with the concepts. However, through lots of research, some solutions were found as seen above. Starting with a base foundation of machine learning concepts and mathematical understanding was extremely beneficial in adapting to future concepts and problems.

We were successful at implementing and understanding the theory of 2D facial detection and recognition but fell a bit short when processing 3D data. The concepts were well understood, but the coding implementation proved challenging given the circumstances, and a proper database of registered users was not able to be created. However, using the pointcloud method, Brendan was able to have his face recognized first in 2D and verified with 3D data as a registered user inconsistently.

We successfully filtered input images and aligned them to a grid, but holes of imperfect image capture are still too frequently present, and the image cropping must be done manually rather than with code. Handling null points seems to be the biggest issue were we to move forward, and more time, testing, and users would be needed in the database to fully finish the system.

Glossary

•**Pretrained Network**: A machine learning network which has been trained on a large and reliable labeled dataset.
CNN: Convolutional Neural Network. A machine learning architecture which uses convolutional layers to filter images into important features.
FaceNet: The pretrained 2D facial recognition algorithm used, which uses deep CNNs to recognize faces then converts them into face embeddings.
Face Embedding: A 128-length vector of numbers representing a person's face.
Triplet Loss Function: A loss function which compares an anchor image of a person to 2 other images: a second image of the same person and an image of a different person. The Euclidean distances of face embeddings are taken to realize a match in the database.
Point Cloud: A 3D mesh of a face that can be aligned to a coordinate system.
Null Point: A datapoint that contains no data, or is a hole in the capture.

Acknowledgements

Professor Subbarao, Professor Dhadwal, and the SBU EE/CE Department

Secure The Crib