Faical recognition on AWS

by

Kaung Htet Cho

A Project Submitted in Partial Fulfillment of the Requirements for the
Cloud computing course

Examination Committee: Prof. A (Chantri Polprasert)

Nationality: Myanmar
Degree: Mechatronics and Machine Intelligence
ISE, AIT
Thailand

Asian Institute of Technology
School of Engineering and Technology
Thailand
March 2024

AUTHOR'S DECLARATION

I, Kaung Htet Cho, declare that the research work carried out for this project was in accordance with the regulations of the Asian Institute of Technology. The work presented in it are my own and has been generated by me as the result of my own original research, and if external sources were used, such sources have been cited. It is original and has not been submitted to any other institution to obtain another degree or qualification. This is a true copy of the thesis including final revisions.

Date: March 7, 2024

Name: Kaung Htet Cho

ACKNOWLEDGEMENTS

I would like to thank Professor Chantri Polprasert for for his invaluable guidance and support throughout the development of this project.

ABSTRACT

This project aims to develop a facial recognition-based access control system utilizing Amazon Web Services (AWS) Rekognition to enhance security measures by accurately identifying and verifying employees' identities in real-time. It involves implementing an efficient registration process to enroll employees' facial images into the system, followed by the creation of a robust authentication mechanism for access attempts. Integration with Amazon API Gateway facilitates user interaction, providing a user-friendly interface for registration, authentication, and access control functionalities. Evaluation of the system's performance and accuracy in real-world scenarios ensures reliability and scalability, offering organizations a comprehensive solution to enhance security and streamline access control processes with cloud-based infrastructure.

CONTENTS

	Page
ACKNOWLEDGEMENTS	**iii**
ABSTRACT	**iv**
CHAPTER 1 INTRODUCTION	**1**
1.1 Background of the Study	1
1.2 Statement of the Problem	1
1.3 Objectives	1
CHAPTER 2 Literature Review	**3**
2.1 Facial Recognition	3
2.2 AWS Rekognition	3
2.2.1 AWS Rekognition - index-faces API	4
CHAPTER 3 METHODOLOGY	**6**
3.1 Methodology	6
3.1.1 Data Collection and Preprocessing	6
3.1.2 Model Selection and Integration	6
3.1.3 Registration Process	6
3.1.4 Authentication Mechanism	6
3.1.5 Integration with EC2 server	7
CHAPTER 4 WELL-ARCHITECTED FRAMEWORK	**8**
4.1 Security foundations	8
4.2 Operational excellence	8
4.3 Reliability	8
4.4 Performance efficiency	8
4.5 Cost optimization	8
4.6 Sustainability	9
REFERENCES	**10**
APPENDICES	**11**
CHAPTER A COST ESTIMATION	**12**
A.1 EC2 t2.micro instance	12
A.2 AWS lambda function	12
A.3 AWS rekognition	13

A.4 DynamoDB 13
A.5 S3 bucket 14

CHAPTER 1
INTRODUCTION

1.1 Background of the Study

In today's digital landscape, secure access to corporate premises is crucial, yet traditional methods like swipe cards or PIN codes often lack robust security. Facial recognition technology, leveraging biometric authentication, offers a promising solution. With advancements in machine learning, facial recognition systems have become more accurate. Amazon Web Services (AWS) Rekognition provides powerful facial recognition capabilities, enabling easy integration into applications. This study proposes developing a facial recognition-based access control system using AWS Rekognition to accurately identify and verify employees' identities in real-time, streamlining access control processes while enhancing security measures and user experience.

1.2 Statement of the Problem

The problem addressed in this project is the need for a secure and efficient access control system for corporate premises. Traditional access control methods like swipe cards or PIN codes often lack robust security and are prone to unauthorized access and security breaches. Additionally, manual verification processes can be time-consuming and error-prone. There is a growing demand for advanced technologies that can enhance security measures while streamlining access control processes. In this context, the problem is how to develop a facial recognition-based access control system using Amazon Web Services (AWS) Rekognition that accurately identifies and verifies employees' identities in real-time, improving security measures and user experience while addressing the limitations of traditional access control methods.

1.3 Objectives

The objective of this project is to develop a facial recognition system that overcomes the limitations of traditional access control methods by leveraging Amazon Web Services (AWS) Rekognition. By utilizing advanced facial recognition algorithms, the system aims to accurately identify and verify employees' identities in real-time, enhancing security measures and streamlining access control processes. The key objectives of the project include:

- Integrating AWS Rekognition into the access control system pipeline to enable

facial recognition capabilities.

- Implementing mechanisms for efficient registration of employees' facial images into the system and generating unique identifiers.
- Creating a robust authentication mechanism for real-time facial recognition during access attempts.
- Integrating the system with Amazon API Gateway to provide a user-friendly interface for registration, authentication, and access control functionalities.

By achieving these objectives, the project aims to deliver an advanced facial recognition-based access control system that enhances security measures and improves the overall access control experience for organizations.

CHAPTER 2
Literature Review

Facial recognition technology has garnered significant attention in recent years due to its wide-ranging applications in security, surveillance, and biometric authentication systems. Several studies have explored the effectiveness and potential challenges associated with implementing facial recognition solutions, particularly those leveraging cloud-based services like Amazon Web Services (AWS) Rekognition.

2.1 Facial Recognition

Facial recognition, essential in various domains like security and authentication, has seen significant advancements with the adoption of deep learning techniques, particularly Convolutional Neural Networks (CNNs). These CNNs excel in learning intricate facial features from images, enabling accurate recognition across diverse conditions. Deep learning models automatically extract relevant features from data, eliminating the need for manual feature engineering and improving adaptability to different environments. Various architectures like Residual Networks (ResNets) and Inception Networks have been proposed, achieving state-of-the-art performance on benchmark datasets. Recurrent Neural Networks (RNNs) and Generative Adversarial Networks (GANs) also contribute to tasks like facial expression recognition and image generation. Despite progress, challenges such as data privacy and bias persist, underscoring the ongoing need for research and innovation in this field. Hu et al. (2015) Onyema et al. (2021)

2.2 AWS Rekognition

AWS Rekognition, a cloud-based image and video analysis service provided by Amazon Web Services (AWS), plays a significant role in advancing facial recognition technology. By leveraging AWS Rekognition, developers gain access to powerful and scalable deep learning models specifically designed for tasks like face detection, face analysis, and face comparison. The service offers a comprehensive suite of features, including real-time face detection and recognition, age and gender estimation, emotion detection, and facial landmark detection. Moreover, AWS Rekognition provides high accuracy and reliability, even in challenging conditions such as low light or occlusions. With its ease of integration, developers can quickly incorporate facial recognition capabilities into their applications without the need for extensive expertise in machine learning or computer

vision. Additionally, AWS Rekognition's robust APIs and SDKs allow for seamless integration with other AWS services, enabling developers to build sophisticated and scalable facial recognition systems with ease. Overall, AWS Rekognition enhances facial recognition technology by providing developers with access to state-of-the-art deep learning models, scalability, reliability, and ease of integration, thereby accelerating the development and deployment of facial recognition applications. Rafael, Kusuma, et al. (2020)

2.2.1 AWS Rekognition - index-faces API

Amazon Rekognition's index-faces API plays a crucial role in the underlying process of facial recognition, utilizing advanced deep learning techniques to extract and index facial features from images. While specific details of the algorithm are proprietary to Amazon, the index-faces API is built on state-of-the-art convolutional neural networks (CNNs) that are trained on vast datasets of facial images. These CNNs are capable of detecting key facial landmarks, capturing facial attributes such as facial expressions, and encoding unique facial features into compact representations known as embeddings.

The index-faces API typically follows a multi-step process, starting with face detection, where the algorithm identifies and localizes faces within an image using sophisticated object detection techniques. Once faces are detected, the algorithm extracts discriminative features from each face region, such as the arrangement of facial landmarks, texture patterns, and color information. These features are then transformed into high-dimensional embedding vectors using neural network architectures optimized for feature extraction tasks.

Next, the extracted embeddings are indexed and stored in a database, enabling efficient retrieval and comparison during the identification process. The indexing process involves encoding each facial embedding with a unique identifier, allowing for fast and accurate retrieval of matching faces from the database. During face identification, the algorithm compares query embeddings with the indexed embeddings using similarity metrics such as cosine similarity or Euclidean distance. Matches are then ranked based on their similarity scores, with higher scores indicating closer matches.

Amazon Rekognition's index-faces API leverages the scalability and computational power of cloud infrastructure to process large volumes of images efficiently, making it suitable for a wide range of applications, including security, authentication, and content

moderation. While the exact implementation details of the index-faces API are proprietary, its underlying process is grounded in cutting-edge deep learning techniques, enabling robust and accurate facial recognition capabilities. Leonor Estévez Dorantes, Bertani Hernández, León Reyes, and Elena Miranda Medina (2022)

CHAPTER 3
METHODOLOGY

3.1 Methodology

3.1.1 Data Collection and Preprocessing

- Gather facial images of company employees for training and testing purposes.
- Ensure that the images cover a diverse range of facial expressions, angles, and lighting conditions.

3.1.2 Model Selection and Integration

- Evaluate various facial recognition models available in AWS Rekognition, considering factors such as accuracy, speed, and scalability.
- IndexFaces API in AWS Rekognition offers automated face printing, high accuracy, scalability, ease of integration, and cost-effectiveness, making it a suitable choice for facial recognition tasks in various applications.
- Integrating storing SearchFacesByImage API from AWS rekognition can find index faces in DynamoDB and retrieving them can have efficiency, scalability, cost-effectiveness, reliability, and seamless integration with other AWS services, making it a powerful solution for facial recognition applications.

3.1.3 Registration Process

- Implement a registration mechanism to enroll employees' facial images into the system.
- Implement AWS KMS service to encrypted the images that will be stored in S3 for security.
- Utilize AWS Rekognition's APIs to extract facial features and generate unique identifiers (Rekognition IDs) for each employee.
- Store the Rekognition IDs along with employees' names in a DynamoDB database for future reference.

3.1.4 Authentication Mechanism

- Develop an authentication mechanism for real-time facial recognition during access attempts.
- Configure Lambda functions to trigger AWS Rekognition's face matching capa-

bilities, comparing captured facial images with the stored database.

- Implement logic to grant access if a match is found or deny access if no match is detected.

3.1.5 Integration with EC2 server

- Integrate the facial recognition system with EC2 server backend with flask to provide a user-friendly interface for interaction.
- Design the model to handle registration, authentication, and access control functionalities, ensuring seamless communication between components.

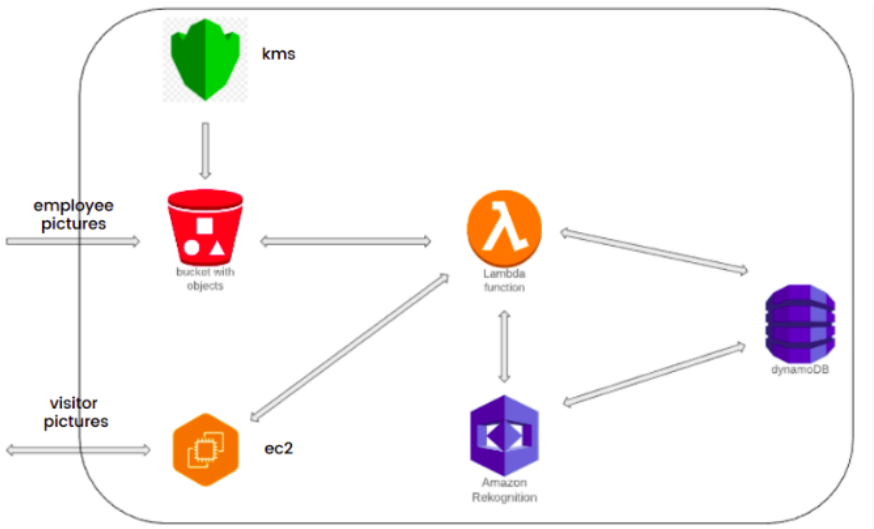


Figure 3.1

Overall Architecture of the System

CHAPTER 4
WELL-ARCHITECTED FRAMEWORK

4.1 Security foundations

- Implement robust identity and access management (IAM) policies to control access to AWS resources used in the project, following the principle of least privilege.
- Encrypt data in transit and at rest, including facial images and other sensitive data, using AWS Key Management Service.

4.2 Operational excellence

- Try to implement AWS CloudFormation template to declare aws services such as S3 bucket, define security groups and lambda functions. This can be well versioning and documented.

4.3 Reliability

- Design the system with redundancy and fault tolerance in mind, utilizing multiple Availability Zones or Regions as necessary.

4.4 Performance efficiency

- AWS CloudWatch will be part of the monitoring system, analyzing the user workload and system performance speed in order to adjust the compute service that is compatible with the demand. Can also set an alarm to remind the usage of our available service when the threshold is reach and scale according to the needs.
- Configure Lambda functions to trigger AWS Rekognition's face matching capabilities, comparing captured facial images with the stored database.
- Implement logic to grant access if a match is found or deny access if no match is detected.

4.5 Cost optimization

- Implement cost-effective architectures by leveraging serverless services like AWS Lambda.
- Implement s3 lifecycle policies to move data that isn't accessed much to cheaper storage options like S3 Infrequent Access or others.

4.6 Sustainability

- Develop an authentication mechanism for real-time facial recognition during access attempts.
- Configure Lambda functions to trigger AWS Rekognition's face matching capabilities, comparing captured facial images with the stored database.
- Implement logic to grant access if a match is found or deny access if no match is detected.

REFERENCES

Hu, G., Yang, Y., Yi, D., Kittler, J., Christmas, W., Li, S. Z., & Hospedales, T. (2015, December). When face recognition meets with deep learning: An evaluation of convolutional neural networks for face recognition. In *Proceedings of the ieee international conference on computer vision (iccv) workshops.*

Leonor Estévez Dorantes, T., Bertani Hernández, D., León Reyes, A., & Elena Miranda Medina, C. (2022). Development of a powerful facial recognition system through an api using esp32-cam and amazon rekognition service as tools offered by industry 5.0. In *2022 the 5th international conference on machine vision and applications (icmva)* (pp. 76–81).

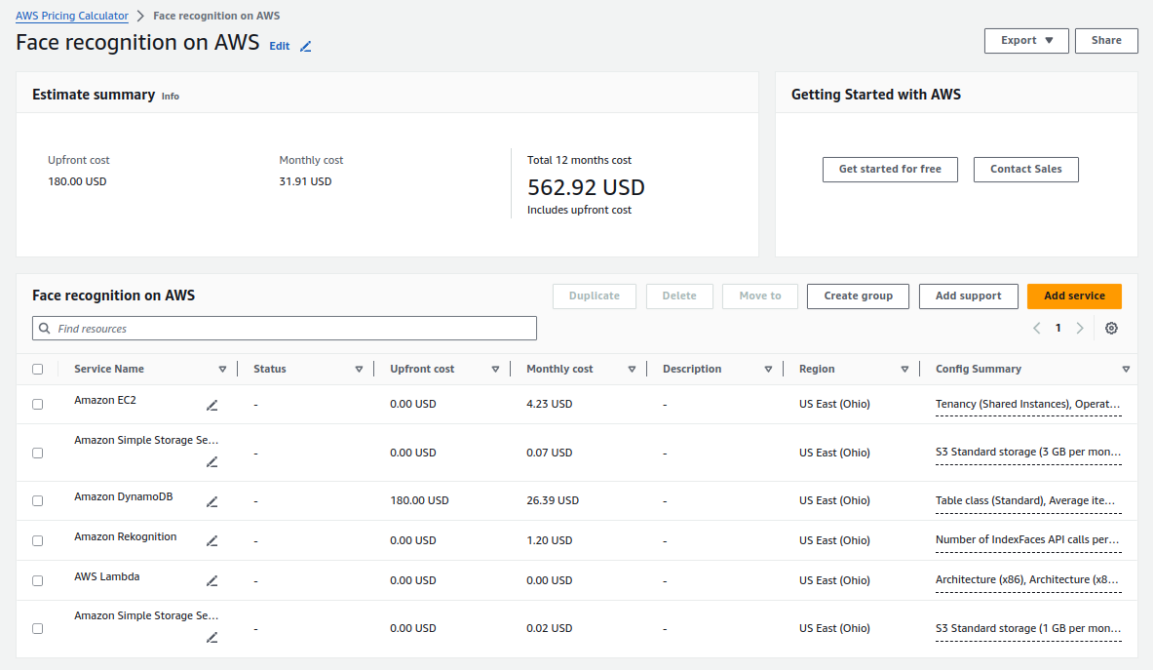
Onyema, E. M., Shukla, P. K., Dalal, S., Mathur, M. N., Zakariah, M., Tiwari, B., et al. (2021). Enhancement of patient facial recognition through deep learning algorithm: Convnet. *Journal of Healthcare Engineering*, *2021*.

Rafael, G., Kusuma, H., et al. (2020). The utilization of cloud computing for facial expression recognition using amazon web services. In *2020 international conference on computer engineering, network, and intelligent multimedia (cenim)* (pp. 366–370).

APPENDICES

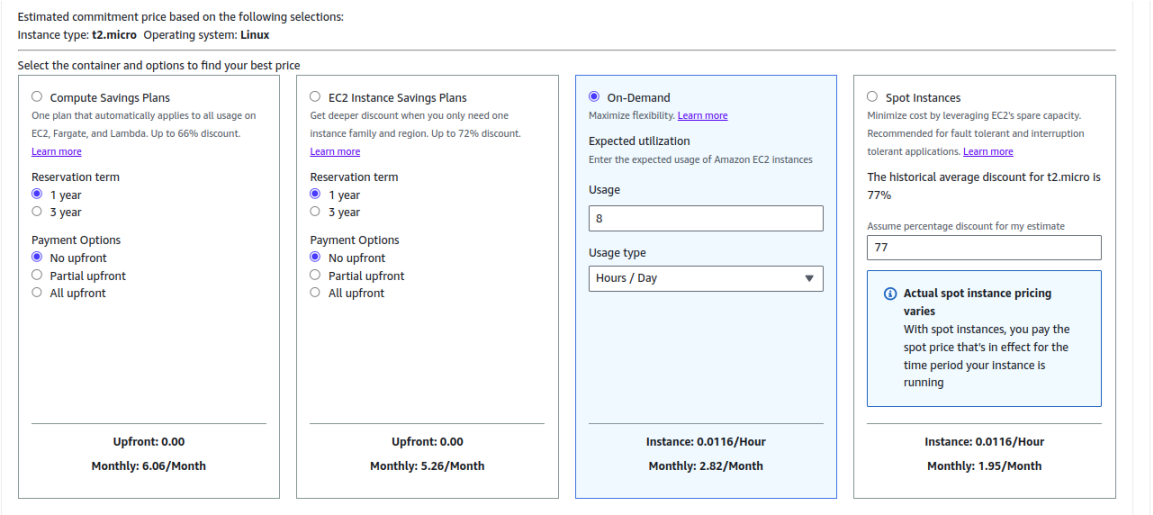
APPENDIX A

COST ESTIMATION



Note. All prices are calculated without considering the free tier eligibility

A.1 EC2 t2.micro instance



- Chose t2.micro instance as high performance is not needed.
- Cost per day: $0.0116/hour * 1hour/day$ =0.0116/day
- Monthly cost: $0.0116/day * 30days/month$ =0.348/month

A.2 AWS lambda function

Lambda functions are priced based on the number of requests and the duration of execution.

▼ Show calculations

Unit conversions

Number of requests: 100 per day * (730 hours in a month / 24 hours in a day) = 3041.67 per month

Amount of ephemeral storage allocated: 512 MB x 0.0009765625 GB in a MB = 0.5 GB

Pricing calculations

3,041.67 requests x 200 ms x 0.001 ms to sec conversion factor = 608.33 total compute (seconds)

3,041.67 requests x 0.0000002 USD = 0.00 USD (monthly request charges)

0.50 GB - 0.5 GB (no additional charge) = 0.00 GB billable ephemeral storage per function

Lambda costs - Without Free Tier (monthly): 0.00 USD

- Number of requests / day = 100
- Time of exceution = 200 ms
- Total amount = 0.000 USD

A.3 AWS rekognition

For the IndexFaces API, it charged based on the number of images processed and SearchFaces-ByImage API,irged based on the number of images compared.

▼ Show calculations

200 index faces + 1,000 search users by image = 1,200 total API calls (group 1)

Tiered price for: 1,200 API calls

1,200 API calls x 0.001 USD = 1.20 USD

Total tier cost = 1.20 USD for images group 1

Image analysis cost group 1 (monthly): 1.20 USD

Image analysis cost group 2 (monthly): 0 USD

Image analysis cost for Image Properties (monthly): 0 USD

Image analysis cost (monthly): 1.20 USD

Face metadata storage cost (monthly): 0 USD

Image pricing (monthly): 1.20 USD

- Number of calling IndexFaces API = 200
- Number of calling SearchFacesByImage = 1000
- Total amount = 1.2 USD

A.4 DynamoDB

DynamoDB pricing can be complex due to its various factors and pricing options.

- Choose mainly focus on the storage size for storing RekognitionIDs and Names.
- Storage 1 GB

A.5 S3 bucket

S3 is only used for storing images for registration purposes.

- Tiered price for: 1 GB ==> 1 GB x 0.023 USD = 0.02 USD