

**REKOGNITION**

*Implementación de una solución de proctoring con análisis biométrico utilizando REKOGNITION*

Brayan Eduardo Duarte / Automatización

09/25/2025

# Resumen

Este proyecto desarrolla un sistema de validación de identidad y prevención de fraude en exámenes universitarios utilizando *Amazon Rekognition*, servicio de inteligencia artificial de AWS especializado en análisis de imágenes y videos.

La solución implementada permite:

* Comparar el rostro del estudiante con su foto oficial.
* Verificar que solo haya una persona en la imagen.
* Detectar obstrucciones en la cámara (manos, paredes, oscuridad).
* Identificar intentos de fraude con fotos estáticas o pantallas.
* Generar alertas automáticas en caso de anomalías.

El desarrollo se realizó en Python con el SDK oficial de AWS (*boto3*), integrando funciones de detección de rostros, comparación de imágenes y análisis de etiquetas. Se diseñó además una matriz de pruebas con múltiples escenarios para validar la eficacia del sistema.

Con esta implementación, la universidad obtiene una herramienta escalable que fortalece la seguridad en los procesos de evaluación, contribuyendo a la reducción del fraude académico y aumentando la confiabilidad de los exámenes en entornos digitales y presenciales.

1. Introducción

2.1. ¿Qué es Amazon Rekognition?

Amazon Rekognition es un servicio de inteligencia artificial de Amazon Web Services (AWS) que permite analizar imágenes y videos mediante algoritmos de aprendizaje profundo. Ofrece capacidades como detección y reconocimiento de rostros, identificación de objetos, escenas y actividades, análisis de emociones, extracción de texto (OCR) y moderación de contenido sensible.

Su principal ventaja es que no requiere entrenar modelos desde cero, sino que proporciona APIs listas para usar (detect\_faces, compare\_faces, detect\_labels, detect\_text, entre otras), lo que facilita la integración en aplicaciones reales. Además, puede escalar para procesar grandes volúmenes de datos y, en casos específicos, permite entrenar modelos personalizados mediante Rekognition Custom Labels.

2.2. Problema a resolver: fraude en exámenes

En los exámenes académicos, especialmente en entornos digitales o con supervisión remota, existe un alto riesgo de fraude. Algunos ejemplos incluyen:

* Suplantación de identidad mediante el uso de otra persona.
* Presencia de múltiples individuos en la misma evaluación.
* Uso de fotografías impresas o pantallas para engañar la cámara.
* Intentos de ocultar la cámara (con manos, objetos o cubriéndola).
* Colocación de una imagen estática frente al lente para evitar ser detectado.

Estos escenarios comprometen la validez del proceso de evaluación y afectan la integridad académica. Por ello, se requiere un sistema automatizado que valide la identidad del estudiante y detecte posibles anomalías en tiempo real o de manera periódica durante la prueba.

2.3. Objetivo del proyecto

Desarrollar un sistema automatizado de validación de identidad y prevención de fraude en exámenes universitarios utilizando Amazon Rekognition, que permita:

* Verificar la identidad del estudiante mediante comparación de rostro con una base de datos oficial.
* Detectar la presencia de más de una persona en la imagen.
* Identificar obstrucciones o anomalías que impidan la validación.
* Reconocer intentos de fraude mediante imágenes estáticas o impresas.
* Generar alertas automáticas para notificar irregularidades.

Con este objetivo, el proyecto busca fortalecer la seguridad en los procesos de evaluación y aumentar la confianza en los resultados obtenidos.

2.4. Alcance

El proyecto se enfoca en implementar un sistema de validación basado en imágenes capturadas periódicamente durante los exámenes. El alcance incluye:

* Implementación local en pruebas piloto con imágenes predefinidas.
* Desarrollo en Python utilizando el SDK oficial de AWS (boto3).
* Uso de APIs estándar de Rekognition para detección de rostros, comparación de identidad, análisis de etiquetas y validación de condiciones de seguridad.
* Matriz de pruebas con escenarios de fraude y obstrucción para evaluar el rendimiento del sistema.
* Integración conceptual con plataformas de proctoring, donde el sistema pueda actuar como módulo de validación de identidad y detección de irregularidades.

No se incluye en este alcance el análisis en tiempo real de video (que corresponde a Rekognition Video), ni el entrenamiento de modelos personalizados con Custom Labels, aunque se identifican como posibles mejoras futuras para ampliar la robustez del sistema.

1. Marco Teórico  
   3.1. Amazon Web Services (AWS)

Amazon Web Services (AWS) es la plataforma de servicios en la nube más utilizada a mundial. Ofrece más de 200 servicios que abarcan cómputo, almacenamiento, redes, bases de datos, inteligencia artificial, seguridad y analítica. Su modelo de pago por uso y su capacidad de escalabilidad la convierten en una opción atractiva tanto para empresas como para instituciones educativas.

3.2. Amazon Rekognition: descripción general

Amazon Rekognition es un servicio de visión por computadora basado en machine learning que permite analizar imágenes y videos sin necesidad de entrenar modelos desde cero. Sus principales ventajas son:

* Facilidad de integración: mediante el SDK (boto3 en Python) o la API de AWS.
* Escalabilidad: puede analizar desde pocas imágenes hasta grandes volúmenes de datos.
* Variedad de funciones: cubre detección de objetos, rostros, emociones, texto y contenido sensible.

Para este proyecto se utilizaron principalmente las siguientes funciones:

**3.2.1. detect\_faces**

Esta función permite identificar rostros en una imagen. Retorna información como:

* Cantidad de rostros detectados.
* Posición (coordenadas) del rostro en la imagen.
* Edad estimada en un rango (ej. 20-30 años).
* Género estimado (masculino o femenino).
* Estado emocional (feliz, triste, enojado, calmado, confundido, asustado, sorprendido, disgustado).
* Rasgos adicionales como presencia de gafas, barba, sonrisa.

En el proyecto se utiliza para:

* Validar que haya un rostro presente en la imagen.
* Detectar si hay más de una persona.
* Obtener el género como atributo de validación adicional.

**3.2.2. compare\_faces**

Permite comparar dos rostros: uno de referencia (por ejemplo, la foto oficial del estudiante) contra un rostro detectado en una imagen de entrada.

* Retorna el porcentaje de similitud entre ambos.
* Es fundamental para la verificación de identidad en este proyecto.
* Se establece un umbral de similitud (ej. 90%) para determinar coincidencia.

**3.2.3. detect\_labels**

Detecta objetos, escenas y conceptos dentro de una imagen.

* Ejemplos: “Person”, “Hand”, “Wall”, “Monitor”, “Dark”.
* Devuelve cada etiqueta con un porcentaje de confianza.
* En este proyecto se usa como apoyo para:
* Detectar obstrucciones en la cámara (mano tapando, cámara frente a una pared).
* Identificar intentos de fraude visual (ej. mostrar una pantalla en vez de la cara del estudiante).

**3.2.4. detect\_moderation\_labels**

Esta función detecta contenido explícito o inapropiado en imágenes (desnudos, violencia, drogas, armas).

* Aunque no es el enfoque principal del proyecto, puede ser útil en un futuro para reforzar la seguridad y evitar uso indebido de la cámara en contextos inapropiados.

**3.2.5. detect\_text (OCR)**

Permite extraer texto de imágenes.

* Se utiliza principalmente en casos donde es necesario leer contenido de carteles, documentos o pantallas.
* En este proyecto no se considera central, pero podría integrarse en fases futuras para validar si el estudiante muestra algún material no autorizado en cámara.

3.3. Limitaciones del servicio

Aunque Rekognition ofrece gran capacidad, tiene algunas limitaciones importantes:

* No identifica personas comunes por nombre, solo rostros contra referencias proporcionadas.
* La estimación de edad y género no es 100% precisa.
* No diferencia entre una persona real y una foto mostrada en pantalla (se requieren validaciones adicionales o Custom Labels).
* No detecta movimiento en imágenes estáticas (para eso se requiere analizar secuencias de imágenes o usar Rekognition Video).

3.4. Comparación con otros servicios relacionados (Textract, Comprehend, SageMaker)

* Amazon Textract: especializado en OCR avanzado para documentos (tablas, facturas, formularios).
* Amazon Comprehend: procesamiento de lenguaje natural (NLP), útil para analizar texto detectado.
* Amazon SageMaker: permite entrenar modelos de machine learning personalizados.
* Rekognition Custom Labels: permite entrenar un modelo de visión específico con tus propios datos (ej. detectar fotos en pantalla vs persona real).

1. Metodología y Desarrollo  
   4.1. Diseño de la solución  
   4.1.1. Diagrama de flujo de validación

Diagrama

El contenido generado por IA puede ser incorrecto.  
  
  
4.1.2. Descripción de cada módulo

* Módulo de captura: se encarga de tomar fotos periódicas.
* Módulo de detección de rostros (detect\_faces): analiza si hay personas y cuántas.
* Módulo de comparación (compare\_faces): valida la identidad del estudiante contra su foto oficial.
* Módulo de validación de género: añade un filtro de seguridad opcional.
* Módulo de obstrucción (detect\_labels): analiza si la cámara está tapada o mirando otra cosa.
* Módulo de movimiento: compara imágenes secuenciales para detectar intentos de fraude con fotos estáticas.
* Módulo de alertas: genera un aviso si alguna condición no se cumple.

4.2. Implementación técnica  
4.2.1. Configuración de AWS y credenciales

Para poder realizar la configuración de AWS se necesitan realizar los siguientes pasos:

1. En la consola AWS:

* Ve a IAM (Identity and Access Management).
* Crea un usuario nuevo:
  + Se le asigna un nombre personalisado
  + Y le indicamos que el tipo de acceso será de tipo Programmatic access (no utilizar acceso a consola).
* Asigna permisos:
  + Crea o selecciona un grupo con la política AmazonRekognitionFullAccess.
* Descarga el archivo .csv con el Access Key ID y Secret Access Key.

1. Instalar las herramientas necesarias en el PC:

Para poder ejecutar los comandos de AWS es necesario descargar el AWS-CLI, desde el siguiente link:

<https://docs.aws.amazon.com/cli/latest/userguide/getting-started-install.html>

4.2.2. Estructura del proyecto en Python  
4.2.3. Explicación de funciones principales  
4.3. Validaciones implementadas  
4.3.1. Validación de identidad  
4.3.2. Detección de múltiples rostros  
4.3.3. Validación de género  
4.3.4. Validación de obstrucciones  
4.3.5. Validación de foto estática (movimiento)

1. Casos de Uso y Escenarios de Prueba  
   5.1. Casos de uso definidos  
   5.2. Matriz de pruebas (checklist)  
   5.3. Evidencias gráficas (capturas de pantalla e imágenes)
2. Resultados y Discusión  
   6.1. Análisis de resultados obtenidos  
   6.2. Limitaciones encontradas  
   6.3. Posibles mejoras técnicas
3. Integración con Proctoring  
   7.1. Flujo de integración en exámenes en línea  
   7.2. Integración con plataformas educativas  
   7.3. Escalabilidad hacia validación en video en tiempo real

# Conclusiones

# Recomendaciones

# Referencias

# Apéndices 11.1. Código fuente simplificado 11.2. Configuración de entorno y dependencias 11.3. Ejemplos de respuestas JSON