

# Reconocimiento Facial

Para Ingreso a Espacios en una Universidad

Integrantes: Sebastián Arancibia  
Rodrigo Catalán  
Camila Salas  
Profesor: Augusto Lucero  
Auxiliar: Jorge Allendes

Fecha de entrega: 22 de mayo de 2024  
Santiago de Chile

# Resumen

A medida que avanza la tecnología surgen cada vez mas soluciones a grandes problemáticas, la seguridad es un tema muy delicado, ya que dada la contingencia es crucial tener un monitoreo de las personas que entran a ciertos espacio, tales como pueden ser ciertas instituciones como universidades y espacios en específicos de estas.

Es por ello que se busca a través de librerías y elementos tecnológicos ya existentes, en este caso se utilizara la librería de *Python mediapipe* y una cámara ESP32, para así llegar a una solución innovadora respecto a como hacer identificación de rostros de personas y a través de estas determinar si pertenecen o no a una base de datos con acceso a ciertos lugares. Esto permitiría que en lugar de tener que utilizar tarjetas para ingresar a espacios en la universidad, baste solo con hacer un escaneo al rostro y con este determinar el acceso, evitando problemas por haber perdido la tarjeta o haberla olvidado en la casa antes de salir.

La implementación de este sistema ofrecerá una oportunidad para agilizar el acceso a espacios dentro de una universidad, lo que le permite a cada departamento tener un mejor control de quienes ocupan sus inmediaciones, ya que todas las personas que quieran hacer ingreso deberán estar registrados en el sistema y no necesitaran ningún elemento adicional.

# Índice de Contenidos

1. Introducción	1
2. Marco teórico	2
3. Desarrollo	3
4. Carta Gantt	7
5. Evaluación económica	8
6. Conclusiones	9
7. Referencias	10

# Índice de Figuras

1. Modelo simplificado de la arquitectura utilizada en <i>Face Mesh</i> . . . . .	2
2. Diagrama del algoritmo de reconocimiento facial . . . . .	3
3. Etapas del Algoritmo de Detección . . . . .	4
4. Captura de funcionamiento de código con orientación y movimiento de rostro.	5
5. Captura de funcionamiento de código con muestra gráfica de métodos imple- mentados. . . . .	5
6. Diagrama funcional del proyecto. . . . .	6
7. Carta Gantt creada para el desarrollo del proyecto para 15 semanas. . . . .	7

# 1. Introducción

A medida que la tecnología va avanzando, la seguridad debe ir adaptándose a este cambio, por esto, se han ido creando diversas maneras de proteger la información o acceso que se tiene de manera digital. Lo que se puede llamar seguridad digital, se ve en empresas, establecimientos privados, aeropuertos, entre otros. De la mano, el procesamiento de imágenes es una herramienta poderosa para poder crear un sistema de acceso o seguridad para el ingreso a cualquier lugar, ya que se pueden utilizar distintas metodologías para detectar el rostro. Además, debido a la inseguridad que se vive como sociedad, es necesario crear un sistema robusto que sea capaz de permitir el acceso solo a las personas que corresponde.

Para poder realizar el ingreso a un espacio, los pasos a seguir son: crear una base de datos que funcione como registro de los usuarios que puedan ingresar al lugar, generar un sistema de reconocimiento de rostros en vivo para evitar problemas de seguridad, comparar rostro encontrado con base de datos y dar acceso o denegar el ingreso al lugar dependiendo de quién quiere ingresar. Lo anterior se realiza principalmente utilizando código en *Python* e implementos como cámara y microprocesador.

A lo largo del proyecto se han tenido que ir realizando cambios, debido a que la idea inicial que se tenía a desarrollar era muy general, por esto, se acotó el problema, y ha sido necesario cambiar métodos para poder utilizar mejores implementaciones.

## 2. Marco teórico

Se estará utilizando la librería *mediapipe* de *Google* [1], la cual proporciona diversas herramientas de *machine learning* para trabajar con audio y video de forma modular y eficiente. En particular, se trabajará con los modelos *Face Detection* y *Face Mesh*, los cuales se describen brevemente a continuación:

- **Face Detection:** Permite detectar rostros en fotos o videos de forma ultra-rápida. Su estructura está basada en *BlazeFace* [2], modelo que utiliza una red neuronal convolucional compacta para la extracción de características de la imagen, la cual fue optimizada para hacer un uso eficiente de GPU (*Graphics Processing Unit*), junto con una estrategia más estable para resolver empates en predicciones superpuestas.
- **Face Mesh:** Estima 468 puntos de referencia sobre un rostro en tiempo real, con los que se puede detectar el movimiento y la orientación del mismo. Este trabaja con el mismo modelo empleado en *Face Detection*, junto con otra red neuronal profunda que ubica y predice la ubicación de los puntos de referencia por medio de regresión. A continuación se muestra un modelo simplificado de la arquitectura empleada:

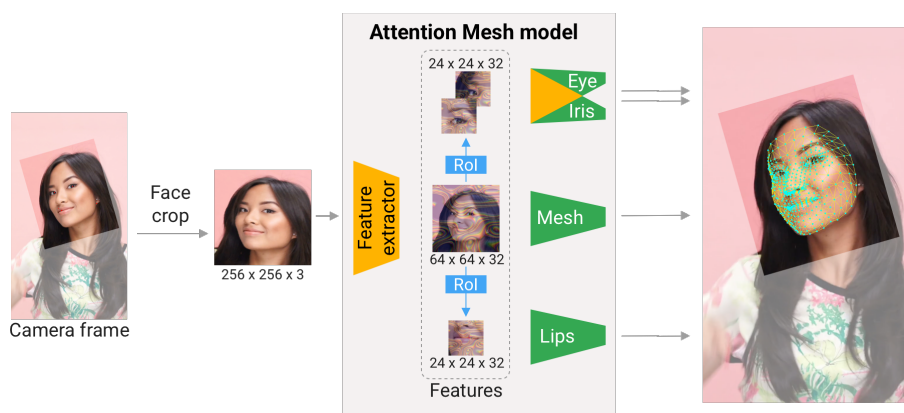


Figura 1: Modelo simplificado de la arquitectura utilizada en *Face Mesh*.

Para dar mayor dinamismo al proyecto se trabajara con una ESP32 Cam, la cual permite trabajar de manera inalámbrica con el centro de detección, puesto a que esta cámara trae consigo un microprocesador que le permite comunicarse vía WiFi o Bluetooth, en este caso se optara por trabajar con la opción de WiFi, ya que es un protocolo mas confiable para la comunicación entre dispositivos y así poder tener una transferencia de datos con menos error, ya que el WiFi tiene una tasa de transmisión de datos de 10Mbps [3]. Además esto nos va a permitir tener cierta distancia y autonomía del sistema, pues la cámara se puede ubicar a cualquier distancia del sistema siempre que tenga cobertura del Router, sin depender del largo de algún cable de Ethernet y/o USB. Otro motivo para trabajar con esta cámara es que solo necesita ser alimentada con 5V, lo que implica un consumo bastante bajo.

### 3. Desarrollo

En esta sección, se presentará como se ha ido desarrollando el proyecto y los resultados obtenidos en el camino.

El inicio del proyecto comenzó con la búsqueda de ideas para elegir cual se iba a realizar, entre las cuales se tenían:

- **Detección de rostros para sistema de seguridad:** trabajar en un sistema de seguridad para ingreso a un lugar, a partir de la detección de rostros.
- **Identificación de especies de plantas y animales:** Trabajar en un sistema que identifique automáticamente especies de plantas y animales a partir de imágenes. Lo anterior podría ser útil en aplicaciones de conservación y biología.

A partir de estas dos ideas, se decidió trabajar en la primera, la cual utiliza procesamiento de imágenes y envío de información.

Con la idea decidida, comenzó el proceso de investigación sobre otros métodos similares para poder guiar el desarrollo del proyecto. La información encontrada se encuentra de manera detallada en la sección “Marco teórico”.

A partir de la investigación previa, se creó un diagrama que presenta tres grandes partes del proyecto, como lo son: los implementos, características y salidas obtenidas. Este diagrama se puede ver en la Figura 2:



Figura 2: Diagrama del algoritmo de reconocimiento facial

Además, se detalla el algoritmo de detección a utilizar, el cual se puede observar en la Figura 3.

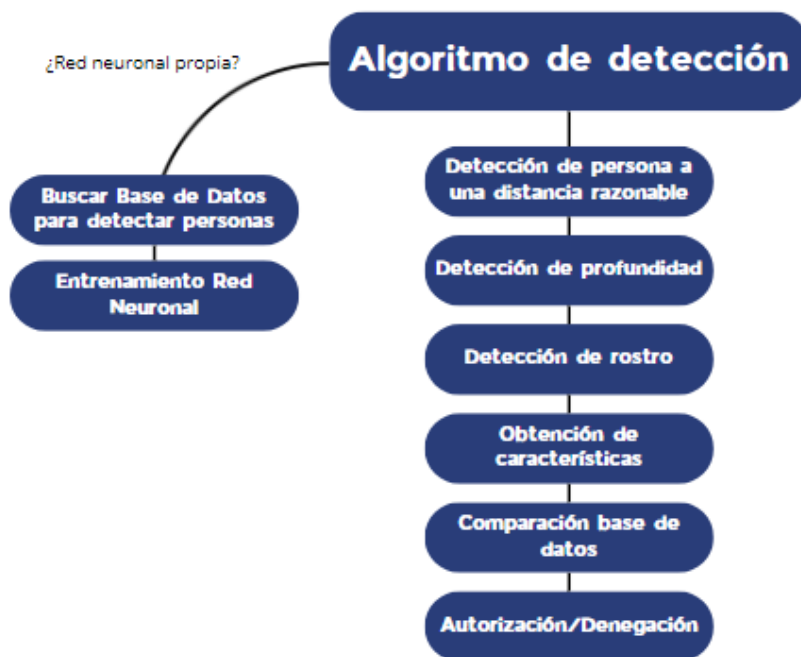


Figura 3: Etapas del Algoritmo de Detección

Debido a que se fue acotando el proyecto, algunas partes de estos diagramas se dejaron de considerar. Lo anterior es debido a que ahora el proyecto se acotó a “Reconocimiento facial para ingreso a espacios en una universidad”, por lo que solo se mantiene como salida el acceso a infraestructuras, no así el acceso a la información y ser multipropósito. También, se realizó un cambio en la detección de profundidad, para ahora utilizar la detección de un rostro a partir del movimiento de este.

A modo de tener todo el trabajo en un solo lugar y hacerlo de manera simultánea, se creó un repositorio en *GitHub* [4]. Aquí se tienen tanto las presentaciones realizadas como los códigos generados para el desarrollo del proyecto.

Luego, siguiendo con lo propuesto en la carta Gantt presentada, se comenzó a crear la base de datos a utilizar para utilizar como registro de usuarios. El código generado crea una base de datos con una cantidad determinada de imágenes del usuario, considerando distintos ángulos del rostro.

Luego, se elaboró un código para experimentar con los modelos *Face Detection* y *Face Mesh*, donde a partir del seguimiento de las posiciones de ciertos puntos claves del rostro

(párpados superior e inferior de ambos ojos, parietales y extremos externos de las cejas), se decía al usuario que mirara hacia al frente y luego que parpadeara 3 veces, para así terminar el programa. A continuación se muestra una captura de su funcionamiento:



Figura 4: Captura de funcionamiento de código con orientación y movimiento de rostro.

Y ahora se presenta gráficamente el funcionamiento interno mediante el dibujado del cuadro delimitador del rostro (*bounding box*), la máscara facial y los puntos de interés:

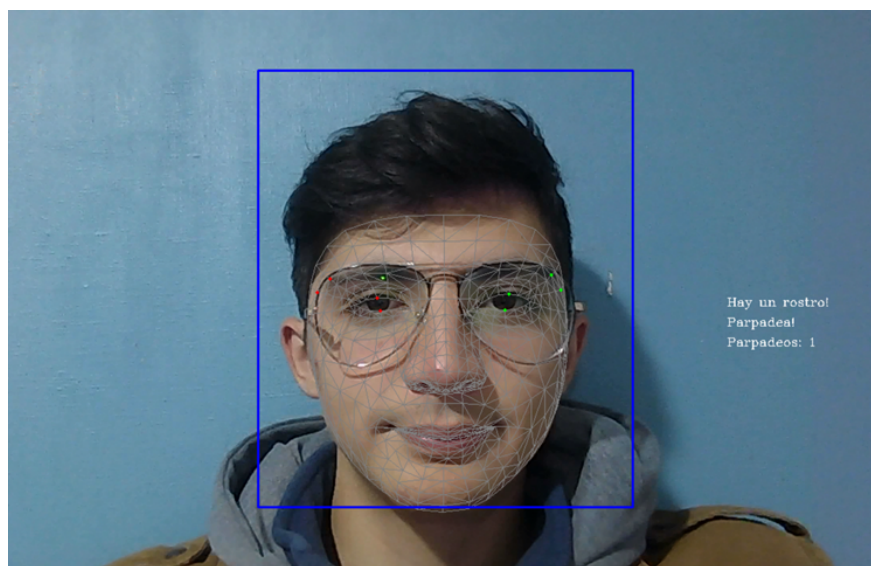


Figura 5: Captura de funcionamiento de código con muestra gráfica de métodos implementados.



A modo de mostrar lo que haría el sistema final de manera general, se tiene un diagrama que representa el proceso completo, este se encuentra en la Figura 6:

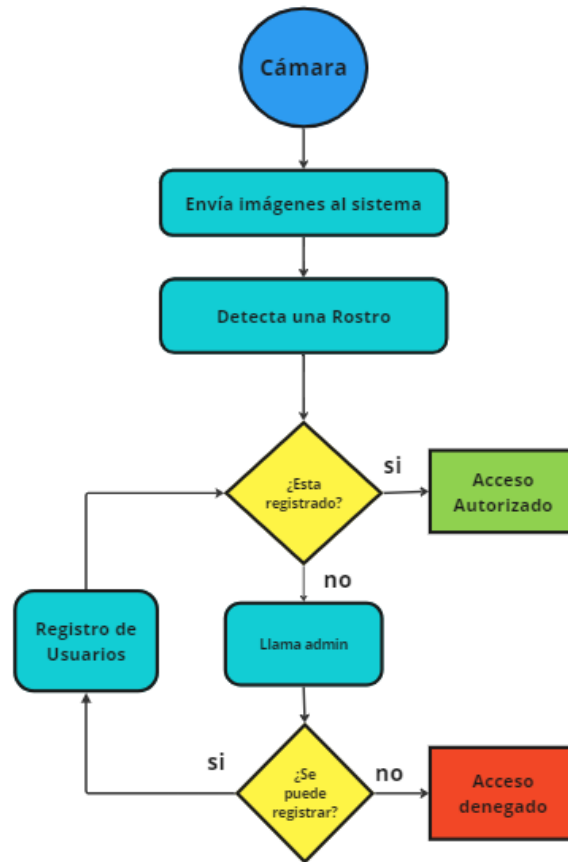


Figura 6: Diagrama funcional del proyecto.

## 4. Carta Gantt

Al comenzar con el desarrollo del proyecto, se creó una carta Gantt para poder organizar las tareas a lo largo del semestre. Esta se presenta en la Figura 7:

Semana	4	5	6	7	8	9	10	11	12	13	14	15
Tareas / Fecha	01/04	08/04	15/04	22/04	29/04	06/05	13/05	20/05	27/05	03/06	10/06	17/06
Buscar base de datos para entrenar la red neuronal.												
Conseguir implementos circuitales tales: como camara, microprocesador, sensor de profundidad.												
Entrenamiento de la red.												
Hacer pruebas con la camara.												
Probar el sistema de camara y red.												
Probar deteccion de movimiento del rostro.												
Obtener las características del rostro.												
Comparacion con la base de datos.												
Buscar actuadores para la deteccion o no deteccion de rostros.												
Elaborar sistema de registros.												
Incorporar respuesta de audio.												
Pruebas finales.												
Puesta en Practica.												
Presentación del proyecto.												

Figura 7: Carta Gantt creada para el desarrollo del proyecto para 15 semanas.

La carta Gantt ha tenido distintas modificaciones a lo largo del desarrollo del proyecto, principalmente debido a cambios en los métodos a utilizar para lograr obtener mejores resultados. Estos se detallan a continuación:

- En lugar de buscar la base de datos para entrenar la red neuronal, se crea en base a los usuarios registrados.
- Ya no se utilizará un sensor de profundidad, pues la corroboración de que efectivamente se encuentra una persona en frente de la cámara se realiza mediante la detección de movimiento.
- Las fechas para las pruebas con la cámara externa fueron aplazadas debido al tiempo que demoró conseguir dicho dispositivo.

A la fecha de elaboración de este informe, se encuentra en la etapa de crear el sistema de registro de usuarios e integración completa de los códigos utilizados, junto con pruebas de la cámara para capturar video de forma inalámbrica, lo que indica un progreso mayoritariamente efectivo al haber cumplido o tener avances significativos en diversas tareas.

## 5. Evaluación económica

Prerrequisitos un computador y un access point, se asume que estos elementos ya se encuentran en los lugares de destino, en caso contrario habría que sumar estos costos a los considerados en la factura del proyecto.

Valores de los insumos:

Valor del programa: \$ 0

Valor cámara: \$ 11900

Fuente de poder o Cargador 5V: \$ 10000

Modulo de cámara: \$ 1900

Costo total: \$ 23800

## 6. Conclusiones

A lo largo del desarrollo del proyecto, ha sido necesario realizar distintos cambios. Dentro de estos se encuentran: enfocar el proyecto a un producto realizable que luego de que este funcione se pueda escalar a algo más grande, cambiar los métodos a utilizar y modificar plazos. Por lo que hace notar que durante un proyecto, aunque se tengan planificaciones, es inevitable tener cambios para poder llegar satisfactoriamente al producto final.

## 7. Referencias

- [1] <https://github.com/google-ai-edge/mediapipe>
- [2] <https://arxiv.org/abs/1907.05047>
- [3] <https://www.entel.cl/hogar/internet-segun-mi-uso/>
- [4] [https://github.com/rorrocatalam/ReconocimientoFacial\\_Grupo3](https://github.com/rorrocatalam/ReconocimientoFacial_Grupo3)
- [5] <https://triacs.cl/modulos/1012--modulo-de-camara-esp32-cam-.html>
- [6] <https://triacs.cl/modulos/951-placa-esp32cam-wifi-.html>
- [7] [https://simple.ripley.cl/cargador-pared-1-usb-carga-rapida-3a-micro-usb-tecnolab-m-pm10000264453?color\\_\\_80=blanco&s=mdco](https://simple.ripley.cl/cargador-pared-1-usb-carga-rapida-3a-micro-usb-tecnolab-m-pm10000264453?color__80=blanco&s=mdco)