

Informe final

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: 15 de julio de 2024
Santiago de Chile

Resumen

A medida que avanza la tecnología surgen cada vez más problemáticas, donde la seguridad es una de ella. Este es un tema muy delicado, ya que dada la contingencia es crucial tener control de las personas que entran a ciertos espacios, tales 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 la librería de *Python mediapipe*, una cámara USB y un microcontrolador Arduino UNO, 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 el hogar 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 quiénes ocupan sus inmediaciones, ya que todas las personas que quieran hacer ingreso deberán estar registrados en el sistema y no necesitarán ningún elemento adicional.

Índice de Contenidos

| | |
|--------------------------|----|
| 1. Introducción | 1 |
| 2. Marco teórico | 2 |
| 3. Desarrollo | 4 |
| 4. Carta Gantt | 9 |
| 5. Evaluación económica | 10 |
| 5.1. Servicios | 10 |
| 5.2. Productos | 10 |
| 6. Conclusiones | 11 |
| 7. Referencias | 12 |

Índice de Figuras

| | |
|---|---|
| 1. Modelo simplificado de la arquitectura utilizada en <i>Face Mesh</i> | 3 |
| 2. Diagrama del algoritmo de reconocimiento facial | 4 |
| 3. Etapas del Algoritmo de Detección | 5 |
| 4. Captura de funcionamiento de código con orientación y movimiento de rostro. | 6 |
| 5. Captura de funcionamiento de código con muestra gráfica de métodos imple- mentados. | 6 |
| 6. Estructura general del circuito | 7 |
| 7. Diagrama funcional para el registro de un nuevo usuario | 8 |
| 8. Diagrama funcional para la detección de un usuario | 8 |
| 9. Carta Gantt creada para el desarrollo del proyecto para 15 semanas. | 9 |

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. Es por esto que el problema fue acotado, logrando aplicar diferentes métodos que interactúen entre sí para tener un sistema efectivo.

2. Marco teórico

El sistema de reconocimiento facial que se pretende implementar se basa principalmente en la identificación de patrones presentes en una imagen. Si se cumplen ciertos criterios establecidos, se determinará si se concede o no el acceso. Las redes neuronales son especialmente eficientes para llevar a cabo esta tarea, y serán consideradas para el desarrollo de este proyecto.

Una red neuronal [1] es un modelo computacional compuesto por varias capas de neuronas artificiales, que están modeladas basándose en el cerebro humano. Estas neuronas están dispuestas en capas donde la información es recibida y procesada para arrojar una respuesta concreta. Cada una de las conexiones entre dos neuronas tiene un peso ajustable que determina la importancia de la respectiva entrada, los que se pueden ajustar mediante el proceso de entrenamiento de la red para mejorar las capacidades de generalización y predicción.

Para el caso particular del procesamiento de imágenes, aquellas que presentan un mejor desempeño son las conocidas como redes neuronales convolucionales (CNN) las cuales poseen una gran capacidad para capturar comportamientos espaciales de los datos de entrada, dada su estructura basada en la convolución matemática.

Considerando la implementación misma, se estará utilizando el modelo *cv2.face.EigenFace Recognizer* de la librería *opencv* [2], el cual está basado en el método de *Eigenfaces* para la reducción de dimensionalidad y facilitar el reconocimiento de patrones.

Por otra parte, también se utilizará la librería *mediapipe* de *Google* [3], 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* que utilizan redes neuronales convolucionales para su funcionamiento, 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* [4], 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:

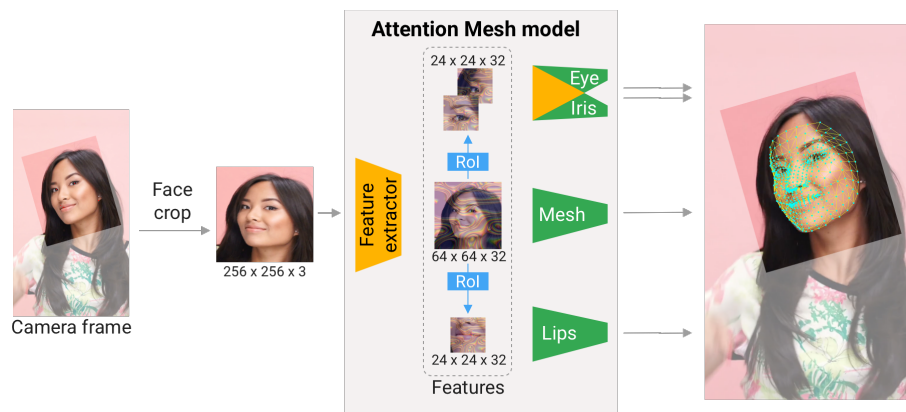


Figura 1: Modelo simplificado de la arquitectura utilizada en *Face Mesh*.

Para dar mayor dinamismo al proyecto, se trabajará con un *Arduino UNO* [5] (placa de desarrollo basada en un microcontrolador, útil para diversos proyectos de electrónica interactiva) para dar una respuesta visual de la salida del sistema de reconocimiento.

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.

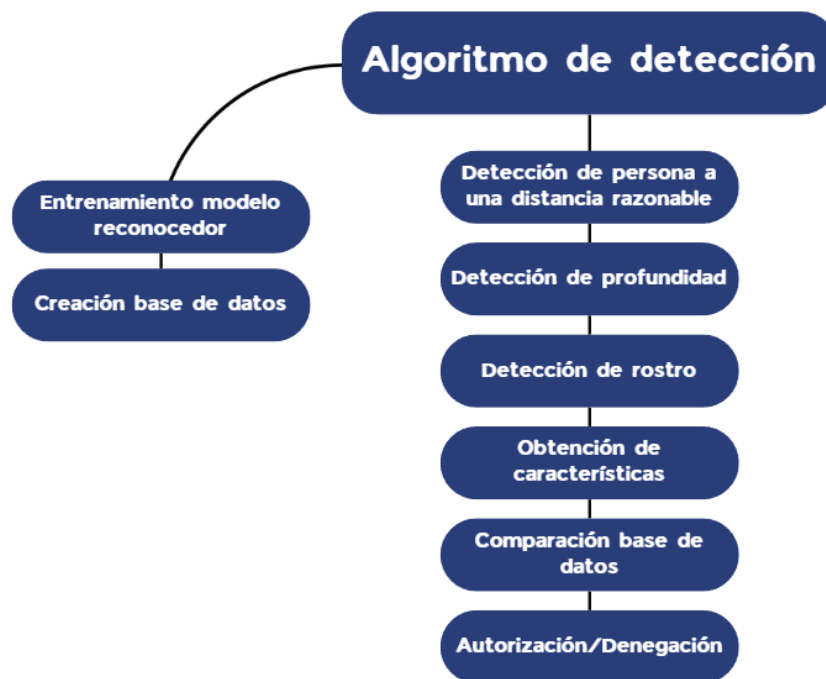


Figura 3: Etapas del Algoritmo de Detección

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* [6]. 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:

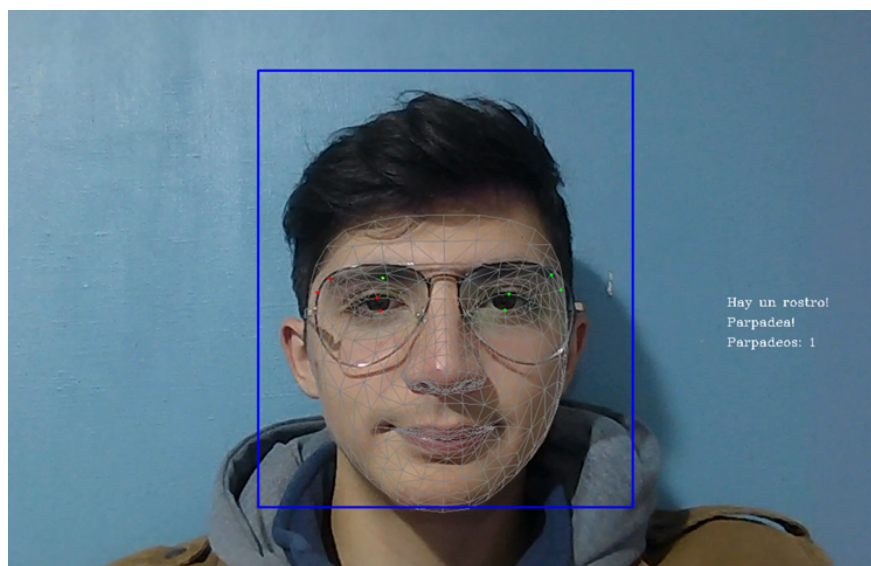


Figura 5: Captura de funcionamiento de código con muestra gráfica de métodos implementados.

Una vez capturado el movimiento, se procede a la etapa de detección del usuario mediante la comparación de las características faciales obtenidas con el modelo *cv2.face.EigenFaceRecognizer* con la base de datos de usuarios registrados. Esta base de datos fue creada previamente con el mismo modelo en un código separado, dedicado exclusivamente a esta tarea. Tras realizar la comparación, se determina si se concede acceso o no, basado en un umbral preestablecido según las condiciones del entorno. Luego, se activan las respuestas auditivas y visuales mediante un Arduino, estructura expuesta en la Figura 6:

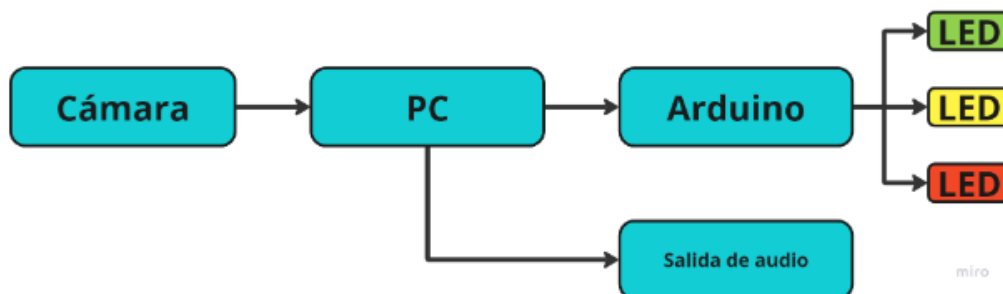


Figura 6: Estructura general del circuito

De la estructura presentada se encuentran los siguientes componentes:

- Cámara: Es el encargado de recibir las imágenes que luego serán procesadas y almacenadas.
- PC: Es el centro del sistema ya que es el encargado de procesar las imágenes, almacenar la base de datos y decidir en base a las opciones que tiene.
- Salida de audio: Entrega un mensaje de manera que pueda escuchar si es reconocido o no el rostro en el sistema.
- Arduino: Es el encargado de activar los actuadores secundarios, los que en este caso están representados como LEDs de distintos colores. LED rojo rostro no es reconocido; LED verde, el rostro si es reconocido; LED amarillo, sistema esta esperando.

Con esto funcionando, se procede a eliminar del código la visualización de los resultados dado que en el resultado final no se tiene contemplada una interfaz gráfica para interactuar con el usuario, sino que este sólo sabrá si se dio acceso o no.

A modo de mostrar lo que hace el sistema final de manera general, se tiene un diagrama que representa el proceso completo presente en las Figuras 7 y 8:

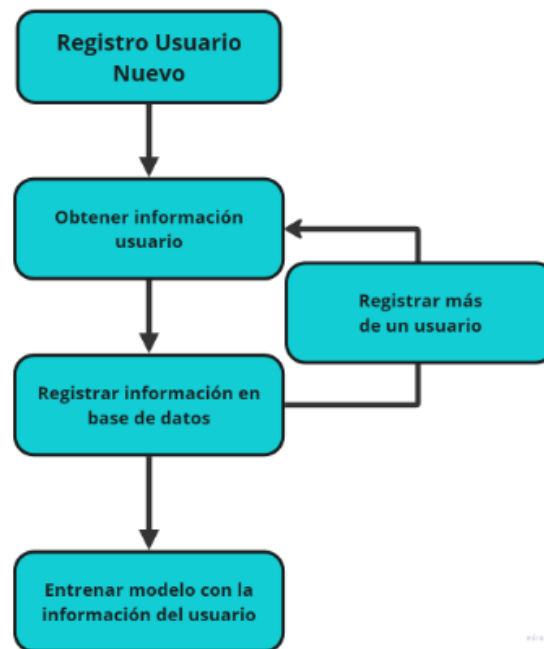


Figura 7: Diagrama funcional para el registro de un nuevo usuario

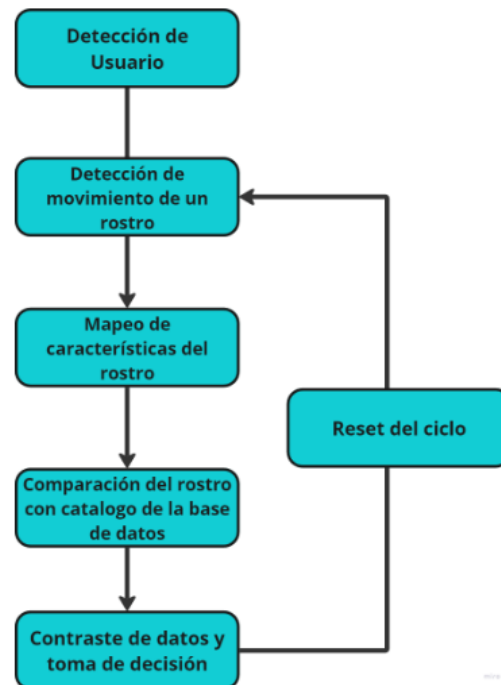


Figura 8: Diagrama funcional para la detección de un usuario

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 9:

| 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 9: 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 y por último descartadas debido a las dificultades presentadas al momento de trabajarla.

Tras haber culminado los plazos predispuestos para la elaboración y culminación, es posible decir que a pesar de los cambios hechos a la planificación y tareas por llevar a cabo, se siguió en gran parte lo inicialmente planeado en la carta Gantt, consiguiendo un sistema de reconocimiento facial acorde al desempeño esperado.

5. Evaluación económica

En esta sección se realizará la evaluación económica de los distintos elementos y servicios utilizados a lo largo del proyecto. Los valores correspondientes se presentan a continuación:

5.1. Servicios

- **Ingeniero/a civil eléctrico/a:** es necesario considerar el servicio que presta cada integrante para la realización del proyecto. Tomando en cuenta lo anterior, se tiene un promedio de \$1.300.000 de sueldo mensual [7], teniendo así un valor final de: $\$1.300.000 \cdot 3 = \$3.900.000$.

Es importante mencionar que el trabajo de cada uno se considera como un trabajo *full time* durante un mes para todo el proceso, por eso en el cálculo solo se considera un sueldo para cada uno de los integrantes.

5.2. Productos

- **Computador:** en este caso es necesario el uso de este, ya que la mayor cantidad de trabajo se realiza a partir de código. Además, se considera que cada integrante del equipo utiliza uno, con lo anterior el valor resultante es [8]: $\$ 629.990 \cdot 3 = \$ 1.889.970$.
- **Programa:** \$0
- **Arduino UNO** [5]: \$30.990
- **Cámara** [9]: \$4.790

Considerando todos los elementos presentados, se tiene un costo total de productos de: \$ 1.925.750.

Si se suman los valores de ambas secciones, se tiene un total de: \$5.825.750.

6. Conclusiones

El proyecto trabajado a lo largo del semestre, buscó mejorar la seguridad y eficiencia en el control de acceso a través del uso de tecnología y procesamiento de imágenes. Mediante el empleo de la librería *mediapipe* de *Python*, una cámara USB y un microcontrolador Arduino UNO, se ha diseñado un sistema que identifica rostros y determina el ingreso al espacio. Este sistema reemplaza la necesidad de tarjetas físicas (TUI), agilizando el proceso de entrada y garantizando un control más riguroso.

Durante el desarrollo del proyecto, se realizaron diversas modificaciones para mejorar la precisión y funcionalidad del sistema, siguiendo una planificación previa mediante una carta Gantt. La evaluación económica indicó un costo total de \$5.825.750, destacando la importancia de una adecuada planificación de recursos.

En conclusión, la implementación de este sistema demuestra cómo la tecnología puede ser implementada en ámbitos de accesos a espacios universitarios. Además, que una planificación meticulosa permite identificar y reducir riesgos que se puedan presentar durante la realización de un proyecto.

7. Referencias

- [1] Colaboradores de los proyectos Wikimedia. “Red neuronal artificial - Wikipedia, la enciclopedia libre”. Wikipedia, la enciclopedia libre. Accedido el 14 de julio de 2024. [En línea]. Disponible: https://es.wikipedia.org/wiki/Red_neuronal_artificial
- [2] “OpenCV: cv::face::EigenFaceRecognizer Class Reference”. OpenCV documentation index. Accedido el 14 de julio de 2024. [En línea]. Disponible: https://docs.opencv.org/3.4/dd/d7c/classcv_1_1face_1_1EigenFaceRecognizer.html
- [3] “GitHub - google-ai-edge/mediapipe: Cross-platform, customizable ML solutions for live and streaming media.” GitHub. Accedido el 14 de julio de 2024. [En línea]. Disponible: <https://github.com/google-ai-edge/mediapipe>
- [4] “BlazeFace: Sub-millisecond Neural Face Detection on Mobile GPUs”. arXiv.org. Accedido el 14 de julio de 2024. [En línea]. Disponible: <https://arxiv.org/abs/1907.05047>
- [5] “Arduino Uno R3 – MCI Electronics”. MCI Electronics – Tu aliado en electrónica. Accedido el 14 de julio de 2024. [En línea]. Disponible: https://mcielectronics.cl/shop/product/arduino-uno-r3-arduino-10230/?utm_source=Google%20Shopping&utm_campaign=Product%20feed%20test&utm_medium=cpc&utm_term=11646&gad_source=1&gclid=CjwKCAjw7s20BhBFEiwABVIMrSPTNJSrnYKpokqoxfQnJEHtemmsd1lxks8Fj8aUwy5v5cJp0aLIWhoC46YQAvD_BwE
- [6] “GitHub - rorrocatalam/ReconocimientoFacial_Grupo3”. GitHub. Accedido el 14 de julio de 2024. [En línea]. Disponible: https://github.com/rorrocatalam/ReconocimientoFacial_Grupo3
- [7] “Sueldo de Ingeniero/a eléctrico/a en Chile”. Indeed.com. Accedido el 14 de julio de 2024. [En línea]. Disponible: <https://cl.indeed.com/career/ingeniero-el%C3%A9ctrico/salaries>
- [8] “Notebook Vivobook 16X M3604 AMD Ryzen 7 16GB RAM 512GB SSD 16"WUXGA 60Hz en Notebooks | Paris”. Paris.cl | Tu experiencia de compra más confiable y segura. Accedido el 14 de julio de 2024. [En línea]. Disponible: https://www.paris.cl/notebook-vivobook-16x-m3604-amd-ryzen-7-16gb-ram-512gb-ssd-16-wuxga-60hz-527292999.html?gad_source=1&gclid=CjwKCAjw7s20BhBFEiwABVIMrRDe4pJ_pxpA6v9W7lsxpgBclRdHVouUK1joIpljQuyZh-SeIU0V4JxoCahgQAvD_BwE&gclidsrc=aw.ds

[9] “Camara Web Usb Con Microfono Teletrabajo Full Hd 1080p”. Paris.cl | Tu experiencia de compra más confiable y segura. Accedido el 14 de julio de 2024. [En línea]. Disponible: https://www.paris.cl/camara-web-usb-con-microfono-teletrabajo-full-hd-1080p-MKT09K6DRP.html?utm_campaign=organicsshopping&utm_medium=organic&utm_source=google