



Instituto Politécnico Nacional

Escuela Superior de Cómputo

ESCOM

Trabajo Terminal II

"Prototipo de sistema de publicidad dirigida a clientes usando visión por computadora y aprendizaje profundo"

TT No. 2019 -B059

Presenta:

Lara Cázares Jaime Arturo
Morales Flores Víctor Leonel
Ramos Diaz Enrique



Directores:

Dr. Jorge Cortés Galicia
Dr. Marco Antonio Moreno Armendáriz

Índice general

1. Introducción	6
1.1. Justificación	7
1.1.1. Originalidad del trabajo	7
1.1.2. Vinculación con usuarios potenciales	8
1.1.3. Mejora a lo ya existente	8
1.2. Planteamiento del problema	8
1.3. Hipótesis	8
1.4. Objetivos	9
1.4.1. Objetivo general	9
1.4.2. Objetivos específicos	9
2. Marco teórico	10
2.1. Introducción al capítulo	10
2.2. Personalización en el marketing	11
2.2.1. Ventajas y desventajas	12
2.2.2. Los deseos y la perspectiva del consumidor	13
2.2.3. Automatización del marketing	15
2.3. Detección de objetos	16
2.3.1. Sustracción de fondo	16
2.3.1.1. Técnica no recursiva	16
2.3.1.2. Técnica recursiva	16
2.3.2. Detección de bordes	17
2.4. Redes neuronales	21
2.4.1. La neurona biológica	21
2.4.2. Función de activación	22
2.4.3. Reglas de aprendizaje	24
2.4.4. Partes de una red neuronal	24
2.4.5. Tipo de aprendizaje	24
2.4.6. Modos de operación	25
2.5. El Perceptrón Multicapa	25
2.6. Redes neuronales convolucionales	25

2.6.1. Capa de agrupación (Pooling)	28
2.7. Transferencia de aprendizaje	29
2.8. Dropout	30
2.9. VGG16	30
2.10. Single Shot Detection	31
2.11. MobileNet	33
2.11.1. Convolución separable en profundidad	34
2.11.2. Comparación entre convolución espacial y convolución separable en profundidad.	35
2.12. Biometría blanda	38
2.12.1. Características de datos biométricos blandos	39
2.12.2. Ventaja de datos biométricos blandos	39
2.13. Edad	39
2.13.1. ¿Por qué son importantes los rangos generacionales?	39
2.13.2. Estimación de la edad	41
2.13.2.1. Métodos no convolucionales	42
2.13.2.2. Métodos con redes neuronales convolucionales	42
2.14. Género	42
2.15. Personalidad: Big Five	44
2.15.1. TPQ	44
2.15.2. Teoría de rasgos	45
2.15.3. Análisis facial	47
2.15.4. Selfies	48
2.16. Recomendador	49
2.16.1. Recomendación basada en contenido	49
2.16.2. Filtrado colaborativo	49
2.16.3. Árbol de decisión	50
2.16.4. Clasificador bayesiano ingenuo	50
2.16.5. KNN (vecino más cercano)	51
2.17. Realidad Aumentada	52
2.17.1. Sistemas de Realidad Aumentada	53
2.18. Bases de datos NoSQL	54
2.19. BubbleTown®	55
2.19.1. Propuesta de valor y ventaja competitiva	55
2.19.2. Misión de la empresa	56
2.19.3. Visión de la empresa	56
2.19.4. Mercado meta	56
2.19.5. Ubicación y sucursal	56
2.19.6. Menú de bebidas	57
2.19.7. Competencia de BubbleTown®	58
2.20. Discusión del capítulo	58
3. Estado del Arte	60
3.1. Introducción al capítulo	60
3.2. Revisión de trabajos de Publicidad Dirigida	61
3.3. Revisión de trabajos de Edad y género	63
3.3.1. Arquitectura propuesta por Orozco et al (2017)	63
3.3.2. Arquitectura propuesta por Ranjan et al (2017)	66
3.3.3. Arquitectura propuesta por Xing et al (2017)	68

3.3.4. Arquitectura propuesta por Vasileiadis et al (2019)	71
3.4. Revisión de trabajos de Big Five	73
3.4.1. Análisis en imágenes	73
3.4.2. El dataset de ChaLearn	74
3.4.2.1. Análisis de video: comparación entre imágenes, audio y texto	76
3.4.3. Nuevo dataset a partir de ChaLearn: PortraitPersonality.v2	78
3.4.4. Modelo de clasificación: CNN-4	80
3.4.5. Modelo de clasificación: FaceNet-1	80
3.5. Revisión de trabajos de Realidad Aumentada	82
3.6. Discusión del capítulo	83
 4. Análisis	85
4.1. Introducción al capítulo	85
4.2. Metodología	85
4.3. Historias de usuario	87
4.4. Reglas del negocio	89
4.4.1. RN1 Nivel de comunicación.	89
4.4.2. RN2 Aviso de privacidad	90
4.4.3. RN3 Tipografía y colores asociados a la marca.	90
4.4.4. RN4 Personal con acceso a datos.	90
4.4.5. RN5 Uso del ancho de banda.	90
4.5. Requerimientos funcionales	91
4.6. Requerimientos no funcionales	92
4.7. Actores del sistema	93
4.8. Diagrama de casos de uso	94
4.9. Descripción de los casos de uso	94
4.9.1. CU-01 Generar publicidad dirigida	95
4.9.1.1. Descripción completa	95
4.9.1.2. Atributos importantes	95
4.9.1.3. Trayectorias del Caso de Uso	96
4.9.2. CU-02 Autenticar administrador	97
4.9.2.1. Descripción completa	97
4.9.2.2. Atributos importantes	97
4.9.2.3. Trayectorias del Caso de Uso	97
4.9.3. CU-03 Agregar publicidad estática	99
4.9.3.1. Descripción completa	99
4.9.3.2. Atributos importantes	99
4.9.3.3. Trayectorias del Caso de Uso	100
4.9.4. CU-04 Extraer datos de la base de datos	101
4.9.4.1. Descripción completa	101
4.9.4.2. Atributos importantes	101
4.9.4.3. Trayectorias del Caso de Uso	102
4.9.5. CU-05 Agregar nueva bebida	104
4.9.5.1. Descripción completa	104
4.9.5.2. Atributos importantes	104
4.9.5.3. Trayectorias del Caso de Uso	105
4.9.6. CU-06 Modificar bebida	106
4.9.6.1. Descripción completa	106

4.9.6.2. Atributos importantes	106
4.9.6.3. Trayectorias del Caso de Uso	107
4.9.7. CU-07 Eliminar bebida	108
4.9.7.1. Descripción completa	108
4.9.7.2. Atributos importantes	108
4.9.7.3. Trayectorias del Caso de Uso	108
4.9.8. CU-08 Eliminar publicidad estática	110
4.9.8.1. Descripción completa	110
4.9.8.2. Atributos importantes	110
4.9.8.3. Trayectorias del Caso de Uso	110
4.10. Gestión de riesgos	112
4.11. Costo estimado de desarrollo	115
4.12. Discusión del capítulo	115
5. Diseño e Implementación	117
5.1. Introducción al capítulo	117
5.2. Diagramas de actividades	118
5.3. Diagrama de clases	122
5.4. Diagramas de secuencia	123
5.5. Diagrama de Despliegue	124
5.6. Especificación del hardware	125
5.6.1. Dispositivos a usar	125
5.6.2. Sistema operativo JetPack y NVIDIA SDK Manager	129
5.7. Herramientas de software a utilizar	130
5.8. Diagrama general del sistema	133
5.9. Captura de video a imagen	135
5.9.1. Establecer comunicación entre la cámara de red y la NVIDIA Jetson	136
5.9.2. Captura del flujo de video y extracción de frames	138
5.10. Preprocesamiento de la imagen	139
5.11. Detección de rostros en la imagen	140
5.11.1. Preprocesamiento del dataset para el entrenamiento de la SSD	140
5.11.2. Configuración del API	141
5.11.3. Entrenamiento	142
5.11.4. Pruebas de detección de rostros	143
5.11.5. Implementación en TensorRT	145
5.11.6. Recorte y procesamiento de los rostros detectados	146
5.12. Red neuronal convolucional para Edad y Género	148
5.12.1. Red neuronal convolucional para edad	148
5.12.2. Red neuronal convolucional mutitarea	152
5.12.3. Implementación en TensorRT	156
5.13. Red neuronal convolucional para Big Five	157
5.13.1. Implementación en TensorRT	157
5.14. Recomendador del sistema	159
5.14.1. Vectorización de bebidas	161
5.15. Base de datos para el sistema	163
5.15.1. Colección de administradores	163
5.15.2. Colección de bebidas	163
5.15.3. Colección de publicidad	164

5.15.4. Colección de recomendaciones	165
5.16. Generación de realidad aumentada	166
5.17. Diseño de la interfaz gráfica	168
5.18. Atributos de calidad	171
5.19. Discusión del capítulo	173
6. Pruebas y Resultados	175
6.1. Introducción al capítulo.	175
6.2. Plan de pruebas	176
6.2.1. RF01 - Detección de rostros	176
6.2.2. RF02 - Preprocesamiento de rostros	179
6.2.3. RF03 - Estimación de edad y género	181
6.2.4. RF04 - Estimación de personalidad	183
6.2.5. RF05 - Obtención de la temperatura	185
6.2.6. RF07 - Generación de realidad aumentada	186
6.2.7. RF08 - Almacenamiento de rostros	188
6.2.8. RF09/RF10 - Almacenamiento y respaldo de recomendaciones	190
6.2.9. RF11 - Autenticación de administrador	192
6.2.10. RF12 - Nueva bebida	194
6.2.11. RF13 - Actualización de bebida	196
6.2.12. RF14 - Eliminación de bebida	197
6.2.13. RF15 - Comprobación de bebida	199
6.2.14. RF16 - Descifrado de rostros	200
6.2.15. RF17 - Despliegue de publicidad estática	202
6.2.16. RF18 - Nueva publicidad estática	204
6.2.17. RF19 - Eliminación publicidad estática	207
6.3. Discusión del capítulo.	209
7. Conclusiones generales	210
7.1. Trabajo a futuro	213
A. Manual de usuario	215
Bibliografía	231

CAPÍTULO 1

Introducción

Se entiende por publicidad a la difusión de mensajes a través de medios de comunicación masiva. Estos medios o canales son de sentido unidireccional; es decir, el mensaje va del emisor al receptor sin que éste último pueda responder en el acto, siendo la única "respuesta" esperada el consumir dicho producto o servicio ofrecido [1]. El principal inconveniente es que esta información presentada está limitada a un contenido estático o generalizado, ya que la mayoría de veces está destinada a un mercado objetivo, pero no a cada uno de los integrantes que pertenecen a él, pues se siente como una tarea imposible e interminable el presentar un anuncio publicitario personalizado para cada uno de los posibles clientes potenciales.

A lo largo de la historia, la forma en generar publicidad para los usuarios también ha evolucionado. Tras la aparición de la primera revista en el año de 1730, los carteles se volvieron muy populares en 1830 para difundir mensajes entre la población. Casi una década después, en 1922 se emitió el primer anuncio a través de la radio y en 1941 se emitió el primer anuncio a través de la televisión, logrando ser transmitido a 4,000 televisores. Acercándose al final del siglo XX surgen dos formas más de hacer publicidad: el spam y el uso de los navegadores como Yahoo o Google. En el siglo XIX surgen las redes sociales como Facebook o Twitter, el E-commerce se acepta por una mayor parte de usuarios y YouTube se consolida, siendo ahora estas formas de generar publicidad las más usadas [2].

En los últimos años, los avances del Deep Learning han ganado una atención significativa en campos como los del procesamiento de imágenes. En general, se pueden generar recomendaciones basadas en las preferencias del usuario, las características del producto o servicio, las transacciones del usuario y otros factores como la hora, el clima y la ubicación [3]. La idea de tener una recomendación publicitaria o anuncio dirigido para cada usuario, según los factores previamente mencionados e incluso otros más de interés para el contexto del producto o servicio (edad, género, rasgos físicos, etc.), no suena tan lejana [1].

Por otra parte, se tiene la creciente popularidad de la realidad aumentada, que consiste en la superposición de información, como imágenes o texto, encima de la realidad, siendo normalmente una fotografía o video captado por una cámara. De la mano va la visión artificial, tecnología que permite identificar un objeto, tomando una fotografía del mismo y relacionándola con un conjunto de datos de referencia [4].

Según Mónica Deza, ex CEO de Universal McCann [2], los anuncios que recurren a la realidad aumentada cumplen con los requisitos básicos de publicidad eficaz:

- La personalización del contenido.
- Que los resultados sean medibles en tiempo real.

- Sorprender al consumidor y que dicho efecto sorpresa fomente el recuerdo de la acción publicitaria en el cliente.

La realidad aumentada llama la atención del consumidor abriéndose paso en su mente, se convierte en un elemento diferenciador con respecto a la competencia y ofrece al usuario la posibilidad de acceder a experiencias visuales llamativas [2].

1.1. Justificación

Como comenta John Sculley en su libro Moonshot: "El futuro pertenece a quienes ven las posibilidades antes de que sean obvias". Se estima que en los próximos años la clase media influirá de forma determinante sobre los mercados emergentes, ya que llegarán a ser más de dos mil millones de personas para 2020 [5].

Esto lleva a la necesidad de abarcar a mayor medida y de diferente manera la forma en la que se llevan a cabo las ventas. En las ventas tradicionales se piensa que se debe mencionar todas las características del producto para lograr que algo capte la atención del cliente, sin embargo, la neurociencia propone que la forma más sencilla es atacar al subconsciente del comprador para lograr la venta de mejor forma, ya que el 85% de las ventas son subconsciente o inconsciente y solo el 15% se realiza de forma consciente [6].

"Ver es creer" y, en muchos casos, "creer es vender". por lo que es importante asegurarse de que el cliente efectivamente ve lo que se quiere que vea y lo interprete de la forma deseada. La mayoría de los consumidores tienen absoluta confianza en la información que reciben a través de la vista, ya que desde la infancia se aprende a interpretar las impresiones visuales [7].

Se puede resumir el impacto del marketing visual en cuatro diferentes áreas: de atención, canalizador de percepciones, estimulador de recuerdos y estimulador de emociones [7].

La visión es uno de los sentidos que cobran más protagonismo en estas nuevas estrategias debido a la gran cantidad de sentimientos y emociones que el sentido de la vista despierta en los humanos. El reto, por tanto, consiste en transmitir al consumidor no sólo una propuesta de valor diferencial y relevante, sino también una propuesta suficientemente atractiva y visualmente capaz de despertar en él las emociones y los sentimientos más genuinos [7].

Actualmente se cuenta con la posibilidad de utilizar sensores como las cámaras para captar la información de los clientes y hacer uso de algoritmos matemáticos con sistemas de Machine Learning que nos permitan analizar el comportamiento del consumidor en un nivel personal y poder predecir resultados de forma automática [5].

Con esto en mente es posible aprovechar las distintas tecnologías existentes para explotar el campo de la publicidad dirigida, la cual da resultados positivos pues como lo menciona Francisco Coll, los anuncios publicitarios intrusivos que se suponen son una forma novedosa resulta ser contraproducentes, provocando en el usuario habitual una constante frustración y malestar por los anuncios y por las empresas que lo bombardean [8].

se propone utilizar la combinación de Machine Learning y realidad aumentada para crear publicidad dirigida la cual sea realmente efectiva, pues como menciona Carmen Abril: "La experiencia visual constituye un valor añadido para el consumidor" [7]. La conjunción de estas tecnologías es nuestro principal diferenciador, pues ya existen distintas implementaciones por separado las cuales no atacan nuestro principal problema, la generación de publicidad dirigida.

En palabras de Carmen Abril "parece evidente que el marketing sensorial se abre paso con fuerza en el marketing actual" [7] y que mejor que en este campo se tenga presencia desde su inicio.

1.1.1. Originalidad del trabajo

Como se explicará de forma más extendida en el capítulo [Estado del Arte](#) del presente documento, algunas de las grandes empresas como lo son Intel, ya han desarrollado sistemas similares como lo es el AIM Suite [9]; sin embargo, durante la investigación no se encontró un trabajo o producto que proporcionara las capacidades que se plantean para la realización de este trabajo terminal.

Anteriormente la empresa de bebidas [BubbleTown®](#) contaba con un sistema de recomendación de bebidas basado en el uso de redes neuronales profundas para que a partir de una foto obtuviera una recomendación, pero no hacía uso de técnicas como la realidad aumentada para generar publicidad dirigida.

El sistema propuesto en este trabajo terminal tiene como objetivo desplegar un anuncio publicitario personalizado para cada cliente potencial que transite por fuera del establecimiento [BubbleTown®](#) haciendo uso de una pantalla para desplegar la publicidad. Con el objetivo de llamar la atención de los espectadores, este anuncio hará uso de tecnologías como la realidad aumentada para mostrarle al usuario la recomendación generada por redes neuronales.

El sistema tendrá como objetivo analizar al cliente por medio de una imagen de su rostro y recomendar uno de los productos de [BubbleTown®](#) que le podría agradar más. Para lograr el objetivo, se hará uso de técnicas de visión artificial a partir de cámaras instaladas estratégicamente en el local.

1.1.2. Vinculación con usuarios potenciales

Para el caso de este trabajo terminal se cuenta con el apoyo de la empresa mexicana [BubbleTown®](#) la cual tendrá el papel de cliente real y colaborador. Por lo anterior se vinculará al personal de la empresa desde la planificación del proyecto y se tendrá contacto continuo hasta la instalación del sistema en el establecimiento.

1.1.3. Mejora a lo ya existente

Al momento de la elaboración del proyecto [BubbleTown®](#) cuenta con una aplicación móvil para recomendar bebidas basadas en redes neuronales profundas [10] del que se ahondará en el capítulo [Estado del Arte](#).

El nuevo sistema permitirá que se genere publicidad dirigida para cualquier peatón que pase por el exterior del establecimiento, haciendo que llegue publicidad a una mayor cantidad de personas sin necesidad de que ingresen al local.

1.2. Planteamiento del problema

Si bien la calidad de los productos son un punto importante para los clientes, hoy en día la experiencia que se le brinda al usuario durante la adquisición de cualquier producto se vuelve un punto crucial. Considerando también que en México las PyMEs requieren tener diferenciadores de su competencia para no desaparecer es que la empresa [BubbleTown®](#) está interesada en aplicar las nuevas tecnologías del campo de la ingeniería en sistemas computacionales para brindarle a los clientes una experiencia inédita.

Con el uso de nuevas áreas de estudio como lo es el Deep Learning se pueden desarrollar herramientas que generen publicidad personalizada en función de características como la temperatura ambiental, el género, edad y dimensiones de la personalidad de las personas que se coloquen afuera del establecimiento. Con esto se generará una recomendación a partir del menú de bebidas de [BubbleTown®](#) y se visualizará con el apoyo de una pantalla y la aplicación de realidad aumentada para generar publicidad atractiva que invite a los usuarios a conocer y probar los productos de la empresa mexicana.

1.3. Hipótesis

Por medio de técnicas de *Deep Learning* y el análisis de algunas características inherentes a los clientes de [BubbleTown®](#), como por ejemplo la edad, el género y la personalidad a partir del análisis de una fotografía de su rostro, además de integrar factores externos, como la temperatura ambiental, el sistema desarrollado en el presente trabajo generará publicidad dirigida, distinta y personalizada para cada uno de ellos, en forma de una recomendación de un producto de forma poco convencional, sin la necesidad de que ingresen al establecimiento. De esta forma, la

experiencia de los posibles consumidores mejorará, además de que la retención de su atención será mucho más fácil de captar y su satisfacción crecerá, lo que se traduce en un aumento significativo de ventas para la empresa con respecto a las cantidades actuales. No se puede ignorar la gran ventaja competitiva que este trabajo le otorgará a la marca **BubbleTown®** en el negocio de bebidas, pues impulsará la popularidad y presencia de su marca y productos entre aquellos posibles clientes que formen parte de su mercado objetivo. Incluso, podrá ser capaz de revolucionar la forma en que se realiza la publicidad para la venta de bienes o productos comerciales.

1.4. Objetivos

1.4.1. Objetivo general

Implementar un prototipo de sistema que permita generar publicidad dirigida a clientes en el establecimiento de bebidas a base de té **BubbleTown®**, haciendo uso de aprendizaje profundo y realidad aumentada.

1.4.2. Objetivos específicos

- Instalar los dispositivos del sistema en el establecimiento de **BubbleTown®** para la operación del prototipo en un ambiente comercial real.
- Detectar el rostro de una persona a partir de una imagen, obtenida por medio de video capturado por fuera del establecimiento de **BubbleTown®**.
- Generar u obtener un conjunto de datos constituido por imágenes del rostro de personas, para el entrenamiento y las pruebas de las redes neuronales.
- Diseñar y entrenar arquitecturas convolucionales de las redes neuronales para obtener estimaciones de atributos de una persona a partir de su rostro, y con base en ellos, generar recomendaciones de bebidas para esta.
- Implementar tecnologías y métodos de realidad aumentada para proyectar las recomendaciones obtenidas en una pantalla publicitaria por fuera del establecimiento, en donde las personas transiten.
- Hacer pruebas del sistema con datos reales de posibles clientes potenciales de **BubbleTown®**, para evaluar el funcionamiento del prototipo de sistema por medio de la satisfacción de los mismos con su recomendación obtenida.

CAPÍTULO 2

Marco teórico

2.1. Introducción al capítulo

Como el principal objetivo de este trabajo está relacionado con el marketing y la publicidad, en la sección de [Personalización en el marketing](#) se habla sobre las ventajas, desventajas, el potencial y el riesgo de personalizar productos y servicios para cada uno de los clientes de una empresa, en función de la comprensión de sus sentimientos, deseos, necesidades, y en general, toda la información disponible (o la mayor cantidad que sea posible obtener sin invadir su privacidad) de ellos. Adicionalmente se habla sobre un marco conceptual que puede ser una referencia para la puesta en ejecución del prototipo: la automatización del marketing.

Al centrarse en un análisis facial para determinar la personalidad del individuo, se debe tener la capacidad de reconocer los rostros dentro de un entorno digital, en este caso una imagen, por ello en la sección [Detección de objetos](#) se da un recorrido por distintas técnicas que se han utilizado para el reconocimiento de objetos, en esta misma sección se habla de cómo identificar el objeto de nuestro interés, el rostro de la persona.

Como comprender al cliente se vuelve un tema crucial, en este capítulo se abordarán las tres características que cobran relevancia y su importancia a lo largo de las secciones [Género](#), [Edad](#) y [Personalidad: Big Five](#). Estas características se han seleccionado porque, como indica la literatura, determinan el comportamiento de un cliente, sin transgredir su privacidad, ya que se usarán técnicas no invasivas para recolectarlas. La razón por la que con estas características no se viola la privacidad de los clientes se explicará en la sección [Biometría blanda](#).

Dentro del área del marketing el conocer a los clientes es una labor crucial, un ejemplo de esta importancia es el estudio de mercado que cada empresa hace para conocer de mejor manera a su público objetivo y construir su publicidad en función de este mercado. Entre más datos se tengan del cliente es más fácil brindarle una recomendación más adecuada [11]. Por ello los especialistas en marketing suelen dirigir sus anuncios a un solo grupo específico usando imágenes icónicas que estén directamente relacionadas con sus experiencias [11].

Distintos trabajos de la literatura indican que la elección de distintos sabores de comida es determinado por el género de la persona, por ello es de interés poderlo reconocer a través de una imagen, pues es la forma menos invasiva de saberlo. En la sección [Género](#) se toma este tema y se muestra distintas formas de lograr la tarea.

La tarea para obtener estos datos no será a través de técnicas tradicionales, sino que se usarán [Redes neuronales](#), sistemas que en la actualidad han cobrado importancia en el área de los sistemas computacionales por su capacidad de "aprender". Igualmente se abordarán varias arquitecturas que auxiliarán a llevar acabo la tarea dentro de las secciones [Redes neuronales convolucionales](#), [VGG16](#), [Single Shot Detection](#) y [MobileNet](#).

En la sección de [Personalidad: Big Five](#) se aborda una de las teorías que mayor aceptación ha tenido en la comunidad especializada para medir de una forma categórica la personalidad de un ser humano: el modelo de los Cinco Grandes, o Big Five. Se habla de como surge y cuales son los instrumentos de medición convencionales, además de qué tan bueno es el análisis facial para estimar el Big Five de una persona.

Una vez que se analiza el [Género](#), [Edad](#) y [Personalidad: Big Five](#) de la persona según su imagen, es momento de hacer la recomendación con base a estos parámetros, por ello en la sección [Recomendador](#) se abordan los formas de realizar un recomendador como es el caso de [Recomendación basada en contenido](#) o [Filtrado colaborativo](#) y algunos algoritmos que se usan para generar la recomendación como pueden ser los [Árbol de decisión](#), [Clasificador bayesiano ingenuo](#) y [KNN \(vecino más cercano\)](#).

La forma en que se presente la publicidad es crucial para este trabajo, por lo que en la sección de [Realidad Aumentada](#) se habla sobre esta tecnología, algunas de las técnicas más implementadas, qué componentes y funcionalidad básica debe de tener un sistema para ser considerado de realidad aumentada.

Al final del capítulo se hace un resumen y discusión de lo más relevante visto en cada una de las secciones abarcadas en este.

2.2. Personalización en el marketing

En un mundo donde los clientes están constantemente conectados a una gran cantidad de información e inteligencia lista para usar, las empresas optan cada vez más por esfuerzos de marketing individualizados, en lugar de esfuerzos masivos [12].

Dado que la competencia entre los productos y servicios es intensa, las empresas han adoptado estrategias de diferenciación para atraer y retener clientes. Una estrategia de diferenciación común es personalizar productos o servicios para satisfacer mejor las necesidades de cada cliente [13]. La adopción de esta estrategia se asocia con un aumento del 26 % en la rentabilidad y un aumento del 12 % en la capitalización de mercado [12].

El marketing uno a uno se centra en el cliente individual y se basa en la idea de que la empresa debe conocer a su cliente. A través de sus relaciones con ese cliente, la empresa puede aprender cómo ese cliente quiere que le traten, lo que le permite tratarle de forma diferente a como trata a otros clientes [13].

La personalización es el procedimiento de recopilación de información de los clientes que ayuda a la empresa a crear productos y servicios que satisfagan perfectamente los deseos y necesidades del cliente [14]. Los datos juegan un papel clave, ya que la personalización sólo es posible si se dispone de datos confiables y proyectables del cliente. Una empresa decide, generalmente basándose en datos de clientes recopilados previamente, qué combinación de marketing es la más adecuada para un cliente individual [13] [15].

El éxito depende de la precisión; los especialistas en marketing y gerentes deben considerar lo que recomiendan a sus clientes al asegurarse de ofrecer el producto o los servicios correctos a la persona adecuada [14].

La personalización de un producto primero invita a cada consumidor individual a revelar su preferencia y luego se produce y entrega un producto con la mayor coincidencia posible [15]. Las empresas no solo buscan aumentar la

probabilidad de compra sino también estimular al cliente a través de la sorpresa de la personalización dinámica en tiempo real [12].

La proliferación de productos o customización no mantiene diálogos uno a uno con los consumidores, en cambio, ofrece muchas variantes y el cliente elige el producto más atractivo [15]. Los grandes aumentos en el poder de cómputo, la robótica de fabricación y el auge de Internet en las últimas décadas le han dado a los especialistas en marketing el poder de personalizar las ofertas a clientes cada vez más exigentes, de formas que antes no podían [16].

2.2.1. Ventajas y desventajas

Las ventajas de la personalización en el marketing son las siguientes:

- Incremento de la lealtad: Un cliente leal realizará más transacciones comerciales con la empresa incluso teniendo ofertas más interesantes de la competencia.
- Mayores ganancias: El conocimiento de más datos del cliente permite ampliar las opciones de venta más allá de la simple transacción comercial inicial.
 - Up-selling: Incita al cliente a comprar algo más de lo que en principio tenía previsto.
 - Cross-selling: Se puede realizar una promoción personalizada para ese cliente de servicios complementarios que él no ha pedido pero que le pueden interesar.
- Captación de clientes de valor: Ayuda a crear clientes que aportan rentabilidad a la relación que se mantiene con ellos.
- Optimización de las estrategias de marketing: El mercado objetivo está mejor definido, la mayoría de mensajes publicitarios llegarán a las personas adecuadas.
- Disminución de costos: Las campañas de publicidad exigen menos recursos, la atención al cliente es más fácil porque se sabe qué necesidades tienen dichos clientes.
- Mejor posicionamiento en el mercado: Se tiene un concepto más claro de lo que es la empresa y que productos vende.

Por el contrario, ocasionalmente los clientes pueden percibir ciertos procesos de personalización como demasiado complejos, lo que puede disminuir su utilidad para la empresa. Algunos simplemente no están dispuestos a invertir el tiempo necesario para personalizar productos, no están seguros de lo que se espera de ellos o simplemente no saben cómo personalizar según sus preferencias. A otros no les gusta que les digan qué hacer y reaccionan negativamente a tales esfuerzos [12].

Los clientes con poca familiaridad pueden tener dificultades para aprovechar al máximo las propuestas de personalización debido a la incertidumbre sobre sus preferencias para las recomendaciones de una empresa o su incapacidad para expresar estas preferencias, el número de elecciones que tienen que hacer, y la cantidad de información que tienen que procesar para tomar una decisión informada [12].

La sobrecarga de información es posiblemente el riesgo más grande en la personalización. Esta se presenta cuando los usuarios reciben más información de la que pueden manejar dentro del tiempo disponible, provocando que no puedan localizar lo que más necesitan y tampoco usen la información relevante [13], resultando en elecciones de baja calidad o que la compra no se lleve a cabo [15].

Otros de los desafíos y riesgos que existen al personalizar productos y servicios son los siguientes:

- Fácilmente se puede herir al consumidor y se puede invadir su privacidad [17].
- Si el costo de una personalización errónea es grande, es mejor para la empresa no personalizar [15] y pueden conducir a experiencias negativas para el cliente [12].

- El costo de la personalización es mayormente elevado debido al software que es necesario para la recolección y extracción de datos [14].
- El análisis de los datos es complicado y requiere de gran intuición para predecir lo que los clientes desean realmente.
- Requiere una estrecha coordinación entre marketing, tecnología de la información y producción, pudiendo superar las capacidades de la empresa [15].
- Al principio, las empresas tienen poca información detallada sobre los clientes, una condición conocida como el problema de arranque en frío [12].

2.2.2. Los deseos y la perspectiva del consumidor

El objetivo fundamental de la personalización radica en aumentar la tasa de retención de clientes al proporcionar un valor competitivo a estos, ya que esta es un factor importante para atraer nuevos usuarios y tiene el efecto de aumentar la satisfacción, confianza y lealtad del cliente, tanto directa como indirectamente [13].

Muchas empresas aprovechan la personalización para aumentar sus clientes leales, particularmente en la industria de servicios. La personalización se utiliza como una táctica y tratan de mejorar la satisfacción de los clientes para obtener más ganancias [14].

Como se observa en el modelo conceptual de la Figura 2.1, obtenido del estudio en [16], la personalización de un producto o servicio genera una serie de sentimientos en el consumidor, siempre y cuando esta sea aplicada y realizada correctamente, en donde se pueden obtener las siguientes premisas:

1. La **comunicación** causa **personalización**.
2. La **personalización** causa **confianza**.
3. La **personalización** causa **satisfacción** [16].
4. La **satisfacción** causa **retención** en el cliente y **lealtad** [14].

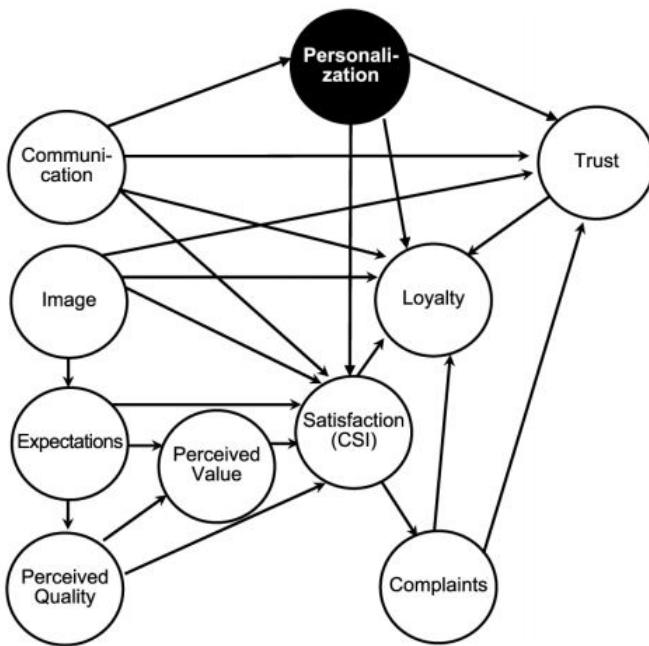


Figura 2.1: Diagrama tomado de [16] que representa el modelo conceptual de los sentimientos que la personalización de un producto o servicio (junto a la comunicación) logra generar en una persona.

La satisfacción afecta la retención del cliente, y es un determinante significativo en las ventas repetidas y su lealtad. Los clientes satisfechos regresan y compran más, y le cuentan a otras personas sobre sus experiencias, lo que lleva a una posición competitiva más fuerte, resultando en una mayor participación en el mercado y ganancias [13]. El costo de crear nuevos clientes es mayor que el costo de mantenerlos [14].

La satisfacción, la confianza, la lealtad emocional y la rentabilidad del cliente deben medirse a través de métodos de observación y encuesta para evaluar los efectos de las estrategias de personalización [16].

Los consumidores a menudo construyen sus preferencias "sobre la marcha" en función de la tarea y las características contextuales, incluida la facilidad con la que se pueden evaluar los atributos, formato de información, modo de respuesta, los atributos particulares utilizados para recomendar productos y las alternativas particulares disponibles para su consideración en un momento dado [15].

Las preferencias del consumidor pueden ser estables o evolucionar de manera predecible. La estabilidad de preferencias es crítica para la personalización porque las elecciones anteriores se utilizan para predecir elecciones futuras. Estas son más estables cuando el entorno de información no cambia y cuando los consumidores han hecho elecciones repetidas en una categoría de producto [15].

La relación entre la empresa y el cliente se va volviendo más estrecha en virtud del uso apropiado del conocimiento de la empresa sobre el cliente, y la creciente confianza y lealtad emocional del cliente, por lo que una publicidad menos estandarizada es necesaria para este tipo de clientes [16].

El peligro para la personalización del producto es que los clientes pueden darse cuenta después de diseñar su producto "ideal" de que sus preferencias reales corresponden más estrechamente a los productos estandarizados. Una forma de reducir esto es proporcionar a los compradores herramientas interactivas que les permitan visualizar y experimentar productos personalizados antes de la compra o aprender de las experiencias de otros [15].

Puede ser más rentable ofrecer a los consumidores un solo producto porque ofrecer múltiples opciones puede conducir a pérdidas percibidas en la mente de los consumidores [15]. Los clientes pueden estar tan satisfechos con los

productos que parecen personalizados incluso si las diferencias con los productos estandarizados son mínimas.

2.2.3. Automatización del marketing

La automatización del marketing es un marco conceptual que promete una mayor productividad, una mejor toma de decisiones y mayores rendimientos en las inversiones. Sugiere la implementación de reglas de decisión en tiempo real que se calibran utilizando datos históricos y se actualizan mediante la experimentación adaptativa. [18].

La automatización de marketing se dirige a los clientes con contenido personalizado y ofertas personalizadas, como cupones o descuentos. Los clientes muestran una mayor participación y prestan más atención a la comunicación de la marca debido a la mayor relevancia de la información proporcionada, y las empresas pueden mejorar su tasa de conversión, up-selling y cross-selling, y de retención [18].

En la Figura 2.2 se presenta este marco conceptual: la información almacenada y la información actual son de vital importancia, pues por medio de un disparador, se le aplica un conjunto de reglas o condiciones al producto o servicio, el cual es presentado al usuario por medio de un medio, donde la mayoría de veces es una interfaz o recurso gráfico.

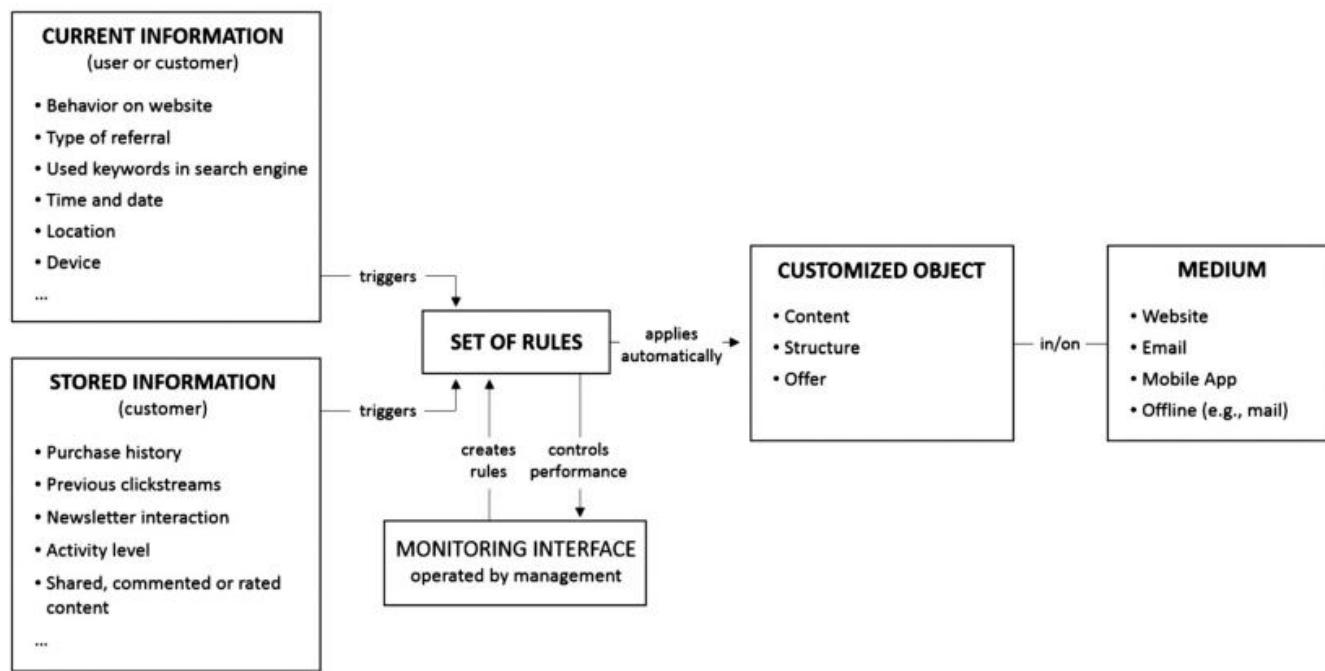


Figura 2.2: Diagrama tomado de [18] que muestra el marco conceptual de la automatización del marketing.

Todas las acciones de marketing automatizadas son una respuesta directa a la información existente, entrante o cambiante del cliente mediante reglas simples: si un cliente muestra un cierto comportamiento, se inicia una acción de marketing predefinida. La mayor ganancia radica en la ejecución automatizada de estas acciones para miles de eventos una vez que se determina una regla [18].

2.3. Detección de objetos

Se entiende por reconocimiento de objetos a la tarea para encontrar e identificar objetos en una imagen o secuencia de video. Los humanos reconocemos una multitud de objetos en imágenes con poco esfuerzo, a pesar del hecho que la imagen del objeto puede variar un poco en diferentes puntos de vista, en diferentes tamaños o escala e incluso cuando están trasladados o rotados. Los objetos pueden ser reconocidos cuando están parcialmente obstruidos desde una vista. No obstante esta tarea es un desafío para los sistemas de visión por computadoras. Para este problema se han implementado muchos métodos durante múltiples décadas [19].

¿Por qué es tan difícil el reconocimiento? El mundo real está hecho de una mezcla de objetos, todos sobre puestos entre sí y aparecen en diferentes poses. Además, la variabilidad intrínseca dentro de una clase (por ejemplo, perros), debido a la articulación compleja no rígida y las variaciones extremas en la forma y la apariencia hace que sea poco probable que podamos realizar una comparación exhaustiva con una base de datos de ejemplos. El problema de reconocimiento puede desglosarse a lo largo de varios ejes. Por ejemplo, si se sabe lo que estamos buscando, implica escanear rápidamente una imagen para determinar dónde puede ocurrir una coincidencia. Si se tiene un objeto rígido específico que se está tratando de reconocer, podemos buscar puntos característicos y verificar que se alineen de una manera geométricamente posible [20].

Si bien la detección de objetos es un tema que es aplicable a diversos problemas con distintos propósitos, el objetivo principal es lograr identificar los rostros existentes dentro de una imagen, por ello a continuación se recapitulan las principales formas del reconocimiento de objetos, partiendo por las técnicas tradicionales hasta los métodos de esta época.

2.3.1. Sustracción de fondo

La sustracción de fondo es el método de separar los objetos del primer plano del fondo que aparece en una secuencia de frames de un vídeo y es altamente usado en sistemas que deban detectar el movimiento de objetos. En la Figura 2.3 se observa el algoritmo general que lleva la sustracción de fondo. Podemos dividirlo en dos grandes categorías, técnica no recursiva y recursiva.

2.3.1.1. Técnica no recursiva

Usualmente se utiliza una técnica de deslizamiento de los distintos frames para tener una estimación del fondo. Esta técnica debe mantener en memoria los frames de vídeo previos lo que ocasiona un gran consumo de recursos, en especial memoria. Algunos ejemplos son diferenciación de frames, median filter, linear filter, entre otros [21].

2.3.1.2. Técnica recursiva

No mantiene el fondo en memoria pero actualiza un fondo simple en cada nuevo frame del vídeo haciendo que ocupe un mínimo de memoria a comparación de una técnica no recursiva, por ende lo vuelve computacionalmente eficiente. Algunos ejemplos son Kalman filter, Mixture of Gausian (MoG), entre otros [21].

El método para detectar el objeto y clasificarlo utiliza un proceso de comparación entre el frame de entrada del vídeo y el modelo previamente obtenido del fondo, identifica los valores cambiantes de los píxeles [21].

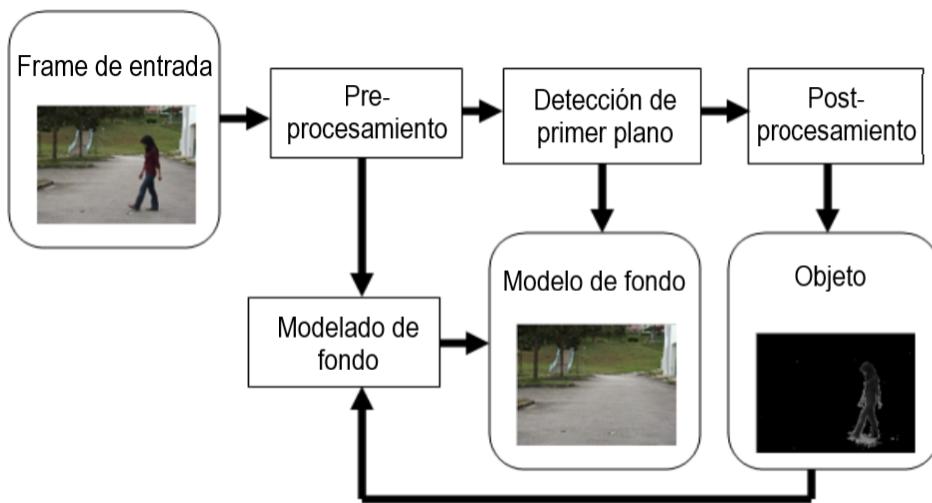


Figura 2.3: Diagrama de flujo del algoritmo general de sustracción de fondo. Imagen tomada de [21].

2.3.2. Detección de bordes

La detección de bordes es una de las operaciones más utilizadas en el análisis de imágenes. La razón de esto es que los bordes forman el contorno de un objeto, en el sentido genérico. Los objetos son temas de interés en el análisis de imágenes y los sistemas de visión por computadora. Un borde es el límite entre un objeto y el fondo, e indica el límite entre los objetos superpuestos. Esto significa que si los bordes de una imagen se pueden identificar con precisión, se pueden ubicar todos los objetos y se pueden medir propiedades básicas como el área, el perímetro y la forma. Dado que la visión por computadora implica la identificación y clasificación de objetos en una imagen, la detección de bordes es una herramienta esencial [22].

La detección de bordes ha pasado por una evolución que abarca más de 30 años. Dos métodos principales de detección de bordes han sido evidentes durante este período, el primero de ellos es el enfoque de coincidencia de plantillas (TM) y el segundo es el enfoque de gradiente diferencial (DG). En cualquier caso, el objetivo es encontrar dónde la magnitud del gradiente de intensidad g es lo suficientemente grande como para ser tomada como un indicador confiable del borde de un objeto, así encontrar los mínimos globales del histograma [22].

Por ejemplo, en la Figura 2.4 se muestra una imagen con un objeto (llave perico), el propósito es detectarlo usando una técnica de binarización.



Figura 2.4: Llave perico en escala de grises.

Al obtener su histograma (Figura 2.5) es posible observar claramente la existencia de un mínimo global y por consiguiente es posible encontrar el objeto estableciendo un umbral de 210. La aplicación de esta técnica consta simplemente de modificar el valor de cada píxel siguiendo la ecuación 2.1.

$$ValorNuevoDelPíxel = \begin{cases} 0 & \text{si } valorActualDelPixel < umbral \\ 1 & \text{si } valorActualDelPixel \geq umbral \end{cases} \quad (2.1)$$

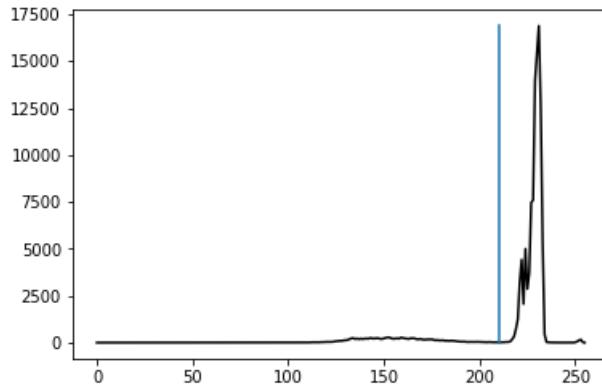


Figura 2.5: Histograma de la Figura 2.4 el cual muestra el umbral de 210 (línea azul) sobre el mínimo global.

Esta técnica muestra el resultado en la Figura 2.6, donde se aprecia una correcta segmentación del objeto al cual se le podría aplicar algún procesamiento extra para su mejora, el caso de este ejemplo es mostrar la aplicación sencilla de la técnica.



Figura 2.6: Objeto segmentado.

Como de costumbre, hay varias definiciones posibles de un borde, cada una aplicable en varias circunstancias específicas. Una de las definiciones más comunes y generales es el borde de paso ideal, ilustrado en la Figura 2.7 [22].

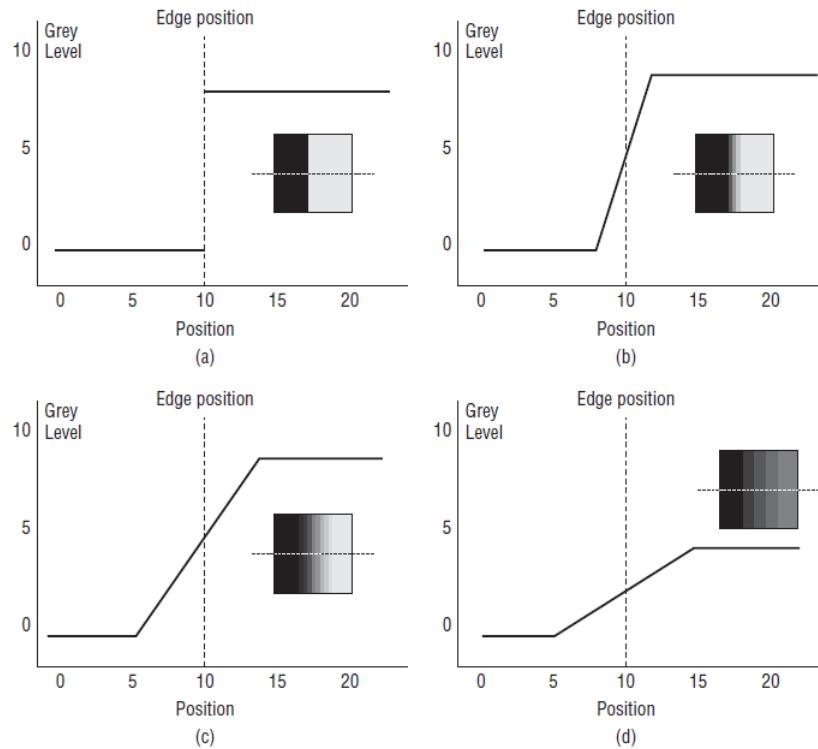


Figura 2.7: Bordes escalonados. (a) El cambio de nivel ocurre exactamente en el píxel 10. (b) El mismo cambio de nivel que antes, pero más de 4 píxeles centrados en el píxel 10. Este es un borde de rampa. (c) Mismo cambio de nivel pero más de 10 píxeles, centrado en 10. (d) Un cambio más pequeño en 10 píxeles. Imagen tomada de [22].

Es claro que esta manera de detectar un objeto tiene muchas limitaciones, en especial depende de un fondo claro y de un cálculo correcto de umbral. Otra complicación ocurre debido a la digitalización. Es poco probable que la imagen se muestre de tal manera que todos los bordes correspondan exactamente con un límite de píxeles. De hecho, el cambio de nivel puede extenderse a través de cierto número de píxeles. Se considera que la posición real del borde es el centro de la rampa que conecta el nivel bajo de gris con el alto. Esta es una rampa solo en el mundo matemático, ya que después de que la imagen se haya digitalizado (muestreada), la rampa tiene la apariencia irregular de una escalera [22].

No se debe olvidar el problema del ruido. Debido a una gran cantidad de factores como la intensidad de la luz, el tipo de cámara y lente, el movimiento, la temperatura, los efectos atmosféricos, el polvo y otros que afectan a la captura de la escena, es muy poco probable que dos píxeles que correspondan exactamente al mismo nivel de gris en la escena tengan mismo nivel en la imagen. El ruido es un efecto aleatorio y solo se puede caracterizar estadísticamente [22].

La utilización de distintos filtros calculados matemáticamente ha dado buenos resultados, entre estos se puede mencionar principalmente el filtro de Roberts (Figura 2.8), Sobel (Figura 2.9) y Canny.

$$R_{x'} = \begin{bmatrix} 0 & 1 \\ -1 & 0 \end{bmatrix} \quad R_{y'} = \begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix}$$

Figura 2.8: Filtro Roberts de dimensión 2x2 [20].

$$S_x = \begin{bmatrix} -1 & 0 & 1 \\ -2 & 0 & 2 \\ -1 & 0 & 1 \end{bmatrix} \quad S_y = \begin{bmatrix} 1 & 2 & 1 \\ 0 & 0 & 0 \\ -1 & -2 & -1 \end{bmatrix}$$

Figura 2.9: Filtro Sobel de dimensión 3x3 [20].

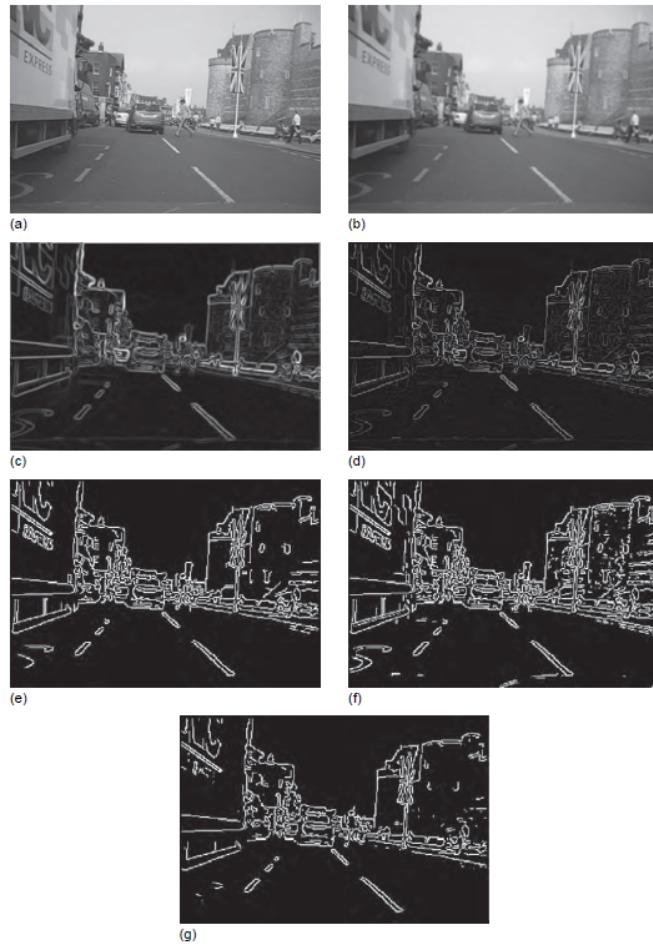


Figura 2.10: Aplicación del detector de bordes Canny. (a) Imagen original. (b) Imagen suavizada. (c) Resultado de la aplicación del operador Sobel. (d) Resultado de la supresión no máxima. (e) Resultado del umbral de histéresis. (f) Resultado de la umbralización solo en el nivel de umbral inferior. (g) Resultado de la umbralización en el nivel de umbral superior. Tenga en cuenta que hay menos salidas falsas o engañosas en (e) que las que resultaría del uso de un solo umbral. Imagen tomada de [20].

Hasta hace poco, el progreso en la visión por computadora se basaba en características de ingeniería manual. Sin embargo, la ingeniería de características es difícil, requiere mucho tiempo y un conocimiento experto sobre el dominio del problema. El otro problema con las funciones diseñadas a mano, como la sustracción de fondo y detección de bordes, es que son demasiado escasos en términos de información que pueden capturar de una imagen [23]. Estas técnicas pueden considerarse clásicas del análisis de imágenes ya que llevan siendo aplicadas desde hace muchas décadas y

aunque han aportado grandes avances tienen limitaciones han forzado a la creación de nuevas maneras de conseguir el objetivo de la detección de objetos, en la sección de [Redes neuronales convolucionales](#) se hablará de una herramienta más reciente y que ha mostrado mejorar los resultados de las antes mencionadas.

2.4. Redes neuronales

Las redes neuronales son más que otra forma de emular ciertas características propias de los humanos, como la capacidad de memorizar y de asociar hechos. Si se examinan con atención aquellos problemas que no pueden expresarse a través de un algoritmo, se observará que todos ellos tienen una característica en común: la experiencia. El hombre es capaz de resolver estas situaciones acudiendo a la experiencia acumulada. Así, parece claro que una forma de aproximarse al problema consiste en la construcción de sistemas que sean capaces de reproducir esta característica humana. En definitiva, las redes neuronales no son más que un modelo artificial y simplificado del cerebro humano, que es el ejemplo más perfecto del que se dispone para un sistema que es capaz de adquirir conocimiento a través de la experiencia [24].

Existen numerosas formas de definir a las redes neuronales; desde las definiciones cortas y genéricas hasta las que intentan explicar más detalladamente qué son las redes neuronales. Por ejemplo:

1. Una nueva forma de computación, inspirada en modelos biológicos [24].
2. Un modelo matemático compuesto por un gran número de elementos procesales organizados en niveles [24].
3. Un sistema de computación compuesto por un gran número de elementos simples, elementos de procesos muy interconectados, los cuales procesan información por medio de su estado dinámico como respuesta a entradas externas [24].
4. Las redes neuronales artificiales son redes interconectadas masivamente en paralelo de elementos simples (usualmente adaptativos) y con organización jerárquica, las cuales intentan interactuar con los objetos del mundo real del mismo modo que lo hace el sistema nervioso biológico [24].

2.4.1. La neurona biológica

Una red neuronal es un nuevo sistema para el tratamiento de la información, cuya unidad básica de procesamiento está inspirada en la célula fundamental del sistema nervioso humano: la neurona [24]. Estas neuronas tienen tres componentes principales de nuestro interés: las dendritas, el cuerpo celular y el axón. Las dendritas son redes receptoras en forma de árbol de fibras nerviosas que llevan señales eléctricas al cuerpo celular. El cuerpo celular produce una señal de salida, el axón es una fibra larga que transporta la señal del cuerpo celular a otras neuronas. El punto de contacto entre un axón de una célula y una dendrita de otra célula se llama sinapsis. Es la disposición de las neuronas y las fuerzas de las sinapsis individuales, determinadas por un proceso químico complejo, lo que establece la función de la red neuronal [25]. Básicamente la neurona es estimulada o excitada a través de sus entradas y cuando se alcanza un cierto umbral, la neurona se activa, pasando una señal hacia el axón. En la Figura 2.11 se puede apreciar un diagrama esquemático simplificado de dos neuronas biológicas.

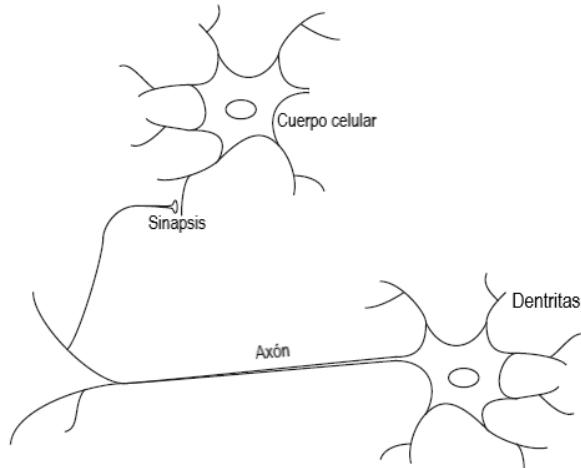


Figura 2.11: Diagrama de dos neuronas biológicas. Imagen tomada de [25].

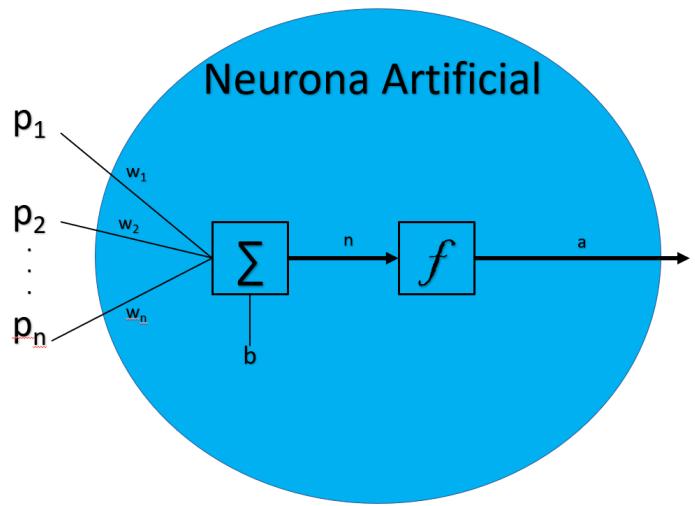


Figura 2.12: Neurona artificial.

La neurona biológica llevó a la creación de la neurona artificial, que se aprecia en la Figura 2.12, donde se puede observar de forma clara cómo está constituida una neurona artificial simple. Estos componentes se explican en la sección de [25] de la siguiente forma:

- Entrada (p_n): Como su nombre lo indica, p_n es un escalar que representa un elemento de la entrada con el que la neurona es alimentada. Al total de elementos a la entrada se les denomina P .
- Pesos (w_n): w_n es un número escalar que representa la fuerza de la sinapsis en la neurona biológica.
- Bias (b): Elemento extra que no se tiene en una neurona biológica pero permite a las redes neuronales artificiales ser más poderosas. Otra forma en la que se hace referencia al bias es como *offset*.
- Sumatoria (Σ): Representa el cuerpo celular de la neurona biológica.
- Función de activación ($f(\cdot)$): Es la representación de la señal emitida por el axón en la neurona biológica. De este componente se hablará más en [Función de activación](#)

Una vez que se entiende qué busca representar cada componente de una neurona artificial, el siguiente paso consiste en explicar cómo interactúan entre sí para llegar a un resultado (a en la Figura 2.12) a partir de las entradas p_n . En la ecuación 2.2 se puede apreciar que la entrada (p_n) es multiplicada por su peso asociado (w_n), posteriormente se le suma el bias (b) y es ingresada a la función de activación ($f(\cdot)$).

$$a = f(wp + b) \quad (2.2)$$

2.4.2. Función de activación

Una neurona biológica puede estar activa (excitada) o inactiva (no excitada), es decir, que tiene un “estado de activación”. Las neuronas artificiales también tienen diferentes estados de activación; algunas de ellas solamente dos, al igual que las biológicas, pero otras pueden tomar cualquier valor dentro de un conjunto determinado [24]. La función

de activación calcula el estado de actividad de una neurona, esta puede ser una función lineal o no lineal y se elige particularmente para satisfacer alguna especificación del problema que la neurona está intentando resolver [25]. Algunas de las funciones de activación más comunes, así como la relación que tienen entre la entrada y salida, se pueden observar en la Tabla 2.1

Nombre de función	Relación entrada/salida
Límite duro	$a = 0$ para $n < 0$ $a = 1$ para $n \geq 0$
Límite duro simétrico	$a = -1$ para $n < 0$ $a = +1$ para $n \geq 0$
Lineal	$a = n$
Lineal saturante	$a = 0$ para $n < 0$ $a = n$ para $0 \leq n \leq 1$ $a = 1$ para $n > 1$
Saturación simétrica lineal	$a = -1$ para $n < -1$ $a = n$ para $-1 \leq n \leq 1$ $a = 1$ para $n > 1$
Log-sigmoide	$a = \frac{e^n - e^{-n}}{e^n + e^{-n}}$
Tangente hiperbólica sigmoidea	$a = \frac{1}{1+e^{-n}}$
Positiva lineal (RELu)	$a = 0$ para $n < 0$ $a = n$ para $0 \leq n$
Competitiva	$a = 1$ Neurona con el máximo n $a = 0$ Todas las demás neuronas

Tabla 2.1: Funciones de activación comunes. Tabla tomada de [25].

2.4.3. Reglas de aprendizaje

Según la Real Academia Española, aprendizaje se define como la "adquisición por la práctica de una conducta duradera". Esta definición es muy ambigua como para poder llevarla directo al ámbito de las RNN.

En términos muy generales, una regla de aprendizaje supervisada permite a una red neuronal calcular nuevos valores de pesos y bias comparando el valor esperado a la salida (targets) con la salida de la red usando los valores actuales de esos pesos (W) y bias (b).

Para implementar una red neuronal se requiere de su modelo matemático y para lograr que su aprendizaje sea automático se necesita de una regla de aprendizaje que está definida en ecuaciones matemáticas. Se tienen que encontrar ecuaciones que satisfagan el propósito de forma adecuada y eficiente. Algunos ejemplos de reglas de aprendizaje en la literatura son el Least Mean Square y el Back propagation [25].

2.4.4. Partes de una red neuronal

Usualmente para tareas complicadas se requerirán más de una neurona para realizar la tarea de forma satisfactoria. Las redes neuronales comprenden una jerarquía de niveles de procesamiento, donde cada nivel se denomina "capa de red" y consta de una serie de neuronas de procesamiento que operan en paralelo (también llamadas nodos) [23] [25]. Típicamente, la entrada se alimenta a través de una capa de entrada, las capas intermedias realizan el procesamiento (se denominan capas ocultas) y nos proporciona un resultado a través de la "capa de salida" [23].

Una **arquitectura de red neuronal** hace referencia al número de capas que la componen, la forma en que las distintas neuronas están conectadas, así como sus pesos y como la forma en que cada neurona se conecta con el resto. A continuación se puede ver, en la Figura 2.13, un ejemplo de una arquitectura de red neuronal. La misma está constituida por neuronas interconectadas y arregladas en cuatro capas (esto último puede variar). Los datos ingresan por medio de la "capa de entrada", pasan a través de las "capas ocultas" (en el caso de la Figura 2.13 solo tiene una capa oculta) y salen por la "capa de salida" [24].

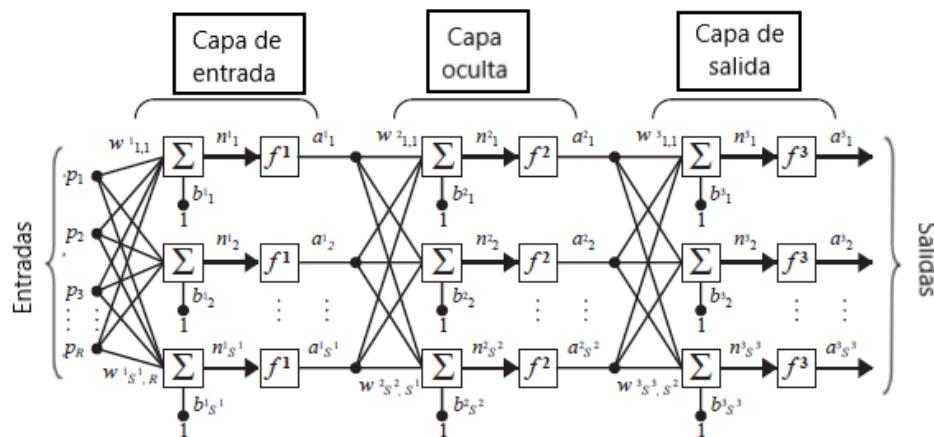


Figura 2.13: Ejemplo de una red neuronal artificial MLP. Imagen tomada de [25].

2.4.5. Tipo de aprendizaje

Es importante señalar que las redes neuronales tienen distintos métodos de aprendizaje. En este caso, solo se explicarán dos de ellos a continuación:

- **Aprendizaje supervisado:** es aquel en el que se conocen la entrada al sistema y las salidas deseadas (etiquetas verdaderas) y el modelo aprende un mapeo entre los dos [23].
- **Aprendizaje no supervisado:** en este no se conocen las etiquetas verdaderas para un conjunto dado de entradas y el modelo apunta a estimar la distribución subyacente de las muestras de datos de entrada [23].

2.4.6. Modos de operación

Una red neuronal funciona principalmente de dos modos: clasificador o regresor.

- **Clasificador:** Cuando la red es configurada en este modo, su labor consiste en separar los datos, pues como su nombre lo dice clasificará ciertos elementos en clases previamente definidas.
- **Regresor:** Por otra parte, cuando se configura en modo regresor, su objetivo será ajustar los datos.

2.5. El Perceptrón Multicapa

El primer modelo de Perceptrón fue desarrollado en un ambiente biológico imitando el funcionamiento del ojo humano respondiendo a señales ópticas y funcionaba de manera maravillosa, por lo que el perceptrón fue utilizado en diferentes ámbitos de investigación. Sin embargo, tenía limitaciones importantes, como la incapacidad de resolver problemas no lineales simples (como la función lógica XOR). Las matemáticas del libro Perceptrons eran indiscutibles y su tono dio el mensaje claro de que los perceptrones eran un camino sin salida, lo cual hizo que se perdiera el interés en el campo de las redes neuronales hasta la década de los 80, en la que el estudio de nuevas arquitecturas de redes y la mayor potencia de los ordenadores permitieron el diseño de redes muy eficaces en tareas en las que otros procedimientos de tipo simbólico encontraban dificultades [25].

Minsky y Papert mostraron en 1969 que la combinación de varios Perceptrones simples, incluso de neuronas ocultas, podría resultar en una solución adecuada para tratar ciertos problemas no lineales. Sin embargo, los autores no presentaron una solución al problema de cómo adaptar los pesos de la capa de entrada a la capa oculta, pues la regla de aprendizaje del perceptrón simple no podía aplicarse en este escenario. No obstante, Hinton y Williams en 1986 presentaron una manera de retropropagación (Back propagation) de los errores medidos en la salida de la red hacia las neuronas ocultas, dando lugar a la llamada regla delta generalizada y con esto el nacimiento del perceptrón multicapa que es capaz de ser usado en modo regresor o clasificador [25].

Si bien, el perceptrón multicapa (también conocido como MLP) ya tiene bastante tiempo de que fue presentado, hoy en día es una arquitectura válida para resolver problemas de aprendizaje supervisado y esta arquitectura, junto con su regla de aprendizaje, han sido la inspiración de muchas arquitecturas nuevas de redes neuronales. Un ejemplo de la arquitectura de un MLP se puede observar en la Figura 2.13 [25].

2.6. Redes neuronales convolucionales

Las CNN (Convolutional Neural Networks) son una de las categorías más populares de redes neuronales, especialmente para datos de alta dimensión (por ejemplo, imágenes y videos). Las CNN funcionan de una manera muy similar a las redes neuronales estándar. Sin embargo, una diferencia clave es que cada unidad en una capa CNN es un filtro bidimensional o de alta dimensionalidad que está conectada con la entrada de esa capa [23].

La CNN aprende a mapear una imagen dada a su categoría correspondiente mediante la detección de una serie de representaciones de características abstractas, que van desde simples hasta más complejas. Estas características discriminatorias se utilizan dentro de la red para predecir la categoría correcta de una imagen de entrada. El clasificador de la red neuronal es idéntico al MLP (explicado en la sección [El Perceptrón Multicapa](#)) [23]. El proceso en general de una CNN se representa en la Figura 2.14.

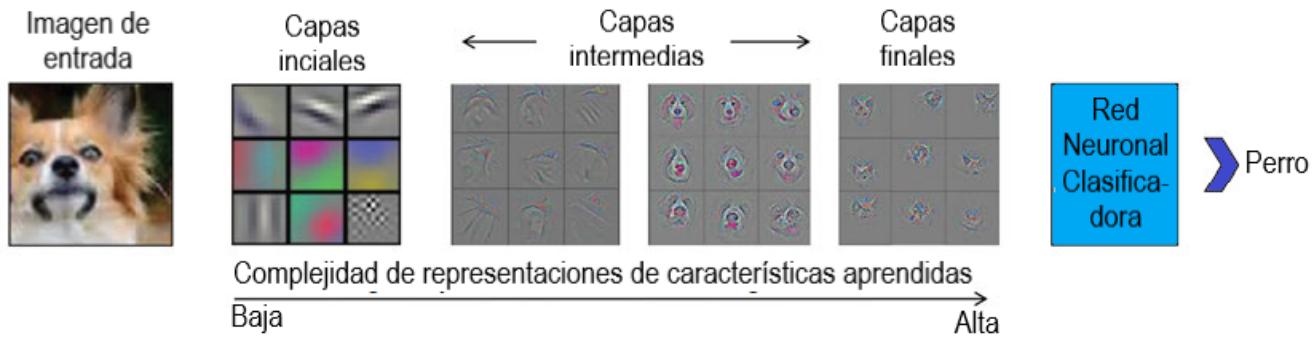


Figura 2.14: Diagrama general de una CNN. Imagen tomada de [23].

Una capa convolucional es el componente más importante de una CNN. Comprende un conjunto de filtros (también llamados núcleos convolucionales o kernels), un filtro en una capa convolucional es una cuadrícula de números discretos que están convolucionados con una entrada dada para generar un mapa de características de salida [23]. Este procedimiento se puede apreciar en la Figura 2.15, donde se muestra la convolución entre una matriz de 4×4 aplicando un filtro de tamaño 2×2 .

Considerese que los valores de los filtros (los números en la matriz) son los pesos de cada neurona y cambian durante el entrenamiento de la CNN. Este procedimiento de aprendizaje implica una inicialización aleatoria de los pesos del filtro al comienzo del entrenamiento. Luego, dados los pares de entrada-salida, los pesos del filtro se ajustan en varias iteraciones diferentes durante el procedimiento de aprendizaje [23].

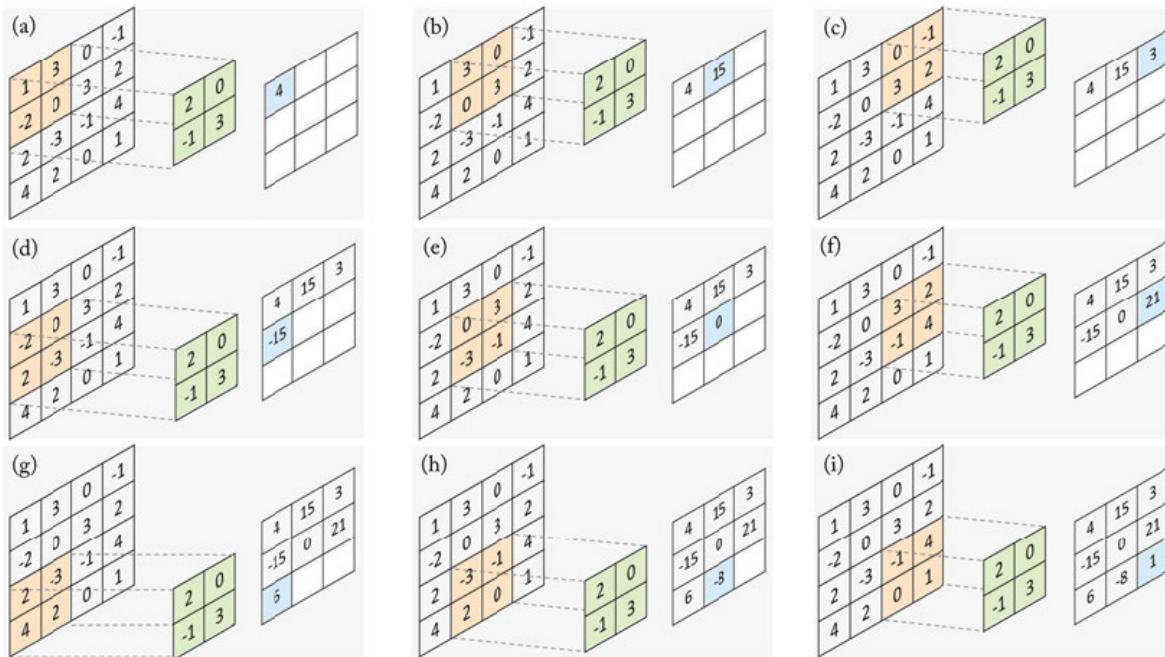


Figura 2.15: Ejemplo de la operación convolución con kernel de 2×2 en una matriz de 4×4 . Imagen tomada de [23].

Si bien el ejemplo representado en la Figura 2.15 sirve para mostrar la operación de convolución en un solo canal (una sola matriz) y un solo kernel, usualmente las imágenes se componen de 3 canales de colores, por ello cabe mencionar cómo se realiza la convolución con más de un canal, la entrada es una imagen con un ancho (w), alto (h) y profundidad (3 canales representados por el *RGB*), esta imagen se muestra en la Figura 2.16.

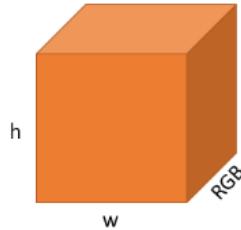


Figura 2.16: Imagen de entrada con de un ancho, alto y profundidad de 3 canales *RGB* (Red, Green, Blue).

Una vez que se establece la imagen de entrada es turno de proponer un kernel, la Figura 2.17 ejemplifica el kernel a usar, considere que si bien el w y h son distintos la profundidad debe ser la misma, en este caso se tiene una profundidad de 3 (RGB) canales.

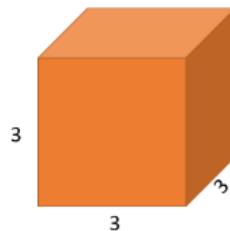


Figura 2.17: Kernel establecido con medidas de 3x3x3.

En la Figura 2.18, se muestra a grandes rasgos la convolución espacial, usualmente es nombrada así en la literatura ya que se mueve en todo el espacio de la imagen, es decir el kernel recorre todo el w y h de la imagen.

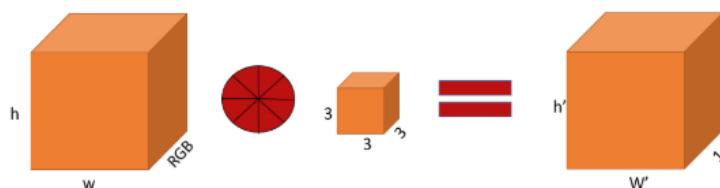


Figura 2.18: Aplicación de convolución espacial con un solo kernel.

Como se observa en la Figura 2.18, la convolución termina con una dimensión de $h' \times w' \times 1$, recordemos que esta aplicación de filtros nos sirven para obtener características de la imagen y no siempre basta un solo filtro, por ello se aplican n cantidad de filtros, para este caso se escogen 64 filtros de convolución. La aplicación de los 64 kernels ocasiona una profundidad de salida de la misma dimensión, pues la convolución con cada uno de los kernels se va apilando una tras otra, vea la Figura 2.19 donde se aplica la convolución de los kernels a la imagen de entrada.

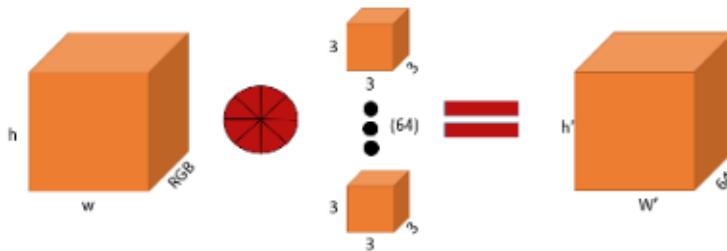


Figura 2.19: Aplicación de convolución espacial con 64 kernels.

2.6.1. Capa de agrupación (Pooling)

Una capa de agrupación opera en bloques del mapa de características de entrada y combina las activaciones de características. Esta operación de combinación está definida por una función de agrupación como la función promedio o máxima (maxpool). Similar a la capa de convolución, se necesita especificar el tamaño de la región agrupada. La Figura 2.20 muestra la operación de agrupación máxima, donde la activación máxima se elige del bloque de valores seleccionado. Esta ventana se desliza a través de los mapas de características de entrada y se mueve cada cantidad n de píxeles, el valor de salto con el que se mueve el filtro se determina "zancada" (1 en el caso de la Figura 2.20) [23].

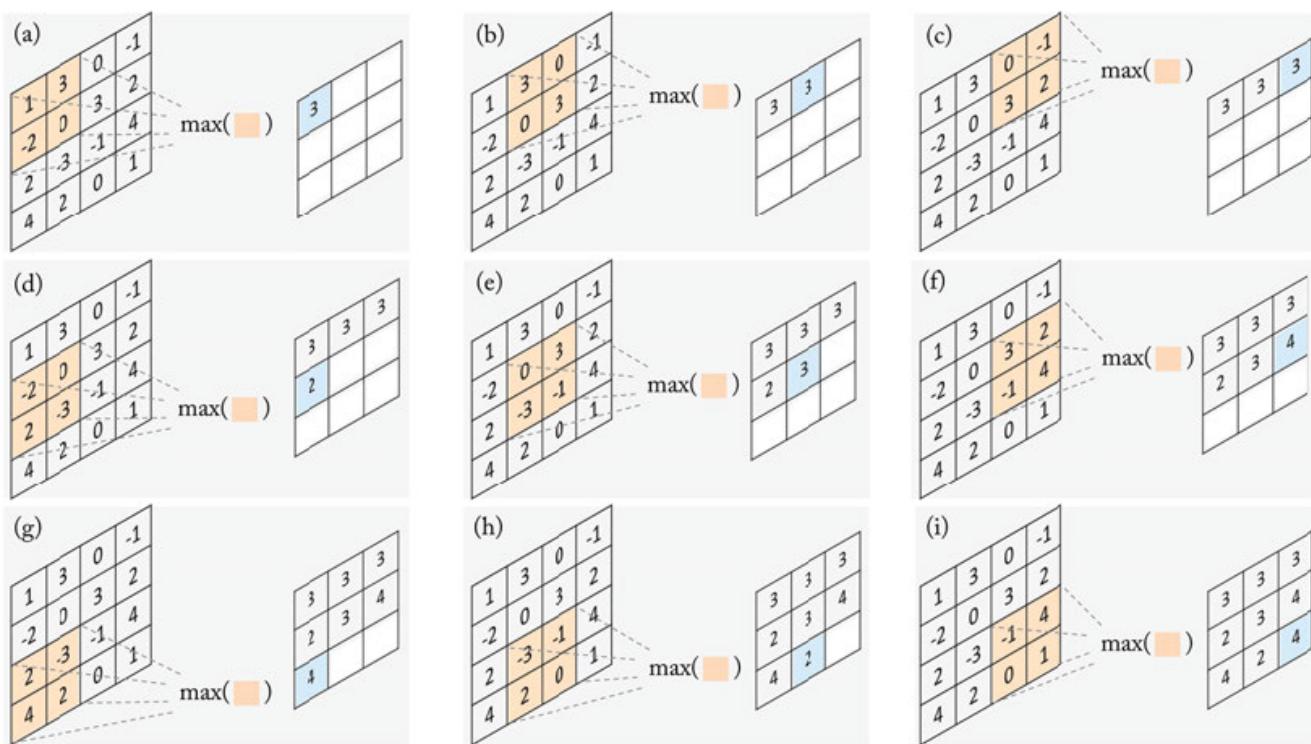


Figura 2.20: Ejemplo de la operación max-pooling con zancada de 1. Imagen tomada de [23].

Hasta el momento se ha explicado el funcionamiento de una red neuronal convolucional, recalmando que a diferencia

de las otras técnicas de detección de objetos, estas no dependen de un filtro antes establecido, sino que se adaptan y buscan los mejores valores de filtros que permitan obtener un mejor resultado por lo que se facilita enormemente la abstracción de características de una imagen.

2.7. Transferencia de aprendizaje

Hasta ahora se tiene en claro lo que son las **Redes neuronales** y se ha recalcado los fabulosos resultados que logran, sin embargo, entrenarlas tiene un costo computacional elevado. Suponga que su finalidad es una crear una red neuronal que detecte y clasifique perros y gatos, esta red es entrenada y se logra el objetivo. La red neuronal cumple su función completamente, pero las circunstancias cambian y ahora debe identificar conejos y loros, ¿es necesario comenzar de cero de nuevo? La respuesta es no, gracias a que ya tiene un previo aprendizaje de la antigua red es posible conseguir cumplir con la nueva tarea sin necesidad de comenzar de cero.

Piense en el conocimiento humano, cuando es necesario enfrentarse a una situación nueva se reconoce y aplica el conocimiento previo que se ha adquirido durante el tiempo de existencia de cada individuo así cuanto más relacionada este una tarea con la experiencia previa más fácilmente se domina. Los algoritmos comunes de inteligencia artificial, por el contrario, abordan tradicionalmente tareas aisladas. La transferencia de aprendizaje intenta cambiar esto mediante el desarrollo de métodos para compartir el conocimiento aprendido en una o más tareas de origen y usarlo para mejorar el aprendizaje en una tarea objetivo relacionada. La transferencia de aprendizaje busca acercar más el cómo aprende una maquina a la manera humana [26].

El objetivo del aprendizaje de transferencia es mejorar el aprendizaje en la tarea objetivo aprovechando el conocimiento de la tarea fuente [26]. A continuación se listan las ventajas que podría otorgar la transferencia de aprendizaje, la Figura 2.21 muestra una comparación entre una red neuronal con transferencia de aprendizaje con otra que no tiene en donde se pueden apreciar sus ventajas.

1. Se puede lograr en la tarea objetivo utilizando solo el conocimiento transferido, por ejemplo en un identificador de manzanas es probable que una misma red neuronal sea capaz de identificar la fruta aunque sea de distintas clases.
2. La cantidad de tiempo que lleva aprender completamente la tarea objetivo dado el conocimiento transferido es menor en comparación con la cantidad de tiempo para aprenderlo desde cero.
3. Mejora del nivel final de rendimiento alcanzable.

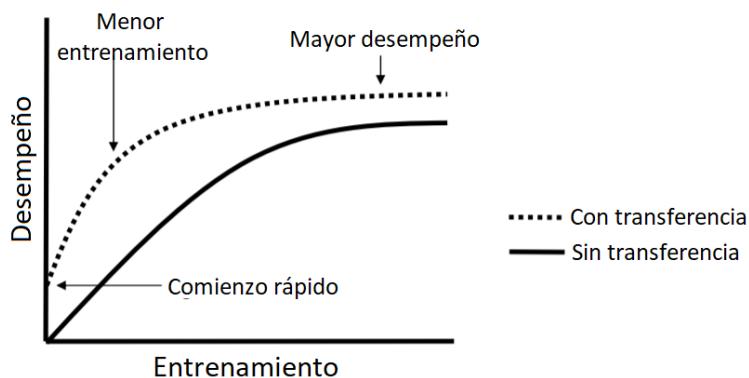


Figura 2.21: Entrenamiento vs desempeño de una red neuronal con y sin transferencia de aprendizaje. Imagen tomada de [26].

Su aplicación es muy fácil de detectar en el aprendizaje supervisado pues cuando se tienen las misma entrada a la red neuronal pero sus etiquetas son de naturaleza distinta es aplicable. La identificación visual es un claro ejemplo dado a que su entrada es usualmente una imagen [27].

2.8. Dropout

Uno de los enfoques más populares para la regularización de redes neuronales es la técnica de dropout. Durante el entrenamiento de la red, cada neurona se activa con una probabilidad fija. Este muestreo aleatorio de una subred dentro de la red a gran escala introduce un efecto de conjunto durante la fase de prueba, donde se utiliza la red completa para realizar la predicción. La Figura 2.22 es exemplificación gráfica del dropout. La desactivación de la activación funciona muy bien con fines de regularización y proporciona un aumento significativo en el rendimiento de los datos no vistos en la fase de prueba [23].

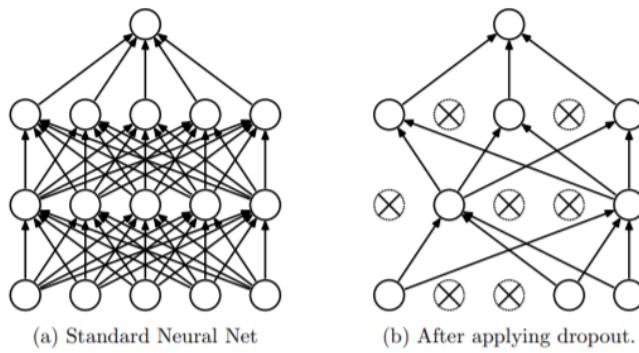


Figura 2.22: Técnica de regularización dropout donde (a) es la red sin dropout y (b) la red con dropout [28].

2.9. VGG16

En la sección [Redes neuronales](#) se introdujo que una arquitectura es la forma de como se componen las capas y como se conectan entre ellas. Para el caso de las [Redes neuronales convolucionales](#) también existen diversas arquitecturas en la literatura y entre ellas cobra relevancia la arquitectura VGGnet, hecha por Simonyan y Zisserman [29] siendo muy popular desde su introducción en 2014 para la labor de detección de objetos dando muy buenos resultados en "El desafío de reconocimiento visual a gran escala de ImageNet" (ILSVRC por sus siglas en inglés). La razón de su popularidad está en la simplicidad de su modelo y el uso de núcleos convolucionales de tamaño pequeño que conducen a redes muy profundas. Los autores introdujeron un conjunto de configuraciones de red, entre las cuales las configuraciones D y E (comúnmente denominadas VGGnet-16 y VGGnet-19 en la literatura) son las más exitosas [23].

La elección del diseño de usar núcleos más pequeños conduce a un número relativamente reducido de parámetros y, por lo tanto, a un entrenamiento y pruebas eficientes. Lo más importante, con filtros más pequeños, se pueden apilar más capas, lo que resulta en redes más profundas y mayor capacidad de extraer características sin necesidad de una exorbitante cantidad de fuerza computacional.

Como se muestra en la Figura 2.23, la arquitectura VGGnet utiliza estrictamente núcleos de convolución de 3x3, seguidas de la aplicación de la [Función de activación ReLu](#) y una capa de maxpool para la extracción de características. A su salida se encuentra un [El Perceptrón Multicapa](#) que se encarga de clasificar las características obtenidas [23].

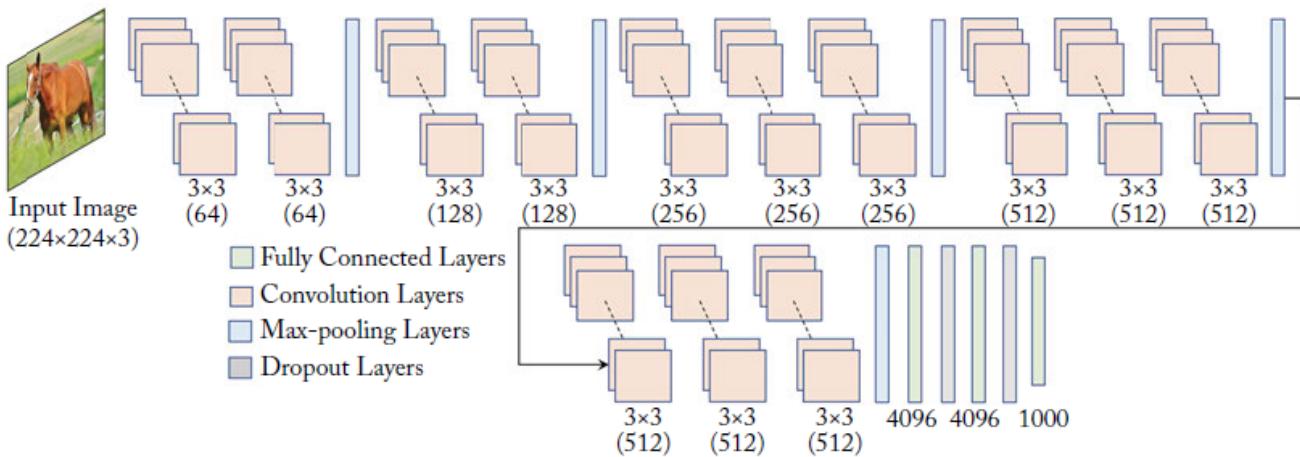


Figura 2.23: Arquitectura VGG-16 (configuración-D) [23].

2.10. Single Shot Detection

Si bien en la actualidad existen distintas arquitecturas de [Redes neuronales convolucionales](#) todas ellas tienen características distintas para cumplir los requerimientos de sus tareas, usualmente el tiempo de procesamiento que se tarda en propagar la imagen en la red neuronal es limitado y por ello un aspecto muy importante en las redes neuronales es cuanto tarda en procesar los datos. Por otro lado, si se desea identificar distintos objetos de una imagen la escala representa un problema pues un objeto pequeño no tiene las mismas características que uno grande, por ejemplo se puede reducir drásticamente su calidad, lo cual dificulta su detección. En sistemas creados para resolver problemas no controlados estos puntos son importantes ya que en ciertos casos es imposible controlar como es que se toma la imagen, si un objeto esta muy lejos o muy cerca, además de necesitar una tiempo de respuesta rápido.

La arquitectura de una Single Shot Detection (SSD) está diseñada para la detección de objetos en tiempo real. Faster R-CNN, una de las principales arquitecturas utilizadas antes de la SSD, ejecuta a 7 frames por segundo, muy por debajo de lo que necesita un procesamiento en tiempo real. La SSD aplica algunas mejoras, como la capacidad de detectar de varias escalas. Estas mejoras permiten que el SSD coincida con la precisión del R-CNN y tener un procesamiento más rápido utilizando imágenes de menor resolución, lo que aumenta aún más la velocidad. La SSD logra la velocidad de procesamiento en tiempo real e incluso supera la precisión de Faster R-CNN [30].

Hay que mencionar que la SSD es una red de aprendizaje supervisado, por ello es necesario tener ciertos datos específicos para poder entrenar, estos comúnmente coinciden con las [Redes neuronales](#) con el propósito de identificar objetos. A continuación se enumeran estos datos:

1. Imagen. Hace referencia a la imagen en la que se quiere identificar los objetos.
2. Coordenadas del objeto. Posición en donde se encuentra el objeto de interés.
3. Clase. Identificador del tipo de objeto, por ejemplo perro, gato, manzana, etc.

La arquitectura SSD tiene la peculiaridad de ocupar otras arquitecturas como base, es decir, dependiendo del problema a tratar se selecciona una red neuronal convolucional y se hace uso de su extracción de características. Es muy usual que la arquitectura [VGG16](#) se ocupe como base y para ello se deja de lado el clasificador para solo ocupar sus capas convolucionales y capas de pooling [30].

La SSD que tiene de base la [VGG16](#) utiliza hasta la capa 4 y tercera convolución (Conv4_3 de la Figura 2.23) de la [VGG16](#), esta salida de la capa se conecta directamente a un nuevo clasificador. Como se ve en la Figura 2.24 la

arquitectura SSD añade más capas convolucionales después de la [VGG16](#), para resolver el problema de la escala de los objetos dentro de la escena, por eso la SSD provee de otras capas para poder detectar distintas escalas de tamaño [31].

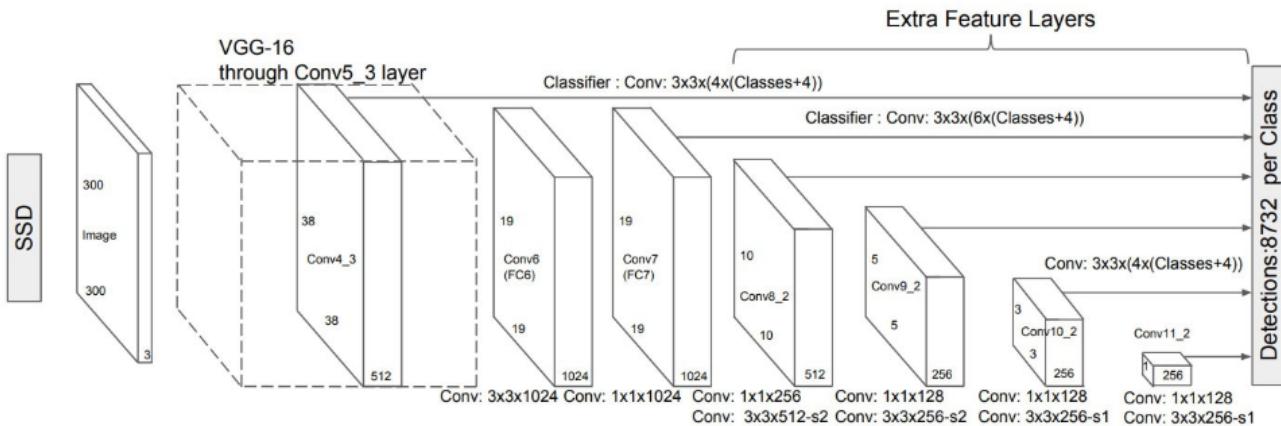


Figura 2.24: Arquitectura Single Shot Multibox Detector [31].

A cada capa convolucional que se conecta con el clasificador (mapas de características) se le asocian un conjunto de cuadros delimitadores, estos son simplemente un recuadro que cubre la posición del objeto. Cada una de estas capas se encarga de distintas escalas las cuales van desde 0.2 (otorgada por la capa más profunda) hasta 0.9 (salida de la arquitectura base) [31]. Los cuadros delimitadores cumplen con 2 características:

- Escala: debido a que hay demasiados tamaños de objetos se ocupan escalas de 0.2 a 0.9, estas responden a una función lineal y su estimación se basa en la aplicación de la Ecuación 2.3.

$$s_k = s_{min} + \frac{s_{max} - s_{min}}{m-1}(k-1), k \in [1, m] \quad (2.3)$$

donde m representa los mapas de características, s_{min} es la mínima escala (0.2) y s_{max} es la máxima escala (0.9) [31].

- Relación de aspecto: se refiere a la proporción (razón) que existe entre el ancho y la altura del cuadro delimitador. Esto debido a que los objetos tienen distintas formas, como por ejemplo un árbol es mucho más largo que ancho y de la forma contraria un automóvil es más ancho que largo. Las razones a ocupar son denotadas por $a_r \in \{1, 2, 3, \frac{1}{2}, \frac{1}{3}\}$ el ancho (w) y largo (h) son calculados por las Ecuaciones 2.4 y 2.5 respectivamente [31].

$$w_a^k = s_k \sqrt{a_r} \quad (2.4)$$

$$h_a^k = \frac{s_k}{\sqrt{a_r}} \quad (2.5)$$

Además para la relación de aspecto 1 se añade un cuadro delimitador más que responde a la escala calculada en la Ecuación 2.6.

$$s'_k = \sqrt{s_k s_{k+1}} \quad (2.6)$$

Por lo tanto estos conjuntos de cuadros delimitadores es donde estará contenida la detección del objeto y la red neuronal deberá buscar distintos valores ("offsets") que coloque correctamente el objeto en los conjuntos de cuadros delimitadores, como se observa en la Figura 2.25 se encuentran los objetos (perro y gato) pero estos están a distintas escalas y por ello se detectan en diferentes mapas de características [31].

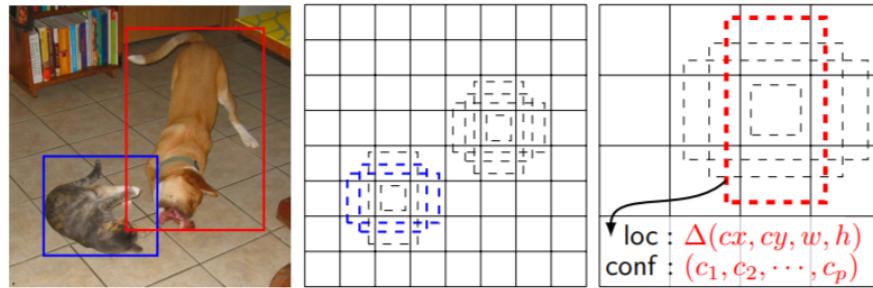


Figura 2.25: Cuadros delimitadores del objeto. Imagen tomada de [31].

Se genera el problema de tener distintos cuadros delimitadores que encierran al objeto, ahora es turno de determinar que cuadro delimitador de su conjunto lo contiene de mejor manera. Se utiliza una simple forma de calcular cual es la mejor, esta es llamada "Intersección sobre la Unión" (IoU), por sus siglas en inglés (Intersection over Union) expresada en la Figura 2.26 [31].

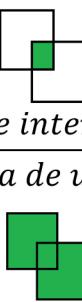
$$IoU = \frac{\text{Área de intersección}}{\text{Área de unión}}$$


Figura 2.26: Intersección sobre la unión.

De esta forma la SSD es capaz de identificar distintas escalas y como ocupa los resultados de las capas de convolución para obtener las siguientes escalas puede determinar de forma muy rápida la posición de los objetos. Sus diferentes razones de cuadros delimitadores hacen posible encerrar correctamente las distintas formas de los objetos sin importar su escala.

2.11. MobileNet

Hasta ahora se ha revisado la arquitectura Single Shot Detection que utiliza como base la red VGG16, pero esta red no es la única que se puede ocupar como base pues hay otras muchas arquitecturas que realizan su misma tarea de

extracción de características que ofrecen distintas ventajas. En esta sección se hablará de la arquitectura MobilNet que se utilizará como base para la SSD que además de ser buena extractora de características es también más veloz por su convolución separable en profundidad, por ello antes de comenzar a hablar de la red neuronal MobileNet es conveniente abarcar la convolución separable en profundidad [32].

2.11.1. Convolución separable en profundidad

Con el objetivo de reducir la complejidad computacional de la convolución espacial pero manteniendo un resultado igual o similar, la convolución profunda se separa en dos etapas [33] y se explicará con un ejemplo:

1. Depthwise convolution: Se parte de una imagen con un ancho, alto y canales establecidos (en este caso 12x12x3) al cual se le aplicará un proceso de convoluciones con distintos números de kernels (igual a los canales) y cada uno de ellos solo se convoluciona con un solo canal de la imagen, como se muestra en la Figura 2.27 [33].

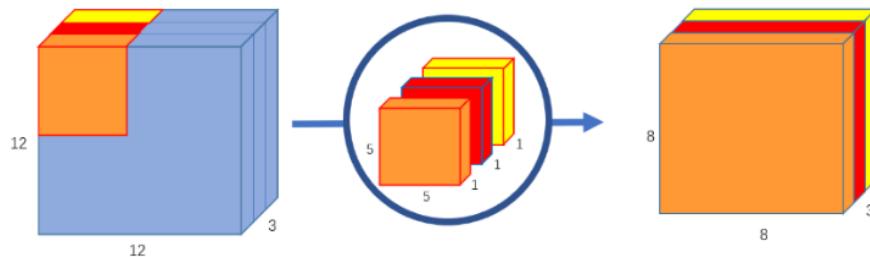


Figura 2.27: Convolución profunda. Imagen tomada de [33].

2. Pointwise convolution: Ahora se aplican otros n kernels para alargar la profundidad, estos kernels tienen la particularidad de ser de dimensiones 1x1xprofundidad. En este ejemplo la profundidad es 3, por lo tanto esto sería de 1x1x3 y se aplicarán 256 kernels, la operación se puede apreciar en la Figura 2.28 [33].

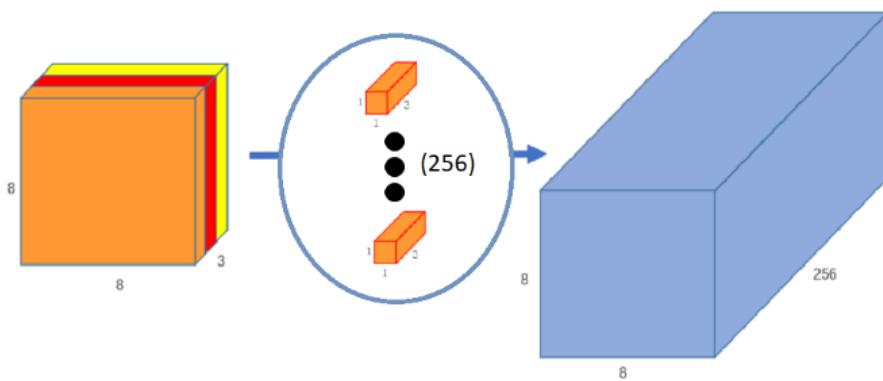


Figura 2.28: Aplicación de 256 kernels. Imagen tomada de [33].

Cuando el tamaño del kernel es de 3x3 los cálculos bajan entre 8 y 9 veces, pero perdiendo un mínimo de precisión [33].

2.11.2. Comparación entre convolución espacial y convolución separable en profundidad.

Si bien, ya se mencionó la forma en que opera la convolución separable en profundidad es necesario señalar sus ventajas sobre la convolución espacial, para esto se aplicará una convolución separable en profundidad a la Figura 2.16 que anteriormente se le aplicó la convolución espacial, en ambos casos se ocupa el filtro mostrado en la Figura 2.17.

Cálculo de operaciones de la convolución en profundidad

Como se vio antes se empieza aplicando un solo kernel a cada una de las capas de imagen de entrada, se puede apreciar en la Figura 2.29.

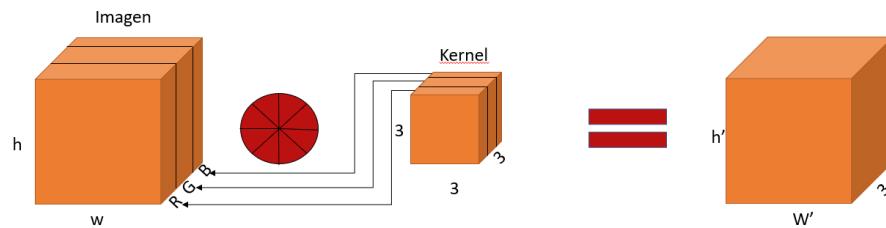


Figura 2.29: Convolución en profundidad.

La métrica de comparación que ocuparemos será el número de operaciones aplicadas para realizar la convolución espacial contra la convolución separable en profundidad. Para ello en esta primera etapa se calcula el número de multiplicaciones siguiendo la Ecuación 2.7, sustituyendo con los valores del ejemplo se obtiene la Ecuación 2.8, este dato se retomará después. Donde c es el número de canales, w el ancho y h el alto de la imagen y d el total de operaciones.

$$wxhxc = d. \quad (2.7)$$

$$3x3x3 = 27 \text{ operaciones.} \quad (2.8)$$

La segunda etapa es la convolución puntual (pointwise convolution), se comienza aplicando un solo kernel de 1x1x3 al resultado de la operación anterior (Figura 2.29).

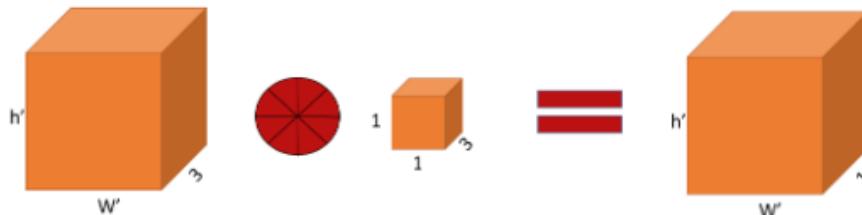


Figura 2.30: Convolución puntual con un solo kernel.

Como se observa en la Figura 2.30 el mapa de características es de profundidad 1, ahora es necesario obtener la profundidad deseada, en este caso 64, es por ello que en lugar de aplicar un solo kernel se aplican 64 kernels, ver la Figura 2.31.

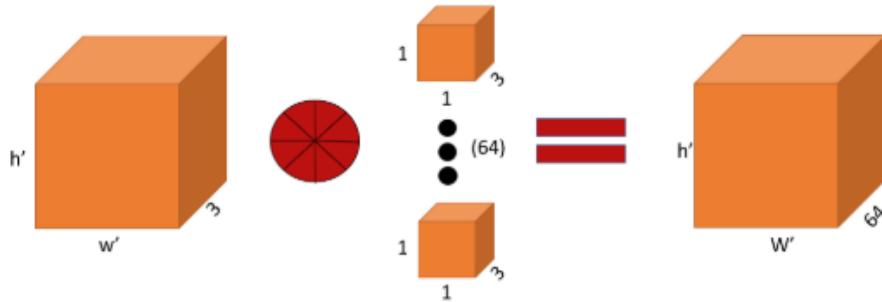


Figura 2.31: Convolución puntual con 64 kernels.

Se puede observar en la Figura 2.31 que se termina con las mismas dimensiones que en la convolución espacial (Figura 2.19) se procede a calcular el número de operaciones de esta etapa siguiendo la Ecuación 2.9. Donde n es el número de filtros.

$$wxhxcxn = d. \quad (2.9)$$

$$1 \times 1 \times 3 \times 64 = 192 \text{ operaciones}. \quad (2.10)$$

Se tienen 64 kernels de $1 \times 1 \times 3$, por lo tanto los parámetros de esta etapa son $1 \times 1 \times 3 \times 64$ lo que nos da 192 operaciones, la Ecuación 2.10 muestra este cálculo. Lo siguiente es sumar las operaciones de las dos etapas de la convolución separable en profundidad, este calculo se muestra la Ecuación 2.11.

$$27 + 192 = 219 \text{ operaciones} \quad (2.11)$$

Entonces obtenemos un total de **219 operaciones de la convolución separable en profundidad**, a continuación calculamos las operaciones de la convolución espacial.

Cálculo de operaciones de la convolución espacial

Para este cálculo es necesario ver la Figura 2.19 en la cual se aplican 64 kernels de dimensiones $3 \times 3 \times 3$, la Ecuación 2.12 muestra el cálculo de las operaciones.

$$3 \times 3 \times 3 \times 64 = 1728 \text{ operaciones} \quad (2.12)$$

Como se observa en la Tabla 2.2 en este ejemplo la convolución en profundidad nos ahorra **9 veces** el número de multiplicaciones realizadas para la misma imagen y kernel.

El costo general de la operación está dado por la Ecuación 2.13:

$$D_K \cdot D_K \cdot M \cdot D_F \cdot D_F + M \cdot N \cdot D_F \cdot D_F \quad (2.13)$$

donde M : número de canales de entrada, N : número de canales de salida, D_K : tamaño del núcleo y D_F : tamaño del mapa de características [34].

Convolución tradicional	Convolución en profundidad
1,728 operaciones	219 operaciones

Tabla 2.2: Comparación de número de operaciones entre convolución tradicional y convolución en profundidad.

Para bajar aún más el costo computacional se puede agregar un multiplicador de ancho α para modelos más delgados, dado por la expresión de la Ecuación 2.14.

$$D_K \cdot D_K \cdot \alpha M \cdot D_F \cdot D_F + \alpha M \cdot \alpha N \cdot D_F \cdot D_F \quad (2.14)$$

donde α está entre 0 y 1, con configuraciones típicas de 1, 0.75, 0.5 y 0.25. Cuando $\alpha = 1$, es la línea base MobileNet. El multiplicador de ancho α se introduce para controlar el ancho de entrada de una capa. El costo computacional y el número de parámetros pueden reducirse cuadráticamente en aproximadamente α^2 [34].

En la tabla 2.3 se puede ver una comparación entre distintos valores de α usando el conjunto de datos ImageNet.

Valor de α	Precisión con dataset ImageNet	Millones de multiplicaciones y sumas	Millones de parámetros
1.0	70.6 %	569	4.2
0.75	68.4 %	325	2.6
0.5	63.7 %	149	1.3
0.25	50.6 %	41	0.5

Tabla 2.3: Comparación de distintos valores α . Tabla tomada de [34].

Se puede apreciar que la precisión disminuye suavemente de $\alpha = 1$ a 0.5 hasta $\alpha = 0.25$, este último tiene una precisión demasiada pequeña [34].

El multiplicador de resolución ρ se introduce para controlar la resolución de la imagen de entrada de la red, con el multiplicador de resolución ρ , el nuevo costo se expresa en la Ecuación 2.15.

$$D_K \cdot D_K \cdot \alpha M \cdot \rho D_F \cdot \rho D_F + \alpha M \cdot \alpha N \cdot \rho D_F \cdot \rho D_F \quad (2.15)$$

donde ρ está entre 0 y 1, y la resolución de entrada es 224, 192, 160 y 128. Cuando $\rho = 1$, es la línea de base MobileNet [34].

En la Tabla 2.4 se puede ver cómo se comporta la red MobileNet distintos valores de ρ en el conjunto de datos ImageNet.

Resolución	Precisión con dataset ImageNet	Millones de multiplicaciones y sumas	Millones de parámetros
224	70.6 %	569	4.2
192	69.1 %	418	4.2
160	67.2 %	290	4.2
128	64.4 %	186	4.2

Tabla 2.4: Comparación de distintos valores ρ . Tabla tomada de [34].

Nótese que la precisión se reduce suavemente en una resolución de 224 a 128 y provocando un menor número de operaciones a realizar [34].

Por último cabe mencionar la arquitectura de la red neuronal MobileNet (Figura 2.32).

Type / Stride	Filter Shape	Input Size
Conv / s2	$3 \times 3 \times 3 \times 32$	$224 \times 224 \times 3$
Conv dw / s1	$3 \times 3 \times 32$ dw	$112 \times 112 \times 32$
Conv / s1	$1 \times 1 \times 32 \times 64$	$112 \times 112 \times 32$
Conv dw / s2	$3 \times 3 \times 64$ dw	$112 \times 112 \times 64$
Conv / s1	$1 \times 1 \times 64 \times 128$	$56 \times 56 \times 64$
Conv dw / s1	$3 \times 3 \times 128$ dw	$56 \times 56 \times 128$
Conv / s1	$1 \times 1 \times 128 \times 128$	$56 \times 56 \times 128$
Conv dw / s2	$3 \times 3 \times 128$ dw	$56 \times 56 \times 128$
Conv / s1	$1 \times 1 \times 128 \times 256$	$28 \times 28 \times 128$
Conv dw / s1	$3 \times 3 \times 256$ dw	$28 \times 28 \times 256$
Conv / s1	$1 \times 1 \times 256 \times 256$	$28 \times 28 \times 256$
Conv dw / s2	$3 \times 3 \times 256$ dw	$28 \times 28 \times 256$
Conv / s1	$1 \times 1 \times 256 \times 512$	$14 \times 14 \times 256$
5× Conv dw / s1	$3 \times 3 \times 512$ dw	$14 \times 14 \times 512$
	$1 \times 1 \times 512 \times 512$	$14 \times 14 \times 512$
Conv dw / s2	$3 \times 3 \times 512$ dw	$14 \times 14 \times 512$
Conv / s1	$1 \times 1 \times 512 \times 1024$	$7 \times 7 \times 512$
Conv dw / s2	$3 \times 3 \times 1024$ dw	$7 \times 7 \times 1024$
Conv / s1	$1 \times 1 \times 1024 \times 1024$	$7 \times 7 \times 1024$
Avg Pool / s1	Pool 7×7	$7 \times 7 \times 1024$
FC / s1	1024×1000	$1 \times 1 \times 1024$
Softmax / s1	Classifier	$1 \times 1 \times 1000$

Figura 2.32: Arquitectura de la red neuronal MobileNet. Imagen tomada de [34].

2.12. Biometría blanda

La biometría blanda o "Soft Biometrics" está definida en [35] como "características que proporcionan alguna información sobre el individuo, pero falta la distinción y permanencia para diferenciar lo suficiente a dos individuos". Esto lo que indica es que si bien, proporciona información de atributos personales asociados a una persona, a través de ellos es imposible identificar de forma única a dicho individuo.

Es importante mencionar que el rostro humano es considerado la mayor fuente de información de atributos ya que provee importantes características geométricas y de textura [36]. Algunos ejemplos de estos atributos son la edad, el género, la estatura, el tipo de cabello, entre otros.

El reconocimiento y análisis facial se han convertido en uno de los problemas más desafiantes en el campo de la visión por computadora y el aprendizaje automático. Como parte del análisis facial, el reconocimiento de género facial también atrae una gran atención de los investigadores. El reconocimiento preciso de género facial tiene una amplia perspectiva de aplicación en áreas como la vigilancia, la indexación de videos y la recopilación de información demográfica [37].

2.12.1. Características de datos biométricos blandos

Como se explica en [36], algunas de las características de los atributos que pertenecen a la biometría blanda son:

- Proporcionan descripciones sencillas que pueden ser interpretadas fácilmente.
- Permite generar deducciones a partir de datos de baja calidad ya que es robusta a la variación de la iluminación, oclusión y baja resolución.
- Es un método no invasivo y que no requiere de consentimiento o cooperación del usuario ya que como se mencionó en el punto anterior, funciona sin restricción de configuraciones, como lo es una larga distancia.
- Debido a que no son atributos únicos de cada individuo, se preserva la privacidad de la identidad para los usuarios.

2.12.2. Ventaja de datos biométricos blandos

Dentro de los sistemas computacionales el almacenamiento de datos es una parte importante. Sin embargo, el almacenar datos personales conlleva a tratarlos de forma adecuada con el fin de cuidar la privacidad de los usuarios que proporcionen sus datos. Al almacenar atributos biométricos blandos, esta privacidad de nuestros clientes no se ve comprometida en ningún momento debido a que aunque lograran obtener los datos, no es posible identificar de forma única a los clientes con los datos.

Otra ventaja que se obtiene es que la forma de obtener estos datos no es invasiva y debido a que no tiene restricciones para obtenerlos no requiere de que el cliente coopere para obtenerlos.

2.13. Edad

El objetivo de esta sección es explicar por qué se considera importante obtener este dato de un cliente, las formas que hasta hoy se han utilizado para estimarla y también algunas razones por las cuales estimar la edad se vuelve un tema complicado.

2.13.1. ¿Por qué son importantes los rangos generacionales?

Las personas son influidas profundamente por los eventos en los que se desarrollaron. Ejemplos de estos eventos son las películas, música, política, entorno social, economía o cultura. Las personas que viven durante la misma época comparten estos eventos y por lo mismo tienen un punto de vista y valores similares [11].

De esta forma, si se logra conocer el rango de edad de una persona se tiene la capacidad de comprender el contexto en el que se desarrolló y como consecuencia se le puede ofrecer un servicio más personalizado [38].

La segmentación mostrada en la Tabla 2.5 fue propuesta por Kotler y Keller [11] y permite entender a qué se refiere al postular que el grupo generacional es un factor grupal para atacar a los diferentes segmentos de mercado.

Un ejemplo claro de que en función del rango generacional las prioridades varían se puede ver entre la "Generación Y" que tiene preocupaciones medioambientales y sociales mientras que la "Generación X" prioriza ganar dinero a la responsabilidad social. De esta forma, una publicidad que muestre las ventajas económicas servirá mejor para la "Generación X" mientras que una publicidad del mismo producto pero que remarque las contribuciones sociales será más efectiva al público de la "Generación Y".

De igual forma se ha encontrado una relación entre los rangos generacionales y las preferencias del sabor. Entre estas relaciones se ha observado que las personas jóvenes prefieren los sabores dulces pero, con el envejecimiento, esta preferencia cambia por sabores más salados. Un ejemplo de esto es que el 55 % de los *Millennials* (*Generación "Y"*) prefieren de sabores dulces, en la *Generación X* disfrutan casi por igual los sabores dulces (47 %) y salados (46 %), mientras que la generación de los *Baby boomers* prefiere los sabores salados (50 %). Con la edad también se aumenta

Rango generacional	Año de nacimiento	Características definitorias
Generación silenciosa	1925-1945	Desafían su edad avanzada manteniendo vidas activas y adquiriendo productos y marketing que les ayuda a lograrlo.
Baby boomers	1946-1964	La mayor parte está todavía en el mejor momento de su ciclo de consumo; adoptan productos y estilos de vida que les permitan retraerse con el paso del tiempo. Están preocupados por la jubilación y el pago de la universidad para sus hijos. Quieren diversificar los activos y amortiguar la volatilidad del mercado. Son activos en las redes sociales.
Generación X	1964-1978	A veces se considera que forman parte de una brecha intergeneracional; tienden puentes entre el conocimiento tecnológico de la "Generación Y" y las realidades adultas de los baby boomers. El 95 % usa Internet, pero no lo valora más que el toque humano. La confianza es lo primero, la tecnología lo segundo. El 90 % usa Smartphone, el 80 % las redes sociales. Usan las redes sociales, el correo electrónico regularmente y herramientas de planificación para ganar confianza. Están dispuesto a aceptar más riesgos y centrarse más en ganar dinero que en la responsabilidad social.
Generación Y	1979-1994	Crecieron con una prosperidad relativa, conectados tecnológicamente y preocupados por asuntos medioambientales y sociales; también tienen un fuerte sentido de independencia y una percepción de inmunidad ante el marketing. Son expertos en tecnología, el 90 % usa Smartphone, el 80 % en las redes sociales. A pesar de que investigan todo en línea, quieren asesoramiento financiero de una persona con experiencia. Sin embargo, piden aplicaciones para poder monitorear el rendimiento de la inversión, gastos, etc, desde su teléfono.
Generación Z	1994- actualidad	Al igual que la Generación Y, la Generación Z como rango generacional se siente cómoda con las tecnologías. Un hecho bastante significativo es que los miembros más jóvenes dentro de este rango no han conocido y les cuesta entender ningún otro entorno.

Tabla 2.5: Perfil de los rangos generacionales estadounidenses. Tabla obtenida con datos de [11] y [38].

el gusto por los sabores amargos y ácido debido a que mientras solo el 3.2 % de los *Millenials* prefieren los sabores ácidos, en el caso de los *Baby boomers* este porcentaje aumenta al 5.2 % [39].

En [40] se llevó a cabo un análisis sobre la relación entre la preferencia de sabor y la edad con sujetos entre 18 y 80 años. Para llevar a cabo el análisis se dividió a los sujetos en tres grupos:

- 18 a 36 años,
- 37 a 50 años,
- más de 50 años.

La Tabla 2.6 muestra el promedio de respuestas a un cuestionario sobre la preferencia de sabor a tres grupos de personas divididos en función de su edad y, al hacer una comparativa entre [39] y [40] se puede observar que ambos coinciden en la disminución de gusto por sabor dulce y el aumento por el gusto de sabores ácidos conforme envejecen, mientras que contrastan en el aumento del gusto por sabores salados y amargos.

Preferencia de sabor	18 a 36 años	37 a 50 años	más de 50 años
Sabor amargo	0.72 (0.82)	0.56 (0.76)	0.51 (0.80)
Sabor dulce	2.55 (0.66)	2.43 (0.83)	2.31 (0.88)
Sabor salado	2.30 (0.84)	2.22 (0.94)	2.16 (0.96)
Sabor ácido	0.61 (0.77)	0.62 (0.85)	0.67 (0.90)

Tabla 2.6: Promedio de preferencia de sabor en función de grupo de edad de un cuestionario clasificado de 0 a 3. Los valores entre paréntesis son la desviación estándar para cada caso. Tabla obtenida con datos de [40].

Por otro lado, la estimación de la edad también cobra relevancia ya que pertenece al ámbito de la gestión de relaciones electrónicas con clientes (ECRM por sus siglas en inglés). Este ámbito se refiere al uso de tecnologías de la información y herramientas de interacción multimedia para gestionar relaciones diferenciadas con los clientes y proporcionarles un trato individualizado en función de sus necesidades específicas [41] [42].

2.13.2. Estimación de la edad

Una vez entendida la razón por la que en el marketing el conocimiento de la edad ayuda de forma crítica en conocer el comportamiento de un cliente, el siguiente paso es identificar de qué forma es posible obtener este dato sin que el usuario tenga que proporcionarlo. Como se mencionó en [Biometría blanda](#), el rostro humano es una gran fuente de datos para estimar estos atributos blandos (edad y género) y por ello estimar la edad de una persona a través de una fotografía es un tema que ha sido abordado en varias ocasiones y haciendo uso de varios métodos para lograrlo; sin embargo, ha sido uno de los problemas más desafiantes para los investigadores del área debido a que expresiones faciales y la edad son transformaciones no lineales que pertenecen a una de las fuentes de variación facial más difíciles de modelar [42].

Antes de explicar los procedimientos por los cuales hoy en día se puede estimar la edad se tiene que explicar algunos conceptos de la edad. En [43] dividen la edad en distintas clases:

- Edad real: Se refiere a los años que tiene de vida.
- Edad aparente: La edad aparentada visualmente.
- Edad percibida: La edad de un individuo según personas a partir de la apariencia visual.
- Edad estimada: La edad de un individuo reconocida por una máquina a partir de la apariencia visual.

En [36], [41] y [42] se han propuesto diversos métodos para lograr la estimación de la edad aparente y no la edad real debido a que los estudios señalan que las personas tienen diferentes ritmos de envejecimiento. Esto se debe a que el ritmo de envejecimiento varía en función de los genes de la persona, el estado de salud, el ambiente de trabajo, estilo de vida, sociabilidad, consumo de tabaco, estrés emocional, cambios drásticos de peso o la exposición a climas extremos [42]. Los principales signos de envejecimiento son el crecimiento craneofacial, envejecimiento de la piel debido a arrugas, cambio de color en la piel y manchas [41].

Entrando en el tema de la visión artificial, algunas de las áreas de investigación son la clasificación, localización, detección de objetos y la segmentación. En el caso del presente trabajo la visión artificial será usada para detectar biometría blanda (edad y género) a través del rostro de una persona. Las personas utilizan más indicadores aparte del rostro para estimar la edad, como son el color del pelo o el tono de la voz y por este motivo un sistema de visión artificial puede no ser tan robusto debido a que carece de estos indicadores [41].

En la literatura se separan los métodos para el reconocimiento del género y la estimación de la edad en métodos convolucionales y no convolucionales.

2.13.2.1. Métodos no convolucionales

Dentro de los métodos no convolucionales, en [42] son divididos en tres tipos:

- Antropometría: Se refiere a métodos basados en la forma. Estos métodos usan las relaciones de distancia entre marcas faciales para describir las diferencias topológicas. Para el caso concreto de la estimación de la edad se suele hacer uso del crecimiento craneofacial pero tiene la limitante de que solo sirve para identificar a niños de personas adultas ya que en la edad adulta la forma facial no varía demasiado [42]. Otro conflicto es que en caras en 2D es difícil obtener el perfil de la cabeza [41].
- Características de texturas: Se refiere a los métodos que radican en la extracción de características de las texturas como las arrugas o marcas faciales a partir de imágenes. Una de las formas de obtener la información es haciendo uso de la intensidad de los píxeles. Con este enfoque se puede obtener la estimación de la edad, reconocimiento del género y la raza pero en general es poco eficiente debido a su alta dimensionalidad de las características [42].
- Apariencia facial: Está basado en la apariencia facial utilizando información de la textura y las formas de la cara para diferenciar los rostros de diferentes grupos demográficos. Tiene la misma desventaja que la basada en antropometría ya que necesita de una localización precisa de las marcas faciales [42].

2.13.2.2. Métodos con redes neuronales convolucionales

Por otra parte tenemos las arquitecturas profundas como las CNN (Redes neuronales convolucionales) que construyen no linealidades una encima de otra por lo que son capaces de aprender transformaciones no lineales de forma más eficiente que los modelos simples; sin embargo, son más vulnerables al sobreentrenamiento. Como se mencionó en [Estimación de la edad](#), la edad requiere de transformaciones no lineales para calcularse y las CNN son capaces de construir estas no linealidades por lo que han logrado superar a los métodos clásicos en la tarea de estimación de edad [42].

En 2011 se propuso en [44] por primera vez utilizar una CNN con cinco capas para la estimación de la edad. El modelo consistía en diferentes salidas que proporcionan el género, raza y edad, aunque esta última estimación no tuvo los mejores resultados con los basados en las características bio-inspiradas; sin embargo, gracias al aumento en la capacidad de procesamiento y la cantidad de datos, hoy en día se usan CNNs mucho más profundas basadas en arquitecturas como la VGG-16 (se habla con más detalle de esta arquitectura en la sección de [VGG16](#)) [45] por su éxito en reconocimiento de objetos en Imagenet [42].

2.14. Género

Para este proyecto en particular se usa el género como un punto importante para poder dar una recomendación del producto que le debería gustar a la persona analizada. Según [46] se ha revelado que el género tiene un factor importante para la elección del consumo de alimentos, influenciado por muchos factores interrelacionados que van desde mecanismos biológicos y perfiles genéticos hasta factores sociales y culturales. Es imposible determinar todos los factores por ello el ocupar ocupar el género de la persona es importante y de gran ayuda.

A lo largo de diversos estudios también se ha concluido que el género es un factor que influye en las preferencias de sabor de cada individuo. Como ejemplo, se puede observar en la Tabla 2.7 que las mujeres suelen preferir los sabores dulces en un 10 % más que los hombres, mientras que en sabores salados y ácidos son muy sutiles las diferencias. En el caso del sabor amargo, el 2.8 % de los hombres lo prefieren contra el 0.6 % de las mujeres, aunque es muy importante remarcar que, en general, un porcentaje muy bajo prefiere sabores ácidos o amargos [39].

En [40] también se realizó un análisis sobre la relación entre el género y la tendencia hacia algún tipo de sabor en donde, como se puede apreciar en la Tabla 2.8, se observó que la calificación asignada a sabores dulces y salados entre

Preferencia de sabor	Mujer	Hombre
Sabor amargo	0.6 %	2.8 %
Sabor dulce	51.2 %	41.2 %
Sabor salado	53.9 %	50 %
Sabor ácido	4.3 %	4.5 %

Tabla 2.7: Preferencia de sabor en porcentaje en función del género. Tabla obtenida con datos de [39].

Preferencia de sabor	Mujer	Hombre
Sabor amargo	0.51 (0.75)	0.75 (0.87)
Sabor dulce	2.42 (0.83)	2.45 (0.74)
Sabor salado	2.22 (0.94)	2.23 (0.86)
Sabor ácido	0.60 (0.86)	0.70 (0.81)

Tabla 2.8: Promedio de preferencia de sabor en función del género de un cuestionario clasificado de 0 a 3. Los valores entre paréntesis son la desviación estándar para cada caso. Tabla obtenida con datos de [40].

hombres y mujeres varía muy poco, mientras que en los hombres tuvieron una mayor aceptación de sabores ácidos o amargos.

Preferencia de sabor	Mujer	Hombre
Sabor amargo	25.3	35.96
Sabor dulce	77.89	72.92
Sabor salado	41.28	49.92
Sabor ácido	37.83	44.72
Sabor picante	54.31	82.88
Sabor umami	44.34	37.52

Tabla 2.9: Preferencia de sabor promedio en función del género. Tabla obtenida con datos de [47]

Por último, en [47] se realizó otro análisis sobre la relación entre la preferencia del sabor y el género. En este estudio se agregaron dos sabores: el *picante* y *umami*.

Al hacer una comparación entre los valores presentados en [39], [40] y [47] se puede apreciar como coinciden en la tendencia en función del género en tres de cuatro sabores, debido a que en el caso del sabor dulce en [39] y [47] se indica que las mujeres prefieren el sabor dulce en más ocasiones que los hombres, mientras que en [40] los hombres dieron una calificación mayor al sabor dulce.

Los métodos convencionales de reconocimiento de género usan la forma de extracción manual de características de las imágenes para representar caras, como SIFT, LBP y HOG, y luego usan estas características en el paso de clasificación [48]. Sin embargo, la ingeniería de características es difícil, requiere mucho tiempo y requiere un conocimiento experto sobre el dominio del problema. El otro problema con las funciones diseñadas a mano, como HOG, SIFT, SURF u otros algoritmos como ellos, es que son demasiado escasos en términos de información que pueden capturar de una imagen. Esto se debe a que las derivadas de imágenes de primer orden no son características suficientes para la mayoría de las tareas de visión por computadora, como la clasificación de imágenes y la detección de objetos. Además, la elección de características a menudo depende de la aplicación y no facilitan el aprendizaje por transferencia, es decir, su aplicación para resolver un problema usualmente no funcionará en resolver otro similar. Además, el diseño de características diseñadas a mano está limitado por la complejidad que los humanos pueden poner en él.

Hay que recordar que en los últimos años, las CNNs han logrado grandes resultados en reconocimiento facial. Con

una arquitectura de redes avanzada y métodos de aprendizaje entusiastas, las CNNs ha mejorado la capacidad de reconocimiento facial a niveles sin precedentes [48].

Distintas investigaciones proponen diversas arquitecturas de redes neuronales artificiales las cuales permiten obtener tanto el género como la localización del rostro y en algunas más complejas se incluyen otras características como la postura, sonrisa y edad, esta última es también de nuestro particular interés.

2.15. Personalidad: Big Five

Los especialistas en marketing creen que las compras de los usuarios, así como la forma en que consumen y cuando lo hacen, pueden verse influidas por la **personalidad**, siendo un tema que puede contribuir a uno de los principales campos en la investigación de marketing: el comportamiento del consumidor.

2.15.1. TPQ

El Cuestionario de Personalidad Tridimensional (TPQ) es un instrumento de 100 elementos de falso/verdadero que toma aproximadamente 15 minutos y permite medir las tres dimensiones de la personalidad de orden superior [49]. Las tres dimensiones genéticas subyacentes de la personalidad son la *Búsqueda de novedad (NS)*, *Evasión del daño (HA)* y *Dependencia de recompensa (RD)* [50].

- NS: Es la tendencia hacia la euforia intensa o la excitación a nuevos estímulos o señales de recompensa, que conduce a una actividad exploratoria frecuente, así como evitar la monotonía y el castigo potencial.
 - NS1: Exitabilidad exploratoria v.s. Rrigidez estoica
 - NS2: Impulsividad v.s. reflexión
 - NS3: Extravagancia v.s. Reserva
 - NS4: Desorden v.s. Regimentación
- HA: Inhibición del comportamiento para evitar el castigo, la novedad y ausencia de recompensa ya que se tiene una tendencia a responder intensamente a estímulos aversivos.
 - HA1: Preocupación anticipada v.s. Optimismo
 - HA2: Hazaña de incertidumbre v.s. Confianza
 - HA3: Timidez con extraños v.s. Gregarismo
 - HA4: Fatiga y astenia v.s. Vigor
- RD: Tendencia a responder a señales de recompensa para así, mantener o resistirse a la conducta que ha sido asociada a una recompensa o un castigo.
 - RD1: Sentimentalismo v.s. Insensibilidad
 - RD2: Persistencia v.s. Indecisión
 - RD3: Apego v.s. Desapego
 - RD4: Dependencia v.s. Independencia.

En [47] se ha estudiado la relación entre la personalidad y su tendencia a preferir ciertos sabores, donde se ha observado que ciertos rasgos sí tienen una preferencia por alguno de los tipos de sabores como se muestra en la Tabla 2.10.

Preferencia de sabor	Dulce	Salado	Umami	Amargo	Ácido	Picante
NS1	5.07	1.92	2.09	1.95	0.19	3.4
NS2	3.07	1.42	2.88	1.36	1.67	5.18
NS3	4.36	1.68	4.31	0.89	0.91	4.17
NS4	5.46	1.56	5.1	1.11	2.42	3.19
HA1	3.36	2.25	1.64	0.63	1.05	5.45
HA2	2.89	1.09	3.72	0.85	0.9	1.49
HA3	3.21	2.53	3.14	0.98	0.87	1.9
HA4	2.29	2.1	1.13	1.26	1.11	2.4
RD1	3.07	3.02	1.22	0.97	0.5	5.93
RD3	5.68	1.5	1.3	1.92	0.01	0.68
RD4	3.04	0.82	1.44	2.2	0.02	6.88

Tabla 2.10: Promedio de preferencia de sabor en función de la personalidad en valores de 0 a 7. Tabla obtenida con datos de [47].

2.15.2. Teoría de rasgos

La teoría de rasgos intenta describir la personalidad con un pequeño número de factores o rasgos clave. Los rasgos son los comportamientos estables que una persona muestra en la mayoría de las situaciones, además se cree que estos rasgos son predisposiciones biológicas, lo que significa que son comportamientos particulares transmitidos por herencia en los seres humanos [51].

Los rasgos se infieren de la forma en que una persona se comporta. En el diccionario del idioma inglés se pueden encontrar unos 18,000 términos que se pueden referir a distintos rasgos de personalidad. El psicólogo Raymond Cattell, por medio de un análisis factorial, descubrió que estos 18,000 rasgos podrían agruparse. Por ejemplo, una persona *perseverante* es probable que también sea *responsable* y *ordenada*, y seguramente no será *desordenada*. Según su investigación, Cattell concluyó que solo se necesitan 16 rasgos para describir completamente una personalidad [51].

El investigador Eysenck propuso que la personalidad podría reducirse a solo tres dimensiones básicas: *estabilidad emocional*, *extraversión* y *psicoticismo*. Basado en los trabajos de Cattell, Eysenck y otros, los teóricos de los rasgos contemporáneos han reducido los rasgos de personalidad a cinco dimensiones básicas presentadas en el modelo de cinco factores [51].

El **modelo de cinco factores** postula que los rasgos de personalidad están organizados en 5 factores o dimensiones básicas. Se ha demostrado que cada una de estas cinco dimensiones tiene al menos 6 facetas o componentes, y 2 rasgos opuestos que clasifican y describen a una persona dentro de cada dimensión [51]. A continuación se enlistan estos cinco factores junto con sus rasgos opuestos, y en la Tabla 2.11 se presentan las respectivas facetas para cada una de las dimensiones.

1. La **inestabilidad emocional** se caracteriza por la calma y la confianza frente a la tensión nerviosa. Puede clasificar a una persona como *relajada*(+) o *inquieta*(-).
2. La **extraversión** explica la cantidad e intensidad de la interacción interpersonal, el nivel de actividad y la necesidad de estimulación. Puede clasificar a una persona como *amistosa*(+) o *reservada*(-).
3. La **apertura a la experiencia** se refiere a la vida experiencial de una persona: aventura, ideas inusuales, curiosidad y variedad de experiencias. Puede clasificar a una persona como *imaginativa*(+) o *práctica*(-).
4. La **amabilidad** implica una orientación compasiva y cooperativa hacia los demás con sospecha y antagonismo. Puede clasificar a una persona como *auténtica*(+) o *egoísta*(-).

5. La **conciencia** se asocia con el comportamiento y el control de los impulsos antes de actuar, siguiendo las normas y reglas, así como la planificación y la organización. Puede clasificar a una persona como *organizada(+)* o *descuidada(-)*. [51].

Inestabilidad emocional	Extraversión	Apertura a la experiencia
<ul style="list-style-type: none"> ● Ansiedad ● Hostilidad - ira ● Depresión ● Autoconciencia ● Impulsividad ● Vulnerabilidad 	<ul style="list-style-type: none"> ● Calidez ● Sociabilidad ● Asertividad ● Actividad ● Búsqueda de júbilo ● Emociones positivas 	<ul style="list-style-type: none"> ● Fantasía ● Estética ● Sentimientos ● Acciones ● Ideas ● Valores
Amabilidad	Conciencia	
<ul style="list-style-type: none"> ● Confianza ● Franqueza ● Altruismo ● Conformidad ● Modestia ● Ternura 	<ul style="list-style-type: none"> ● Competencia ● Orden ● Obediencia ● Esfuerzo por el logro ● Autodisciplina ● Deliberación 	

Tabla 2.11: Las cinco dimensiones de personalidad, cada una con sus seis facetas [51].

Para medir el Big Five en una persona se hacen uso de pruebas o cuestionarios especializados. La personalidad de un individuo se define como una colección de 5 puntajes en el rango [0, 1] para cada dimensión [52]. Los valores cercanos o iguales a 0 corresponden al rasgo negativo de la dimensión, mientras que aquellos cercanos o iguales a 1 corresponden al rasgo positivo [51].

El Inventario de Personalidad NEO (NEO PI) consta de 180 elementos que miden cada rasgo con 6 escalas de facetas para cada uno [51].

El más común es el Inventario de Personalidad Neo Revisado (NEO PI-R). Posee 240 elementos en escalas tipo Likert de 5 puntos (0 - 4) que miden 5 factores y 6 facetas por factor [51].

El Inventario de Personalidad de Diez Artículos (TIPI) es una escala más corta. Se puede completar en aproximadamente 1 minuto, posee 10 elementos que constan de 2 adjetivos y cada uno de los 5 factores principales está representado por 2 elementos [51].

Sin embargo, el llenado de un cuestionario es una tarea tediosa que consume mucho tiempo, además de que la información proporcionada en ellos nunca es completamente real, pues las personas manipulan las respuestas para

generar mayor probabilidad de obtener un resultado favorable para ellos. Por esto, se han buscado otras alternativas para obtener la personalidad en términos de Big Five de una persona, como la **apariencia y vestimenta** por ejemplo, en donde se han obtenido resultados mostrando que las personas nerviosas y tensas (*inestabilidad emocional*) así como las convencionales (*apertura a la experiencia*) tienen un bajo interés en su apariencia, y que la *extraversión* tiene una relación con el énfasis en la apariencia [51].

Albert Mehrabian descubrió que el 7 % de la impresión de una persona depende de sus palabras, el 38 % de las expresiones vocales y el 55 % de las **expresiones faciales** [53]. Entonces, gracias a este dato, a continuación se da un vistazo a qué tanto influye el análisis facial con la detección de Big Five en un individuo.

2.15.3. Análisis facial

Según el estudio en [54], la cara contiene suficiente información para lograr la precisión en regresiones aparentes de la personalidad, como es el caso de los trabajos de ChaLearn [55]. En la Figura 2.33 se muestran los resultados de este estudio, en el cual las respuestas de activación más altas de las unidades ubicadas en las capas dentro de redes neuronales convolucionales (ver sección [Redes neuronales convolucionales](#)) se enfocan principalmente en regiones de la cara para discriminar entre los diferentes rasgos de personalidad, representadas como mapas de temperatura, en donde la información más importante en la que una red neuronal convolucional se centra principalmente (el rostro), representada con una mayor temperatura.



Figura 2.33: Imagen tomada de [54] con las zonas de activación resultantes en las 20 imágenes con el valor predicho más alto del rasgo de *amabilidad* en el subconjunto de prueba del dataset de ChaLearn [55], indicando que el rostro es imprescindible para la detección de personalidad.

Estudios de psicología han encontrado que las expresiones faciales dinámicas de un individuo, como **la dominación facial, el tipo de mirada, la sonrisa o un aspecto tenso**, se reflejan en facetas como la *extraversión*, la *inestabilidad emocional* y la *amabilidad* [56]. Esto gana mayor justificación y apoyo con los resultados en [54], mostrados en la Figura 2.34, que respalda la aseveración de que los puntos de activación más altos en un rostro para discriminar entre personalidades son las secciones previamente mencionadas, representadas de igual forma en un mapa de calor.

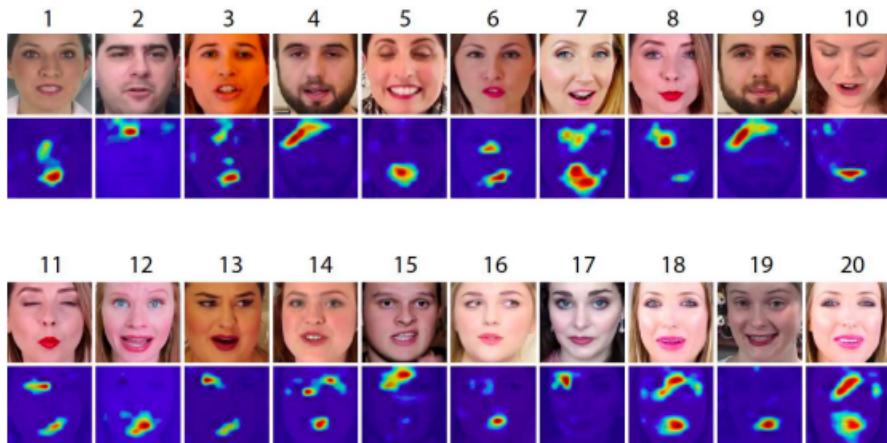


Figura 2.34: Imagen tomada de [54]. La red utilizada en este trabajo se enfoca principalmente en los **ojos y la boca** para discriminar entre los diferentes rasgos de personalidad.

En [57] se muestra una recopilación y comparación de todos los trabajos existentes hasta 2019 en donde se ha intentado identificar el Big Five en una persona. En todos ellos el análisis facial es uno de los principales enfoques; siendo las características más relacionadas con cada uno de los 5 factores de personalidad **la sonrisa y el contacto visual**, para la personalidad real en una persona. Dichos trabajos pueden limitarse al análisis realizado en pequeños conjuntos de datos, un pequeño número de participantes y sin protocolos de evaluación estándar, que no garantizan la generalización a diferentes escenarios y contextos.

Para el análisis facial se necesita un medio no invasivo, que posea de la totalidad del rostro de una persona. Uno de estos medios, que probablemente es el que mayor relación tiene con el rostro, pues es toda la base y naturaleza del mismo, es la *selfie*. En el siguiente bloque se habla sobre sus características, ventajas y desventajas.

2.15.4. Selfies

Una *selfie* es un **autorretrato** de una o un grupo de personas, que es tomada con una cámara o un teléfono con cámara con el **brazo extendido o apuntando a un espejo**, y puede aparecer una sola persona, una persona y un compañero, o un grupo de personas. Para todos los casos, **los rostro completos son visibles** [51].

Las ventajas de usar selfies para este tipo de análisis, como lo es la predicción del Big Five, se enlistan a continuación:

1. Transmiten mucha **información sobre una persona** (la atractividad de una persona o para analizar sentimientos, por ejemplo).
2. Están directamente relacionadas con la formación de **impresiones y la personalidad**.
3. **No es necesario procesar tanta información** como en los videos (análisis del audio, extracción de frames, etc).
4. Es un recurso **no invasivo y de fácil acceso** en comparación con conversaciones o textos personales.
5. Se estudia la **composición de la imagen**, y no su contenido.
6. La extracción de características de la selfie se hace de manera automática.
7. Un único modelo puede realizar la extracción de características y la clasificación [51].

Por otro lado, algunas de las desventajas son que no se sabe qué característica exacta de una selfie es la que revela información sobre la personalidad de una persona, y además el juicio humano de personalidad de una persona con base a una selfie no es exacto. Desgraciadamente, hay pocos datasets de selfies e imágenes etiquetadas por Big Five [51].

2.16. Recomendador

Un sistema de recomendación es una aplicación que filtra información personalizada, buscando entender los gustos del usuario para sugerirle cosas apropiadas (en este caso una bebida de [BubbleTown®](#)) considerando los patrones exclusivos del mismo [58].

Los sistemas de recomendación que proporcionan sugerencias personalizadas aumentan en gran medida la probabilidad de que un cliente realice una compra en comparación con los sistemas que proporcionan sugerencias no personalizadas. Esto es especialmente importante en mercados donde la variedad de opciones es grande, el gusto de los clientes es importante [59]. Considerando que [BubbleTown®](#) tiene una amplia gama de bebidas que ofrece a sus clientes, la recomendación personalizada es de especial interés.

2.16.1. Recomendación basada en contenido

Un sistema de recomendación basado en contenido examina las características de los productos para poder identificar los que podrían ser de interés para el usuario. Una práctica común es tener la información de los productos almacenados en una base de datos y con la descripción de estos junto con el perfil del usuario generar la recomendación [60].

Al realizar una recomendación se basa en la descripción de los elementos y el perfil de preferencia del usuario. Es usual que el perfil de preferencia de un usuario sea generado a través de datos que ingresan los usuarios pero también hay otras formas de generar los perfiles de preferencia de un usuario como es la retroalimentación que el usuario da con respecto a los productos. Es posible adaptar algoritmos de aprendizaje que permitan obtener perfiles de usuarios pero la selección de estos depende de la forma en la que está representado el contenido [60].

2.16.2. Filtrado colaborativo

Se entiende por filtrado colaborativo al proceso en el que se evalúa o filtran diferentes artículos auxiliándose de la opinión generada por los usuarios. Para su correcto funcionamiento el sistema debe contar con puntuaciones o calificaciones del artículo que se quiere recomendar, por lo que requiere que los usuarios asignen calificaciones a los artículos que consumen [60].

De forma muy resumida el filtrado colaborativo funciona prediciendo la calificación que el usuario daría al producto que se desea recomendar y si la calificación obtenida es alta se lo muestra al usuario. Para poder predecir la calificación de un artículo el sistema hace uso de una matriz como la mostrada en la Figura 2.35, con la que identificará usuarios que consuman productos similares al usuario en cuestión y que también hayan calificado el producto que se planea recomendar.



Figura 2.35: Ejemplo de matriz usada en filtrado colaborativo. Imagen obtenida de [Stat Developer](#)

A través de las calificaciones dadas por cada uno de los usuarios similares es posible pronosticar la calificación que el usuario dará al producto; sin embargo, no todos los sistemas predicen una calificación para hacer la recomendación, sino que simplemente muestran los productos mejor puntuados por los clientes similares [60].

Para que un recomendador de filtrado colaborativo funcione correctamente es recomendable que se cumplan los siguientes escenarios:

- Debe haber varios artículos o productos,
- Por cada artículo debe haber varias calificaciones,
- Hay más calificaciones de usuarios que artículos,
- El usuario ha calificado varios artículos del catálogo,
- Para cada usuario debe haber otros usuarios con necesidades o gustos similares,
- Los artículos deben ser de la misma categoría,

Uno de los grandes problemas a los que se enfrentan estos sistemas es cuando se presenta a un nuevo usuario ya que al no tener registros anteriores de sus gustos o preferencias, no es posible compararlo con usuarios ya existentes por lo que algunos sistemas suelen poner un *top* de los artículos mejor calificados por los usuarios para lograr comenzar a obtener información sobre el perfil del nuevo usuario [60].

2.16.3. Árbol de decisión

Los árboles de decisión son una técnica estadística para la segmentación, la estratificación, la predicción, la reducción de datos y el filtrado de variables, la identificación de interacciones, la fusión de categorías y la discretización de variables continuas. Provienen del área de la inteligencia artificial y se basan en aplicar algoritmos de aprendizaje sobre un conjunto de datos para encontrar resultados en base a estos datos. Los árboles de decisión son usualmente ocupados en la minería de datos para la clasificación [61].

Algunas ventajas de los árboles de decisión son:

- Facilita la interpretación de la decisión adoptada [61].
- Facilita la comprensión del conocimiento utilizado en la toma de decisiones [61].
- Explica el comportamiento respecto a una determinada decisión [61].
- Reduce el número de variables independientes [61].

Un punto importante a tomar en cuenta es que se necesita una cantidad considerable de muestras para que el árbol de decisión aprenda de forma correcta, por lo menos obtener 1,000 muestras [61].

2.16.4. Clasificador bayesiano ingenuo

Los clasificadores bayesianos son clasificadores estadísticos. Pueden predecir las probabilidades de pertenencia a una clase, como la probabilidad de que una muestra determinada pertenezca a una clase en particular [62]. El clasificador bayesiano se basa en el teorema de Bayes. Los clasificadores bayesianos ingenuos asumen que el efecto de un valor de atributo en una clase dada es independiente de los valores de los otros atributos. Esta suposición se llama independencia condicional de clase. Está hecho para simplificar el cálculo involucrado y, en este sentido, se considera "ingenuo" [62].

Algunas ventajas son:

- No requiere gran cantidad de datos para entrenar [63].

- Entrenamiento rápido (comparado a otros algoritmos) [63].
- Utiliza poca memoria [63].
- Fácil implementación [63].

Claramente este algoritmo también cuenta con desventajas como:

- Si dos o más atributos están altamente correlacionados, reciben demasiado peso en la decisión final, esto conduce a una disminución en la precisión de la predicción en dominios con características correlacionadas [63].
- La estimación de probabilidad es menos precisa y el rendimiento se degrada cuando no se mantiene la independencia de atributo [63], es decir, las clases deben ser completamente separables para no caer en errores de clasificación.

2.16.5. KNN (vecino más cercano)

El algoritmo vecino más cercano simplemente almacena todos sus datos de entrenamiento para clasificar un elemento nuevo sin etiquetar, el algoritmo lo compara con todos los elementos almacenados usando una función de similitud y determina el "vecino más cercano" o los k vecinos más cercanos. La etiqueta de clase o la puntuación numérica de un elemento nunca antes visto se puede derivar de las etiquetas de clase de los vecinos más cercanos. La función de similitud utilizada por el algoritmo de vecino más cercano depende del tipo de datos. Para datos estructurados, a menudo se usa una métrica de distancia euclídea [64].

En este algoritmo se necesita determinar el valor de los k vecinos a seleccionar, principalmente se tienen las siguientes opciones en la práctica:

- $k=1$: este es el caso más básico, el patrón desconocido que se desea clasificar se asigna a la clase del vecino más cercano [65], en la Figura 2.36 se observa un ejemplo de esta situación.
- $k=n$: este caso puede generar complicaciones dado a que se llegan a generar empates, es decir, se seleccionan puntos con distintas clases y se debe de implementar un método para desempatar o elegir la clase a la cual se clasificará el patrón desconocido [65]. La Figura 2.37 representa este problema donde las muestras clasificadas en el rango k pertenecen a distintas clases.

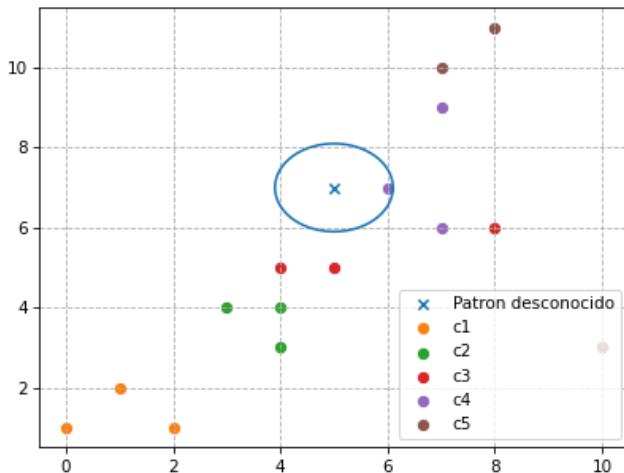


Figura 2.36: Algoritmo de clasificación KNN con $k=1$.

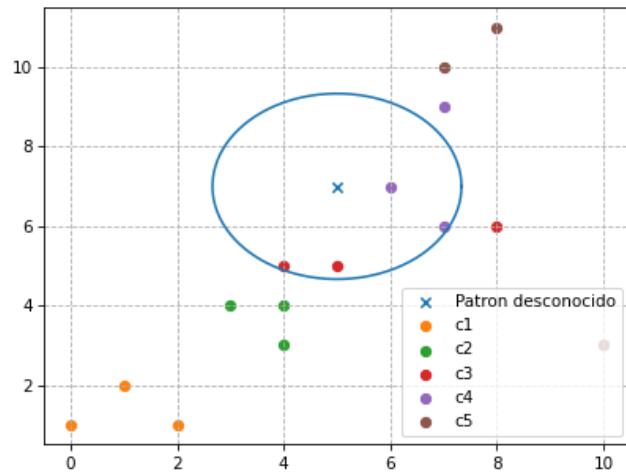


Figura 2.37: Algoritmo de clasificación KNN con $k=4$.

2.17. Realidad Aumentada

La realidad aumentada (en inglés *augmented reality* ó *AR*) es una tecnología que integra señales captadas del mundo real (típicamente video y audio) con señales generadas por computadores (objetos gráficos bidimensionales o tridimensionales); las hace corresponder para construir nuevos mundos coherentes, complementados y enriquecidos. En otras palabras, hace coexistir objetos del mundo real y objetos del mundo virtual [66].

A diferencia de las otras técnicas de realidad mezclada, la realidad aumentada permite que el usuario perciba el entorno real "aumentado" con algunos objetos virtuales (creados por computadora). En circunstancias ideales, le debería parecer al usuario que los objetos reales y virtuales coexisten en el mismo espacio [67].

Mientras que la tecnología de realidad virtual (VR), sumerge completamente a los usuarios en un mundo sintético sin ver el mundo real, la tecnología AR aumenta el sentido de la realidad al superponer objetos virtuales y señales sobre el mundo real en tiempo real [68]. En la Figura 2.38 se presenta un ejemplo de realidad aumentada.



Figura 2.38: Imagen obtenida de Google Imágenes, en donde se muestra a Pokemon Go, la cual fue la aplicación de realidad aumentada más famosa y popular del mundo en el año 2016, la cual muestra y ejemplifica de manera sencilla este concepto.

Esta tecnología aprovecha las tecnologías derivadas de la visualización para construir aplicaciones y contenidos con las cualidades que estas áreas han madurado en las últimas décadas [66].

Del procesamiento de imágenes toma la cualidad de resaltar aspectos en las imágenes captadas por la cámara de video; estos rasgos son analizados por procesos de visión por computadora para extraer propiedades geométricas del entorno y los objetos (posición, patrones fiduciarios para el reconocimiento y ubicación de objetos susceptibles a sustitución, etc.) [66].

De los gráficos por computadora toma la síntesis de objetos bidimensionales o tridimensionales y sus transformaciones, mientras que gracias a la teoría de interfaces gráficas ha sido posible la construcción de nuevas metáforas dentro de estos mundos mixtos [66].

De hecho, eliminar objetos del mundo real corresponde a cubrir el objeto con información virtual que coincida con el fondo para dar al usuario la impresión de que el objeto no está allí. Los objetos virtuales agregados al entorno real muestran información al usuario que este no puede detectar directamente con sus sentidos [68].

2.17.1. Sistemas de Realidad Aumentada

Según [67], para hacer una buena aplicación de realidad aumentada se necesitan al menos cuatro componentes básicos:

1. Un elemento que capture las imágenes de la realidad que están viendo los usuarios. Generalmente se utiliza una cámara actual de cualquier tipo.
2. Un elemento sobre el que proyectar la mezcla de las imágenes reales con las imágenes sintetizadas. Para ello se puede utilizar una pantalla, ya sea un televisor o un monitor.
3. Un elemento de procesamiento, o varios de ellos que trabajan conjuntamente. Su cometido es el de interpretar la información del mundo real que recibe el usuario, generar la información virtual que cada servicio concreto necesite y mezclarla de forma adecuada.
4. Un elemento al que llamaremos "activador de realidad aumentada". En un mundo ideal el activador sería la imagen que están visualizando los usuarios, ya que a partir de ella el sistema debería reaccionar.

Un sistema de realidad aumentada inicia con el registro de las señales del mundo real (normalmente video y audio). Estas señales son procesadas por un sistema de realce de orillas de objetos para preparar la imagen para la segmentación o extracción de objetos y el reconociendo de patrones. Este proceso permite determinar en dónde hay que remplazar un objeto real por uno virtual, cuál objeto virtual debe colocarse sobre el espacio real (el espacio de video), en qué posición y en qué perspectiva [66].

Para agilizar el proceso y permitir la interactividad, la cual requiere de gráficos en tiempo real, es conveniente que la correspondencia entre patrones, rasgos geométricos del entorno, la posición dimensional y la perspectiva de dibujado de los objetos virtuales sea preparada con anticipación. Esto puede ser con datos que entrenen al sistema para evitar muchos cálculos en tiempo de ejecución. Algunos de estos cálculos pueden ser preparados y dicha inteligencia artificial puede ser parte del sistema. Sin embargo, la síntesis de imágenes en su posición y perspectiva correctas, que permita lograr una correspondencia geométrica entre los mundos virtual y real (que el nuevo espacio secoherente para usuario), debe realizarse en tiempo real y de forma interactiva. Esta alineación de ambos mundos se logra extrayendo información tridimensional de las imágenes de video (en dos dimensiones) a partir de marcos de referencia en el mundo real y de rasgos de perspectiva que pueden ser extraídos del entorno real (los contornos de muros, geometrías simples y conocidas o medidas previo a la generación del contenido, entre otras técnicas). Esta característica restringe las aplicaciones de realidad aumentada en mundos virtuales específicos a mundos reales; de otra forma, la demanda de cálculo exige computadoras poderosas [66].

Cuando se conocen las propiedades del dibujado (se establecen la serie de transformaciones que han de aplicarse al objeto virtual; así como los parámetros de la cámara virtual y la iluminación), se crean o sintetizan y se pasan al proceso de composición de la señal de video de salida; la cual fusiona siguiendo reglas (de oclusión, por ejemplo) la señal de video original con la señal de la escena virtual. Esta nueva señal, mezcla de ambos mundos, es transferida a los monitores o proyectores [66].

Los sistemas de AR se pueden construir utilizando configuraciones basadas en monitor, como la que se muestra en la Figura 2.39. En este caso, una o dos cámaras de video ven el entorno. Las cámaras pueden ser estáticas o móviles. El video del mundo real y las imágenes gráficas generadas por un generador de escena se combinan y se muestran en un monitor frente al usuario [69].

Por último, esta señal que contiene una reconstrucción visual de la escena objetivo de la aplicación, es dirigida al sistema visual humano. Si el sistema genera una perspectiva única para ambos ojos, el usuario verá una imagen bidimensional (mono); mientras que para ver en tercera dimensión (estereografía) es necesario generar un par de imágenes, cada una con la perspectiva correspondiente a cada ojo y solo dejar ver a cada ojo su imagen correspondiente [66].

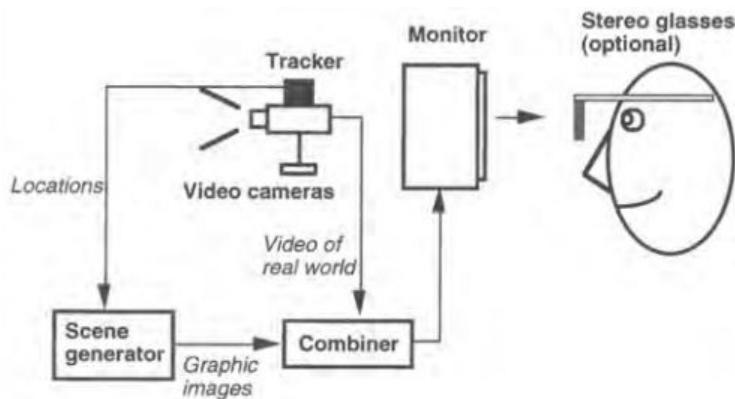


Figura 2.39: Imagen tomada de [69] que representa el diagrama conceptual de realidad aumentada basada en monitor. Las gafas estereográficas (tres dimensiones) son opcionales.

La concatenación de estos procesos según [66] resulta en un sistema con las siguientes características, las cuales definen la realidad aumentada:

- Combina objetos reales y virtuales en nuevos ambientes integrados.
- Las señales y su reconstrucción se ejecutan en tiempo real.
- Las aplicaciones son interactivas.
- Los objetos reales y virtuales son registrados y alineados geométricamente entre ellos y dentro del espacio, para darles coherencia espacial.

2.18. Bases de datos NoSQL

Durante mucho tiempo los sistemas almacenaban la información haciendo uso de bases de datos relacionales. Sin embargo, en la década del 2000 surgieron nuevos modelos de datos que comenzaron a cobrar relevancia en el desarrollo de sistemas que requerían almacenar información, generando que se acuñara el término "NoSQL" o "no relacional" para referirse a ellos [70]. Este tipo de bases de datos están optimizadas para manejar un gran volumen de datos, una baja latencia y esquemas flexibles a cambio de reducir las restricciones de cohesión que las bases de datos relacionales tienen. Algunas de las características de las bases de datos NoSQL son:

- **Flexibilidad:** Permiten un desarrollo más rápido e iterativo, volviéndolas excelentes para datos semi-estructurados o no estructurados [70].
- **Escalabilidad:** Están diseñadas para poder escalar añadiendo clusters distribuidos de hardware en vez de añadir servidores sólidos [70].
- **Alto rendimiento:** Están optimizadas para modelos de datos específicos y patrones de acceso que permite brindar un mayor rendimiento [70].

Aún entre las bases de datos NoSQL existen diferentes tipos como lo son las bases de datos "clave-valor", "de documentos" o "gráficos". Las bases de datos de documentos están diseñadas para almacenar y consultar los datos en forma de documentos tipo JSON, permitiendo un buen funcionamiento en casos donde se hace uso de catálogos, perfiles de usuario y sistemas de administración de contenido en donde el documento evoluciona con el tiempo.

2.19. BubbleTown®

BubbleTown®: Bubble Tea & Yogurt Drinks es una empresa registrada en México ante el Instituto Mexicano de la Propiedad Industrial (IMPI) con la clase 43 que hace referencia a Servicios de Restauración (alimentación). Esta empresa tiene como objetivo brindar a sus clientes una auténtica experiencia al degustar deliciosas bebidas a base de té o yogurt preparadas al instante, dando la posibilidad de elegir entre una extensa variedad de toppings y numerosas opciones de personalización, dando como resultado un producto que es original, saludable y delicioso. Para el caso del presente trabajo BubbleTown® tiene el papel del cliente real, por lo que la publicidad personalizada será en función del menú de bebidas presentado en [Menú de bebidas](#).



Figura 2.40: Logotipo de BubbleTown®

2.19.1. Propuesta de valor y ventaja competitiva



Ventajas competitivas

- Concepto propio en que el cliente selecciona del mostrador los toppings que desea agregar a su bebida, los cuales vienen incluidos sin costo adicional.
- Sabores únicos y originales. Categorías de productos exclusivas
- Mayor nivel de personalización y productos especiales con vistosas decoraciones.
- Gama de toppings extensa y diversa

Figura 2.41: Ventajas competitivas de BubbleTown®

Como se muestra en la Figura 2.41, través de sus diferentes bebidas, BubbleTown® proporciona la posibilidad de "armar" un producto personalizado ya que se pueden escoger entre los distintos toppings del mostrador por el mismo precio.

2.19.2. Misión de la empresa

"Para nosotros los placeres simples de la vida son los que la hacen significativa y memorable. Nuestra misión es traer a nuestros clientes aquellas emociones que hagan de su vida una experiencia más placentera y satisfactoria."

2.19.3. Visión de la empresa

"Creemos que toda persona tiene derecho a llevar una vida plena y saludable. Esta es la filosofía que nos motiva para trabajar y seguir nuestro camino. A través de nuestras relaciones con clientes y empleados, esperamos tocar vidas e inspirar sueños."

2.19.4. Mercado meta

El mercado meta son personas de 20 a 44 años que habiten en la delegación Cuauhtémoc de la Ciudad de México, con un ingreso socio económico C+, que tengan ingresos mensuales superiores a \$15,000 y que disfruten de opciones gastronómicas diferentes y/o novedosas.

2.19.5. Ubicación y sucursal

La sucursal se encuentra en la calle Marsella, dentro de la delegación Cuauhtémoc en la Ciudad de México y brinda atención al cliente en un horario de 10:00 am - 9:00 pm entre semana y de 10:30 am - 9:20 pm los sábados y domingos. La ubicación se puede apreciar en la Figura 2.42. El local tiene la apariencia como se muestra en la Figura 2.43 y al entrar se encontrará con un ambiente colorido como el mostrado en la Figura 2.44.

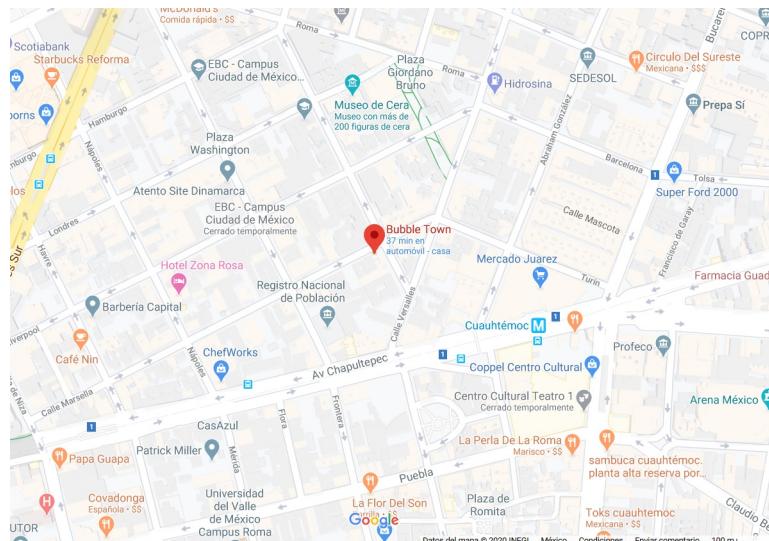


Figura 2.42: Ubicación de BubbleTown® (Calle Marsella 14, Juárez, Cuauhtémoc, 06600 Col Juárez, CDMX). Tomada haciendo uso de Google Maps



Figura 2.43: Exterior de la sucursal. Obtenida de Google Imágenes



Figura 2.44: Interior de la sucursal. Obtenida de Google Imágenes

2.19.6. Menú de bebidas

<p>Bubble Tea Original Series bebidas a base de té con leche</p> <ul style="list-style-type: none"> • Bandung Mix pétalos de rosa • Bandung & Coco té de rosas con crema de coco • Passion Shake fruta de la pasión • Taro Shake delicioso tubérculo oriental • Thai Mix té tailandés con crema de coco y almendra • Thai Supreme té tailandés con crema de coco, flor de azahar y anís • Especial de Almendra almendra, vainilla y canela • Delicia de Milán almendra, avellana y chocolate • Falooda Mix té de rosas con pistache • Sharbat Mix almendra, pistache y hierbas frescas • Sharbat Supreme almendra, pistache y flores aromáticas • Té de Violeta & Lavanda • Chai Tradicional • Choco Chai • Té Verde - Sandía • Matcha Tradicional • Choco Matcha • Mento Matcha • Matcha & Coco • Choco Avellana • Choco Menta • Especial de Oreo® 	<p>Bubble Refresh Series bebidas a base de té sin leche</p> <ul style="list-style-type: none"> • Lychee Punch deliciosa fruta tropical • Passion Punch fruta de la pasión • Té Negro & Limonada <p>Bubble Natural Series Tisanas e infusiones saludables, veganas y bajas en calorías</p> <ul style="list-style-type: none"> • Té Verde - Clorofila • Té Verde con Arándano • Té Rojo con Frutos del Bosque • Té Blanco con Frambuesa • Kombucha con Limón y Jengibre té fermentado con limón, jengibre y chía • Kombucha con Guanábana té fermentado con guanábana y chía • Café verde con Pepino y Limón 	<p>Bubble Tea Espresso Series bebidas a base de café</p> <ul style="list-style-type: none"> • Café Latte & Matcha café latte con té matcha • Mocaccino & Matcha capuccino, mocha y té matcha • Caramel Macchiato • Vanilla Macchiato <p>Yogurt Drinks bebidas a base de yogur natural</p> <ul style="list-style-type: none"> • Bosque Secreto fresa, granadina y almendra • Pasión Tropical fruta de la pasión y granadina • Tutti-Mora frutos del bosque • Orange Blend naranja y fresa • Dragón Rosa pétalos de rosa 	<ul style="list-style-type: none"> • Doble Dragón pétalos de rosa y crema de coco • Brisa Fusión pétalos de rosa y hierbas frescas • Coco Ártico crema de coco y hierbas frescas • Sueño Invernal yogur natural con toque cítrico • Calpimix Original coctel lácteo a base de Calpis® • Calpimix de Mango coctel de Calpis® y mango • Calpimix de Lichi coctel de Calpis® y lichi • Calpimix de Coco coctel de Calpis® y coco • Caplimix de Menta coctel de Calpis® y menta • Calpimix de Rosas coctel de Calpis® y rosas
--	--	---	---



Figura 2.45: Menú de bebidas

Como se muestra en la Figura 2.45, BubbleTown® cuenta con 51 bebidas, segmentadas en 5 categorías diferentes. Una vez escogida la bebida se puede escoger entre distintos toppings que se encuentran en el mostrador y escoger una

"Modalidad". Esta modalidad puede ser "Zen" (bebida caliente), "Frozen" (frappé) o "Iced" (fría).

2.19.7. Competencia de BubbleTown®

Dentro del mercado de las bebidas con opción de personalización ya se encuentran algunos competidores, entre ellos, los cuatro principales se muestran a continuación:

- BUBBA TEA & CO.®
- BOBA Fusión: Bubble Tea Bar.®
- Cassava Roots®
- Starbucks Coffee®

2.20. Discusión del capítulo

En este capítulo se ha hablado del impacto de la [Personalización en el marketing](#), de la cual se puede concluir que la personalización de un producto o servicio solamente funciona si esta llega a tener un impacto real en los sentimientos, comportamientos y sobretodo en las acciones de los clientes de una empresa, ya que este proceso consume mucho tiempo, recursos humanos y materiales, y que nunca será completamente precisa. El principal objetivo es generar un sentimiento de lealtad en el cliente, para que así, este continúe consumiendo en dicha empresa, a pesar de lo llamativo que pueden ser las ofertas o productos de otra, y para esto, previamente se necesita generar satisfacción y retención en él.

Otra aspecto muy importante es el marco conceptual de la [Automatización del marketing](#), pues es casi idéntico a cómo se va a comportar el prototipo resultante de este trabajo: con base en la estimación de los atributos de edad, género y personalidad (información actual), y siguiendo una serie de condiciones o reglas, se va a ofrecer una recomendación de una bebida para cada uno de los clientes que interactúen con dicho prototipo de sistema.

En la sección [Detección de objetos](#) se habló de distintas formas para detectar un objeto, empezando con técnicas simples como sustracción de fondo o detección de bordes, se debe recordar que estas técnicas son susceptibles a diversos fallos e implican tener a un experto del tema de interés, además que su costo computacional es alto.

Se introdujeron las [Redes neuronales](#), que son sistemas inspirados en [La neurona biológica](#) y que son capaces de emular el comportamiento humano hasta cierto punto. Se aborda a [El Perceptrón Multicapa](#), primera arquitectura capaz de resolver algunos problemas no lineales, que junto a su regla de aprendizaje Back propagation, se han vuelto muy importantes en el campo de estudio, al grado de funcionar como clasificador en las etapas finales de las [Redes neuronales convolucionales](#).

Las [Redes neuronales convolucionales](#) se vuelven un tema bastante atractivo para el presente trabajo debido a sus buenos resultados al tratar con imágenes, cuestión relevante en el proyecto para la detección de rostros, clasificación del [Género](#), estimación de la [Edad](#) y [Personalidad: Big Five](#). Debido a estos buenos resultados, en la literatura se encuentran variantes de este tipo de arquitecturas como lo son la arquitectura [VGG16](#), [Single Shot Detection](#) y la [MobileNet](#), una variante para dispositivos con bajo poder computacional.

Como se abordó en la sección de [¿Por qué son importantes los rangos generacionales?](#), el saber la edad da una ventaja competitiva ya que es más fácil determinar el comportamiento y preferencia hacia algún tipo de sabor de un cliente y, obtener este dato a partir de su rostro es posible; sin embargo, no es una tarea fácil aunque haya distintos métodos para obtenerlo. Por otro lado el determinar el género de una persona también permite tener conocimiento sobre su

tendencia a preferir algún tipo de sabor, lo que lleva a una recomendación mucho más exacta. Gracias a la evolución en la capacidad de procesamiento de cómputo y la capacidad de obtener una gran cantidad de datos, el entrenamiento de una red CNN profunda para la obtención del género y edad a partir de una imagen del rostro es una tarea posible y viable debido a que permite aprender transformaciones no lineales, a diferencia de los métodos no convolucionales.

Por otro lado, la detección de personalidad en un individuo fue tratada en la sección de [Personalidad: Big Five](#) por medio de este modelo, pues al ser una característica de una naturaleza tan compleja en una persona, además de sus escasos métodos de medición, e incluso de definición misma, el contar con un modelo que clasifica la mayoría de rasgos de comportamiento con la premisa de que algunos de ellos tienden a expresarse en actitudes y reacciones muy similares (como la agrupación de términos de la que se habló en la sección [Teoría de rasgos](#)), facilita la forma de definir, medir y estimar la personalidad de un individuo.

Si bien la adquisición de los rasgos distintivos de cada persona ([Edad, Género y Personalidad: Big Five](#)) es importante, es igualmente importante el saber cómo utilizar estos datos, por ello en la sección de [Recomendador](#) se presentan dos formas de llevar a cabo el recomendador y distintos algoritmos usados para clasificar a un patrón desconocido a una clase, entre estos se abordaron los [Árbol de decisión](#), [Clasificador bayesiano ingenuo](#) y [KNN \(vecino más cercano\)](#). Más adelante en el capítulo de [Diseño e Implementación](#) se elige a uno de estos para ser implementado en el sistema tomando en cuenta las circunstancias actuales del proyecto.

En la sección de [Realidad Aumentada](#) se habló sobre la diferencia entre este concepto y el de Realidad Virtual, que frecuentemente son confundidos, siendo la primera el agregar elementos digitales a un entorno de nuestra realidad, y la segunda recrear o sustituir completamente un entorno real por uno digital. Se identificaron los principales componentes que un sistema debe tener para ser considerado de realidad virtual, los cuales son: un elemento que capture imágenes del mundo real, un elemento sobre el cual se va a desplegar el resultado, un elemento que haga la combinación y procesamiento de las imágenes reales y las digitales, y por último, un disparador que active la funcionalidad de los elementos anteriores y por ende, al sistema en general. Para fines de este trabajo, los componentes que corresponden a los mencionados anteriormente son respectivamente: la cámara de video, la pantalla, la unidad de procesamiento central y la detección de rostros humanos con la red neuronal [Single Shot Detection](#).

CAPÍTULO 3

Estado del Arte

3.1. Introducción al capítulo

Para comprender el panorama y entender el contexto relacionado al objetivo que se busca alcanzar con este trabajo, en la sección del [Revisión de trabajos de Publicidad Dirigida](#) se hace una recopilación de sistemas en el mercado o académicos cuya función es similar a la establecida. A lo largo de este capítulo, se habla sobre otros trabajos previos, aplicaciones relacionadas o antecedentes a los distintos temas que se manejaron en el capítulo pasado de [Marco teórico](#), para ser tomados como referencia en la elaboración del sistema propuesto, siendo estos la estimación de edad, de género y de personalidad a partir del análisis facial, y los sistemas de realidad aumentada.

Partiendo con el conocimiento de [Biometría blanda](#) y dejando en claro la importancia de la obtención de datos relevantes, como la edad y género, a partir de una imagen la sección [Revisión de trabajos de Edad y género](#) muestra trabajos basados en [Redes neuronales convolucionales](#) que han mostrado muy buenos resultados para la obtención de dichas características del rostro. En la sección de [Revisión de trabajos de Big Five](#) se exponen fuentes de información no invasivas y con buenos resultados para detectar la personalidad de un individuo, así como algunos métodos y formas de ser medida a través de este, remarcando las ventajas de reemplazar el uso de métodos tradicionales, como encuestas especializadas, por métodos de [Deep Learning](#), que se enfocan en el análisis facial, mostrando la comparación entre todos estos, sus ventajas, desventajas y complejidad para cada uno.

Respecto a la presentación de publicidad utilizando [Realidad Aumentada](#), en la sección de [Revisión de trabajos de Realidad Aumentada](#) se abordan brevemente un par de aplicaciones de este tipo en el ámbito comercial, publicitario, y de entretenimiento, con el fin de remarcar algunas técnicas, métodos o formas de implementación y despliegue para los diseños e información de las recomendaciones obtenidas para cada cliente.

Al final del capítulo se hace un resumen y discusión, en donde se elige la información, modelos, metodologías y propuestas finales o definitivas que se toman como referencia y apoyo para alcanzar los objetivos de este trabajo. Adicionalmente, se da una idea más clara de en qué consistirá el prototipo a desarrollar.

3.2. Revisión de trabajos de Publicidad Dirigida

En primer lugar se debe de hablar de aquellos sistemas existentes cuyo principal enfoque sea la generación de publicidad dinámica a partir de la detección de un individuo frente de este, el cual es detectado ya sea por medio de cámaras, sensores, u otro tipo de herramientas, resaltando las características existentes para cada uno, lo que resulta atractivo y lo que se considera de suma importancia para alcanzar los objetivos de este proyecto (es decir, la posibilidad de tomar elementos de estos como base para diseñar e implementar los de este trabajo), y aquellas funcionalidades que les faltarían, o deficiencias en ellas, de las cuales se pueden aprovechar para implementarlas o mejorarlas.

Después de una exhaustiva revisión en la literatura, se encontraron dos trabajos con conceptos similares a las de este trabajo:

1. Una suite de publicidad dirigida de Intel® llamada **Audience Impression Metrics Suite** [9] [71].
2. Un Trabajo Terminal de la ESCOM Llamado **Aplicación móvil para recomendación de bebidas basada en redes neuronales profundas** (Trabajo Terminal 19-1-004) [10].

La **suite de Intel®** [9] [71], distribuida en 2011 en EUA y cuya apariencia es mostrada en la Figura 3.1, es un dispositivo de publicidad dirigida que hace uso de sistemas automatizados para detectar posibles consumidores a través de visión por computadora. Entre sus características más llamativas están el uso de sensores anónimos que buscan y capturan de forma temporal patrones de caras o cuerpos dentro de un alcance de visión predeterminado, en otras palabras, la capacidad para detectar rostros; el análisis de rasgos antropométricos para proporcionar anuncios a través de pantallas, dependiendo de la persona que se colocara frente a ella, y además esta es generada en función de atributos como la edad, la altura, la raza y el género de dicha persona. Por otra parte, esta suite incluye gran variedad de funciones de estadísticas de datos, que incluye desde generación de reportes, análisis y almacenamiento de datos en la nube de Intel®, hasta conteo de rostros detectados, medición del tiempo que un espectador permanece viendo el anuncio desplegado antes de irse, y por último, cálculo de la cantidad de personas que transitan frente a la pantalla simultáneamente, aunque esta operatividad solamente está disponible en entornos limitados o beta [9].

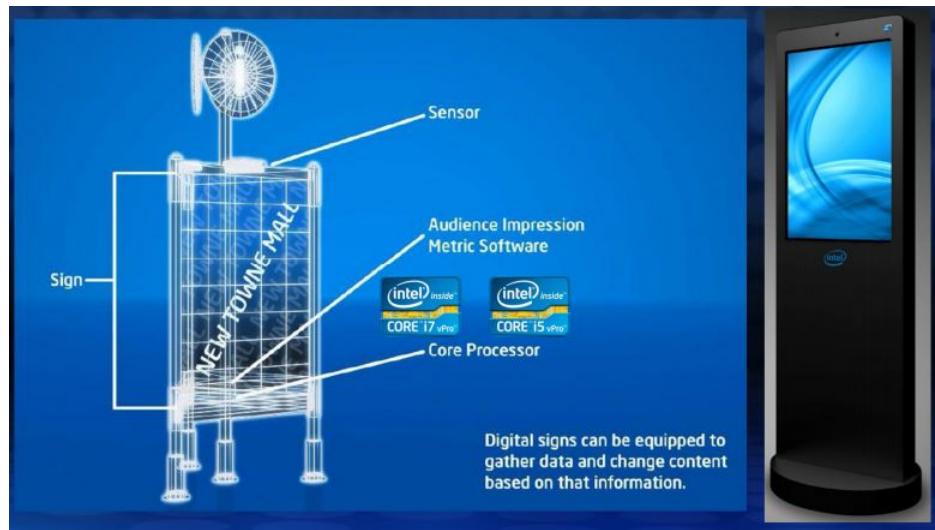


Figura 3.1: Imagen tomada de [9] que muestra el diseño del dispositivo que integra a la **Intel® Audience Impression Metrics Suite**. Sus principales componentes son: un sensor para la detección de personas y una pantalla para mostrar todos los carteles publicitarios.

Al revisar los resultados documentados en [9], se tiene que los niveles de exactitud en la detección de rostros son de aproximadamente 98 %, en la estimación del género son de aproximadamente 86 %, y en la estimación de la edad están dentro de un rango de 70 % a 80 %. No obstante, una de las principales desventajas de esta suite es el costo, el cual consiste en una renta mensual de \$200.00 USD [71].

Por otro lado, el **trabajo de J. M. Balderas Sánchez** en [10], realizado en el año 2018 y el cual es ilustrado en la Figura 3.2, consiste en una aplicación móvil que permite recomendar a los clientes de la empresa BubbleTown® una bebida a partir de su rostro. La generación de dicha recomendación se hace utilizando factores como la edad, el género y el estilo de vida de una persona, además de la temperatura ambiental. Estos factores son obtenidos mediante un proceso basado en redes neuronales convolucionales, las cuales estiman la edad, el género y el estilo de vida a partir de una fotografía del rostro. Aunado a esto, esta el despliegue de dicha recomendación con diseños previamente establecidos, es decir, se seleccionaba un cartel o imagen disponible desde un catálogo y esta es mostrada al cliente, la cual contiene el nombre de la bebida, la modalidad correspondiente y una fotografía real de la bebida [10]. Al revisar los resultados finales de la aplicación, se tiene una efectividad por encima del 60 % en cada una de las redes neuronales profundas.

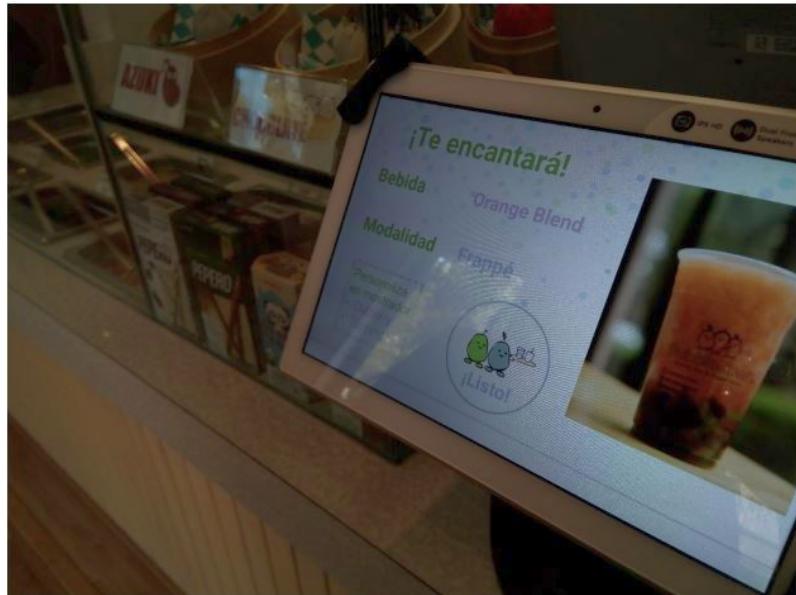


Figura 3.2: Imagen tomada de [10] que muestra una tablet con la **Aplicación móvil para recomendación de bebidas basada en redes neuronales profundas** del Trabajo Terminal 19-1-004, instalada en el local de BubbleTown®. En el dispositivo móvil aparece un ejemplo del diseño y de los datos que se muestran al cliente al generar una recomendación de una bebida.

Sin embargo, una de las desventajas de este trabajo es que es una aplicación móvil: una vez que se implementó en el local de BubbleTown®, era necesario tener un dispositivo móvil (en este caso, una tablet) y que el cliente tuviera que interactuar de forma directa con ella. Es decir, por si solo el dispositivo móvil con la aplicación debe de retener la atención del cliente potencial. Luego, este debe de acercarse e interactuar con ella, para después de mostrar unos avisos de privacidad y del uso de sus datos, solicitar una fotografía al usuario, tomarla y pedirle la confirmación al mismo para hasta justo ahora sí finalmente brindarle la recomendación, todo esto suponiendo que el cliente potencial no desvió la atención de ella o continuo por su propio camino, lo cual es lo más común. Como se vio en la sección de **Los deseos y la perspectiva del consumidor** en el capítulo del **Marco teórico**, esto más que aumentar la probabilidad

de compra de algún producto, la reduce significativamente, pues se invierte demasiado tiempo en la interacción con la aplicación móvil, lo que puede desanimar a clientes potenciales que en un inicio estaban dispuestos a realizar una compra en [BubbleTown®](#).

A raíz de esto, entonces se busca tener esta misma funcionalidad, pero ahora con la propuesta de que todo el proceso este automatizado, y además, brindarle al usuario un equipo físico que capte su atención de inmediato, y al mismo tiempo, la interacción con este sea prácticamente nula.

Aquí es donde la [suite de Intel®](#) [9] [71] puede solucionar este problema, pues si se tuviera una hipotética combinación de ambos trabajos, resultaría en un sistema compuesto de un sensor o cámara que detecte a aquellas personas que se detengan a ver dicho sistema, y que a partir de su rostro obtuviera estimaciones de determinadas características, para así generar una recomendación de [BubbleTown®](#) personalizada para este posible consumidor, desplegándola en una pantalla.

Las características que tienen en común estos dos sistemas son: reconocimiento y análisis facial, detección de factores externos e internos, y recomendaciones basadas en análisis e integración de estos últimos.

De entre todos los factores o atributos de la persona que ambos sistemas utilizan, hay dos que tienen en común: la edad y el género. Eso habla de la importancia de estos en las preferencias que puede tener un individuo, por lo que se ha decidido hacer una revisión de los trabajos que tratan de extraer ambas características, con la condición de que sea por medio de análisis fácil.

Además de estos, en el trabajo de [trabajo de J. M. Balderas Sánchez](#) [10] también se utiliza el estilo de vida para la recomendación. A pesar de esto, como se habló en la sección de [Personalidad: Big Five](#) en el capítulo de [Marco teórico](#), la personalidad influye demasiado en las preferencias, gustos y deseos del consumidor, por lo que se ha decidido hacer una revisión de aquellos trabajos que miden la personalidad de una persona, utilizando el modelo de Big Five, también manejado en dicha sección.

Así pues, en las siguientes secciones se habla sobre trabajos previos de extracción de género, edad y Big Five por medio del análisis facial.

3.3. Revisión de trabajos de Edad y género

Como ya se ha mencionado en el [Marco teórico](#), desde 2011 se propuso por primera vez el uso de una CNN para la estimación de la edad a través de un rostro [44] pero los esfuerzos no se detuvieron con esto, sino que aparecieron varias propuestas para estimar la edad de un individuo a través del mismo. Inicialmente la extracción de características eran concentrándose solo en una de ellas, ya sea la edad, género o raza. Sin embargo, nuevas investigaciones mostraron que el aprendizaje de las características de forma conjunta puede mejorar el rendimiento de cada tarea individual [36], por ello es que a lo largo de las distintas arquitecturas de las que se hablarán a continuación, la mayoría serán redes neuronales convolucionales multitarea, lo que significa que cumplirán con dos o más tareas.

3.3.1. Arquitectura propuesta por Orozco et al (2017)

El sistema propuesto por Orozco et al (2017) [72] es una red neuronal con solo un propósito: obtener el género de la persona a través de una imagen; consta de dos etapas principales:

A.- Generación de regiones candidatas (ROI) delimitadas por cuadros delimitadores (BBs).

B.- Clasificación de las regiones candidatas en dos clases: persona masculina y persona femenina.

En la Figura 3.3 se expresa con un ejemplo el proceso que sigue el sistema, empezando con una imagen de entrada, seguida por la etapa A de generación de regiones candidatas (encerradas dentro de un cuadro verde) y finalizando en la etapa B, clasificación del género perteneciente al rostro (encerrando en cuadro azul el rostro masculino y en rojo el femenino).



Figura 3.3: Proceso del sistema presentado en [72].

La arquitectura (llamada Ubunsa) empleada en [72] se puede observar en la Figura 3.4 y consta de 8 capas. Cabe señalar que la [Función de activación](#) en todas las capas es una unidad lineal rectificada (ReLU), excepto la capa de salida del [El Perceptrón Multicapa](#) que utiliza una [Función de activación](#) softmax.

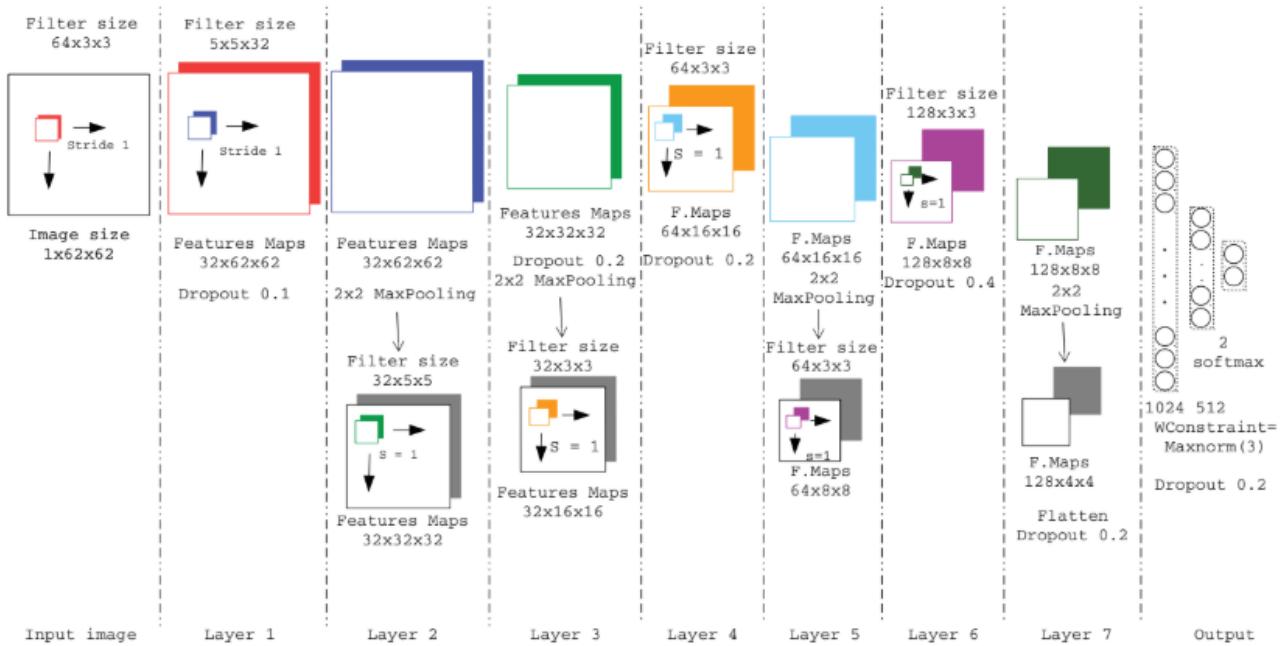


Figura 3.4: Arquitectura Ubunsa. Las primeras 7 capas se utilizan para la extracción de características. La capa de salida es [El Perceptrón Multicapa](#) con 1 capa oculta para la clasificación [72].

La misma arquitectura vista en la Figura 3.4 es expresada de forma más detallada en la Figura 3.5. La primera columna indica el número de capa. La columna "Tipo" indica el tipo de subcapas (es decir, convolución, deserción, agrupación máxima, aplanamiento o densidad). La forma de salida indica el tamaño de los mapas de características de salida. # Param indica el número de parámetros a entrenar. La última columna muestra las dimensiones del filtro (para convolución) o máscara (para maxpooling).

Layer	Type	Output shape	#Param	Filter (conv.) Mask (pool.)
1	conv.	(32, 62, 62)	1600	(7, 7, 3)
	drop.	(32, 62, 62)	0	-
2	conv.	(32, 62, 62)	25632	(5, 5, 32)
	pool.	(32, 31, 31)	0	(2, 2)
3	conv.	(32, 31, 31)	9248	(3, 3, 32)
	drop.	(32, 31, 31)	0	-
	pool.	(32, 15, 15)	0	(2, 2)
4	conv.	(64, 15, 15)	36928	(5, 5, 32)
	drop.	(64, 15, 15)	0	-
5	conv.	(64, 15, 15)	18496	(3, 3, 32)
	pool.	(64, 7, 7)	0	(2, 2)
6	conv.	(128, 7, 7)	73856	(3, 3, 64)
	drop.	(128, 7, 7)	0	-
7	conv.	(128, 7, 7)	147584	(3, 3, 128)
	pool.	(128, 3, 3)	0	(2, 2)
8	flatt.	(1152)	0	-
	drop.	(1152)	0	-
	dense	(1024)	4195328	-
	drop.	(1024)	0	-
	dense	(512)	524800	-
	drop.	(512)	0	-
	dense	(2)	1026	-

Figura 3.5: Resumen de la información sobre la arquitectura presentada en [72].

Para entrenar la red presentada en [72] se utilizó el dataset LFW [73] y el Gallagher's [74]. La red se configuró con los hiper-parámetros mostrados en la Tabla 3.1.

Hiper-parámetros	Valor
Épocas	50
Factor de aprendizaje	0.01
Momentum	0.9
Tamaño de batch	100

Tabla 3.1: Hiper-parámetros usados en Ubunsa [72].

Por último, los resultados de la exactitud obtenida por Ubunsa [72] están presentados en la Tabla 3.2.

Autor	Exactitud en LFW [73]	Exactitud en Gallagher's [74]
Dago-Casas [74]	89.77	86.61
Fazl-Ersi [75]	91.59	91.59
Mansanet [76]	94.48	90.58
Ubunsa [72]	95.42	91.48

Tabla 3.2: Resultados de Ubunsa [72].

3.3.2. Arquitectura propuesta por Ranjan et al (2017)

Dado a que la red Ubunsa de [72] demuestra muy buenos resultados, es necesario investigar trabajos que puedan hallar más datos de la persona dada una imagen. La red convolucional multi propósito de Ranjan et al (2017) [77] es capaz de detectar caras, extraer puntos clave, plantear ángulos, determinar la expresión de la sonrisa, el género y estimar la edad, todo simultáneamente [77].

Además, asigna un descriptor de identidad a cada rostro que se puede utilizar para el reconocimiento y la verificación de rostros. El algoritmo propuesto se entrena en un framework Multi-Task Learning (MTL) que crea una sinergia entre diferentes tareas relacionadas con la cara, mejorando el rendimiento de cada una de ellas [77]. Dicho framework se puede apreciar en la Figura 3.6.

Goodfellow [27] interpreta MTL como una metodología de regularización para CNN profundas. Por lo general, una tarea de análisis de caras requiere una región de cara recortada como entrada. La CNN procesa la cara para obtener una representación y extraer información significativa relacionada con la tarea. Las capas inferiores de la CNN aprenden características comunes a un conjunto general de tareas de análisis de caras, mientras que las capas superiores son más específicas para tareas individuales. Por lo tanto, los parámetros de las capas inferiores de CNN se comparten entre diferentes tareas para producir una representación facial genérica que posteriormente es procesada por las capas específicas de la tarea para generar los resultados requeridos [77]. La propia Figura 3.6 muestra un claro ejemplo de esto.

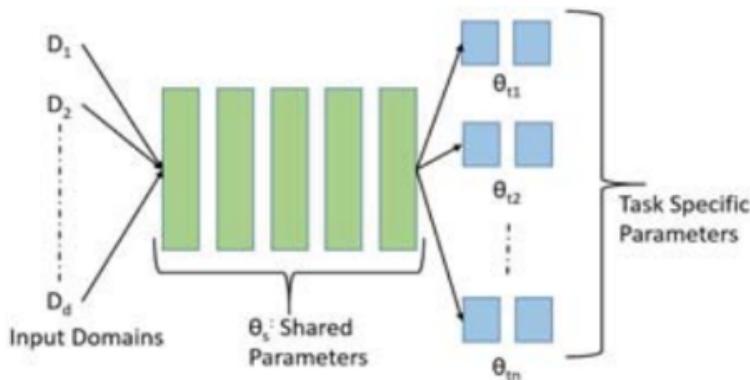


Figura 3.6: Framework general de aprendizaje multitarea para una arquitectura CNN profunda. Las capas inferiores se comparten entre todas las tareas y dominios de entrada [77].

La Figura 3.7 representa la arquitectura de la CNN usada en [77]. En ella se ve claramente como las multitareas ocupan una misma extracción de características iniciales, y luego estas se dividen con distintos objetivos, entre ellos la obtención de la edad y género.

El reconocimiento de género fue tratado como un problema de clasificación binaria similar a la detección de rostros utilizando entropía cruzada. Por otro lado, la tarea de estimación de edad se plantea como una regresión en la que la red aprende a predecir la edad a partir de una imagen facial.

Se evaluó el rendimiento del género en los conjuntos de datos CelebFaces Attributes (CelebA) [78] y ChaLearn Faces of the World [79] a gran escala. Si bien CelebA [78] tiene una amplia variedad de temas, en su mayoría contienen caras frontales. Faces of the World [79] tiene amplias variaciones en escala y puntos de vista de caras. La tabla 3.3 muestra una comparación de los resultados obtenidos en el trabajo contra otros utilizando como dataset a CelebA [78]. El resultado es muy convincente, logrando un 99 % de exactitud.

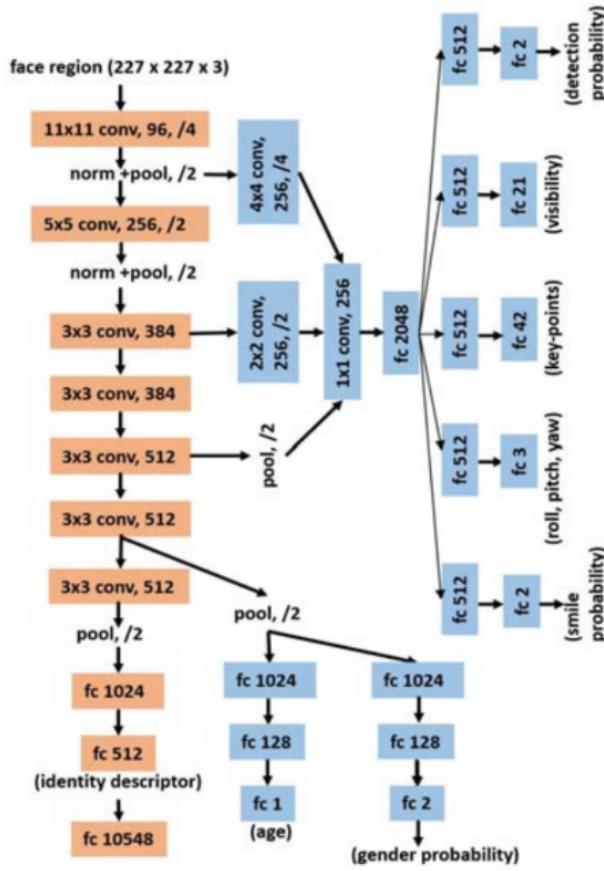


Figura 3.7: Arquitectura CNN. Cada capa está representada por el tamaño del kernel filtro, el tipo de capa (convolucional o completamente conectada), el número de mapas de características y el paso del filtro. Las capas naranjas representan la red pre-entrenada de Sankaranarayanan et al. [80], mientras que el azul representa capas añadidas para el framework MTL [77].

Método	Porcentaje de exactitud
PANDA-1 [81]	97 %
MT-RBM [82]	90 %
LNets+ANet [78]	98 %
HyperFace [83]	97 %
Walk & Learn [84]	96 %
Ranjan et al [77]	99 %

Tabla 3.3: Resultados de la detección de género utilizando CelebA [78], tomado de [77].

Por otro lado, se utilizó el conjunto de datos de desafío de estimación de edad aparente Chalearn LAP2015 [85] y la base de datos de envejecimiento FG-Net [86] para evaluar la tarea de estimación de edad. La Tabla 3.4 es un resumen de los resultados obtenidos usando como métrica el error. Nótese que aunque el error de [77] no es el más pequeño,

sigue siendo un buen resultado.

Método	Error
UMD [87]	0.359
Human	0.34
CascadeAge [88]	0.297
CVL ETHZ [89]	0.278
x ICT-VIPL [90]	0.292
Ranjan et al [77]	0.293

Tabla 3.4: Resultados de la detección de edad, tomado de [77].

3.3.3. Arquitectura propuesta por Xing et al (2017)

La CNN multi propósito de [77] demuestra buen desempeño en sus resultados, en el trabajo de Xing et al (2017) [91] se realizó un diagnóstico sobre los tres tipos de formulaciones (clasificación, regresión o regresión ordinal) para estimar la edad usando cinco funciones de costo así como tres distintas arquitecturas multitarea que incluyen la estimación de la edad, la clasificación de género y raza.

La regresión ordinal, también denominada clasificación ordinal, se encuentra entre la clasificación multiclas y la regresión métrica. Su configuración del problema es exactamente la misma que la de la clasificación multiclas, que es predecir la etiqueta de categoría para una instancia de entrada. Sin embargo, las categorías en el problema de regresión ordinal tienen una relación ordinal. La regresión ordinal también puede verse como una versión discreta de la regresión métrica, pero la diferencia es que el número de categorías en la regresión ordinal es finito y las distancias entre categorías son indefinidas [92].

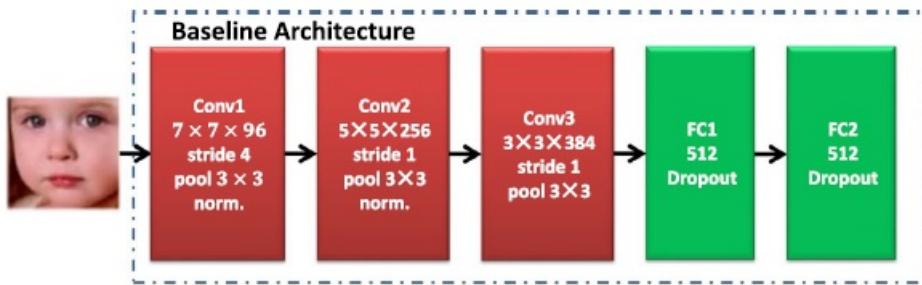
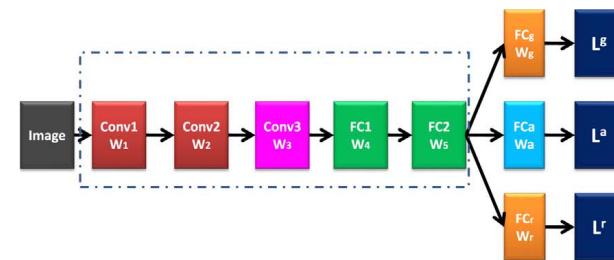
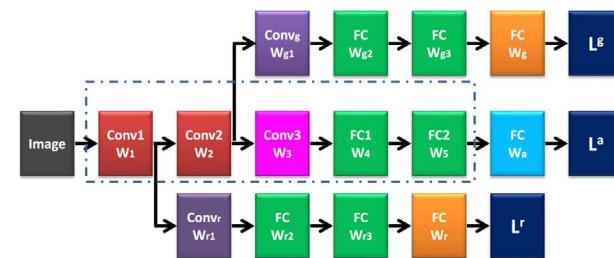


Figura 3.8: Arquitectura base en [91].

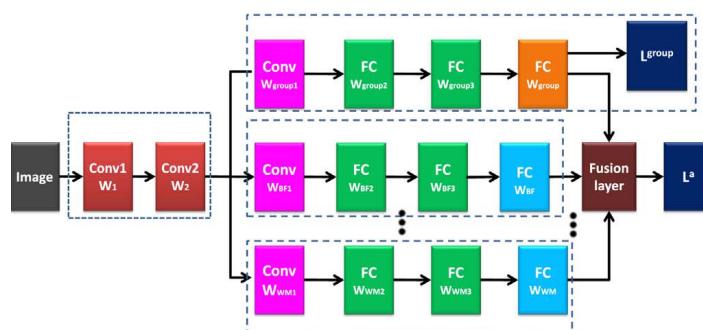
En la Figura 3.8 se puede observar la estructura de la arquitectura base del artículo que, a su vez, está basada en la AlexNet[93] y fue seleccionada debido a que ha demostrado tener buen rendimiento en la clasificación de la edad. A la red entra una imagen de 227×227 a color, que será propagada a través de tres capas convolucionales y dos capas completamente conectadas. La primera capa convolucional (Conv1) tiene un filtro de $7 \times 7 \times 96$ con un stride de cuatro. La segunda capa convolucional (Conv2) tiene un filtro de $5 \times 5 \times 256$ con una zancada de uno. La última capa convolucional (Conv3) tiene un filtro de $3 \times 3 \times 384$ con un stride de uno. Todas las capas convolucionales están seguidas de una capa Max-Pooling de 3×3 con una zancada de 2. Seguido por la última capa convolucional se tienen dos capas completamente conectadas con 512 neuronas cada una. Todas las capas (convolucionales y completamente conectadas) usan ReLU como función de activación.



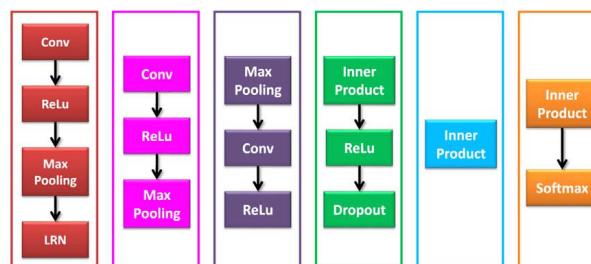
(a) Arquitectura paralela de aprendizaje multitarea



(b) arquitectura de aprendizaje multitarea profundamente supervisada



(c) Arquitectura de aprendizaje multitarea híbrida



(d) Estructura interna de cada bloque

Figura 3.9: Tres distintas arquitecturas multitarea para la estimación de edad y clasificación de género y raza [91].

Haciendo uso de la arquitectura base (Figura 3.8) se proponen tres arquitecturas para llevar acabo el aprendizaje multitarea de la edad, género y raza, como se puede ver en la Figura 3.9.

La arquitectura de aprendizaje multitarea paralela (Figura 3.9 (a)) se ha usado en muchos trabajos anteriores y ha demostrado muy buen rendimiento en la tarea de reconocimiento y detección de objetos. La diferencia sobre la arquitectura base es que al final se le agregan tres capas completamente conectadas. Para el caso del género y la raza se utiliza con una función softmax, mientras que para la estimación de la edad se tiene una función de error MAE.

La arquitectura de aprendizaje multitarea profundamente supervisada (Figura 3.9 (b)) es la segunda opción, donde la idea es fusionar las distintas tareas de forma progresiva a lo largo de la arquitectura base. Como se puede observar en la Figura 3.9 (b), en este caso se le agregan ramas intermedias para la tarea del género y raza, mientras que la tarea de la edad sigue usando toda la arquitectura base. Esta idea surge debido a que la estimación de la edad es una tarea más complicada de aprender que las tareas de clasificación de raza y género. Debido a que las características extraídas en las principales capas convolucionales pueden tener ruido y no ser lo suficientemente discriminatorias, se agrega una capa de reducción dimensional ($Conv_g$ y $Conv_r$) con un kernel de 1x1.

La última opción es una arquitectura de aprendizaje multitarea híbrida que fue diseñada con la premisa de que la estimación de la edad puede ser drásticamente influenciada por el género y la raza, por lo que se propone la arquitectura mostrada en la Figura 3.9 (c) donde se puede observar que se encuentra una capa de fusión antes de estimar la edad. La arquitectura se divide en 4 secciones:

- Parte compartida: Consta de las dos primeras capas convolucionales de la arquitectura base (3.8).
- Grupo de género racial: Consta de la tercera capa convolucional y las dos capas completamente conectadas de la arquitectura base. Por último cuenta con una función softmax.
- Parte de estimación de edad específica del grupo: Están compuestas por una capa convolucional y dos capas completamente conectadas. Al final tiene una función de error MAE. Hay un bloque de este tipo por cada grupo de género racial (Femenino-blanco, Femenino-Negro, Masculino-blanco, Masculino-negro)
- Capa de fusión: En esta capa se fusionan las edades específicas de los cuatro estimadores de edad para cada grupo de género racial.

Para el entrenamiento de todas las redes propuestas se configuraron minibatches de 127 elementos, Como optimizador se usó el descenso de gradiente estocástico con impulso de 0.9 y disminución de peso de 5×10^{-4} . En el caso de las capas conectadas, se les aplicó a todas un dropout de 0.5. La tasa de aprendizaje se estableció en 10^{-3} y cada 10,000 iteraciones era dividido entre 10 hasta llegar a 50,000 iteraciones.

El primer experimento realizado fue para evaluar el mejor método para estimar la edad entre la clasificación, regresión y regresión ordinal. La primera prueba realizada fue estimando la edad a través de métodos de clasificación, para lo que se usaron dos formas de codificar las clases de la red: one hot y una distribución gaussiana de etiquetas. De esta prueba se obtuvo que la distribución de etiquetas es mejor debido a que mientras con la codificación One-Hot asume que cada clase es independiente, en el caso de la distribución de etiquetas sí se considera la correlación entre las distintas edades. Haciendo el estudio con métodos de regresión se obtuvo que la estimación del error a través de MAE (Error absoluto medio) es mejor de MSE (Error cuadrático medio) debido a que la primera opción es más robusta a los outliers. Por último se analizó la regresión ordinal, donde fue comparada con la clasificación usando One-hot y distribución gaussiana y resultó ser mejor que la clasificación One-hot pero comparable a la clasificación haciendo uso de la distribución gaussiana.

De este primer experimento los resultados indicaron que la mejor forma de estimar la edad es haciendo uso del método de regresión con MAE.

En el segundo experimento se evaluaron las arquitecturas mostradas en las Figuras 3.8 y 3.9, donde el objetivo era comprobar si una arquitectura multitarea brindaba mejores resultados al estimar la edad que la arquitectura base propuesta en la Figura 3.8. De este experimento se obtuvo que aunque la arquitectura 3.9 (b) obtuvo mejores resultados que 3.9 (a) y 3.8, estos no fueron significativos. Sin embargo, utilizando la arquitectura de la Figura 3.9 (c), se obtuvieron mejoras con respecto a la arquitectura base.

3.3.4. Arquitectura propuesta por Vasileiadis et al (2019)

Como se mencionó al inicio, el reconocimiento de rasgos faciales por medio de una imagen es un tema muy investigado y por ende es muy común que cada año se encuentren nuevos hallazgos, Vasileiadis et al (2019) [36] propone una red convolucional MobileNet con TensorFlow Lite adecuada para dispositivos de bajo poder computacional que estima de forma simultánea características como la edad, género, raza, estado del ojo, si está sonriendo o si tiene barba, bigote o lentes. Esta metodología se combina con un clasificador de varias partes totalmente conectado que permite estimar 8 rasgos, los cuales se muestran en la Tabla 3.5

Rasgo facial suave	Clases
Género	Masculino/femenino
Grupo de edad	Infante/ niño/adulto joven/ adulto/persona mayor
Raza	Caucásico/Africano/Asiático/ Hindú/ Mezcla
Estado del ojo	Abiertos, parcialmente abiertos, cerrados
¿Lentes?	Con lentes/ sin lentes
Sonrisa	Sonriendo/ No sonriendo
Barba	Con barba/Sin barba
Bigote	Con bigote/sin bigote

Tabla 3.5: Rasgos suaves estimados por [36]

La arquitectura mostrada en la Figura 3.10 está conformada por dos etapas: extracción de características y clasificación. La etapa de extracción de características, en color azul claro, es una versión inconclusa de una MobileNet que solo usa las capas convolucionales completamente conectadas. Se compone de una capa densa inicial de 3x3x32, seguida de 12 capas convolucionales separables en profundidad (En [Convolución separable en profundidad](#) se explica este tipo de convolución) que se encargan de reducir gradualmente la dimensión espacial del mapa de características. Al final de las capas convolucionales se pasa a través de una capa Average Pooling para reducir el espacio dimensional del mapa de características a 1x1x1024.

La etapa de clasificación recibe como entrada el vector generado anteriormente (1x1x1024) y lo pasa a 8 estructuras de clasificación. Cada estructura está compuesta de dos capas completamente conectadas: una capa de 256 neuronas que reduce la dimensionalidad del vector, seguida de una segunda capa que produce la probabilidad final. Durante la etapa de entrenamiento se usó la función softmax con entropía cruzada como función de pérdida. Como optimizador se hizo uso de Adam con un índice de aprendizaje de $\alpha = 0,0004$ y un momentum de $b = 0,9$.

Con el objetivo de reducir la complejidad computacional se introduce un parámetro de multiplicador de ancho a entre (0,1] a todas las capas convolucionales completamente conectadas para reducir el ancho de la red en cada capa. Los valores menores a 1 en el parámetro de multiplicador se usa para definir modelos más pequeños.

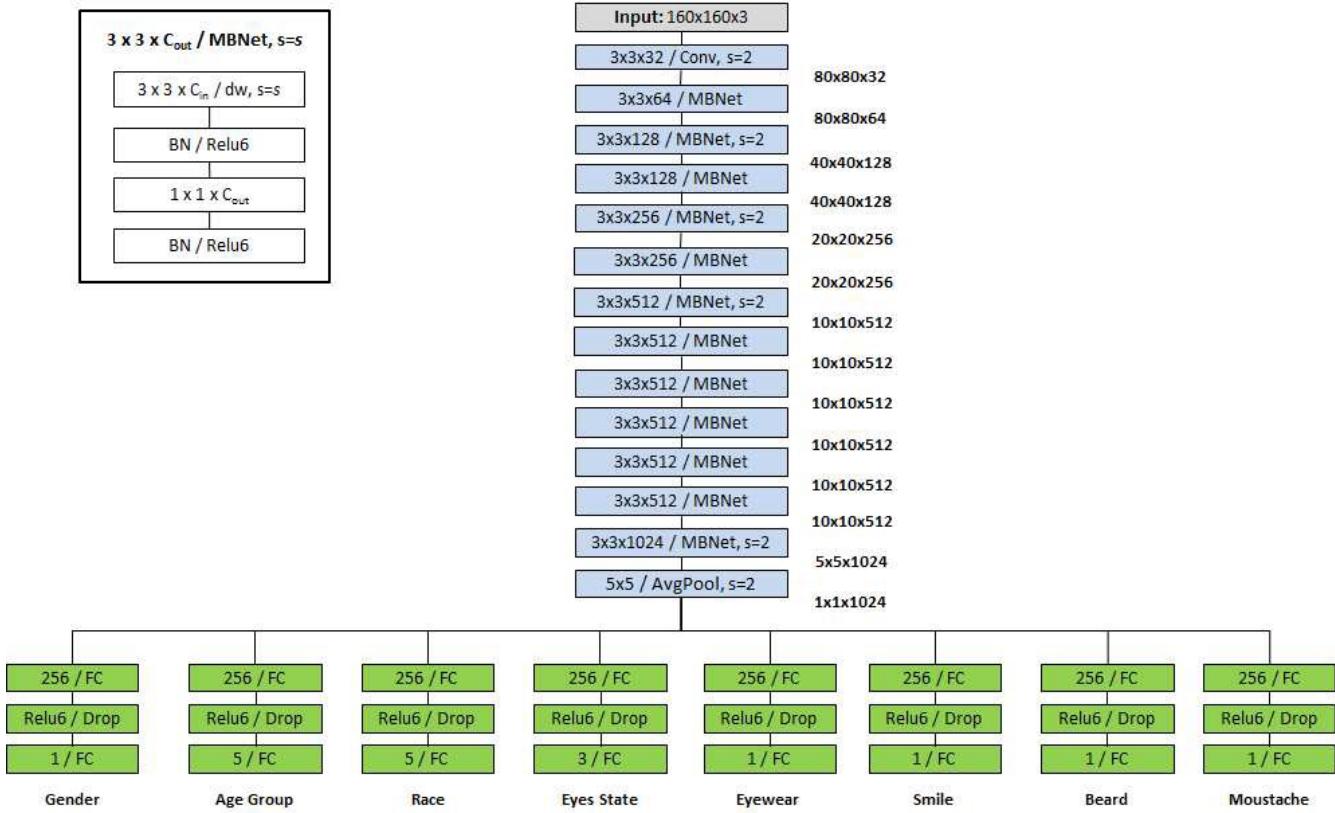


Figura 3.10: Arquitectura de CNN multi-task para dispositivos con poco poder computacional. Imagen tomada de [36]

Además se hace una cuantización post-entrenamiento para reducir el costo al sustituir las operaciones de punto flotante por enteros de 8 bits.

Para la evaluación del modelo se entrenó y evaluó con el dataset LFW Soft Biometrics. Este dataset es una extensión del LFW (Labelled Faces in the Wild) que contiene 11 atributos faciales (género, edad, raza, uso de lentes, barba, bigote, frente, boca, ojos, sonrisa y pose) de 13,233 imágenes de rostro con un tamaño de 250x250px sin restricciones de 5,749 personas.

Con esto se obtuvo que la mayoría de los estimadores tuvieron una exactitud por encima del 90%; sin embargo, el peor estimador fue precisamente el de la edad, con un 75.6% debido a que hay un gran solapamiento entre las categorías de adulto (adulto joven, adulto, persona mayor). Todo esto dejando el parámetro multiplicadora en 1.

Usando el parámetro de multiplicador con valores menores a 1 se redujo la exactitud en menos del 1% para valores de a iguales a 0.75 y 0.5, mientras que con un valor de a igual a 0.25 se mantuvo una exactitud promedio del 90%.

Por otro lado, la cuantización presentó altos valores de exactitud, reduciéndose en promedio solo el 1% con referencia a sus modelos con punto flotante. Estos resultados se muestran en la Figura 3.11

Model	Gender	Age	Race	Eyes state	Eyewear	Smile	Beard	Moustache	MEAN
a=1.00	0.967	0.756	0.897	0.941	0.980	0.891	0.979	0.961	0.921
a=0.75	0.957	0.752	0.893	0.915	0.979	0.871	0.975	0.962	0.913
a=0.50	0.961	0.740	0.886	0.936	0.974	0.884	0.976	0.967	0.915
a=0.25	0.958	0.728	0.858	0.898	0.963	0.876	0.963	0.956	0.900
a=1.00 quan	0.968	0.745	0.892	0.942	0.979	0.891	0.936	0.928	0.910
a=0.75 quan	0.958	0.751	0.890	0.918	0.978	0.874	0.935	0.935	0.905
a=0.50 quan	0.962	0.736	0.886	0.937	0.973	0.884	0.935	0.931	0.906
a=0.25 quan	0.959	0.726	0.855	0.904	0.958	0.878	0.936	0.919	0.892
Hyperface	0.979	0.772	0.901	0.936	0.980	0.902	0.981	0.970	0.928
Face++	0.911	0.388	0.874	-	0.922	-	-	-	-
MS Cognitive Services	0.929	0.593	-	-	0.917	-	0.938	0.941	-

Figura 3.11: Resultados de CNN multi-task para dispositivos con poco poder computacional con distintos valores de a . Tabla tomada de [36]

3.4. Revisión de trabajos de Big Five

El principal conflicto para el empleo de la personalidad del consumidor como una herramienta de generación de recomendaciones o publicidad dirigida en el marketing es la complejidad para obtener información privada y confidencial sobre los clientes potenciales. La personalidad no es un dato accesible que se puede encontrar en documentos, sino que es una característica que requiere profesionales e investigación personalizada en el comportamiento humano [51].

La fuente de información más trivial es el perfil psicológico de cada cliente, pero no todos lo tienen; un enfoque tradicional para medir la personalidad consiste en encuestas con preguntas sobre el comportamiento y el estilo de vida de los clientes, como se mencionó en la sección de [Personalidad: Big Five](#) del capítulo de [Marco teórico](#), aunque en muchas ocasiones estos se consideran invasivos y las personas no se sienten cómodas con ellos, además de ser lentos y poco prácticos [51].

La investigación en el campo ha demostrado que los rasgos de personalidad pueden extraerse de otras fuentes de información como el habla, la escritura a mano o el uso de las redes sociales, pero esta información privada no es accesible, al menos no sin el consentimiento de los clientes. En general, la personalidad de una persona se puede encontrar de distintas maneras, pero es necesario encontrar datos públicos que contengan suficiente información para describirla [51].

Se ha descubierto que los rasgos de personalidad pueden predecirse con precisión en función de características de una imagen, como la media de saturación, la variación del valor, la temperatura, el número de caras o el nivel de color (filtros de Instagram) [51].

También existen modelos que han utilizado imágenes de perfil de Facebook, encontrando los siguientes resultados:

1. Las imágenes de perfil de personas que no están *abiertas a la experiencia* tienden a tener puntos fuertes de luz y figuras oscuras, mientras que las que sí lo están a veces aparecen con otras personas.
2. Las personas poco *amables* tienden a usar imágenes de perfil con pocos colores [51].

3.4.1. Análisis en imágenes

En [56] se utiliza un dataset de 120 videos de personas con formato de tipo entrevista de aproximadamente 20 minutos cada uno, a los cuales se les aplicó el cuestionario IPIP de 50 elementos para ser etiquetados por Big Five. Se

extraen todos los frames dentro de un periodo de 5 segundos por video, teniendo como resultado un total de 10,000 imágenes en escala de grises y con un ancho de 640 píxeles; estas imágenes pueden ser mejor visualizadas en la Figura 3.12 como los datos de entrada.

Utilizando una red neuronal convolucional en Tensorflow, cuya arquitectura también es mostrada en la Figura 3.12 que consiste en un extractor de características y en un clasificador para cada dimensión de personalidad, **se obtiene una precisión por encima del 95 %**. Se utiliza el 50 % de los datos para entrenamiento y la mitad restante para las pruebas [56].

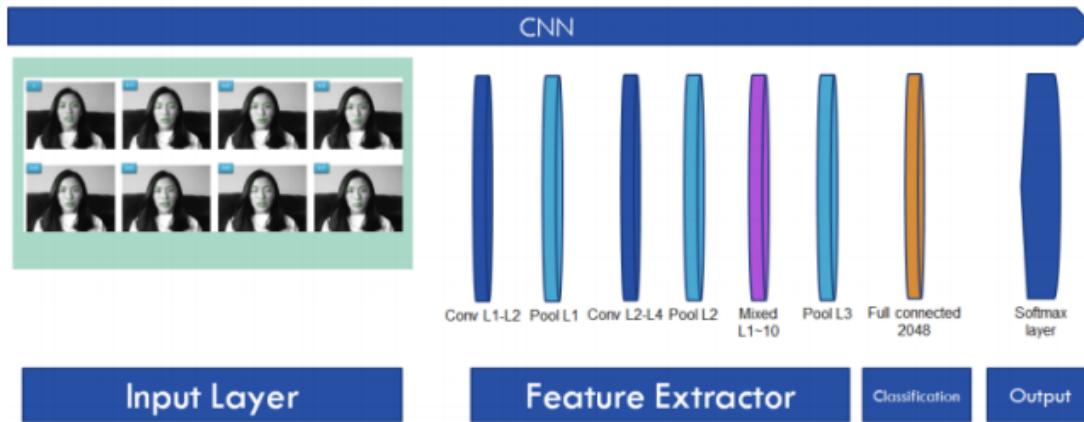


Figura 3.12: Diagrama tomado de [56] que muestra los frames extraídos de videos con formato tipo entrevista, y la arquitectura de la red neuronal convolucional utilizada para este trabajo.

Los humanos no pueden evaluar con precisión la personalidad de un persona simplemente mirando un video grabado de ellos en formato de entrevista, pues esto resulta en etiquetados de manera subjetiva, de forma involuntaria o por subconsciente [57]. La principal ventaja del estudio en [56] es que la etiqueta de Big Five en el dataset es con base al inventario IPIP, y no en impresiones subjetivas como en el dataset de Chalearn o en Amazon Mechanical Turk [55].

3.4.2. El dataset de ChaLearn

En 2016 se creó para un concurso, cuyo objetivo era identificar el Big Five en una persona a través de videos, el **dataset ChaLearn** [55], compuesto de 10,000 videos de personas hablando de frente de 15 segundos de duración obtenidos de Youtube en resolución 720p, cada uno etiquetado por Big Five utilizando Amazon Mechanical Turk. En la Figura 3.13 se presenta una muestra de las imágenes con sus respectivas etiquetas de Big Five del dataset de ChaLearn 2016, en donde se muestran resultados para cada uno de los 2 rasgos opuestos en cada una de las 5 dimensiones, como se explica en la sección Personalidad: Big Five del Marco teórico.

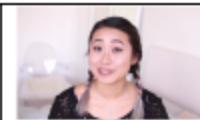
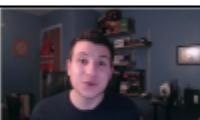
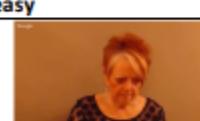
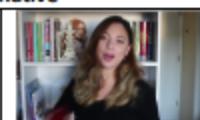
Agreeableness			
Authentic		Self-interested	
			
0.9230	0.9340	0.1098	0.0879
Conscientiousness			
Organized		Sloppy	
			
0.9708	0.9514	0.0873	0.1068
Extraversion			
Friendly		Reserved	
			
0.9158	0.9252	0.0521	0.0933
Neuroticism			
Comfortable		Uneasy	
			
0.9585	0.9791	0.1005	0.0872
Openness			
Imaginative		Practical	
			
0.9777	0.9582	0.0549	0.1113

Figura 3.13: Imagen tomada de [55] con capturas de pantalla de videos de muestra y etiquetas de personalidad del dataset ChaLearn 2016

Por medio de regresiones profundas y redes neuronales convolucionales, el ganador de ChaLearn, cuya arquitectura para la modalidad de video se presenta en la Figura 3.14, combina los resultados de **análisis de imagen** (detección de rostros en frames) y **análisis de características de audio** (dividido en N pedazos) extraídos de los videos del dataset, para obtener una **precisión media final ligeramente por encima del 91 %** [51] [55].

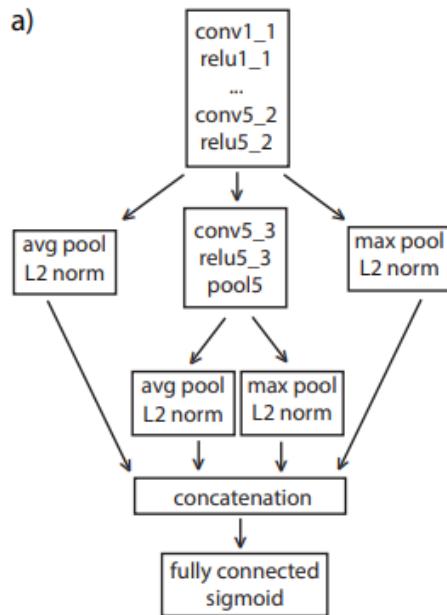


Figura 3.14: Diagrama tomado de [54] que muestra la arquitectura de la red neuronal convolucional utilizada para la modalidad de video del ganador de ChaLearn 2016 [55]

A pesar de los buenos resultados, este dataset contiene algunas deficiencias y observaciones que se enlistan a continuación:

- El **fondo en los videos** abarca un área muy grande y pueden haber más personas en él (multitud, audiencia sin hablar, rostros con baja resolución).
- Es necesario **segmentar los rostros** para que el fondo no afecte en las predicciones.
- Las personas sólo hablan en **inglés**.
- Se consideraron videos en donde **por lo menos un ojo puede ser observado**, es decir, en algunos **no está presente el rostro frontal completo, pero sí de perfil**.
- Fue etiquetado en Amazon Mechanical Turk por personas, por lo que **la etiqueta de Big Five es aparente**, y no tan segura como si se hubiera aplicado una prueba o cuestionario.
- Para reducir este variación de los datos etiquetados, se utilizó un **modelo estadístico para ajustar los valores**.
- Las personas están en un rango de **edad de entre 13 y 15 años**.
- Algunos videos solo muestran personas sentadas, **sin realizar movimiento o decir ninguna palabra** [53].

3.4.2.1. Análisis de video: comparación entre imágenes, audio y texto

En [53] y [52] se extrae audio, imágenes (utilizando OpenFace) y texto hablado de los videos del dataset de ChaLearn. Las Figuras 3.15 y 3.16 muestran y detallan las arquitecturas de estos dos trabajos respectivamente.

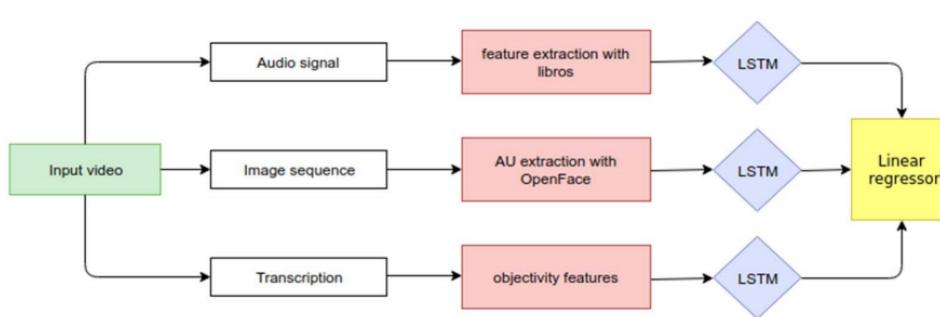


Figura 3.15: Diagrama tomado de [53] que muestra la arquitectura de este trabajo: contiene un componente de audio, de video y de procesamiento de lenguaje natural, que son procesados por tres celdas de red neuronal de memoria a corto plazo (LSTM) separadas; las características ocultas son procesadas por un regresor lineal y luego alimentadas a una capa de salida.

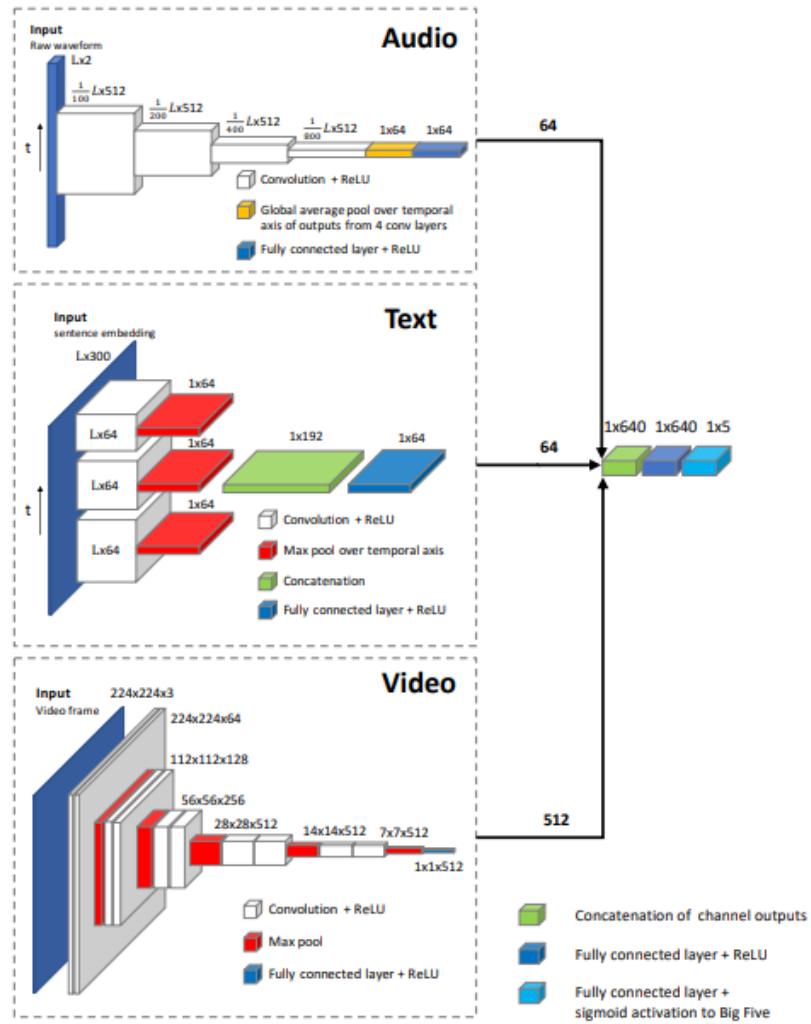


Figura 3.16: Diagrama tomado de [52] que muestra la arquitectura trimodal para la predicción de Big Five a partir de la entrada de audio, texto y video.

En ambos se tienen 3 componentes o canales separados de procesamiento y extracción de características, uno para cada modalidad tomada, y al final se combinan los resultados de cada componente para obtener una predicción de personalidad. Para el análisis visual en [52], se toma un frame aleatorio de los videos de ChaLearn, no se le aplica ningún preprocessamiento y se extraen representaciones de las imágenes utilizando el modelo de redes neuronales convolucionales VGG-Face (ver subsección [VGG16](#)).

La Tabla 3.6 presenta los resultados completos de [52]; estos demuestran que **utilizando sólo imágenes se obtiene una precisión media del 89 %**, la cual no es muy lejana de la que se obtiene en [53] **combinando análisis de audio, texto e imagen** o de los ganadores de ChaLearn 2016 [55] (**audio + imagen**).

Modelo	Precisión media	Apertura	Conciencia	Extraversión	Amabilidad	Inestabilidad
Imagen [52]	89.6 %	89.6 %	90.4 %	89.1 %	89.3 %	89.7 %
Audio + Texto + Imagen [53]	90.6 %	90.4 %	90.9 %	90.7 %	90.3 %	90.6 %
Ganador ChaLearn [55]	91.3 %	91.2 %	91.7 %	91.3 %	91.3 %	91.0 %

Tabla 3.6: Comparación entre las precisiones resultantes obtenidas utilizando video, imágenes, audio y la combinación de estos.

De forma similar, la recopilación en [57] demuestra que la precisión de trabajos en donde sólo se usan imágenes contra aquellos en donde las combinan con audio e incluso con texto (lenguaje natural) varía muy poco, en no más de 1 %, y que el modelo implementado no significa un gran impacto o aumento en ella.

En [54] se hace una comparación al utilizar frames del dataset de ChaLearn [55] completos contra sólo los rostros recortados (240x240 px) de los mismos en la misma red neuronal convolucional, cuyos resultados representados en la Tabla 3.7 indican que el rostro recortado provee más información.

Método	Precisión media	Apertura	Conciencia	Extraversión	Amabilidad	Inestabilidad
Ganador ChaLearn [55]	91.3 %	91.2 %	91.7 %	91.3 %	91.3 %	91.0 %
Imagen [54]	90.9 %	90.9 %	91.1 %	90.9 %	91.0 %	90.5 %
Rostro [54]	91.2 %	91.0 %	91.4 %	91.5 %	91.2 %	90.7 %

Tabla 3.7: Comparación entre distintos métodos de análisis (frame completo y rostro recortado) para obtener el Big Five.

Como se revisó en la sección de [Selfies](#) en el capítulo de [Marco teórico](#), estas proveen una gran cantidad de información, además de ser un recurso no invasivo y de fácil acceso. Se han utilizado *selfies* de personas para predecir y comprender su personalidad, por medio de características como la presencia de duckface, si el usuario está solo, positividad emocional, etc.

3.4.3. Nuevo dataset a partir de ChaLearn: PortraitPersonality.v2

En [51] y [94] se construyó un nuevo dataset a partir del de ChaLearn que consiste de **imágenes tipo selfie donde sólo aparece una persona y su rostro es visible**, etiquetado con el Big Five de la persona en la foto.

Para cada uno de los 8,000 videos disponibles, se toman de 3 a 4 frames, lo que resulta en un total de 30,935 nuevas imágenes a color. Las dimensiones de estas imágenes son 180x254x3 (alto, ancho, canales de color) [51] [94].

En la Figura 3.17 se muestra la redimensión de un par de frames extraídos de los videos de ChaLearn, para darles el formato tipo selfie deseado.

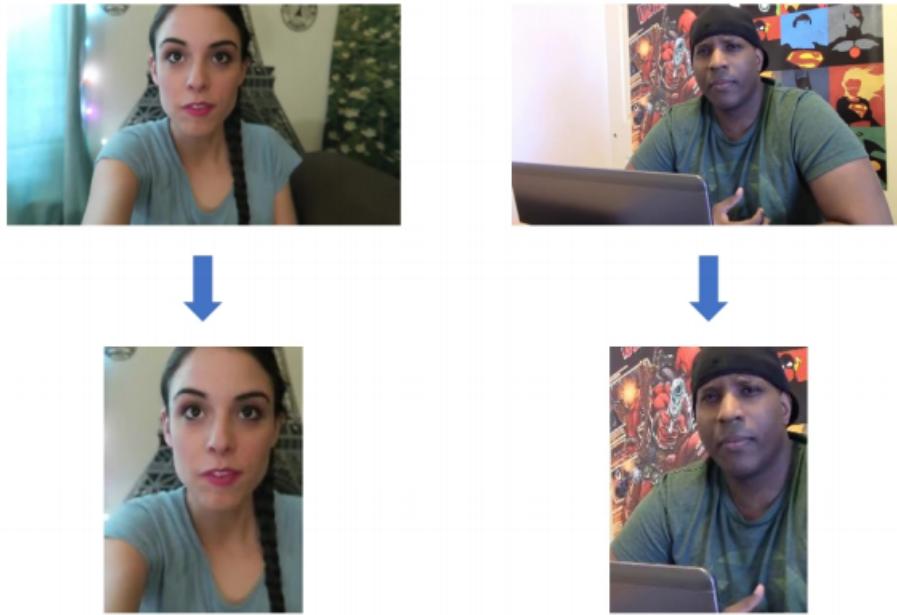


Figura 3.17: Imagen tomada de [51] que muestra el formato de una selfie por cada frame extraído.

Posteriormente, se hace la detección de rostros usando OpenCV en Python en cada imagen y luego se recortan para que tengan un formato tipo selfie, dando como resultado al **dataset PortraitPersonality.v2**. Las dimensiones de estas selfies son 228x170x3 (alto, ancho, canales de color) [51] [94].

En este trabajo se hicieron pruebas con diversos modelos pilotos (redes neuronales profundas), de regresión (redes neuronales convolucionales y autoencoders) y de clasificación (redes neuronales convolucionales y redes FaceNet) para comparar los resultados de cada uno de ellos, y llegar a la conclusión de cuál es el mejor para obtener el Big Five [51]. Estos son presentados en la Tabla 3.8, donde el mejor modelo fue el que utilizó la arquitectura FaceNet, cuyos resultados obtenidos son mostrados y comparados con el prejuicio humano en la Tabla 3.9.

Modelo	Precisión media
Regresor: ACoder	50.05 % (con tolerancia)
Regresor: CNN-2 (B)	52.06 % (con tolerancia)
Clasificador: CNN-4	65.77 %
Clasificador: FaceNet-1	65.86 %

Tabla 3.8: Comparación de los mejores resultados obtenidos de los distintos modelos desarrollados en [51]

A continuación se exponen los 2 modelos entrenados como clasificadores binarios [51] [94] con mejores resultados en este trabajo: FaceNet-1 y la red neuronal convolucional que tiene el resultado más alto, CNN-4.

3.4.4. Modelo de clasificación: CNN-4

Este modelo de clasificación es el segundo que mejor resultados tiene en el trabajo en [51] y [94], una red neuronal convolucional cuya arquitectura completa se muestra en la Figura 3.18

Las entradas son las selfies del dataset PortraitPersonality2.0 en blanco y negro (un solo canal de color) pero recortadas, normalizadas y redimensionadas a 208x208x1.

Los datos de salida de este modelo son un vector de cinco dimensiones, cada valor corresponde a un rasgo de personalidad binario en el modelo Big Five que toma 0 o 1 como valor, donde 0 indica la ausencia o negación del rasgo evaluado y 1 indica la presencia o confirmación [51].

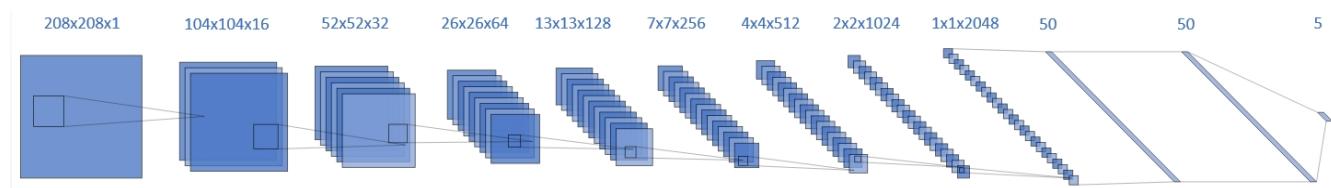


Figura 3.18: Imagen tomada de [51] que muestra la arquitectura del modelo de clasificación CNN-4. Esta consiste en: a) 8 capas convolucionales; b) un kernel de tamaño 3x3 con stride igual a 1 en cada capa convolucional; c) La función de activación ReLu en cada capa convolucional; d) Una capa Max-Pooling con un stride de 2x2 para cada capa convolucional; e) 2 capas completamente conectadas, cada una con la función de activación ReLu; f) Una capa de salida de 5 neuronas sin función de activación; g) 28,152 imágenes para entrenamiento y 2,783 para pruebas; h) Batch size de 100; i) Error cuadrático medio como función de pérdida; j) Optimizador Adam con velocidad de aprendizaje de 0.000001; k) Entrenada durante 200 épocas.

En la Tabla 3.9 se muestran las precisiones medias obtenidas para cada dimensión de personalidad de este modelo, y se comparan con el prejuicio humano y los resultados del modelo que se expondrá en [Modelo de clasificación: FaceNet-1](#) a continuación.

3.4.5. Modelo de clasificación: FaceNet-1

Este modelo de clasificación es el que mejor resultados tiene en el trabajo en [51] y [94]. FaceNet es una red para la verificación, el reconocimiento y la agrupación de rostros entrenada con millones de imágenes de estos, que se basa en aprender una incrustación euclíadiana por imagen utilizando una red convolucional muy profunda (más de 20 capas profundas) [51]. A diferencia del modelo CNN-4, que es una red neuronal convolucional "pura", el conocimiento de FaceNet es transferido a este modelo, utilizando la arquitectura de FaceNet como un extractor de características y una red entrenada completamente conectada a los datos de personalidad; la forma en que FaceNet observa los rostros puede ayudar a determinar los rasgos de personalidad. La arquitectura completa de este modelo resultante se muestra en la Figura 3.19.

Las entradas son las selfies del dataset PortraitPersonality2.0 a color pero recortadas y redimensionadas a 160x160x3.

Los datos de salida de este modelo son un vector de cinco dimensiones, cada valor corresponde a un rasgo de personalidad binario en el modelo Big Five que toma 0 o 1 como valor, donde 0 indica la ausencia o negación del rasgo evaluado y 1 indica la presencia o confirmación [51].

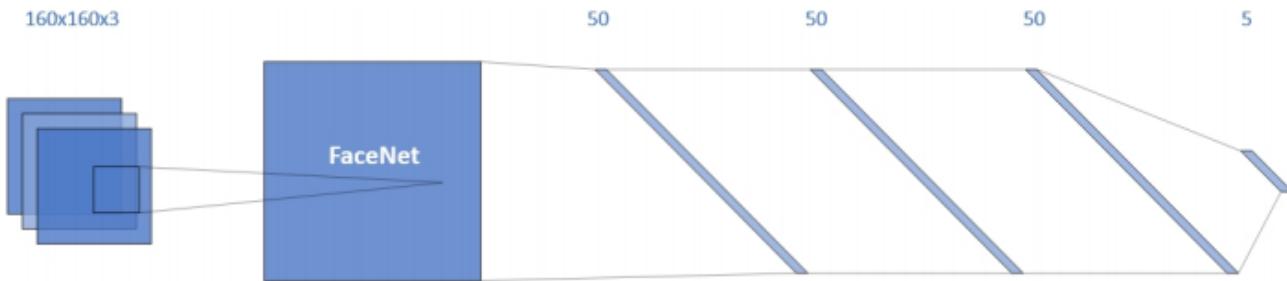


Figura 3.19: Imagen tomada de [51] que muestra la arquitectura del modelo de clasificación FaceNet-1. Esta consiste en: a) la red FaceNet; b) 3 capas completamente conectadas, cada una con la función de activación ReLu; c) Una capa de salida de 5 neuronas con función de activación sigmaidea; d) 28,152 imágenes para entrenamiento y 2,783 para pruebas; e) Batch size de 100; f) Error cuadrático medio como función de pérdida; g) Optimizador Adam con velocidad de aprendizaje de 0.001; h) Entrenada durante 100 épocas.

Modelo	Precisión media	Apertura	Conciencia	Extraversión	Amabilidad	Inestabilidad
Humano	56.66 %	58.33 %	50.00 %	33.33 %	83.33 %	58.33 %
CNN-4	65.77 %	62.01 %	67.94 %	72.15 %	61.44 %	65.28 %
FaceNet-1	65.86 %	61.48 %	69.56 %	73.23 %	60.68 %	64.35 %

Tabla 3.9: Comparación entre los resultados medidos y por factor de personalidad del mejor modelo de [51] y de una prueba con personas que de igual forma se realizó en este trabajo. Los mejores resultados para cada columna están en negritas.

Los bajos resultados en la *amabilidad* mostrados en la Tabla 3.9 se pueden justificar diciendo que esta tiende a manifestarse en situaciones sociales, a través de la cooperación y la empatía. Según [57] la *amabilidad* es el rasgo más difícil de reconocer cuando se considera la personalidad real.

Las señales verbales como palabras son más adecuadas para predecir la *amabilidad*, la *conciencia* y la *apertura a la experiencia* [52] mientras que la *extraversión* se puede predecir sin señales verbales [57].

Una vez que se han tratado la extracción de características de una persona a partir de una imagen de su rostro, es momento de hablar sobre las herramientas para presentar el resultado de la recomendación a los clientes. Como se menciona en la sección de *Realidad Aumentada* del capítulo de *Marco teórico*, esta tecnología puede ser muy atractiva para el público en general, por lo que vale la pena hacer una revisión de un par de sistemas que hacen un uso de ella similar al que se tiene planeado.

3.5. Revisión de trabajos de Realidad Aumentada

El uso de **Realidad Aumentada** para aplicaciones publicitarias y comerciales radica en reemplazar por completo la necesidad de probar cualquier cosa en las tiendas, ahorrando así una cantidad considerable de tiempo para los clientes, que probablemente se usaría para observar, decidir y seleccionar un producto (no siempre concluyendo en la venta del mismo) y así aumentan las posibilidades de venta de las tiendas [68].

El ejemplo más conocido, mostrado en la Figura 3.20, es **Magic Mirror** [95] [96], dispositivos que básicamente son una pantalla de dimensiones largas donde se puede interactuar con diversos objetos simulados, proporcionados por otro determinado dispositivo (marcadores). Según [95] y [96], las principales características de estos dispositivos pueden ofrecer son las siguientes: detección de rostros sobre un cuadro delimitador, es decir, el sistema establece coordenadas para seleccionar únicamente el área que marque la cara en una imagen; reconocimiento de expresiones faciales, definiendo el borde a lo largo de la cara, cejas, ojos, nariz y boca; modelado y superposición dinámica de objetos bidimensionales y tridimensionales en las fotografías tomadas, en diferentes ubicaciones, una vez que estas fueron analizadas y procesadas; reconocimiento de gestos y experiencia interactiva con el usuario; el Kinect es un claro ejemplo de esto último.



Figura 3.20: Imagen tomada de [68] que muestra a un usuario probándose zapatos virtuales frente a un dispositivo de tipo **Magic Mirror**.

Sin embargo, el enfoque de marketing que se le da en el trabajo [68] consiste en que el usuario puede ver su reflejo en el **Magic Mirror** con un modelo virtual de la ropa o producto que le gustaría probar. La ventaja de este sistema sobre ir a la tienda es que una vez que el usuario selecciona la prenda para la prueba, tiene la capacidad de cambiar algunos detalles, como el color, la talla e incluso las costuras. Para hacerlo, se utilizan unos marcadores esféricos reflectantes infrarrojos pintados que sirven como un sistema de seguimiento para el **Magic Mirror**, que de hecho es una pantalla que realmente procesa la información del catálogo electrónico e ingresa datos del cliente para ver si el modelo elegido está aprobado, además de detectar y reflejar los movimientos de este [68].

Por otro lado, es importante definir la forma en que la recomendación será presentada, es decir, hallar la forma de colocar los diseños y datos de la bebida obtenida en la recomendación cerca del rostro de la persona, y que de

preferencia se este constantemente actualizando la posición del mismo rostro para lograr crear un efecto de realidad aumentada en tiempo real.

Una aplicación que puede ser tomada como referencia o punto de partida es **Snapchat Lenses** [97], la cual es una conocida aplicación móvil que aplica filtros al rostro, como por ejemplo: cambiar el color de ojos, la forma de la cara, añadir accesorios, hacer que se inicien animaciones cuando se abre la boca o se levanten las cejas, así como también intercambiar una cara por la de alguien más. Otras de sus principales funcionalidades son la detección de rostros frontales por medio de la cámara del dispositivo móvil, así como la aplicación de filtros sobre una máscara tridimensional sobrepuerta en estos en tiempo real. En la Figura 3.21 se presenta una comparativa entre una fotografía con y sin filtros.



Figura 3.21: Imagen obtenida de Google Imágenes, que muestra la comparativa entre la aplicación de filtros por parte de la aplicación móvil **Snapchat Lenses** con la fotografía original tomada por la cámara un dispositivo móvil. Lo que llama la atención es la capacidad de la aplicación para detectar la ubicación del rostro de la persona en la fotografía en tiempo real, y la colocación de diseños en esta, además de estar constantemente actualizando esta incrustación de elementos visuales.

3.6. Discusión del capítulo

En la sección de [Revisión de trabajos de Publicidad Dirigida](#) se expuso la **Aplicación móvil para recomendación de bebidas basada en redes neuronales profundas** del Trabajo Terminal 191004 [10], la cual es el antecedente directo de este trabajo, pues nace de ella y la principal motivación es mejorarla.

Como consecuencia, se ha decidido que el sistema de este trabajo generará la recomendación de una bebida de la misma empresa **BubbleTown®** a partir de las estimaciones de las características de **edad, género, y personalidad en forma de Big Five**, siendo el estilo de vida reemplazado por esta última; todo esto a partir del análisis de una fotografía del rostro de una persona. También se rescata la idea de incluir la **temperatura ambiental** en la recomendación, para así tener un criterio de selección en la modalidad de la bebida.

Adicional a esta convicción, se pretenden tener una serie de dispositivos, de modo similar a los que tiene la **Intel® Audience Impression Metrics Suite** [9], los cuales son: una cámara para obtener video e imagen de aquellas personas que transiten por fuera del local de **BubbleTown®** en tiempo real, y una pantalla en donde se despliegue la recomendación obtenida, de forma similar a como lo hace **Magic Mirror**. Más adelante retomaremos este punto.

En la sección de [Revisión de trabajos de Edad y género](#) se abordan algunos trabajos presentados que muestran una muy buena exactitud en el reconocimiento del género y edad. En la mayoría de ellos se presentan arquitecturas

multitarea, que permiten aprender más de una tarea(en este caso la estimación de edad y clasificación del género) con la misma red neuronal. Este dato es relevante porque al ser un sistema que se colocará en un local, los recursos computacionales serán limitados y la opción de usar una sola red neuronal para desempeñar dos tareas permite reducir la cantidad de recursos que se requerirían.

En la sección de [Revisión de trabajos de Big Five](#) intencionalmente se hizo una búsqueda de trabajos cuyo principal enfoque fuera el análisis facial, para de esta forma lograr un conjunto de datos más homogéneo en todos los atributos que se pretenden identificar de una persona.

De aquí se puede resumir que muchos han intentado identificar la personalidad utilizando videos como la principal fuente de información; se expusieron los modelos y pruebas para tres tipos de componentes en estos: imagen, audio y transcripción de texto (lenguaje natural), cuyos resultados muestran que el análisis combinado de los tres genera una mayor precisión respecto a otras combinaciones o tomando solamente uno de ellos, sin embargo, estas precisiones no son significativamente altas respecto a las de otros análisis. Por este motivo, se decidió seleccionar únicamente el componente de imagen, pues además de que sus resultados no son muy lejanos a los mejores trabajos revisados en la literatura, se debe tomar en cuenta que se requiere una detección y recomendación de forma casi inmediata, y el análisis de muchos componentes combinados consume tiempo y poder computacional, que no vale la pena con respecto al ligero aumento en los resultados.

Adicional a esto, de todas las arquitecturas que únicamente realizan análisis de imagen para la estimación de personalidad por medio de Big Five, se opta por seleccionar el [Modelo de clasificación: CNN-4](#) del trabajo en [51] y [94] como punto principal de referencia o partida para nuestro objetivo, pues a pesar de que el modelo presentado en [Análisis en imágenes](#) del trabajo en [56] tiene la mejor precisión de todos, su dataset no está disponible para uso público ni comercial; y el [Modelo de clasificación: FaceNet-1](#) también en [51] y [94], a pesar de ser el que mejores resultados tiene de dicho trabajo, cae en el mismo problema de capacidad computacional y tiempo de convergencia, que probablemente los dispositivos de hardware seleccionados (ver sección [Especificación del hardware](#)) no serán capaces de soportar, tomando entonces el segundo mejor modelo del trabajo, cuya precisión obtenida tampoco se aleja significativamente de este, y a cambio ofrece un mejor desempeño y velocidad.

La mayoría de los trabajos expuestos en la sección de [Revisión de trabajos de Big Five](#) giran en torno a [El dataset de ChaLearn](#), y para este trabajo, se pretende utilizar la variación del mismo en formato de selfies del trabajo en [51] y [94], debido a las ventajas, el buen desempeño que muestra, y para que sea compatible con el modelo tomado del mismo trabajo.

Retomando la forma en que se desplegará la recomendación obtenida al posible cliente potencian una vez obtenida, se ha decidido hacer uso de la [Realidad Aumentada](#), con las condiciones de que los diseños incrustados serán bidimensionales, y estos serán colocados al lado del rostro de la persona, de forma similar a como lo hace [Snapchat Lenses](#). La información que incluirá al diseño será: la imagen de la bebida, su nombre y su modalidad.

Se estima que el añadido de la presentación de dicha recomendación utilizando [Realidad Aumentada](#) puede ser una gran mejora al trabajo de [trabajo de J. M. Balderas Sánchez](#) en [10], y podrá solucionar de forma contundente el problema de la retención e inversión del tiempo del cliente, para así generar más ventas en [BubbleTown®](#).

En el siguiente capítulo de [Análisis](#), se habla más formalmente de las características y requerimientos que el sistema final tendrá.

CAPÍTULO 4

Análisis

4.1. Introducción al capítulo

En este capítulo se especifican los elementos necesarios para llevar a cabo la planeación correcta del proyecto y su correspondiente esquema de análisis.

La [Metodología](#) que se pretende usar a lo largo de todo el desarrollo del proyecto es explicada por medio de su marco de trabajo, premisa, pautas y reglas de ejecución, así como herramientas de administración y distribución de las tareas entre los integrantes del equipo de trabajo.

Se incluyen las [Historias de usuario](#) por parte de la empresa cliente [BubbleTown®](#), a las cuales se les asocia una complejidad de desarrollo y la prioridad para ellos. También se enlistan las [Reglas del negocio](#) detectadas que determinan la funcionalidad en los procesos de la empresa con los cuales este sistema se va a involucrar directa o indirectamente.

De las [Historias de usuario](#) se desembocan los requerimientos del sistema, los cuales fueron divididos en [Requerimientos funcionales](#) y en [Requerimientos no funcionales](#), como se acostumbra a clasificar en trabajos de esta área; a estos también se les asignó una prioridad relacionada a su importancia para satisfacer los objetivos del sistema.

Por otro lado, se hace una [Gestión de riesgos](#) y obstáculos que pudiesen surgir en el desarrollo del presente trabajo, ya sean relacionados con recursos materiales, humanos, de tiempo e incluso limitantes tecnológicas.

Adicionalmente, se presentan a los principales [Actores del sistema](#) que tendrán una interacción constante con los procesos que este trabajo pretende implementar, mejorar o actualizar.

Finalmente, se hace la [Descripción de los casos de uso](#) que guiarán al diseño y desarrollo del prototipo del sistema en sí, que surgen a partir de las [Historias de usuario](#), las [Reglas del negocio](#), los [Requerimientos funcionales](#) y los [Requerimientos no funcionales](#); resaltando la información de entrada y salida, las [Reglas del negocio](#) que participan en ellos, y el flujo de información o las trayectorias, ya sean alternativas o la principal, que describen la forma en que el sistema operará.

4.2. Metodología

En el desarrollo de este trabajo se utilizará la metodología SCRUM que es un proceso de gestión que reduce la complejidad en el desarrollo de productos para satisfacer las necesidades de los clientes. La gerencia y los equipos

de SCRUM trabajan juntos alrededor de requisitos y tecnologías para entregar productos funcionando de manera incremental usando el empirismo [98].

Se comienza por definir los roles básicos, a saber: Product Owner (Propietario del Producto), SCRUM Master (Líder del Equipo) y Team (el Equipo) [99].

- Product Owner (PO). Es quien tiene la visión del producto final. Está en contacto continuo con los clientes, conoce la tendencia de los mercados y de la competencia. Así mismo, tiene bien en claro las prioridades y el valor que agregará al negocio el producto final
- SCRUM Master (SM). Es el protector del equipo. A diferencia del clásico rol de líder de proyectos, este rol está para servir al equipo y garantizar que se cumplan las prácticas dictadas por SCRUM.
- Team (Equipo). Es un conjunto multifuncional y autónomo de desarrolladores.

El proceso se inicia con el SCRUM Planning (SP). En la primera parte de esta reunión el PO presenta ante el SCRUM Master y el Equipo el Product Backlog (lista de requerimientos ordenados por importancia para el negocio). El PO comparte con el Equipo la visión que se pretende lograr con el producto de software. Enlista cada una de las funcionalidades que formarán el nuevo producto y permite al Equipo seleccionar los elementos con los que desean trabajar. Al estar debidamente ordenado el Product Backlog por prioridades, se garantiza que el Equipo seleccionará de la lista aquellas funcionalidades más urgentes. Entonces inicia la segunda parte de la reunión, donde el Equipo y el SM toman cada uno los elementos seleccionados previamente del Product Backlog, detallándolos lo más posible a fin de conseguir unidades individuales de trabajo, también conocido como Sprint Backlog [99].

Un Sprint, es un ciclo de trabajo, el objetivo es finalizar cada una de las tareas listadas en el Sprint Backlog durante el período de tiempo correspondiente a 2 semanas. El Sprint ha de finalizar en la fecha estipulada aún cuando no se haya logrado el objetivo [99].

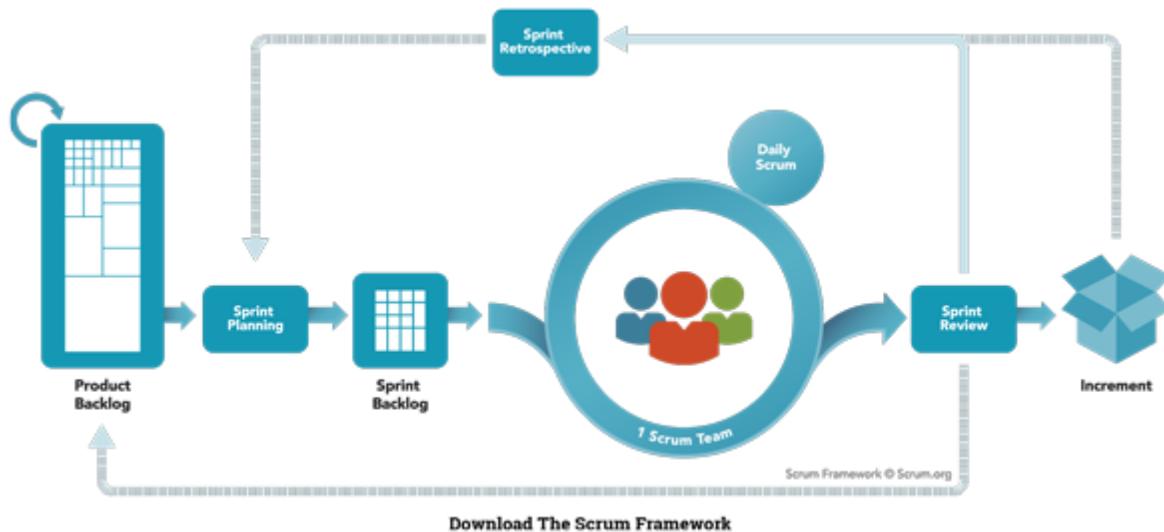


Figura 4.1: Framework de SCRUM [98].

La razón por la que se decidió hacer uso de esta metodología es porque se cuenta con BubbleTown® como cliente real y es fundamental entregar productos funcionales lo antes posible. Considerando que el producto de software solicitado es susceptible al cambio, no realizará operaciones críticas, el equipo de desarrollo es pequeño y se tiene el

apoyo del cliente durante todo el desarrollo se ha optado por SCRUM ya que se cumplen con todas las características para las que fue desarrollada esta metodología.

Por otra parte, se usará un pizarrón virtual llamado ClickUp [100] como herramienta de administración de tareas. Un ejemplo de la interfaz gráfica de esta herramienta se encuentra en la Figura 4.2.

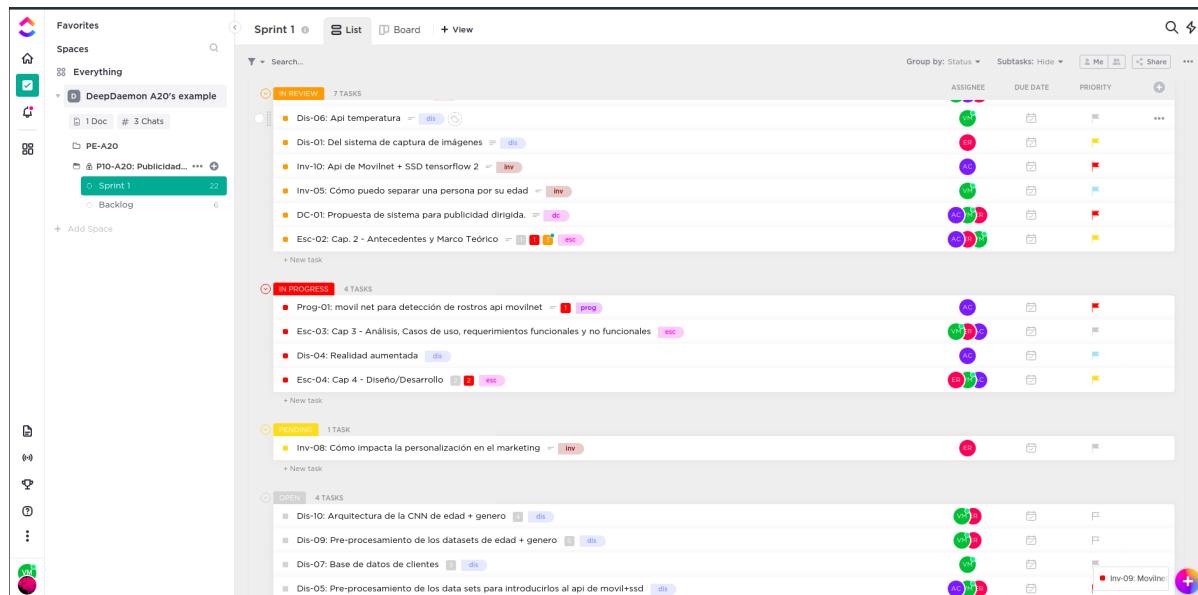


Figura 4.2: Ejemplo de interfaz gráfica de ClickUp

4.3. Historias de usuario

Las historias de usuario son una herramienta similar a los caso de uso, ya que describen la salida necesaria, características o funcionalidad del software que se elaborará y buscan obtener los requerimientos de negocio pertinentes. Cada historia es escrita por el cliente junto con su prioridad para el negocio, para después ser colocada en una tarjeta indexada. Sin embargo, es necesario enfatizar que las historias de usuario solo son un recordatorio y están basadas en la interacción oral [101] [102].

A continuación se presentarán historias de usuario que servirán como base para el análisis de requerimientos del presente trabajo.

HU-01:Sistema de publicidad

Como **Director de BubbleTown®** quiero **un sistema de recomendación que ofrezca al cliente publicidad de nuestras bebidas** para dar a conocer más la marca y sus productos.

Complejidad asociada: **Alta**
 Prioridad para el negocio: **Alta**

HU-02: Recomendación de una bebida a partir del rostro

Como Director de BubbleTown® quiero que el sistema obtenga características a partir del rostro, además de la temperatura ambiental para recomendar una bebida personalizada.

Complejidad asociada: **Alta**
Prioridad para el negocio: **Alta**

HU-03: Publicidad estática

Como Director de BubbleTown® quiero que muestre publicidad estática si no detecta personas para obtener la atención del cliente.

Complejidad asociada: **Alta**
Prioridad para el negocio: **Alta**

HU-04: Instalación del sistema en tienda

Como Director de BubbleTown® quiero que el sistema de recomendación sea instalado en un lugar visible y ocupando solo el espacio necesario para que no interfiera con las actividades dentro del establecimiento.

Complejidad asociada: **Baja**
Prioridad para el negocio: **Media**

HU-05: Tamaño de publicidad

Como Director de BubbleTown® quiero que la publicidad se muestre a través de una pantalla del tamaño adecuado para que la publicidad pueda observarse con claridad a una distancia aceptable.

Complejidad asociada: **Baja**
Prioridad para el negocio: **Media**

HU-06: Identidad de marca

Como Director de BubbleTown® quiero que la publicidad mostrada use los colores, tipografías y personajes de la marca para reforzar la identidad de mi marca

Complejidad asociada: **Baja**
Prioridad para el negocio: **Alta**

HU-07:Legislación

Como **Director de BubbleTown®** quiero que el sistema cumpla con la ley de protección de datos personales para que mis clientes se sientan seguros al interactuar con él.

Complejidad asociada: **Media**

Prioridad para el negocio: **Alta**

HU-08:Almacenamiento de datos

Como **Administrador del sistema** quiero que se almacene el rostro, los resultados del análisis y la recomendación propuesta para darle mantenimiento en el futuro y mejorar el funcionamiento.

Complejidad asociada: **Baja**

Prioridad para el negocio: **Baja**

HU-09:Documentación

Como **Administrador del sistema** quiero la documentación del software y manuales para que pueda ser fácilmente instalado, utilizado y mantenido en el futuro.

Complejidad asociada: **Media**

Prioridad para el negocio: **Media**

HU-10:Sistema de recomendación

Como **Cliente de BubbleTown®** quiero un sistema que proponga bebidas de forma divertida y poco convencional para tener nuevas posibles bebidas que sean de mi agrado.

Complejidad asociada: **Alta**

Prioridad para el negocio: **Alta**

4.4. Reglas del negocio

Las reglas de negocio son herramientas que ayudan a los desarrolladores para comprender las políticas, normas, operaciones, definiciones y restricciones presentes dentro de la empresa. Con ellas se tiene una guía de lo que puedo o no puede hacer el sistema, por lo que a continuación se presentan las reglas del negocio identificadas que el sistema del presente trabajo debe cumplir.

4.4.1. RN1 Nivel de comunicación.

Descripción: La publicidad debe contar con un nivel de comunicación informal sin hacer uso de palabras altisonantes.

Motivación: Hacer sentir al cliente cercano a la marca.

Ejemplo Positivo: El cliente se identifica con la marca y decide comprar.

Ejemplo Negativo: El cliente se siente ofendido y se retira, quedando con una mala imagen de la empresa.

4.4.2. RN2 Aviso de privacidad

Descripción: Se debe mostrar un mensaje al cliente indicándole los datos que se están almacenando junto con el motivo ya sea a través de la misma pantalla o haciendo uso de un cartel físico. No se debe almacenar en ningún momento datos que permitan identificar de forma única a las personas a excepción del rostro. Este debe ser tratado por métodos criptográficos para garantizar la seguridad de su almacenamiento.

Motivación: El cliente tiene derecho a saber qué datos se están recolectando y el motivo. Por otro lado la empresa tiene la obligación legal de informarlo.

Ejemplo Positivo: El cliente comprende qué datos se están almacenando y entiende que su privacidad no está en riesgo.

Ejemplo Negativo: El cliente desconoce qué información del él se está almacenando, por lo que siente que la privacidad de sus datos está siendo violada.

4.4.3. RN3 Tipografía y colores asociados a la marca.

Descripción: Se deben utilizar los colores en hexadecimal #F9F3B8, #B5D590, #91C361, #5D918D, #B8CCE1 y #DEB5D5 en la publicidad mostrada y las tipografías autorizadas para la publicidad deben ser "Century Gothic" o "Segoe Print".

Motivación: La empresa usa estos elementos como parte de su identidad de marca. Cambiarlos provocaría un efecto negativo en el reforzamiento de la identidad de marca sobre el cliente.

Ejemplo Positivo: La publicidad mostrada combina perfectamente con la demás publicidad creada por agentes externos, por lo que la imagen de la marca no se pierde.

Ejemplo Negativo: La publicidad mostrada parece extraña a la marca ya que no combina con la publicidad ya existente.

4.4.4. RN4 Personal con acceso a datos.

Descripción: Solo el personal registrado en la base de datos como administrador debe tener acceso a la información recolectada y almacenada en la base de datos del sistema.

Motivación: Es responsabilidad de la empresa proteger los datos que se están recolectando, por lo que debe haber algún método de control para acceder a estos datos.

Ejemplo Positivo: Un agente no autorizado intenta obtener los datos pero el sistema no lo deja ingresar porque no se ha autenticado.

Ejemplo Negativo: Un agente externo logra acceder a los datos almacenados en la base de datos sin necesitar autenticarse.

4.4.5. RN5 Uso del ancho de banda.

Descripción: El sistema solo debe hacer operaciones de carga o descarga de información a la base de datos fuera de la jornada de trabajo en el establecimiento.

Motivación: Los recursos para acceder a internet son limitados y otros sistemas ya hacen uso del ancho de banda disponible. Saturar el ancho de banda limitaría la capacidad para prestar el servicio a los clientes.

Ejemplo Positivo: El sistema usa el ancho de banda cuando ningún cliente está siendo atendido, por lo que no afecta a los demás sistemas destinados a la atención del cliente.

Ejemplo Negativo: El sistema hace uso del ancho de banda y alienta los demás sistemas, imposibilitándolos de cumplir con sus tareas para atender a los clientes de forma adecuada.

4.5. Requerimientos funcionales

Dentro de los sistemas computacionales se entiende por Requerimientos Funcionales a las declaraciones sobre las funciones o los servicios que el sistema debe proporcionar a los usuarios, la forma de reaccionar a cada una de las entradas y el cómo debe comportarse tras determinados eventos [103]. En la Tabla 4.1 se describen los requerimientos funcionales del sistema de publicidad personalizada.

Requerimientos Funcionales				
Id	Nombre	Descripción	Prioridad	Origen
RF01	Detección de rostros	El sistema debe identificar los rostros humanos en todo momento, a partir de las imágenes proporcionadas por la cámara, respetando una determinada distancia entre la persona y este.	Alta	HU-02
RF02	Preprocesamiento de rostros	Cada rostro detectado debe ser recortado de la imagen original para su procesamiento.	Alta	HU-02 RF01
RF03	Estimación de edad y género	A través del rostro humano se debe estimar la edad y género de la persona en cuestión.	Alta	HU-02
RF04	Estimación de personalidad	A través del rostro humano se debe estimar la personalidad por medio del modelo Big5 de la persona.	Alta	HU-02
RF05	Obtención de la temperatura ambiental	El sistema debe obtener la temperatura ambiental al momento de generar una recomendación.	Baja	HU-02
RF06	Generación de recomendación	Con los datos obtenidos a partir del rostro humano, y además integrando la temperatura ambiental, el sistema generará una recomendación distinta de una bebida para cada cliente potencial, además de proporcionar su imagen, nombre y modalidad.	Alta	HU-01 RF03 RF04 RF05
RF07	Generación de realidad aumentada	El sistema debe colocar al lado de cada rostro la recomendación generada por medio de un globo de pensamiento y este se debe mover junto con la persona a la que se le generó dicha recomendación. Todos estos serán elementos en 2D.	Media	HU-01 HU-10
RF08	Almacenamiento de rostros	El rostro debe ser almacenado en un archivo cifrado.	Media	HU-07
RF09	Almacenamiento de recomendación	El sistema debe almacenar el archivo de la imagen del rostro recortado y sus estimaciones obtenidas de edad, género, personalidad, la temperatura ambiental y las coordenadas del rostro. También se debe almacenar la bebida y la modalidad, es decir, la recomendación obtenida.	Media	HU-08
RF10	Respaldo de recomendaciones	El sistema debe permitir recuperar el historial de todas las recomendaciones realizadas. Estas a su vez deben poder ser almacenadas junto con los rostros de las personas a las que se les realizó la recomendación.	Baja	HU-08
RF11	Autenticación de administrador	El sistema debe comprobar que el usuario existe y que la contraseña introducida coincide con la almacenada al intentar acceder a este.	Alta	RN4
RF12	Nueva bebida	Debe ser posible, agregar la información de una nueva bebida que haya sido puesta a la venta por parte de la empresa.	Baja	HU-08

Requerimientos Funcionales				
Id	Nombre	Descripción	Prioridad	Origen
RF13	Actualización de bebida	Un administrador debe tener la facultad de actualizar la información de una bebida específica a través del nombre de la misma. Los datos modificables son la modalidad, ruta de la imagen o tipo de bebida.	Media	HU-02
RF14	Eliminación de bebida	Debe ser posible que un administrador pueda eliminar una bebida del catálogo disponible.	Media	HU-08
RF15	Comprobación de bebida	Cuando se genere una recomendación, únicamente podrá seleccionar una bebida que esté registrada en el catálogo.	Media	HU-02
RF16	Descifrado de rostros	El sistema debe descifrar las imágenes de los rostros de forma automática, sólo si se ha autenticado un administrador.	Media	HU-07
RF17	Despliegue de publicidad estática	El sistema debe mostrar publicidad estática cuando no se detecten rostros humanos.	Media	HU-03 HU-05 HU-06
RF18	Nueva publicidad estática	El sistema debe permitir agregar nueva publicidad estática únicamente a un administrador autenticado correctamente.	Media	HU-03 HU-05 HU-06
RF19	Eliminación de publicidad estática	El sistema debe permitir eliminar publicidad ya existente únicamente a un administrador autenticado correctamente.	Media	RF17
RF20	Uso de redes neuronales	La detección de rostros y las estimaciones de edad, género y personalidad de una persona a partir de este se harán por medio de redes neuronales convolucionales.	Alta	RF01 RF03 RF04

Tabla 4.1: Requerimientos funcionales

4.6. Requerimientos no funcionales

Por otro lado, los Requerimientos No Funcionales no hacen referencia a funciones del sistema, sino a las propiedades emergentes que este debe tener, como es el caso de la factibilidad, usabilidad, tiempos de respuesta, capacidad de almacenamiento, seguridad, disponibilidad, entre otros. Si bien no son funciones, usualmente son más críticos que los requerimientos funcionales debido a que el incumplimiento de estos puede resultar en un sistema inutilizable [103]. En la Tabla 4.2 se muestran los requerimientos no funcionales del sistema a desarrollar por el presente trabajo.

Requerimientos No Funcionales				
Id	Nombre	Descripción	Prioridad	Origen
RNF01	Tiempo de respuesta para publicidad	El tiempo que el sistema puede tardar como máximo para generar la recomendación desde que reconoce un rostro debe estar en un rango de entre aproximadamente 2 a 4 segundos.	Media	RF06 RF07

		Requerimientos No Funcionales		
Id	Nombre	Descripción	Prioridad	Origen
RNF02	Lenguajes de programación	El lenguaje de programación utilizado deberá ser Python 3.7 o superior junto con TensorFlow para la implementación de las redes neuronales. También se utilizará la biblioteca OpenCV en su versión 4.0 o superior.	Alta	RF01 RF02 RF03 RF04 RF06 RF07
RNF03	Cifrado de la información	El algoritmo de cifrado para el archivo del rostro a almacenar debe ser AES. Para el caso de la contraseña, solo debe ser almacenado su hash criptográfico usando SHA256.	Media	RF08 RF09 RF11 RF16
RNF04	Sistema operativo	El sistema debe funcionar sobre Linux, en la distribución de Ubuntu.	Baja	No Aplica.
RNF05	Distancia mínima de visibilidad de los transeúntes	La publicidad estática se debe poder ver a 2 metros de la pantalla por la mayoría de los transeúntes.	Media	HU-04 HU-05 RF01 RF07 RF17
RNF06	Tamaño de letra	El tamaño de letra debe ser legible a 1 metro del establecimiento por la mayoría de las personas que transiten cerca del local.	Baja	RF07 RF17
RNF07	Cámaras	El sistema debe hacer uso de cámaras para obtener las imágenes que se usarán en la red de detección de rostros.	Alta	HU-04 RF01
RNF08	Pantalla	La publicidad dirigida será mostrada auxiliándose de una pantalla de 50 pulgadas.	Media	HU-04 RF07 RF17
RNF09	Instalación de hardware	El hardware debe ser instalado en la entrada del establecimiento con vista hacia el exterior del local para poder ser usado sin necesidad de que los clientes potenciales ingresen al local.	Media	HU-04
RNF10	Base de datos y gestor	Para el almacenamiento de las recomendaciones obtenidas, el catálogo de bebidas disponibles y el manejo de las cuentas de administrador, se deberá utilizar una base de datos NoSQL, y se utilizará el gestor MongoDB.	Media	RF09 RF10 RF11 RF12 RF13 RF14 RF15

Tabla 4.2: Requerimientos no funcionales

4.7. Actores del sistema

Para el presente trabajo se han identificado dos actores principales:

- **Cliente potencial:** Se entiende por cliente potencial a cualquier persona que dirija su atención a la pantalla o cámara que se instalará frente al local.

- **Administrador:** Toda persona autorizada por el director de BubbleTown® a través de las credenciales de acceso para realizar acciones como agregar o eliminar nueva publicidad estática o generar una copia de la base de datos.

4.8. Diagrama de casos de uso

Los diagramas de caso de uso permiten visualizar, especificar y documentar el comportamiento de algún elemento del sistema. Su uso facilita abordar y comprender el sistema al presentar una vista externa de cómo puede utilizarse[104]. En la Figura 4.3 se presenta el diagrama de casos de uso de mayor importancia identificados en el sistema del presente trabajo.

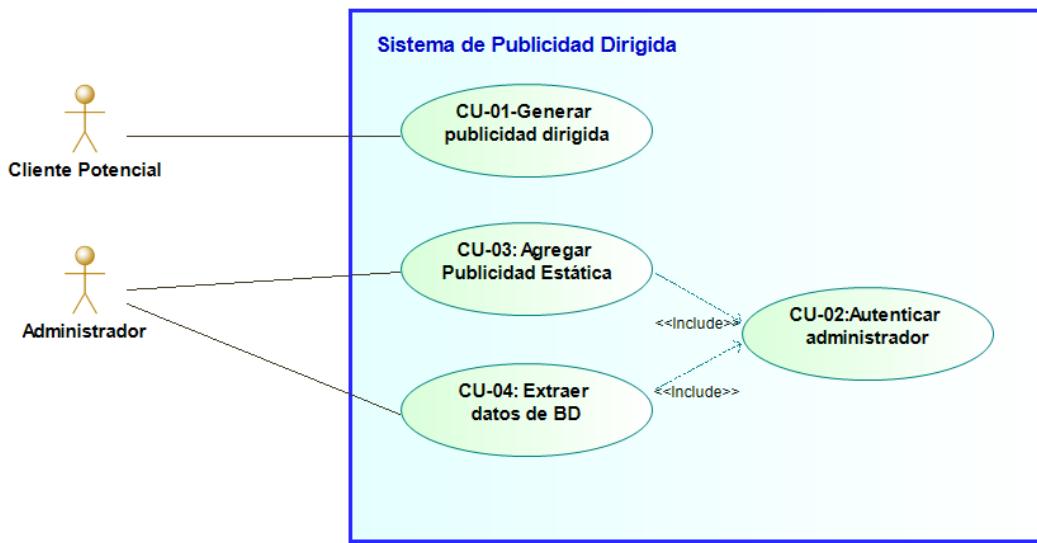


Figura 4.3: Diagrama de los casos de uso del sistema de publicidad dirigida.

4.9. Descripción de los casos de uso

En esta sección se exponen a detalle los casos de uso identificados en [Diagrama de casos de uso](#) para este sistema. La simbología de sus trayectorias principales y alternativas es la siguiente:

- corresponde al sistema en general. Puede ser cualquier proceso, método, función, algoritmo, modelo, hardware o software que lo integre.
- corresponde al actor que utiliza o participa en el cada caso de uso.
- corresponde a una interfaz de usuario.



4.9.1. CU-01 Generar publicidad dirigida

4.9.1.1. Descripción completa

El cliente potencial podrá hacer uso del sistema colocándose frente a la cámara, para posteriormente recibir una recomendación de bebida personalizada a través de la pantalla.

4.9.1.2. Atributos importantes

Caso de Uso:	CU-01 Generar publicidad dirigida
Versión:	1.1
Autor:	Morales Flores Víctor Leonel
Supervisa:	Lara Cázares Jaime Arturo
Actor:	Cliente Potencial.
Propósito:	Hacer un análisis del rostro del cliente para proponer un producto de su agrado en función de sus características y de la temperatura ambiental.
Entradas:	Frame captado por la cámara
Origen:	RF01, RF02, RF03, RF04, RF05, RF06, RF07, RF08, RF09, RF15, RF17, RF20, RN1 , RN2 , RN3
Salidas:	Frame de la cámara con una recomendación de bebida para cada rostro dentro del frame, por medio de realidad aumentada.
Destino:	No aplica
Precondiciones:	<ul style="list-style-type: none"> • Una persona debe pararse frente a la pantalla con la cámara. • La persona debe permanecer por lo menos 4 segundos enfrente de la pantalla para obtener su recomendación.
Postcondiciones:	<ul style="list-style-type: none"> • El cliente potencial contará con una recomendación personalizada para comprar en BubbleTown®. • El rostro detectado, edad, género, personalidad, temperatura ambiental y la recomendación obtenida serán almacenados en la base de datos.
Errores:	<ol style="list-style-type: none"> 1. El sistema detecta erróneamente un rostro. 2. Se genera un error entre la comunicación de la cámara con el sistema. 3. La conexión con la base de datos es perdida.
Estado:	Revisión.

Caso de Uso:	CU-01 Generar publicidad dirigida
Observaciones:	—

4.9.1.3. Trayectorias del Caso de Uso

Trayectoria principal:

- 1 ○ Muestra publicidad estática en la pantalla, que se encuentra fuera del establecimiento de BubbleTown®.
 - 2 ⚙ Pasa frente a la pantalla con publicidad estática.
 - 3 ⚙ Se detiene a observar la pantalla con publicidad estática.
 - 4 ○ Detecta uno o varios rostros de la imagen obtenida por medio de la cámara a través de una red neuronal *Single Shot Detection + MobileNet*.
 - 5 ○ Recorta los rostros de la imagen.
 - 6 ○ Procesa los rostros recortados.
 - 7 ○ Propaga los rostros procesados a través de las redes neuronales de edad y género para obtener un resultado estimado de cada uno.
 - 8 ○ Propaga los rostros procesados a través de la red neuronal de personalidad para obtener un resultado estimado de cada uno.
 - 9 ○ Obtiene la temperatura ambiental.
 - 10 ○ Ingresa los resultados obtenidos de edad, género, personalidad y la temperatura ambiental al recomendador.
 - 11 ○ Obtiene una recomendación de bebida para cada uno de los rostros, a partir de los datos ingresados en el paso 10.
 - 12 ○ Utiliza el resultado del recomendador para obtener la imagen específica del producto de BubbleTown que se recomendó desde un catálogo.
 - 13 ○ Guarda el o los rostros detectados cifrados, sus características estimadas de edad, género y personalidad, la medida de la temperatura ambiental y la o las recomendaciones obtenidas en la base de datos. [Trayectoria A]
 - 14 ○ Agrega un globo de pensamiento al lado de los rostros con el nombre del producto recomendado a la imagen de salida a la pantalla. [Trayectoria A]
 - 15 ○ Agrega iconos de BubbleTown y la imagen del producto específico, obtenido de la recomendación, a la imagen de salida a la pantalla. [Trayectoria A]
 - 16 ○ Transmite la imagen con la recomendación a la pantalla. [Trayectoria A]
 - 17 ○ Actualiza la ubicación del rostro de la persona constantemente y de forma periódica para mostrar los elementos referentes a la realidad aumentada. [Trayectoria A]
- - - - *Fin del caso de uso.*

Trayectoria alternativa A:

Condición: ⚙ Ya no se encuentra frente a la pantalla y sí se completa la generación de una o varias recomendaciones.

- A1** ○ No muestra la publicidad personalizada generada.
 - A2** ○ Continúa en el paso 1 de la trayectoria principal del CU-01.
- - - - *Fin de la trayectoria.*



4.9.2. CU-02 Autenticar administrador

4.9.2.1. Descripción completa

El administrador requiere acceder al sistema para realizar operaciones especiales (modificar catálogo de bebidas o publicidad estática del local).

4.9.2.2. Atributos importantes

Caso de Uso:	CU-02 Autenticar administrador
Versión:	1.1
Autor:	Morales Flores Víctor Leonel
Supervisa:	Ramos Díaz Enrique
Actor:	Administrador.
Propósito:	Proteger información del sistema de personas que no sean administradores.
Entradas:	Usuario y contraseña del administrador.
Origen:	RF11, RN4
Salidas:	Sesión de administrador iniciada.
Destino:	CU-03, CU-04, CU-05, CU-06, CU-07, CU-08
Precondiciones:	<ul style="list-style-type: none"> • No debe existir una sesión de administrador activa. • El sistema no debe permitir hacer operaciones propias del administrador a usuarios que no tengan este rol o estos permisos.
Postcondiciones:	<ul style="list-style-type: none"> • Se iniciará una sesión de administrador para realizar las operaciones deseadas.
Errores:	<ol style="list-style-type: none"> 1. Los datos de entrada poseen caracteres inválidos para la base de datos. 2. El administrador ingresa datos erróneos. 3. No se ingresan todos los datos de entrada (se dejan campos vacíos). 4. La conexión con la base de datos es perdida.
Estado:	Revisión.
Observaciones:	—

4.9.2.3. Trayectorias del Caso de Uso

Trayectoria principal:

- 1  Muestra la interfaz de usuario  IU1 Login de acceso.
 - 2  Ingresa su usuario.
 - 3  Ingresa su contraseña.
 - 4  Presiona el botón **Acceder** de la interfaz de usuario  IU1 Login de acceso.
 - 5  Busca en la base de datos los datos asociados al usuario ingresado. [Trayectoria A]
 - 6  Calcula el hash criptográfico de la contraseña ingresada.
 - 7  Verifica el hash calculado en el paso 6 con el hash de la contraseña almacenado en la base de datos. [Trayectoria A]
 - 8  Muestra la interfaz de usuario  IU2 Pantalla principal.
- - - - *Fin del caso de uso.*

Trayectoria alternativa A:

Condición: El usuario y/o contraseña ingresados son incorrectos:  no es autenticado satisfactoriamente como administrador.

- A1**  Muestra un mensaje indicando que alguno de los datos ingresados es incorrecto.
- A2**  Continúa en el paso 1 de la trayectoria principal del CU-02.
- - - - *Fin de la trayectoria.*



4.9.3. CU-03 Agregar publicidad estática

4.9.3.1. Descripción completa

Permite al administrador del sistema agregar nueva publicidad estática de forma sencilla para ser proyectada en la pantalla.

4.9.3.2. Atributos importantes

Caso de Uso:	CU-03 Agregar publicidad estática
Versión:	1.1
Autor:	Lara Cázares Jaime Arturo
Supervisa:	Ramos Díaz Enrique
Actor:	Administrador.
Propósito:	Disponer de publicidad tradicional en el sistema cuando no estén usuarios directamente frente a la cámara.
Entradas:	Ruta donde se ubica la nueva imagen que desea proyectarse.
Origen:	RF18, RN1 , RN3
Salidas:	Consulta a la base de datos para inserción de nuevo "cartel" de publicidad.
Destino:	No aplica
Precondiciones:	<ul style="list-style-type: none"> ● La nueva imagen con publicidad debe cumplir con las características necesarias para ser mostrada como salida en la pantalla del sistema. ● Un nuevo diseño se quiere agregar al sistema para ser mostrado cuando este no esté generando publicidad dirigida. ● El usuario administrador debe haber sido autenticado correctamente.
Postcondiciones:	<ul style="list-style-type: none"> ● La imagen de la nueva publicidad se podrá visualizar en la pantalla.
Errores:	<ol style="list-style-type: none"> 1. El archivo de publicidad estática no puede ser encontrado. 2. El archivo de publicidad estática esta corrupto. 3. La memoria es insuficiente. 4. No se puede guardar la publicidad en la base de datos. 5. Ocurre un problema de conexión con la base de datos.
Estado:	Revisión.

Caso de Uso:	CU-03 Agregar publicidad estática
Observaciones:	—

4.9.3.3. Trayectorias del Caso de Uso

Trayectoria principal:

- 1  Muestra la interfaz de usuario  IU2 Pantalla principal.
 - 2  Presiona el botón **Publicidad Estática** de la interfaz de usuario  IU2 Pantalla principal.
 - 3  Muestra la interfaz de usuario  IU3 Pantalla para gestión de publicidad.
 - 4  Presiona el botón **Agregar** de la interfaz de usuario  IU3 Pantalla para gestión de publicidad.
 - 5  Muestra la interfaz de usuario  IU4 Pantalla para agregar publicidad
 - 6  Agrega un nombre de referencia para ubicar la publicidad posteriormente.
 - 7  Presiona el botón **Cargar imagen** y selecciona la ruta de la ubicación de la imagen que se quiere agregar.[Trayectoria A]
 - 8  Muestra una vista previa de la imagen seleccionada.
 - 9  Presiona el botón **Guardar**. [Trayectoria B]
 - 10  Indica que la nueva imagen fue cargada exitosamente.
- - - - *Fin del caso de uso.*

Trayectoria alternativa A:

Condición: La imagen que se intenta cargar no cumple con las características necesarias para ser mostrada adecuadamente en la pantalla publicitaria.

- A1**  Indica al  que la imagen no cumple con las características y la imagen no es cargada.
 - A2**  Continúa en el paso 3 de la trayectoria principal del CU-03.
- - - - *Fin de la trayectoria.*

Trayectoria alternativa B:

Condición:  Presiona el botón **Regresar** de la interfaz  IU3 Pantalla para gestión de publicidad.

- B1**  No se carga la imagen.
 - B2**  Continúa en el paso 3 de la trayectoria principal del CU-03.
- - - - *Fin de la trayectoria.*



4.9.4. CU-04 Extraer datos de la base de datos

4.9.4.1. Descripción completa

El administrador requiere de una copia de las recomendaciones realizadas, junto con los datos usados para generarlas (imagen del rostro de la persona junto con edad, género, personalidad-BigFive estimadas y temperatura ambiental al momento de la recomendación) para darle mantenimiento al sistema o respaldar los datos.

4.9.4.2. Atributos importantes

Caso de Uso:	CU-04 Extraer datos de la base de datos
Versión:	1.1
Autor:	Ramos Diaz Enrique
Supervisa:	Morales Flores Víctor Leonel
Actor:	Administrador.
Propósito:	Permitir realizar un respaldo al administrador de los datos almacenados.
Entradas:	Fecha a partir de la que se realizará el respaldo de las recomendaciones.
Origen:	RF10, RF16, RN5
Salidas:	Imágenes descifradas y archivo con la información de las recomendaciones en el plazo de tiempo solicitado.
Destino:	No aplica
Precondiciones:	<ul style="list-style-type: none"> • Se requiere realizar de una copia de las recomendaciones almacenados en la base de datos. • El usuario deber haber sido autenticado correctamente.
Postcondiciones:	<ul style="list-style-type: none"> • Se contará con una copia de las recomendaciones de la base de datos.

Caso de Uso:	CU-04 Extraer datos de la base de datos
Errores:	<ol style="list-style-type: none"> 1. Ocurre un problema de conexión con la base de datos. 2. Ocurre un error en el descifrado de los rostros almacenados. 3. La ruta seleccionada para guardar la copia de seguridad no posee permisos de escritura. 4. La copia de seguridad es interrumpida por un problema desconocido del sistema. 5. La copia de seguridad resulta corrupta o incompleta. 6. La fecha ingresada es anterior al funcionamiento del sistema. 7. La fecha ingresada es posterior a la fecha en que se solicita la copia de seguridad.
Estado:	Revisión.
Observaciones:	—

4.9.4.3. Trayectorias del Caso de Uso

Trayectoria principal:

- 1 ⓘ Muestra la interfaz de usuario IU2 Pantalla principal.
 - 2 ⚙ Presiona el botón **Respaldo** de la interfaz de usuario IU2 Pantalla principal.
 - 3 ⓘ Muestra la interfaz de usuario IU6 Pantalla para hacer respaldo.
 - 4 ⚙ Indica la ruta de la carpeta donde se almacenará el respaldo de los datos a través del botón **Seleccionar ruta**.
 - 5 ⚙ Indica la fecha a partir de la que quiere generar el respaldo por medio del selector **A partir de**. [Trayectoria A]
 - 6 ⚙ Ingresa sus credenciales (usuario y contraseña) para verificar que es un administrador quien está intentando realizar el respaldo. [Trayectoria B]
 - 7 ⚙ Presiona el botón **Iniciar descarga**. [Trayectoria C] [Trayectoria D]
 - 8 ⓘ Descifra los rostros almacenados.
 - 9 ⓘ Almacena una copia de seguridad sin cifrado en la ruta indicada.
- - - - Fin del caso de uso.

Trayectoria alternativa A:

Condición: ⚙ Desea hacer un respaldo de toda la información de la base de datos.

- A1** ⚙ Marca la casilla **Descargar todo**.
 - A2** ⓘ Continúa en el paso 5 de la trayectoria principal del CU-04.
- - - - Fin de la trayectoria.

Trayectoria alternativa B:

Condición: ⚙ Ingresó credenciales erróneas en tres ocasiones

B1 ○ El sistema se cierra por completo por motivos de seguridad.

- - - - *Fin de la trayectoria.*

Trayectoria alternativa C:

Condición: ♂ Ingresó una fecha anterior a la fecha de funcionamiento del sistema.

C1 ○ Almacena una copia de seguridad de toda la base de datos.

- - - - *Fin de la trayectoria.*

Trayectoria alternativa D:

Condición: ♂ Ingresó una fecha posterior a la fecha en que se está solicitando la copia de seguridad.

D1 ○ No descarga ningún archivo.

D2 ○ Continúa en el paso 3 de la trayectoria principal del **CU-04**.

- - - - *Fin de la trayectoria.*



4.9.5. CU-05 Agregar nueva bebida

4.9.5.1. Descripción completa

El administrador requiere poder agregar la información de una nueva bebida que se pondrá a la venta.

4.9.5.2. Atributos importantes

Caso de Uso:	CU-05 Agregar nueva bebida
Versión:	1.1
Autor:	Ramos Diaz Enrique
Supervisa:	Lara Cázares Jaime Arturo
Actor:	Administrador.
Propósito:	Mantener el catálogo actualizado en caso del lanzamiento de una nueva bebida por parte del negocio.
Entradas:	Nombre, descripción, categoría y modalidades de la bebida. Ruta de la imagen de la nueva bebida.
Origen:	RF12
Salidas:	Consulta de inserción a la base de datos en la colección de bebidas.
Destino:	CU-06 , CU-07
Precondiciones:	<ul style="list-style-type: none"> ● La imagen de la bebida debe cumplir con las características establecidas. ● La bebida no debe estar ya registrada en la base de datos. ● El usuario administrador debe haber sido autenticado correctamente.
Postcondiciones:	<ul style="list-style-type: none"> ● La base de datos contará con una nueva bebida para recomendar.
Errores:	<ol style="list-style-type: none"> 1. Los datos de texto de entrada poseen caracteres inválidos para la base de datos. 2. No se ingresan todos los datos de entrada (se dejan campos vacíos). 3. La ruta de la imagen ingresada es inválida o no existe. 4. Ocurre un error el subir la imagen, o esta es almacenada de forma corrupta.
Estado:	Revisión.
Observaciones:	—

4.9.5.3. Trayectorias del Caso de Uso

Trayectoria principal:

- 1 ⓘ Muestra la interfaz de usuario  IU2 Pantalla principal.
 - 2 ♂ Presiona el botón **Bebidas** de la interfaz de usuario  IU2 Pantalla principal.
 - 3 ⓘ Muestra la interfaz de usuario  IU7 Pantalla de gestión de bebidas.
 - 4 ⓘ Despliega una lista de las bebidas registradas en la base de datos, con una pequeña imagen a su izquierda.
 - 5 ♂ Presiona el botón **Agregar** de la interfaz de usuario  IU7 Pantalla de gestión de bebidas.
 - 6 ⓘ Muestra la interfaz de usuario  IU8 Pantalla para agregar o modificar bebida.
 - 7 ♂ Ingresa el nombre de la bebida.
 - 8 ♂ Ingresa una breve descripción de la bebida.
 - 9 ♂ Elige de una lista desplegable la categoría a la que pertenece la bebida.
 - 10 ♂ Selecciona las casillas correspondientes de la o las modalidades en las que se encuentra disponible la bebida.
 - 11 ♂ Presiona el botón **Cargar Imagen** y selecciona la ruta de la imagen de la bebida con las características establecidas. [Trayectoria A]
 - 12 ⓘ Muestra una vista previa de la imagen seleccionada.
 - 13 ♂ Presiona el botón **Guardar**. [Trayectoria B]
 - 14 ⓘ Indica que la nueva bebida fue agregada exitosamente.
- - - - *Fin del caso de uso.*

Trayectoria alternativa A:

Condición: La imagen que se intenta subir no cumple con las características necesarias para ser mostrada adecuadamente en la pantalla.

- A1** ⓘ Indica al ♂ que la imagen no cumple con las características y la imagen no es cargada.
 - A2** ⓘ Continúa en el paso 11 de la trayectoria principal del CU-05.
- - - - *Fin de la trayectoria.*

Trayectoria alternativa B:

Condición: ♂ Presiona el botón **Regresar** de la interfaz  IU8 Pantalla para agregar o modificar bebida.

- B1** ⓘ No se carga la bebida en la base de datos.
 - B2** ⓘ Continúa en el paso 3 de la trayectoria principal del CU-05.
- - - - *Fin de la trayectoria.*



4.9.6. CU-06 Modificar bebida

4.9.6.1. Descripción completa

El administrador requiere poder modificar la información que ha ingresado de una bebida.

4.9.6.2. Atributos importantes

Caso de Uso:	CU-06 Modificar bebida
Versión:	1.1
Autor:	Lara Cázares Jaime Arturo
Supervisa:	Morales Flores Víctor Leonel
Actor:	Administrador.
Propósito:	Mantener información sobre el catálogo de bebidas de la base de datos actualizada.
Entradas:	Nombre, de la bebida. Campos de la bebida que quieren ser actualizados(descripción, categoría, modalidades de la bebida y/o ruta de la nueva imagen de la bebida).
Origen:	RF13
Salidas:	Consulta de modificación a la base de datos en la colección de bebidas.
Destino:	No aplica.
Precondiciones:	<ul style="list-style-type: none"> • La bebida debe estar registrada en la base de datos. • De quererse modificar la imagen de la bebida, la nueva imagen debe cumplir con las características establecidas. • El usuario administrador debe saber que no puede modificar el nombre de la bebida, ya que este actúa como llave primaria en la base de datos*. • El usuario administrador debe haber sido autenticado correctamente.
Postcondiciones:	<ul style="list-style-type: none"> • La información de la bebida seleccionada será modificada.
Errores:	<ol style="list-style-type: none"> 1. El archivo de publicidad estática no puede ser encontrado. 2. La memoria es insuficiente. 3. No se puede guardar la publicidad en la base de datos.
Estado:	Revisión.
Observaciones:	—

4.9.6.3. Trayectorias del Caso de Uso

Trayectoria principal:

- 1 ○ Muestra la interfaz de usuario  IU2 Pantalla principal.
 - 2 ♂ Presiona el botón **Bebidas** de la interfaz de usuario  IU2 Pantalla principal.
 - 3 ○ Muestra la interfaz de usuario  IU7 Pantalla de gestión de bebidas.
 - 4 ○ Despliega una lista de las bebidas registradas en la base de datos, con una pequeña imagen a su izquierda.
 - 5 ○ Mantiene los botones **Modificar** y **Eliminar** de la interfaz de usuario  IU7 Pantalla de gestión de bebidas bloqueados mientras no se seleccione una bebida.
 - 6 ♂ Selecciona la bebida (de la lista de bebidas) cuya información desea modificar.[Trayectoria C]
 - 7 ○ Activa los botones **Modificar** y **Eliminar** de la interfaz de usuario  IU7 Pantalla de gestión de bebidas.
 - 8 ○ Muestra una imagen ampliada de la bebida que se seleccionó en el lado derecho de la interfaz de usuario.
 - 9 ♂ Presiona el botón **Modificar** de la interfaz de usuario  IU7 Pantalla de gestión de bebidas.[Trayectoria C]
 - 10 ○ Muestra la interfaz de usuario  IU8 Pantalla para agregar o modificar bebida con el campo del nombre de la bebida bloqueado para edición.
 - 11 ○ Llena los campos de la  IU8 Pantalla para agregar o modificar bebida con la información de la bebida seleccionada.
 - 12 ♂ Modifica la información que deseé (de la bebida) exceptuando el nombre de la bebida. [Trayectoria A]
 - 13 ♂ Presiona el botón **Guardar**. [Trayectoria B]
 - 14 ○ Indica que la información de la bebida fue actualizada exitosamente.
- - - - *Fin del caso de uso.*

Trayectoria alternativa A:

Condición: La imagen que se intenta subir no cumple con las características necesarias para ser mostrada adecuadamente en la pantalla.

- A1 ○ Indica al ♂ que la imagen no cumple con las características y la imagen no es cargada.
 - A2 ○ Continúa en el paso 12 de la trayectoria principal del CU-06.
- - - - *Fin de la trayectoria.*

Trayectoria alternativa B:

Condición: ♂ Presiona el botón **Regresar** de la interfaz  IU8 Pantalla para agregar o modificar bebida.

- B1 ○ No actualiza la información la bebida en la base de datos.
 - B2 ○ Continúa en el paso 3 de la trayectoria principal del CU-06.
- - - - *Fin de la trayectoria.*

Trayectoria alternativa C:

Condición: ♂ Decide no realizar ninguna acción y presiona el botón **Regresar** de la interfaz  IU7 Pantalla de gestión de bebidas.

- C1 ○ Regresa a la interfaz  IU2 Pantalla principal.
- - - - *Fin de la trayectoria.*



4.9.7. CU-07 Eliminar bebida

4.9.7.1. Descripción completa

El administrador requiere poder eliminar una bebida que ya no estará a la venta.

4.9.7.2. Atributos importantes

Caso de Uso:	CU-07 Eliminar bebida
Versión:	1.1
Autor:	Ramos Diaz Enrique
Supervisa:	Morales Flores Víctor Leonel
Actor:	Administrador.
Propósito:	Mantener el catálogo actualizado en caso de que una bebida sea descontinuada.
Entradas:	Nombre de la bebida a eliminar.
Origen:	RF14
Salidas:	Consulta de eliminación a la base de datos.
Destino:	No aplica.
Precondiciones:	<ul style="list-style-type: none"> ● La bebida debe estar registrada en la base de datos. ● El usuario administrador debe haber sido autenticado correctamente.
Postcondiciones:	<ul style="list-style-type: none"> ● La bebida seleccionada será eliminada de la base de datos. ● Ya no se hará la recomendación de esta bebida a los clientes potenciales.
Errores:	<ol style="list-style-type: none"> 1. Ocurre un error al eliminar la bebida desde la base de datos.
Estado:	Revisión.
Observaciones:	—

4.9.7.3. Trayectorias del Caso de Uso

Trayectoria principal:

- 1  Muestra la interfaz de usuario  IU2 Pantalla principal.
- 2  Presiona el botón **Bebidas** de la interfaz de usuario  IU2 Pantalla principal.
- 3  Muestra la interfaz de usuario  IU7 Pantalla de gestión de bebidas.
- 4  Despliega una lista de las bebidas registradas en la base de datos, con una pequeña imagen a su izquierda.

- 5 ⓘ Mantiene bloqueados los botones **Modificar** y **Eliminar** de la interfaz de usuario  IU7 Pantalla de gestión de bebidas.
- 6 ♂ Selecciona la bebida, de la lista de bebidas, que desea eliminar de la base de datos. [Trayectoria A]
- 7 ⓘ Activa los botones **Modificar** y **Eliminar** de la interfaz de usuario  IU7 Pantalla de gestión de bebidas.
- 8 ⓘ Muestra una imagen ampliada de la bebida que se seleccionó.
- 9 ♂ Presiona el botón **Eliminar** de la interfaz de usuario  IU7 Pantalla de gestión de bebidas. [Trayectoria A]
- 10 ⓘ Muestra la interfaz de usuario  IU5 Pantalla de confirmación para eliminar.
- 11 ♂ Presiona el botón **Aceptar** de la interfaz de usuario  IU5 Pantalla de confirmación para eliminar. [Trayectoria B]
- 12 ⓘ Indica que la bebida fue eliminada exitosamente de la base de datos.
- - - *Fin del caso de uso.*

Trayectoria alternativa A:

Condición: ♂ Decide no realizar ninguna acción y presiona el botón **Regresar** de la interfaz  IU7 Pantalla de gestión de bebidas.

- A1** ⓘ Continúa en el paso 1 de la trayectoria principal del CU-07
- - - *Fin de la trayectoria.*

Trayectoria alternativa B:

Condición: ♂ Presiona el botón **Cancelar** de la interfaz  IU5 Pantalla de confirmación para eliminar.

- B1** ⓘ No se elimina la bebida en la base de datos.
B2 ⓘ Continúa en el paso 3 de la trayectoria principal del CU-07.
- - - *Fin de la trayectoria.*



4.9.8. CU-08 Eliminar publicidad estática

4.9.8.1. Descripción completa

El administrador requiere poder eliminar una imagen de publicidad estática por cuestiones relacionadas al negocio.

4.9.8.2. Atributos importantes

Caso de Uso:	CU-08 Eliminar publicidad estática
Versión:	1.1
Autor:	Morales Flores Víctor Leonel
Supervisa:	Lara Cázares Jaime Arturo
Actor:	Administrador.
Propósito:	Permitir eliminar publicidad que se considere obsoleta o errónea.
Entradas:	ID de la publicidad que se desea eliminar
Origen:	RF19
Salidas:	Consulta de eliminación de publicidad a la base de datos.
Destino:	No aplica.
Precondiciones:	<ul style="list-style-type: none"> • La publicidad debe estar registrada en la base de datos. • El usuario administrador debe haber sido autenticado correctamente.
Postcondiciones:	<ul style="list-style-type: none"> • La publicidad ya no está registrada en la Base de Datos. • Ya no se hará la recomendación de esta bebida a los clientes potenciales.
Errores:	<ol style="list-style-type: none"> 1. Los archivos como las imágenes no son encontrados por lo que el sistema no puede completar la operación con éxito.
Estado:	Revisión.
Observaciones:	—

4.9.8.3. Trayectorias del Caso de Uso

Trayectoria principal:

- 1  Muestra la interfaz de usuario  IU2 Pantalla principal.
- 2  Presiona el botón **Publicidad Estática** de la interfaz de usuario  IU2 Pantalla principal.
- 3  Muestra la interfaz de usuario  IU3 Pantalla para gestión de publicidad.

- 4 ○ Despliega una lista de la publicidad registrada en la base de datos, con una pequeña imagen a su izquierda.
- 5 ○ Mantiene bloqueado el botón **Eliminar** de la interfaz de usuario  IU3 Pantalla para gestión de publicidad.
- 6 ♂ Selecciona la publicidad que desea eliminar de la base de datos.[Trayectoria A]
- 7 ○ Activa el botón **Eliminar** de la interfaz de usuario  IU3 Pantalla para gestión de publicidad.
- 8 ○ Muestra una imagen ampliada de la publicidad que se seleccionó.
- 9 ♂ Presiona el botón **Eliminar** de la interfaz de usuario  IU3 Pantalla para gestión de publicidad. [Trayectoria A]
- 10 ○ Muestra la interfaz de usuario  IU5 Pantalla de confirmación para eliminar.
- 11 ♂ Presiona el botón **Aceptar** de la interfaz de usuario  IU5 Pantalla de confirmación para eliminar. [Trayectoria B]
- 12 ○ Indica que la publicidad fue eliminada exitosamente de la base de datos.
- - - - *Fin del caso de uso.*

Trayectoria alternativa A:

Condición: ♂ Decide no realizar ninguna acción y presiona el botón **Regresar** de la interfaz  IU3 Pantalla para gestión de publicidad.

- A1** ○ Continúa en el paso 1 de la trayectoria principal del CU-08
- - - - *Fin de la trayectoria.*

Trayectoria alternativa B:

Condición: ♂ Presiona el botón **Cancelar** de la interfaz  IU5 Pantalla de confirmación para eliminar.

- B1** ○ No se elimina la bebida en la base de datos.
B2 ○ Continúa en el paso 3 de la trayectoria principal del CU-08.
- - - - *Fin de la trayectoria.*

4.10. Gestión de riesgos

La gestión de riesgos es un factor importante a considerar desde el inicio del proyecto para buscar alternativas de prevención, monitorización y planes de contingencia en caso de que estos ocurran. Al realizar un adecuado análisis de riesgos reducimos la posibilidad de que estos ocurran y con ello que el calendario para el desarrollo del sistema no se vea afectado radicalmente.

Los principales riesgos identificados en el proyecto se muestran de forma detallada en las siguientes tablas:

Riesgo:	Riesgo-01 Falta de recursos humanos.
Tipo:	Proyecto.
Descripción:	Aún hay muy pocos recursos humanos capaces de diseñar e implementar soluciones personalizadas con el uso de <i>Deep Learning</i> , por lo que la ausencia de alguna de las personas que actualmente componen el equipo podría afectar al desarrollo del sistema y las entregas.
Probabilidad de ocurrencia:	40%
Afectación al proyecto:	Seria.
Estrategia de prevención:	Mantener un buen ambiente de trabajo entre los integrantes del equipo para no perder recursos humanos ya capacitados y procurar que el equipo esté al tanto y actualizado en las técnicas de <i>Deep Learning</i> que se implementan en el proyecto.
Estrategia de monitoreo:	Sprints con el equipo entero, así como sesiones scrum para solucionar de manera oportuna cualquier problema entre los recursos humanos del proyecto, ya sea en cuanto a relaciones sociales, de salud o problemas de conocimiento.
Plan de contingencia:	Reducción de tareas por sprint y posible incorporación de un nuevo recursos para ser capacitado por los integrantes restantes del equipo.

Tabla 4.11: Riesgo-01: Falta de recursos humanos.

Riesgo:	Riesgo-02 Estimación de tiempo errónea.
Tipo:	Proyecto.
Descripción:	Es posible que alguna de las actividades identificadas en la calendarización no haya sido correctamente clasificada en cuanto a costo de desarrollo, provocando que a la hora de realizar la tarea el equipo se encuentre con que se requiere mucho más tiempo para desarrollarla de forma adecuada, por lo que la calendarización original tendrá que ser modificada y se puede generar retraso en la entrega y culminación del proyecto.
Probabilidad de ocurrencia:	50%
Afectación al proyecto:	Seria.
Estrategia de prevención:	Informarse sobre las tecnologías necesarias y disponibles para desarrollar el proyecto, proponer tiempos de desarrollo para cada tarea ligeramente más largos a los estimados en un inicio para tener tiempo extra en caso de que esto ocurra.
Estrategia de monitoreo:	Avances en cada sprint y tener un seguimiento de los problemas que se reportaron tras cada sesión de scrum y sprint.
Plan de contingencia:	Ajustar el calendario en caso de que ocurra y notificar de inmediato a todos los involucrados. Esto incluye al equipo de desarrollo, cliente, etc.

Tabla 4.12: Riesgo-02: Estimación de tiempo errónea.

Riesgo:	Riesgo-03 Costo de tecnología.
Tipo:	Proyecto.
Descripción:	Los dispositivos capaces de llevar acabo tareas de <i>Deep Learning</i> de forma eficiente son extremadamente costosos y escasos, dejando pocas alternativas de hardware que sirvan para implementar el software desarrollado y cumpla con los requerimientos de tiempo de procesamiento.
Probabilidad de ocurrencia:	80 %
Afectación al proyecto:	Seria.
Estrategia de prevención:	Comenzar con tareas ligeras y hacer uso de equipo sencillo. Usar equipo especial solo cuando se necesite realizar entrenamientos de redes neuronales. Buscar algoritmos o implementaciones que satisfagan las necesidades del trabajo y sean las más óptimas en cuanto aprovechamiento del hardware.
Estrategia de monitoreo:	Evaluar el incremento de recursos de hardware necesarios tras cada sprint.
Plan de contingencia:	Uso de servidores especiales durante la fase de desarrollo para no tener que realizar la compra del hardware en etapas tempranas.

Tabla 4.13: Riesgo-03: Costo de tecnología.

Riesgo:	Riesgo-04 Tiempo de despliegue.
Tipo:	Producto.
Descripción:	Aún con la metodología usada, debido a la tecnología que se implementará, el tiempo que tardará en desplegarse una primera versión funcional será alto.
Probabilidad de ocurrencia:	80 %
Afectación al proyecto:	Tolerable.
Estrategia de prevención:	Tener un cronograma con un margen de error e intentar cumplirlo.
Estrategia de monitoreo:	Verificar que el avance que se consigue tras cada sprint permitirá que la fecha de entrega del sistema se cumpla o, en su defecto, reestructurar actividades tras cada sprint para cumplir con la fecha de entrega final.
Plan de contingencia:	Trabajar horas extra para poder adecuarse al tiempo de entrega.

Tabla 4.14: Riesgo-04: Tiempo de despliegue.

Riesgo:	Riesgo-05 Herramientas para desarrollo y mantenimiento.
Tipo:	Producto.
Descripción:	Durante el desarrollo y/o mantenimiento del proyecto existe la posibilidad de que algunas de las herramientas usadas para el desarrollo del sistema sufra alguna actualización que afecte al curso original del desarrollo. Esto porque se puede liberar una nueva versión que elimine o modifique el funcionamiento de alguna de sus características y con ello se tengan que hacer modificaciones al trabajo ya desarrollado.
Probabilidad de ocurrencia:	30 %
Afectación al proyecto:	Tolerable.
Estrategia de prevención:	Mantener al equipo informado sobre la documentación de las herramientas, así como de liberaciones de versiones beta o similares por parte de los proveedores oficiales de las herramientas.
Estrategia de monitoreo:	Revisión de cambios en versiones betas o similares que hayan sido anunciadas por los proveedores oficiales de las herramientas que se utilizan.

Riesgo:	Riesgo-05 Herramientas para desarrollo y mantenimiento.
Plan de contingencia:	Actualizar el componente que haya sido afectado por la nueva versión u optar por no actualizar la versión de la herramienta utilizada.

Tabla 4.15: Riesgo-05: Herramientas para desarrollo y mantenimiento.

Riesgo:	Riesgo-06 Daño de hardware.
Tipo:	Negocio.
Descripción:	El sistema se instalará en un dispositivo local que cumpla con las características para llevar a cabo las tareas de <i>Deep Learning</i> , por lo tanto se corre el riesgo de que en caso de necesitar un repuesto nos se encuentre con que ya no se encuentra disponible en el mercado, la obtención del repuesto sea demasiado costosa y/o tardada.
Probabilidad de ocurrencia:	20%
Afectación al proyecto:	Catastrófica.
Estrategia de prevención:	Realizar el mantenimiento adecuado del dispositivo, verificar su funcionamiento adecuado, protegerlo contra descargas eléctricas a través de dispositivos específicos para ese fin y corregir los errores de funcionamiento que se presenten a lo largo del tiempo.
Estrategia de monitoreo:	Establecer periodos para revisar el buen funcionamiento del hardware. Inspección visual y mantenimiento preventivo del hardware.
Plan de contingencia:	Tener una copia offline lista para instalarse en cuanto se compre un repuesto del hardware.

Tabla 4.16: Riesgo-06: Daño de hardware.

Riesgo:	Riesgo-07 Desaparición de la empresa.
Tipo:	Negocio.
Descripción:	Debido a que nuestro cliente es una startup, según las estadísticas, la empresa podría ir a la quiebra o desaparecer por falta de capacidades competitivas contra los competidores.
Probabilidad de ocurrencia:	70%
Afectación al proyecto:	Catastrófica.
Estrategia de prevención:	Establecer cláusulas que protejan al equipo de desarrollo en dado caso.
Estrategia de monitoreo:	Plática periódica con el cliente sobre el estado general en términos financieros de la empresa.
Plan de contingencia:	Proteger al equipo de desarrollo de sanciones o responsabilidades no acordados.

Tabla 4.17: Riesgo-07: Desaparición de la empresa.

4.11. Costo estimado de desarrollo

Es necesario estimar el costo del desarrollo del proyecto, por ello en la Tabla 4.18 se muestran los costos en pesos mexicanos de los componentes de hardware necesarios para la implementación, los cuales se abarcan de una mejor y más detallada manera en la sección [Especificación del hardware](#) del capítulo de [Diseño e Implementación](#). Cabe destacar que el costo del equipo de cómputo para el desarrollo del sistema no se refleja en la Tabla 4.18 ya que fue proporcionado por el Centro de Investigación en Computación (CIC) de forma presencial y remota.

Concepto:	Costos de desarrollo
Cámara HIKVISION DS-2CD2143G0-IS:	\$2,278.00
NVIDIA Jetson AGX Xavier Developer Kit:	\$30,920.80
Switch PoE HIKVISION DS-3E0105P-E:	\$999.00
Hisense Smart TV 50R6000FM:	\$7,299.00
3 Cables Ethernet:	\$81.00
1 Cable HDMI:	\$40.00
Mouse USB:	\$80.00
Teclado USB:	\$150.00
Electricidad:	\$5,400.00
Servicio de internet:	\$13,320.00
Pedestal:	\$3,300.00
Total:	\$63,867.80

Tabla 4.18: Costos estimados del desarrollo.

4.12. Discusión del capítulo

Siendo que este proyecto tiene un fin de producción en la vida real es importante establecer una [Metodología](#) que satisfaga la necesidad de entregas funcionales que puedan otorgar valor al cliente. El uso de SCRUM permite el correcto desarrollo del proyecto ya que da la facilidad de hacer cambios constantes que [BubbleTown®](#) solicite, siempre y cuando estos estén a tiempo y sean viables de realizar.

Es imprescindible satisfacer las necesidades de [BubbleTown®](#) establecidas en las [Historias de usuario](#), entre las cuales sobresale obtener una recomendación de bebida personalizada a través de solamente el rostro, pues da la funcionalidad principal a este sistema. Las [Reglas del negocio](#) se encargan de dar un margen específico a lo que el sistema puede y no hacer, así como la naturaleza y el manejo de los datos de entrada y salida. Se recalca la importancia de manejar la tipografía y colores asociados a la marca para mantener su personalidad (descritas en las [Reglas del negocio](#)) dentro y fuera del establecimiento, con ello se logra un estilo homogéneo para el cliente.

Las [Historias de usuario](#) dan lugar a distintos [Requerimientos funcionales](#). Dado a que solicitan un análisis facial, es importante informar al usuario del manejo de la privacidad de este dato y así brindarle confianza de que sus datos serán resguardados y usados adecuadamente, única y exclusivamente para m con los objetivos de este trabajo; esto recalca la necesidad de utilizar métodos criptográficos (reflejado el RF-08) en la tarea del almacenamiento de rostros en la base de datos y de la correcta autenticación de los administradores para evitar usuarios sin permisos en el sistema.

Los [Requerimientos no funcionales](#) hacen referencia a las propiedades emergentes que este sistema debe tener. Entre ellas se identifica la necesidad de cumplir con un mínimo tiempo de procesamiento, lo cual da un indicio de qué especificaciones de equipo se deben ocupar (esto se trata en el capítulo de [Diseño e Implementación](#)). También se habla de características fundamentales para la publicidad, con el fin de lograr un efecto positivo en distintas personas y que estas pasen a convertirse en clientes que realicen una compra.

La [Descripción de los casos de uso](#) permite visualizar, especificar y documentar el comportamiento de los elementos

del sistema. Entre ellos destaca el **CU-01**, que representa la principal funcionalidad de este. Generar la publicidad dirigida o personalizada implica obtener bastantes datos de forma muy rápida desde una imagen, por lo que esto conlleva a primero leer la imagen desde la cámara, seguido de identificar el rostro con el uso de **Redes neuronales**, posteriormente a partir de la estimación de la **Edad, Género, Personalidad: Big Five** y lectura de la temperatura ambiental, generar una recomendación de una bebida disponible en el menú de **BubbleTown®**, y por último, plasmar esa recomendación utilizando **Realidad Aumentada**.

La clara necesidad de tener un sistema estable, seguro y rápido obliga a establecer la **Gestión de riesgos**. Se descubrió que existe una vulnerabilidad alta, que va desde aspectos económicos como el cierre de la empresa, el costo que implica la obtención de un equipo de cómputo especializado que cumpla con los requerimientos solicitados o bajo el presupuesto que se tiene disponible para el proyecto, hasta aspectos de la conformación del equipo, ya que es sumamente complicado encontrar nuevos recursos humanos con los conocimientos, actitudes y aptitudes que complementen al equipo en dado caso de que algún integrante no pueda o quiera continuar.

CAPÍTULO 5

Diseño e Implementación

5.1. Introducción al capítulo

En este capítulo se abordará el diseño e implementación pertinente del proyecto. Al tratarse de un sistema que debe realizar muchas funciones, y basándose en el paradigma de programación orientada a objetos, se necesita determinar la ejecución que se llevará a cabo dentro de cada una las clases que integran al sistema, por ello se presenta un [Diagramas de actividades](#), en donde se muestran estos flujos de actividades. Este paradigma hace imprescindible el diseño de un [Diagrama de clases](#), necesario para definir el diseño de cada clase, además de su comportamiento e interacción con el resto de ellas. Por otro lado, nos apoyamos del [Diagramas de secuencia](#) para representar los flujos de control del sistema a través de un tiempo determinado.

Para señalar los dispositivos de hardware se utiliza el [Diagrama de Despliegue](#), que a su vez, muestra las conexiones físicas entre estos. Sin embargo, dado a que en dicho diagrama no se establecen todas las especificaciones técnicas de los dispositivos de hardware, en la sección de [Especificación del hardware](#) se tratan todos los aspectos técnicos importantes de estos dispositivos, los cuales son: la cámara de red, el switch PoE, la NVIDIA Jetson y la pantalla LED.

En las [Herramientas de software a utilizar](#) se muestran todas las herramientas de software necesarias para el desarrollo de este sistema (lenguajes de programación, librerías, plugins, APIs, frameworks, SDKs, etc.), además de una breve explicación de su fin y una justificación de su elección, con base al rendimiento y compatibilidad con el sistema operativo instalado en la NVIDIA Jetson, tomando en cuenta la misma arquitectura que esta posee.

El [Diagrama general del sistema](#) expresa con una serie de bloques los diferentes módulos y su correspondiente interpretación para el proyecto. Esta sección da pie a todas las demás, donde se aborda a fondo el diseño e implementación de los distintos módulos.

En el módulo de la [Captura de video a imagen](#) se habla de la forma de obtener cada frame capturado por la cámara de video, abarcando la configuración necesaria entre los dispositivos y los protocolos de comunicación que se emplean para dicho fin.

La sección del [Preprocesamiento de la imagen](#) abarca las operaciones que se deben aplicar a estos frames, debido a

que las arquitecturas de las [Redes neuronales convolucionales](#) utilizadas en este sistema dictan una serie de requisitos que una imagen debe de cumplir para poder ser ingresada a estos modelos.

En la [Detección de rostros en la imagen](#) se explica la manera en que se realiza un entrenamiento personalizado en el modelo de la red neuronal [Single Shot Detection](#) basado en la arquitectura [MobileNet](#) para el reconocimiento de rostros, además de presentar los pasos y solucionar los principales problemas que surgen al implementar dicho modelo en la arquitectura de la NVIDIA Jetson. Asimismo, se explica que una vez detectados los rostros en la imagen, es necesario recortarlos, extraerlos y también procesarlos.

Es necesario extraer las características de la edad, género y personalidad dada la imagen del rostro de la persona, por ello se diseñan, construyen, entranan y ponen en funcionamiento [Redes neuronales convolucionales](#) que sean capaces de llevar a cabo esta tarea; estas se dividen en la [Red neuronal convolucional para Edad y Género](#) y [Red neuronal convolucional para Big Five](#). Similar a la red [Single Shot Detection](#), se presentan los pasos para su implementación en la arquitectura de la NVIDIA Jetson, así como la solución de los principales problemas que pudiesen surgir durante la misma.

Un aspecto que se deberá abordar es el diseño del [Recomendador del sistema](#), el cual usará todos los datos previamente obtenidos para brindarle al cliente una opción de bebida adecuada dadas sus características. En esta sección se describirán los factores que intervienen al momento de realizar la recomendación así como la [Vectorización de bebidas](#) para posteriormente abordar el diseño de la [Base de datos para el sistema](#) debido a que es de interés guardar la información de los datos que participan en la recomendación de la bebida, además del rostro de la persona correspondiente.

Otro punto del sistema es la [Generación de realidad aumentada](#), pues otorga una publicidad dinámica para el cliente. Es en esta parte del diseño donde todos los elementos cobran sentido, ya que tras todos los procesos realizados, en este paso el cliente puede observar la bebida propuesta.

No hay que olvidar que este sistema está planeado para el uso de [BubbleTown®](#) lo que implica brindarle una forma amigable de poder cambiar la publicidad estática, las imágenes de las bebidas y generar un respaldo de la información guardada en la [Base de datos para el sistema](#), para ello se plantea un [Diseño de la interfaz gráfica](#), donde se puedan hacer las acciones mencionadas y además hacer uso de todo el sistema de forma sencilla y cómoda para los usuarios con permisos de administrador.

Finalmente, se exhiben los [Atributos de calidad](#) corresponden y dejan en claro qué puntos de calidad cumple o no el sistema una vez terminado

5.2. Diagramas de actividades

Los diagramas de actividades son, en esencia, diagramas de flujo que muestra el flujo de control entre actividades. A diferencia de un diagrama de flujo tradicional, estos muestran tanto la concurrencia como las bifurcaciones del control. Mientras que los diagramas de interacción (como lo son los [Diagramas de secuencia](#)) apoyan en la visualización del flujo de control entre objetos, los diagramas de actividades nos ayudan con el flujo de control entre los distintos pasos [104]. A continuación se presentarán los diagramas de actividades asociados al sistema que se consideran más relevantes.

En la Figura 5.1 se observa claramente cómo el proceso comienza cuando el cliente o los clientes potenciales observan la publicidad estática mostrada en la pantalla. Entonces, el sistema reconoce uno o varios rostros, los recorta y los procesa. Dichos rostros procesados son propagados a través de dos redes neuronales, una para obtener la edad y el género de la o las personas y otra para obtener su personalidad; concurrentemente, el sistema obtiene la temperatura

ambiental actual mediante una API. Una vez recabados los resultados anteriores, estos son ingresados al recomendador, del cual se obtendrá una bebida junto con su modalidad (Zen, Frozen o Iced). Tras obtener el nombre de la bebida, se obtiene la imagen de la bebida de la base de datos; también, se guarda el o los rostros detectados cifrados junto con todos los resultados obtenidos.

Por último, se mostrará un globo de pensamiento con la bebida recomendada junto al o a los rostros.



Figura 5.1: Diagrama de actividades para Caso de uso [CU-01](#): Generar publicidad dirigida.

Como se ve en la Figura 5.2, al iniciar el proceso el sistema mostrará la  IU1 Pantalla principal, donde el usuario

administrador seleccionará “Publicidad estática”, lo cual mostrará la IU8 Pantalla de gestión de publicidad. En la IU8 se selecciona la opción “Agregar” y será desplegada la IU3 Pantalla para agregar publicidad. Posteriormente, el administrador cargará la imagen de publicidad e ingresará un nombre de referencia para dicha publicidad. Si la imagen que fue cargada no cumple con las características establecidas, se tendrá que cargar una imagen que sí cumpla con tales características. El sistema mostrará una vista previa de la imagen que se desea cargar. Para terminar el proceso, el administrador selecciona “Guardar” y el sistema indicará a este que la imagen fue cargada exitosamente.

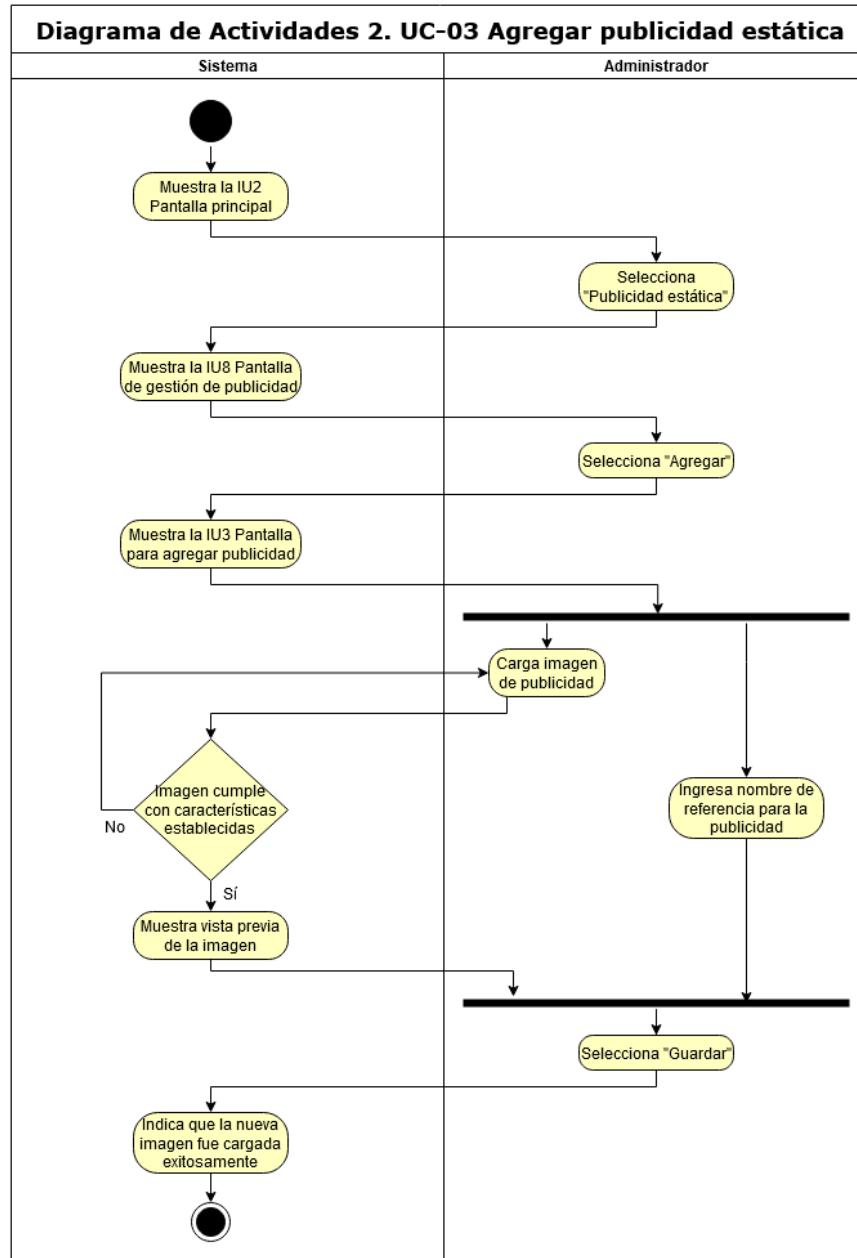


Figura 5.2: Diagrama de actividades para Caso de uso **CU-03: Agregar publicidad estática**.

En la Figura 5.3 se muestran los pasos de otra función importante que es la de respaldar las recomendaciones. Al iniciar el proceso, el sistema mostrará la IU1 Pantalla principal, donde el usuario administrador seleccionará “Respaldo”, lo cual mostrará la IU4 Pantalla para hacer respaldo.

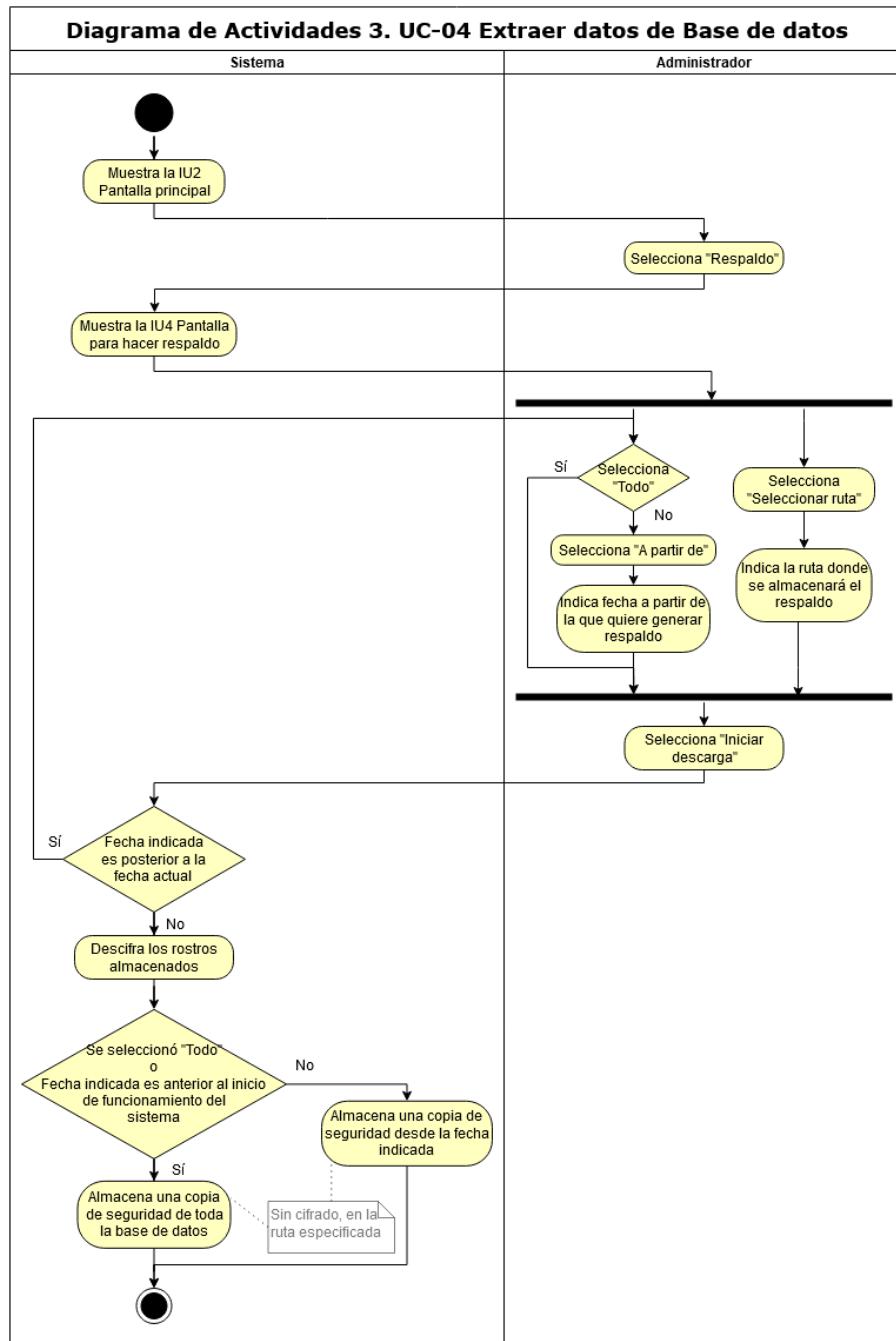


Figura 5.3: Diagrama de actividades para Caso de uso **CU-04**: Extraer datos de base de datos.

En la  IU4 el administrador selecciona la opción “Seleccionar ruta”, en la cual se indica la ruta donde se almacenará el respaldo. Después, si el usuario desea hacer un respaldo de toda la base de datos seleccionará la opción “Todo”; de lo contrario, seleccionará la opción “A partir de”, en donde se indicará la fecha a partir de la cual se desea generar el respaldo. Una vez hecho esto, el administrador selecciona “Iniciar descarga”, con una fecha válida elegida, se descifrarán todos los rostros y si el administrador selecciona la opción “Todo”, se almacenará una copia de seguridad de todas las recomendaciones de la base de datos. La copia de seguridad se almacena sin cifrado en la ruta especificada.

5.3. Diagrama de clases

Los diagramas de clases son herramientas que permiten modelar una vista de diseño estática, así como modelar el vocabulario de un sistema, sus colaboraciones y esquemas. Con un diagrama de clases se tiene el aspecto estático de los bloques pero también se cuenta con las relaciones y detalles para su construcción [104]. Por ello es que en la Figura 5.4 se muestra el diagrama de clases del sistema, con el objetivo de auxiliar en estructurar las bases para una correcta construcción de los elementos fundamentales del sistema.

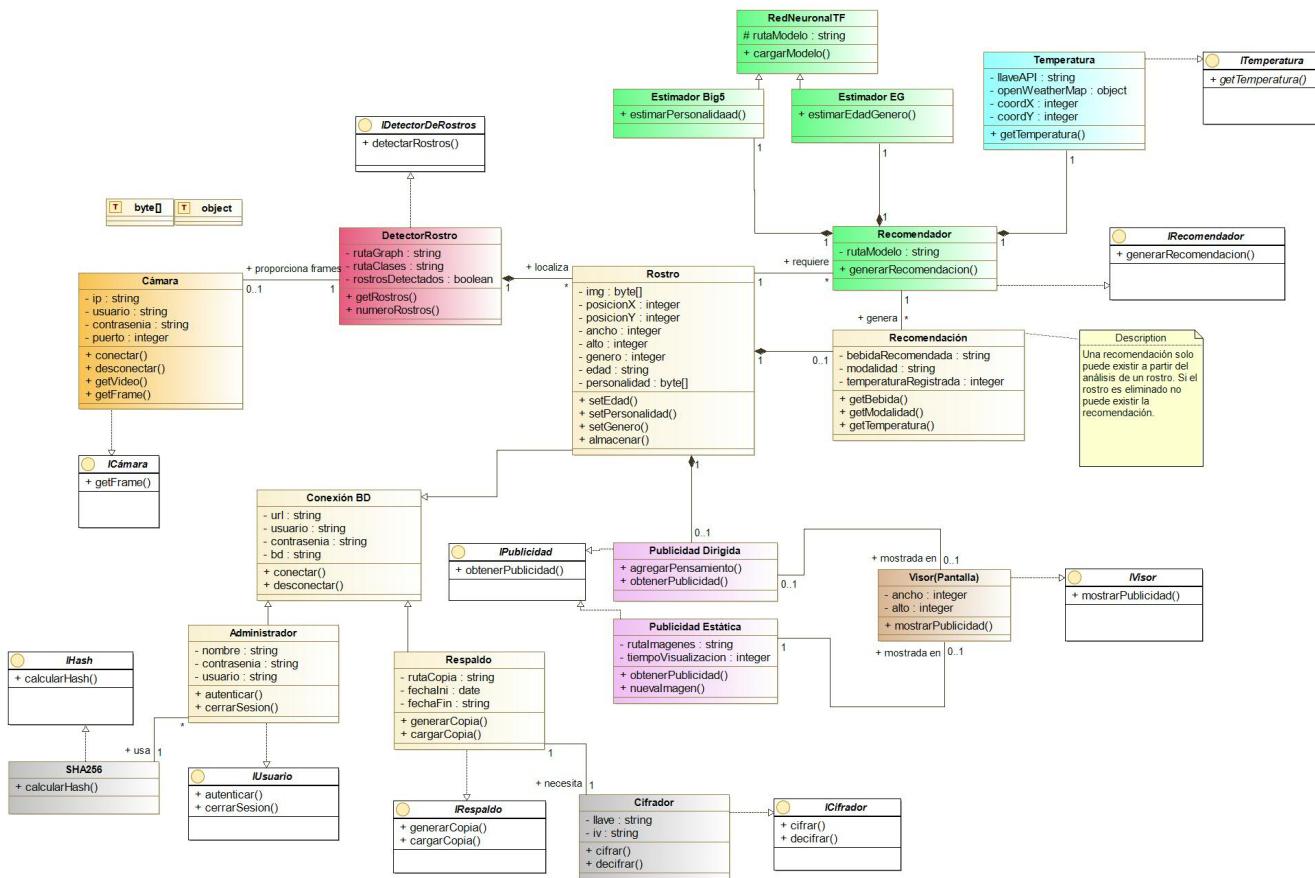


Figura 5.4: Clases importantes del sistema con atributos y operaciones.

5.4. Diagramas de secuencia

Los diagramas de secuencia son una buena herramienta para comprender el flujo de control a lo largo de un periodo de tiempo. El contenido principal de estos diagramas son los mensajes y permite al lector entender cómo interactúan los objetos para realizar una operación [104]. Estos diagramas son utilizados en el trabajo para auxiliar al lector a comprender cómo interactúan los objetos del [Diagrama de clases](#).

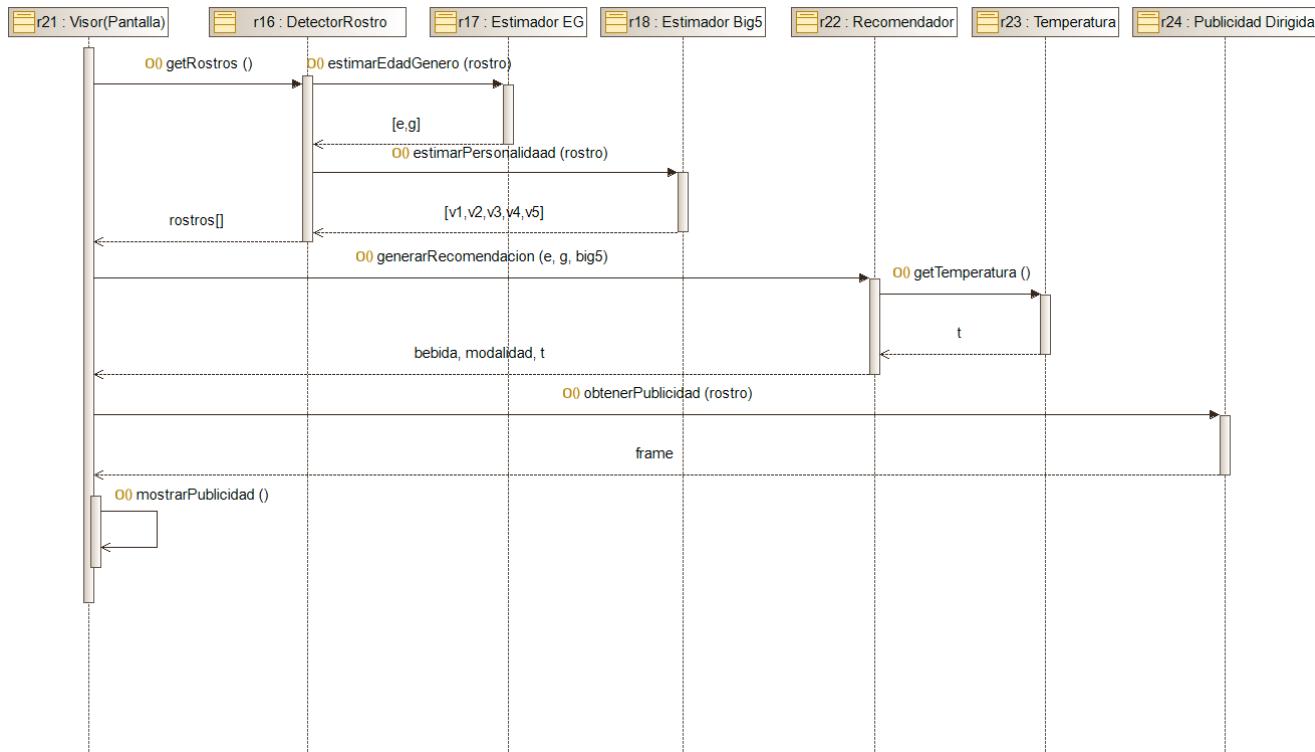


Figura 5.5: Diagrama de secuencia para generar la publicidad dirigida en el CU-01

En la Figura 5.5 se muestra el proceso que se lleva a cabo cuando el Detector de Rostros detecta alguno. En esta figura se puede observar cómo el el Detector solicita los valores de edad, género y personalidad-Big5 a los estimadores, para posteriormente retornar un arreglo de rostros al "Visor". Una vez obtenido el rostro, se envían los datos al "Recomendador" para que, junto con la temperatura ambiental, obtenga una bebida y modalidad apta para esas características. Por último, se envía a "Publicidad Dirigida" los rostros para que se encargue de generar un frame con la realidad aumentada y pueda ser debidamente proyectada.

5.5. Diagrama de Despliegue

Los diagramas de despliegue permiten modelar la topología del hardware sobre el que se ejecuta el sistema. En la Figura 5.5 se proporciona al lector los nodos que conformarán al sistema cuando este haya sido puesto en producción. Los nodos que se encuentran dentro del cuadrado verde son los que pertenecen al proyecto, mientras que los nodos externos representan nodos necesarios pero que son externos al sistema; es decir, su administración o configuración no fueron realizados durante el desarrollo de este proyecto como es el caso del “Modem” que es proporcionado por el proveedor internet o el “ServidorOWM” que se refiere al servidor que provee al sistema de la temperatura ambiental actual en el sitio a través de una API.

Entre los nodos que competen al sistema se encuentran la “Cámara Ethernet” que será la encargada de proporcionar al sistema las imágenes, el nodo “Jetson Xavier AGX Developer Kit” será el encargado de hacer todo el procesamiento y almacenamiento de los datos generados, el “Switch PoE” es un switch con tecnología PoE (Power over Ethernet) que permitirá la alimentación de la cámara y la comunicación entre ella y la “Jetson Xavier AGX Developer Kit”; por último, la “Pantalla” será el dispositivo encargado de mostrar al usuario la publicidad (ya sea estática o personalizada). Todos estos componentes se explican de forma más detallada en [Especificación del hardware](#).

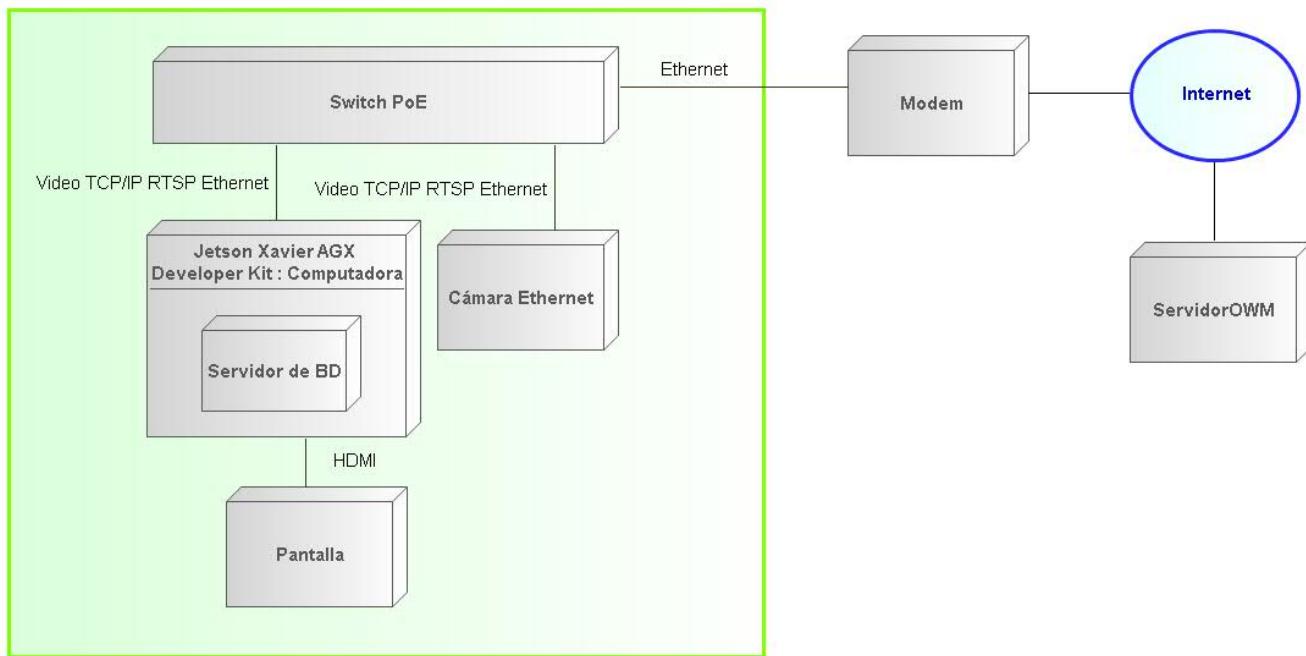


Figura 5.6: Diagrama de despliegue del sistema.

5.6. Especificación del hardware

En el [Diagrama de Despliegue](#), representado por la Figura 5.6, se presenta el flujo de comunicación del sistema desde una perspectiva de los dispositivos y componentes de hardware que lo integran, así como el tipo de información que viaja entre ellos, el protocolo de comunicación que utilizan y la forma de conectarlos. Existen dos tipos de componentes: los dispositivos críticos, aquellos que son la columna vertebral de todo el sistema, los cuales son: la cámara de red, la NVIDIA Jetson, el switch PoE y la pantalla LED; y luego esta el modem, cuya única función es proveer de Internet a la NVIDIA Jetson.

En la siguiente subsección [Dispositivos a usar](#), se abordan a fondo cada uno de los dispositivos críticos en el [Diagrama de Despliegue](#), y en la sección [Captura de video a imagen](#) se explica y justifica el uso de los protocolos TCP/IP y RTSP.

5.6.1. Dispositivos a usar

En este apartado se exponen los dispositivos a usar, así como una explicación detallada de cada uno de ellos y su principal papel en el presente trabajo. Se incluyen su información de entrada, de salida, los protocolos de comunicación que utilizan y los puertos (tipo de cable) entre conexiones, la cual fue extraída directamente de las hojas de especificación de los distintos proveedores para cada dispositivo.

En la Tabla 5.1 se describen los principales atributos de interés de la cámara de red a utilizar, y en la Figura 5.7 se muestra el modelo seleccionado.

Cámara de red	
Modelo:	HIKVISION DS-2CD2143G0-IS [105]
Características:	Resolución máxima: 2688 x 1520 @ 30fps Lente fija de 2.8 mm Consumo eléctrico PoE: 36V - 57V 0.2A, máximo 75W Flujo de video con codificación H.264
Función:	Captura del video a procesar
Puertos:	RJ45 10M/100M auto adaptable para cable Ethernet
Protocolos:	TCP/IP, HTTP, IEEE 802.1X, RTSP
Entradas:	Alimentación: Switch PoE
Salidas:	Video: Switch PoE

Tabla 5.1: Especificaciones de la cámara de red a utilizar.



Figura 5.7: Imagen tomada de [\[105\]](#) del modelo de cámara de red a utilizar.

En la Tabla 5.2 se describen los principales atributos de interés del switch PoE a utilizar, y en la Figura 5.8 se muestra el modelo seleccionado.

Switch PoE	
Modelo:	HIKVISION DS-3E0105P-E [106]
Características:	No requiere configuración Detecta y suministra energía automáticamente a dispositivos con alimentación IEEE 802.3AF/AT La energía eléctrica se transmite junto con los datos en un solo cable Proporciona 15.8 W por puerto PoE, en total 58 W Consumo eléctrico: 51V 1.25A, máximo 63W
Función:	Alimentar la cámara de red y ser un medio de conexión a Internet para el resto de los dispositivos.
Puertos:	4 puertos RJ45 10/100 Mbps para cable Ethernet 1 puerto Ethernet 100M
Protocolos:	IEEE 802.3X, TCP/IP, RTSP
Entradas:	Video: Cámara de red
Salidas:	Alimentación: Cámara de red Video + Internet: NVIDIA Jetson

Tabla 5.2: Especificaciones del switch PoE a utilizar.



Figura 5.8: Imagen tomada de [106] del modelo de switch PoE utilizado.

En la Figura 5.9 se muestra cómo se debe realizar la conexión física entre la cámara de red y la NVIDIA Jetson por medio del switch PoE. Posteriormente, en la sección [Captura de video a imagen](#) se explica la configuración necesaria para capturar el video de la cámara de red en la NVIDIA Jetson para procesar sus frames.

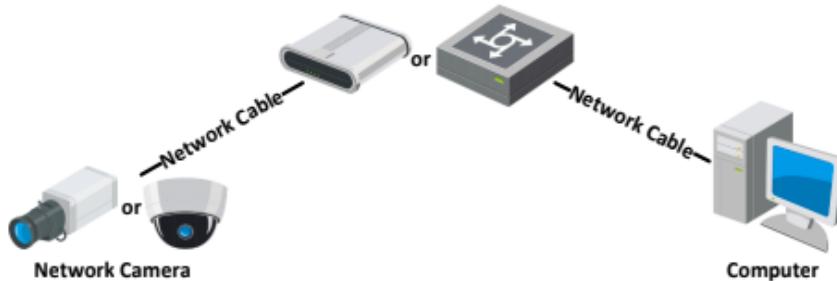


Figura 5.9: Diagrama tomado de [107] que ilustra la conexión entre la cámara de red seleccionada y una computadora por medio de un switch o router.

En la Tabla 5.3 se describen los principales atributos de interés de la unidad de procesamiento central seleccionada: la NVIDIA Jetson, y en la Figura 5.10 se muestra el contenido del paquete o modelo elegido para este trabajo.

NVIDIA Jetson	
Modelo:	AGX Xavier Developer Kit [108]
Características:	<p>Es una pequeña pero potente computadora para tareas específicas de inteligencia artificial. Está potenciada por aceleradores de aprendizaje profundo NVIDIA y procesadores para imagen y video. [109]</p> <p>GPU Volta de 512 núcleos tensoriales 1GB VRAM CPU ARM v8.2 de 8 núcleos, 64 bits, 8MB L2 + 4MB L3 32 GB RAM 32 GB almacenamiento Salida de video con resolución máxima de 4K 60fps 30 trillones de operaciones por segundo Adaptador de CA externo de 19V</p>
Función:	Es el componente más importante del sistema, ya que se va a encargar de la mayoría de las tareas, como la recepción de video, extracción de frames, pre procesamiento de imágenes, detección de rostros, además de contener todas las redes neuronales, el recomendador, los catálogos de diseños y la generación de realidad aumentada.
Puertos:	RJ45 para cable Gigabit Ethernet HDMI 2.0 USB 3.1 Type-C y Type-A para teclado y mouse
Protocolos:	TCP/IP, RTSP
Entradas:	Video + Internet: Switch PoE
Salidas:	Video: Pantalla LED

Tabla 5.3: Especificaciones de la NVIDIA Jetson.



Figura 5.10: Imagen tomada de [108] que muestra el contenido de la NVIDIA Jetson AGX Xavier Developer Kit.

Se eligió a la NVIDIA Jetson como unidad de procesamiento central ya que muestra un desempeño sobresaliente ejecutando redes neuronales profundas, además de que es un dispositivo enteramente dedicado y optimizado para este tipo de tareas. Como prueba de esto, en la Figura 5.11 se muestra una comparación de rendimiento entre dispositivos

similares (mini-computadoras dedicadas para *Machine Learning* e Inteligencia Artificial) ejecutando modelos de redes neuronales utilizando TensorFlow, en donde los dispositivos NVIDIA Jetson presentan un rendimiento muy alto.

Model	Application	Framework	NVIDIA Jetson Nano	Raspberry Pi 3	Raspberry Pi 3 + Intel Neural Compute Stick 2	Google Edge TPU Dev Board
ResNet-50 (224x224)	Classification	TensorFlow	36 FPS	1.4 FPS	16 FPS	DNR
MobileNet-v2 (300x300)	Classification	TensorFlow	64 FPS	2.5 FPS	30 FPS	130 FPS
SSD ResNet-18 (960x544)	Object Detection	TensorFlow	5 FPS	DNR	DNR	DNR
SSD ResNet-18 (480x272)	Object Detection	TensorFlow	16 FPS	DNR	DNR	DNR
SSD ResNet-18 (300x300)	Object Detection	TensorFlow	18 FPS	DNR	DNR	DNR
SSD Mobilenet-V2 (960x544)	Object Detection	TensorFlow	8 FPS	DNR	1.8 FPS	DNR
SSD Mobilenet-V2 (480x272)	Object Detection	TensorFlow	27 FPS	DNR	7 FPS	DNR
SSD Mobilenet-V2 (300x300)	Object Detection	TensorFlow	39 FPS	1 FPS	11 FPS	48 FPS

Figura 5.11: Tabla tomada de [110] que muestra los resultados de rendimiento ejecutando diversos modelos de SSD y Mobilenet con TensorFlow, posiblemente usando C++, en la Jetson Nano (un modelo de la misma familia Jetson pero inferior al utilizado en este trabajo), la Raspberry Pi 3, la Intel Neural Compute Stick 2 y la Google Edge TPU Coral Dev Board.

Finalmente, en la Tabla 5.4 se describen los principales atributos de interés de la pantalla LED a utilizar.

Pantalla LED	
Modelo:	Hisense Smart TV 50R6000FM
Características:	Resolución 4K 60Hz 49.5 pulgadas
Función:	Desplegar la publicidad personalizada, es decir, las recomendaciones obtenidas para el usuario en forma de realidad aumentada.
Puertos:	HDMI
Protocolos:	No aplica
Entradas:	Video: NVIDIA Jetson
Salidas:	No aplica

Tabla 5.4: Especificaciones de la pantalla LED a utilizar.

5.6.2. Sistema operativo JetPack y NVIDIA SDK Manager

NVIDIA JetPack es la solución más completa para crear aplicaciones de inteligencia artificial. Es un sistema operativo basado en Linux que además incluye bibliotecas, APIs, muestras, herramientas de desarrollo y documentación.

Proporciona el gestor de arranque, el kernel de Linux, los firmwares necesarios, los controladores NVIDIA y el sistema de archivos de muestra.

Con el uso de **Jetpack 4.3** de NVIDIA, la Jetson AGX Xavier es compatible con **Ubuntu 18.04, OpenCV, CUDA y cuDNN**; se pueden instalar librerías que usen la GPU como TensorFlow, PyTorch, Caffe, ROS, etc. Hace uso de **TensorRT**, que ayuda a compilar y optimizar redes neuronales [111].

Para la instalación de **Jetpack 4.3** en la NVIDIA Jetson AGX Xavier se hace uso de la herramienta **NVIDIA SDK Manager**, cuya interfaz es mostrada en la Figura 5.12, como se indica en la guía oficial en la web de NVIDIA [108] [112], además de ser ilustrado en la Figura 5.13.

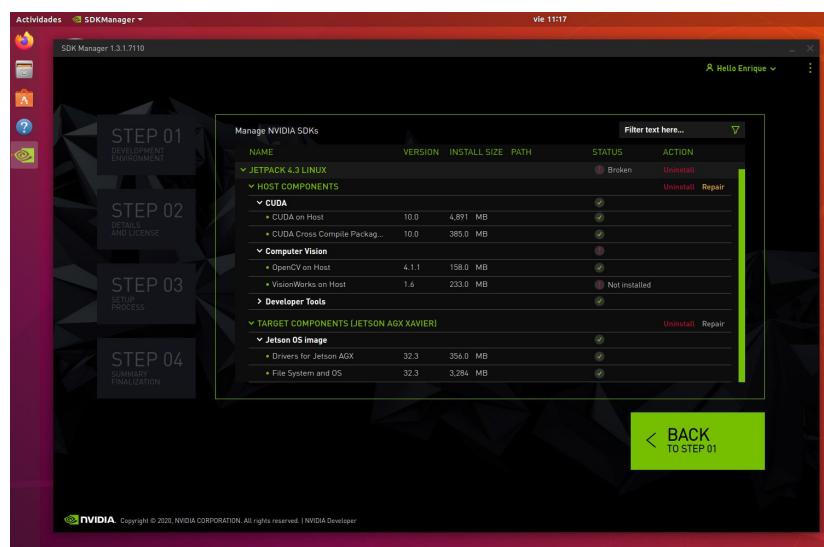


Figura 5.12: Interfaz gráfica de la herramienta NVIDIA SDK Manager para instalar Jetpack en la NVIDIA Jetson.

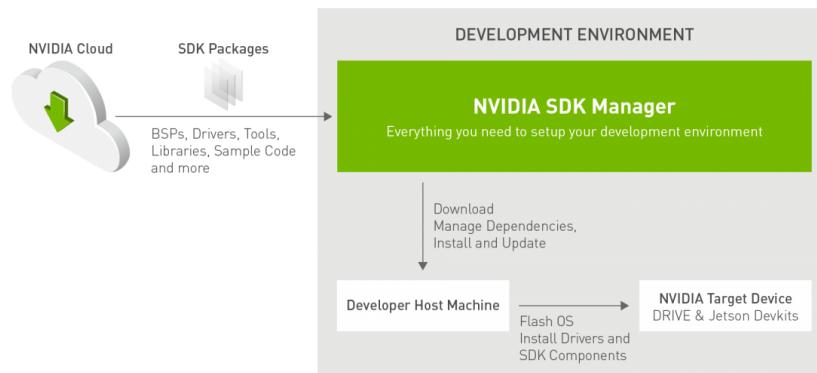


Figura 5.13: Diagrama tomado de [113] que muestra la instalación de Jetpack en la NVIDIA Jetson utilizando NVIDIA SDK Manager.

5.7. Herramientas de software a utilizar

En esta sección se abordan las principales herramientas de software, lenguajes de programación, APIs y bibliotecas que se emplean para el desarrollo e implementación de este proyecto; cada uno incluye una breve justificación de su elección, basándose principalmente en los trabajos revisados en el capítulo del [Estado del Arte](#).

- **Python 3:** El lenguaje de programación mayor usado para [Redes neuronales](#) es Python, el cuál es un lenguaje interpretado y multiparadigma [10]; esto último se aprovecha para utilizar una programación orientada a objetos. Los modelos de [Single Shot Detection](#) principalmente vienen en formato para ser leídos y ejecutados en este lenguaje, además de que es la base para la red de Big Five del trabajo en [51]. Por estas razones se selecciona este lenguaje de programación en su versión 3, además de ser fácil de utilizar y con muchas bibliotecas de utilidad, lo que agilizará los tiempos de desarrollo.
- **TensorFlow 1 y 2:** TensorFlow es una biblioteca de software de código abierto para el cálculo numérico por medio de operaciones matemáticas, que utiliza gráficos de flujo de datos como lo son las matrices de datos multidimensionales o tensores [114]. Esta es la biblioteca por excelencia para la programación de [Redes neuronales](#) de todo tipo. La versión 2 será utilizada para el entrenamiento de la red [Single Shot Detection](#) y la [Red neuronal convolucional para Edad y Género](#) (además de la creación de esta última), mientras que la versión 1 servirá como apoyo para su implementación en la NVIDIA Jetson (más no será la librería principal de ejecución, como normalmente se pensaría). La [5.14](#) Figura ilustra de forma muy conceptual el entrenamiento de una red neuronal en TensorFlow.

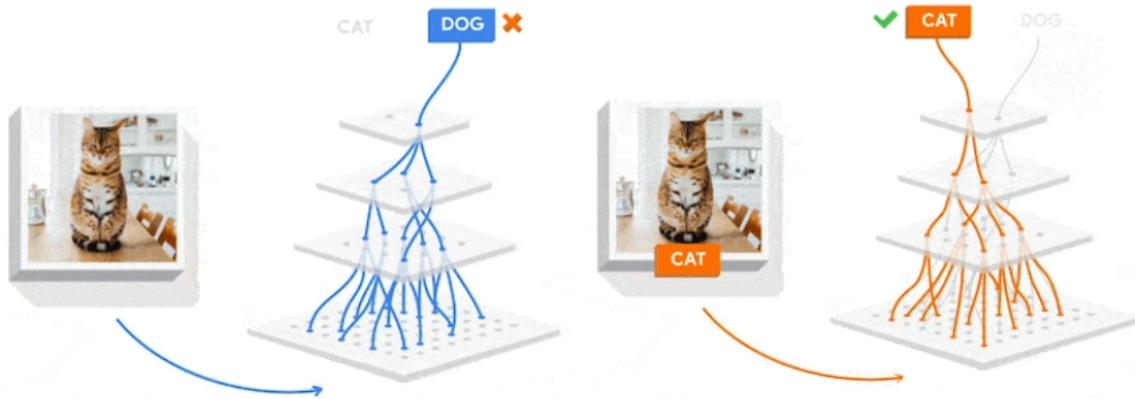


Figura 5.14: Imagen tomada de [114] que simula el entrenamiento de un modelo de red neuronal realizado por TensorFlow.

- **Google Colaboratory:** Colaboratory permite escribir y ejecutar código de Python en un navegador sin configuración requerida con acceso gratuito a GPUs. Utiliza documentos en una página web que no es estática, sino un entorno interactivo denominado notebook de Colab, que permite escribir y ejecutar código [115].

Permite importar conjuntos de datos de imágenes, entrenar un modelo de aprendizaje automático y evaluarlo. Los notebooks de Colab ejecutan código en los servidores alojados en la nube de Google, lo que significa que se aprovecha al máximo el hardware de Google, incluidas GPUs y TPUs remotas [115]. Entre los usos que se da a Colab en la comunidad de aprendizaje automático se encuentran: introducción a TensorFlow, desarrollo y entrenamiento de redes neuronales, experimentación con TPU [115], etc. Se decide utilizar Colaboratory para construir, diseñar, entrenar, evaluar y exportar los modelos de las redes neuronales en TensorFlow 2 que se necesitan para este trabajo.

- **MongoDB:** Es una base de datos distribuida que se basa en documentos (JSON); además ofrece una gran escalabilidad y flexibilidad [116]. La razón por la que se seleccionó a este gestor de base de datos es debido a que su flexibilidad al construir la base de datos y almacenar valores coincide con la filosofía de la metodología seleccionada, debido a que en caso de que se requiera de un cambio durante el desarrollo, será muy fácil implementarlo. Por otro lado, debido a que el giro del negocio es de productos alimenticios, la base de datos puede ser adaptada fácilmente a nuevos productos que sean distintos a los que se encuentran en el momento del desarrollo del sistema.
- **OpenCV:** OpenCV es una biblioteca de software de visión artificial y aprendizaje automático de código abierto [117]. La versión 4 viene incluida en la instalación de **Jetpack 4.3** (ver sección [Sistema operativo JetPack y NVIDIA SDK Manager](#)), por lo que módulos como el de [Captura de video a imagen](#), además de todos los que involucran procesamiento de imágenes, toman ventaja de sus métodos. Incluye operaciones como: decodificación de flujo de video, redimensión y recorte de imágenes, aplicación de filtros, colocación de máscaras, etc.
- **TensorRT:** NVIDIA TensorRT es un kit de desarrollo de software que facilita la inferencia de alto rendimiento en GPUs de NVIDIA, diseñado para funcionar de manera complementaria con frameworks de entrenamiento como TensorFlow. Se enfoca específicamente en ejecutar una red ya entrenada de manera rápida y eficiente en una GPU con el propósito de generar un resultado (un proceso que se puede denominar como puntuación, detección, regresión o inferencia) [118]. Las aplicaciones basadas en TensorRT funcionan hasta 40 veces más rápido que aquellas que solo utilizan el CPU durante la inferencia [119].

Según los [Requerimientos funcionales](#), la generación de publicidad dirigida debe ser en un lapso muy corto de tiempo, por lo que se opta por implementar todos los modelos de redes neuronales en esta librería en vez de TensorFlow. TensorRT optimiza redes neuronales combinando capas para mejorar la latencia, el rendimiento, la eficiencia energética y el consumo de memoria [118]. NVIDIA recomienda utilizar TensorRT en modelos de redes neuronales que están destinados a ponerse en producción [119], justamente como lo plantean los objetivos de este trabajo. La Figura 5.15 muestra la migración de TensorFlow a TensorRT.

La versión 6 viene incluida en la instalación de **Jetpack 4.3**, por lo que se aprovechan los motores de inferencia que hacen uso de la GPU de la NVIDIA Jetson [119] para reducir tiempos de propagación en las redes neuronales, y también para evitar comprometer el rendimiento y la fluidez del sistema.

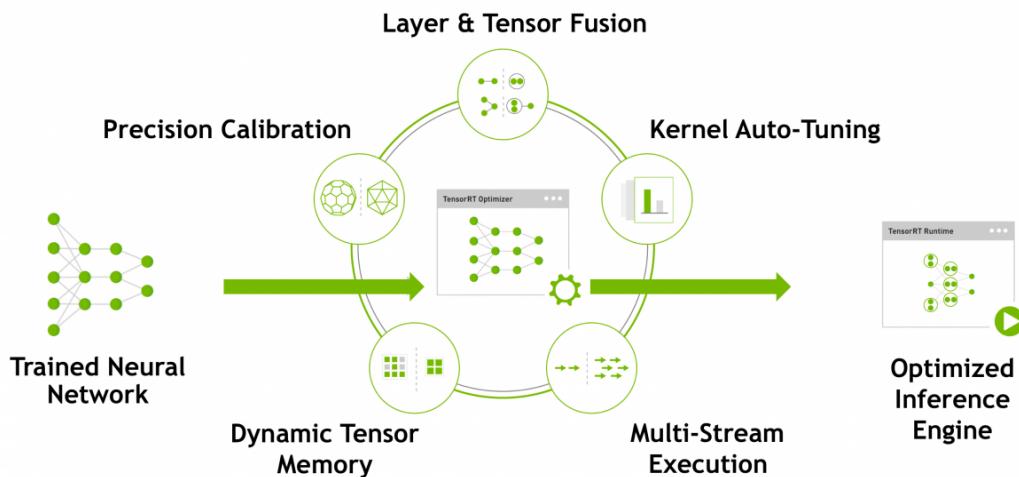


Figura 5.15: Imagen tomada de [119] que ilustra el proceso de conversión de un modelo de red neuronal entrenado en TensorFlow a un motor optimizado de inferencia de TensorRT.

- **NumPy:** NumPy es un proyecto de código abierto que tiene como objetivo permitir la computación numérica con Python [120]; utilizan arreglos basados en los del lenguaje de programación C. Debido a que las [Redes neuronales](#) necesitan una representación numérica de los datos que van a manejar, se usan las bondades que esta biblioteca dispone para convertir las imágenes a vectores o matrices (tensores) que TensorFlow y TensorRT pueda entender y procesar. Posee la capacidad de normalizarlas y manejarlas con distintos tipos de datos.
- **API OWM:** OpenWeatherMap ofrece una API que permite obtener datos meteorológicos actuales como la temperatura [121]. En el caso de este proyecto, la temperatura es utilizada para generar la recomendación de una modalidad de bebida adecuada (Iced, Frozen o Zen) en función de la temperatura ambiental. Para poder consumir esta API se debe registrar en la página web de Open Weather Map y generar una llave que, posteriormente será usada para realizar las peticiones de los datos con el uso de la librería para Python ("pyowm"). Es importante destacar que el plan usado es gratuito, por lo que tiene limitaciones como un número máximo de peticiones por día y mes.

5.8. Diagrama general del sistema

El sistema propuesto está compuesto por redes neuronales profundas que se encargan de analizar características independientes del usuario por medio de su rostro, además de obtener la temperatura ambiental, para poder ser ingresadas a un sistema recomendador de bebidas y desplegar la recomendación obtenida a través de un módulo encargado de agregar la realidad aumentada a la salida.

En la Figura 5.16 se representan con bloques esta funcionalidad, que es muy importante mencionar que toda será realizada dentro de la NVIDIA Jetson (ver sección [Especificación del hardware](#), en donde se hablan de las especificaciones de este equipo). Como se puede observar, hay tres tipos de bloques: los de color verde representan componentes externos, cuya implementación fue utilizada para fines propios, pero no es de nuestra autoría; los de color rojo, que representan acciones que se realizan por medio de protocolos de comunicación entre componentes de hardware (ver sección [Especificación del hardware](#)); los de color azul, que son los módulos que se implementaron en este trabajo; y finalmente los de color naranja, que son los requerimientos funcionales cumplidos por estos módulos.

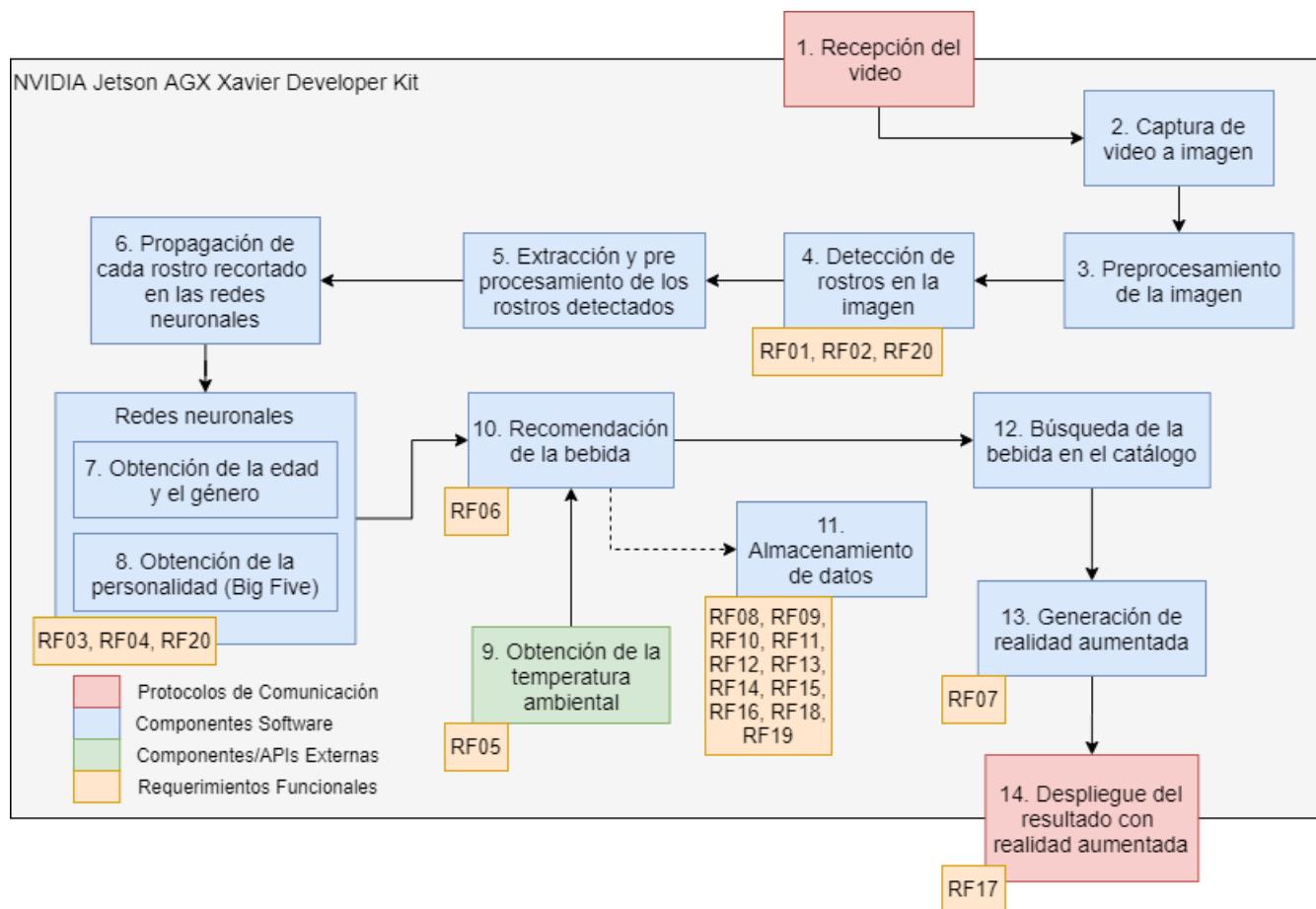


Figura 5.16: Diagrama global del sistema.

A continuación, se hace una descripción de cada uno de los módulos que aparecen en el diagrama de la Figura 5.16.

1. **Recepción del video:** El video del exterior de la tienda es enviado desde la cámara hasta la NVIDIA Jetson, de forma continua, para su procesamiento. Ver sección del diseño de la [Captura de video a imagen](#).

2. **Captura de video a imagen:** Una vez obtenido el vídeo en la NVIDIA Jetson desde la cámara, es necesario extraer frame por frame del mismo, ya que estos son la base de todo el procesamiento. Ver subsección de la [Captura del flujo de video y extracción de frames](#).
3. **Preprocesamiento de la imagen:** Los frames obtenidos deben ser procesados antes de ingresarlos a la detección de rostros. Aquí se incluyen acciones como el ajuste en la resolución de la imagen, redimensión de su tamaño, etc. Ver sección del diseño del [Preprocesamiento de la imagen](#).
4. **Detección de rostros en la imagen:** Por medio de la red neuronal SSD + MobileNet, el sistema identifica los rostros humanos en cada imagen, estableciendo las coordenadas dentro de esta que delimiten el área en donde el o los rostros estén ubicados. Ver sección del diseño de la [Detección de rostros en la imagen](#).
5. **Extracción y preprocesamiento de los rostros:** Ya que se conoce la ubicación de los rostros en la imagen, estos son recortados y son preprocesados para poder ser analizados y propagados en redes neuronales convolucionales, de forma independiente. Ver subsección del [Recorte y procesamiento de los rostros detectados](#).
6. **Propagación de cada rostro recortado en las redes neuronales:** Cada uno de los rostros obtenidos en el bloque anterior será propagado a través de varias redes neuronales convolucionales para estimar las características de: edad, género y personalidad a partir de este, en ese orden. Ver secciones del diseño de la [Red neuronal convolucional para Edad y Género](#) y de la [Red neuronal convolucional para Big Five](#).
7. **Obtención de la edad y el género:** Utilizando una sola red neuronal convolucional, se obtienen las estimaciones de edad y género a partir de un rostro. Ver sección del diseño de la [Red neuronal convolucional para Edad y Género](#).
8. **Obtención de la personalidad (Big Five):** Por medio de la red neuronal convolucional de personalidad, se obtiene el resultado de Big Five a partir de un rostro. Ver sección del diseño de la [Red neuronal convolucional para Big Five](#) y la [Discusión del capítulo del Estado del Arte](#).
9. **Obtención de la temperatura ambiental:** Se obtiene la temperatura ambiental utilizando una API externa. Ver sección de las [Herramientas de software a utilizar](#) y del diseño del [Recomendador del sistema](#).
10. **Recomendación de la bebida:** Los resultados de las estimaciones de edad, género, personalidad, y la temperatura ambiental son ingresadas a un recomendador, que ofrecerá una bebida al cliente. Ver sección del diseño del [Recomendador del sistema](#).
11. **Almacenamiento de datos:** Una vez obtenida la recomendación de bebida, esta es almacenada en la base de datos junto a la imagen del rostro, el resultado de la estimación de edad, género, personalidad, y la medida de temperatura ambiental. Ver sección del diseño de la [Base de datos para el sistema](#).
12. **Búsqueda de la bebida en el catálogo:** Por medio de un identificador, la bebida obtenida por el recomendador se buscará en un catálogo de diseños previamente construido. Ver sección del diseño del [Recomendador del sistema](#) y de la [Base de datos para el sistema](#).
13. **Generación de la realidad aumentada:** El diseño de la bebida obtenida es incrustado al del rostro de la persona, junto con otros datos. Ver sección del diseño de la [Generación de realidad aumentada](#).
14. **Despliegue del resultado con realidad aumentada:** Finalmente la imagen con la recomendación de la bebida obtenida en forma de realidad aumentada es desplegada en la pantalla de publicidad de Bubble Town®. Es importante mencionar que periódicamente se irá refrescando la salida en pantalla con un frame nuevo (obtenido desde el video de la cámara). Además, este bloque sobresale de la NVIDIA Jetson porque también tiene que ver con la pantalla LED. Ver sección del diseño de la [Generación de realidad aumentada](#).

5.9. Captura de video a imagen

Esta sección está destinada al cumplimiento del bloque **1. Recepción del video** y **2. Captura de video a imagen** del [Diagrama general del sistema](#), representado por la Figura 5.16.

En la Figura 5.17 se muestra el diagrama correspondiente al diseño del sistema de captura de video desde la cámara de red en la NVIDIA Jetson, cuyo objetivo es la extracción de frames o imágenes desde este para su procesamiento y despliegue continuo en una pantalla. Como se puede observar, existen cuatro tipos de componentes: los que están en un cuadro verde corresponden a librerías externas, software de terceros y lenguajes de programación necesarios para lograr el objetivo del diseño; los que están en cuadros grises corresponden a los componentes de hardware que van a participar en este diseño, siendo la cámara de red y la NVIDIA Jetson, junto a los protocolos de comunicación que usan entre ellos (ver sección [Especificación del hardware](#)). Y finalmente, se encuentran un par de cuadros azules dentro de otro más grande en amarillo; estos en especial representan los bloques y módulos que se implementan por cuenta propia en este diseño del sistema.

Los datos de entrada del diseño son los siguientes:

- El flujo de video codificado capturado por la cámara.
- La dirección IP de la cámara.
- La dirección IP de la NVIDIA Jetson.
- El puerto de la cámara.
- El usuario para la cámara.
- La contraseña para la cámara.

Los datos de salida son frames del mismo video pero decodificado, listos para ser procesados o manipulados.

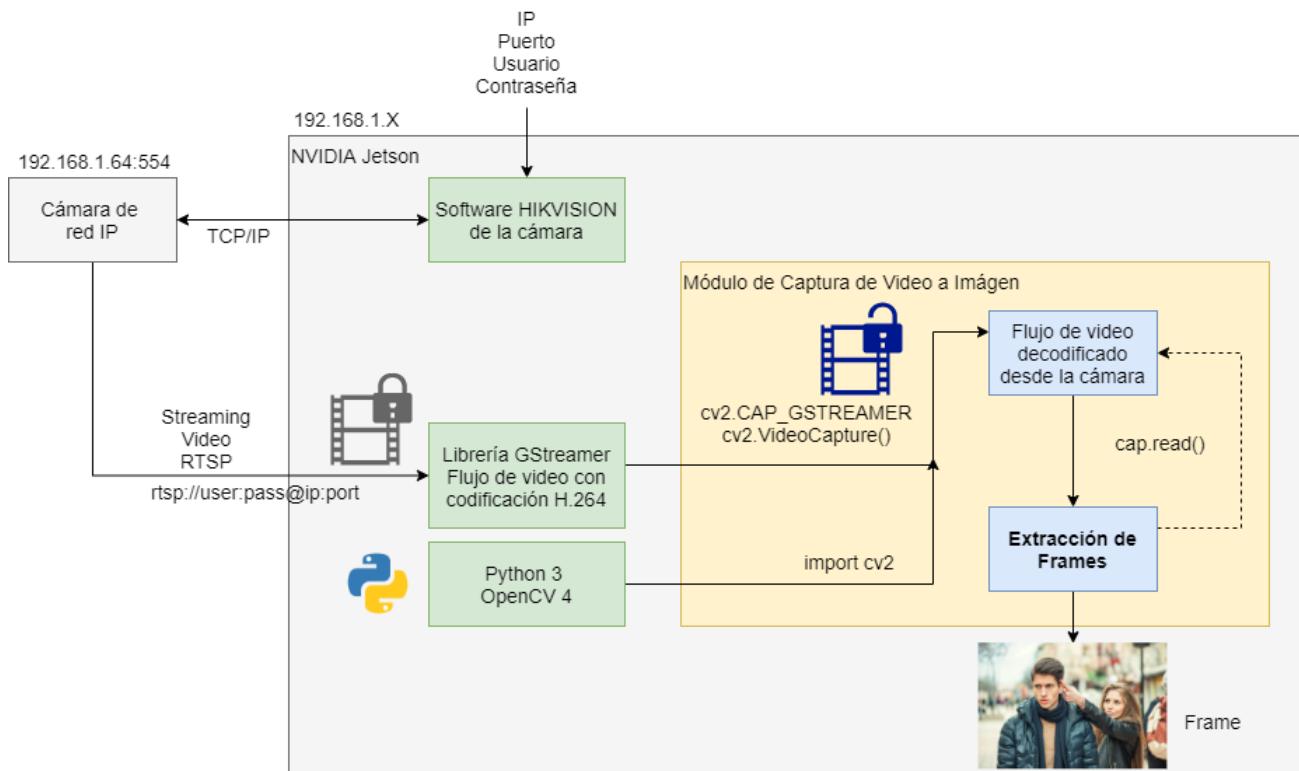


Figura 5.17: Diseño del sistema de captura de video a imágenes (frames)

A lo largo de esta sección se explican y detallan cada uno de los bloques de dicho diagrama, su funcionamiento, sus configuraciones y los protocolos necesarios.

En primer lugar, para lograr este propósito, previamente es necesario haber cumplido con lo siguiente:

1. Todos los componentes de hardware deben estar conectados con sus correspondientes cables y puertos (ver sección [Especificación del hardware](#)).
2. Jetpack 4.3 debe estar instalado y funcionando correctamente en la NVIDIA Jetson [122] (ver subsección [Sistema operativo JetPack y NVIDIA SDK Manager](#)).
3. La NVIDIA Jetson y la cámara de red deben tener direcciones IP que pertenezcan a la misma subred, como se indica en [107].

Teniendo lo anterior logrado, el siguiente paso es conseguir que la NVIDIA Jetson reconozca a la cámara de red y que por medio de esta se pueda visualizar el video que la cámara este capturando en tiempo real, es decir, que exista comunicación entre ambos dispositivos, por medio del protocolo TCP/IP (ver Figura 5.17). A continuación se describe dicho proceso.

5.9.1. Establecer comunicación entre la cámara de red y la NVIDIA Jetson

Esta configuración se llevó a cabo siguiendo el procedimiento proporcionado por el manual de usuario de la cámara de red HIKVISION [107].

Utilizando el cliente web de HIKVISION [107], el cual está disponible ingresando la dirección IP por defecto de la cámara de red (192.168.1.64) en la barra de búsqueda de cualquier navegador web instalado en la NVIDIA Jetson, se debe activar y configurar la cámara de red por medio de la inicialización de los siguientes parámetros:

- **Dirección IP:** Por defecto es **192.168.1.64/24**
- **Usuario:** Por defecto es **admin**
- **Puerto:** Por defecto es **554** para el protocolo RTSP
- **Contraseña**

La Figura 5.18 muestra la interfaz del login al cliente web de HIKVISION [107]. La Figura 5.19 muestra la interfaz de activación de la cámara de red, que es donde se debe realizar la asignación del usuario y la contraseña. Y por último, la Figura 5.20 muestra la interfaz de modificación de los parámetros de red, en donde se deben inicializar o modificar (según sea el caso de la dirección IP seleccionada para la NVIDIA Jetson) la dirección IP y el puerto de la cámara de red, sin olvidar el requisito de que ambos dispositivos deben pertenecer a la misma red.



Figura 5.18: Imagen tomada de [107] que muestra la interfaz del login al cliente web de HIKVISION en un navegador web.

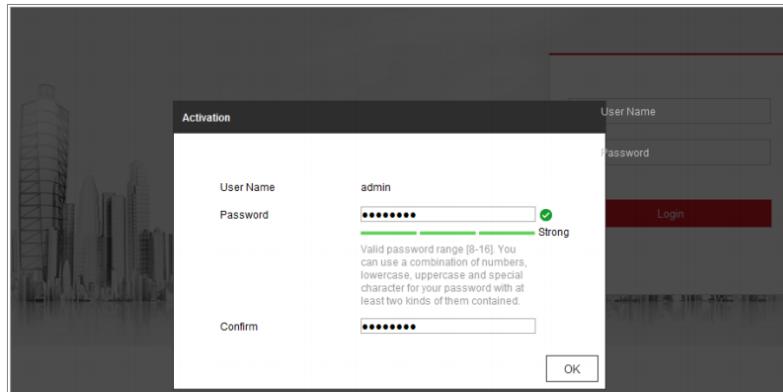


Figura 5.19: Imagen tomada de [107] que muestra la interfaz de activación de la cámara de red (usuario y contraseña) en el cliente web de HIKVISION.

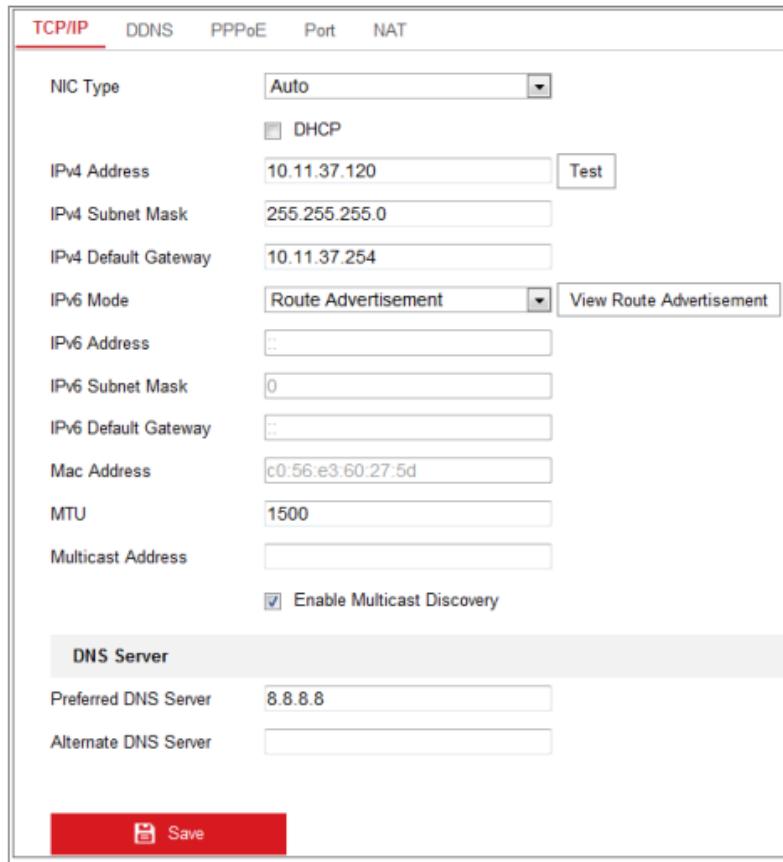


Figura 5.20: Imagen tomada de [107] que muestra la interfaz de la modificación de los parámetros de red de la cámara de red (dirección IP y puerto) en el cliente web de HIKVISION.

Para verificar que existe comunicación entre la cámara de red y la NVIDIA Jetson, se debe de visualizar el video

capturado en tiempo real por la cámara de red en el cliente web de HIKVISION [107], por medio de la interfaz mostrada en la Figura 5.21.

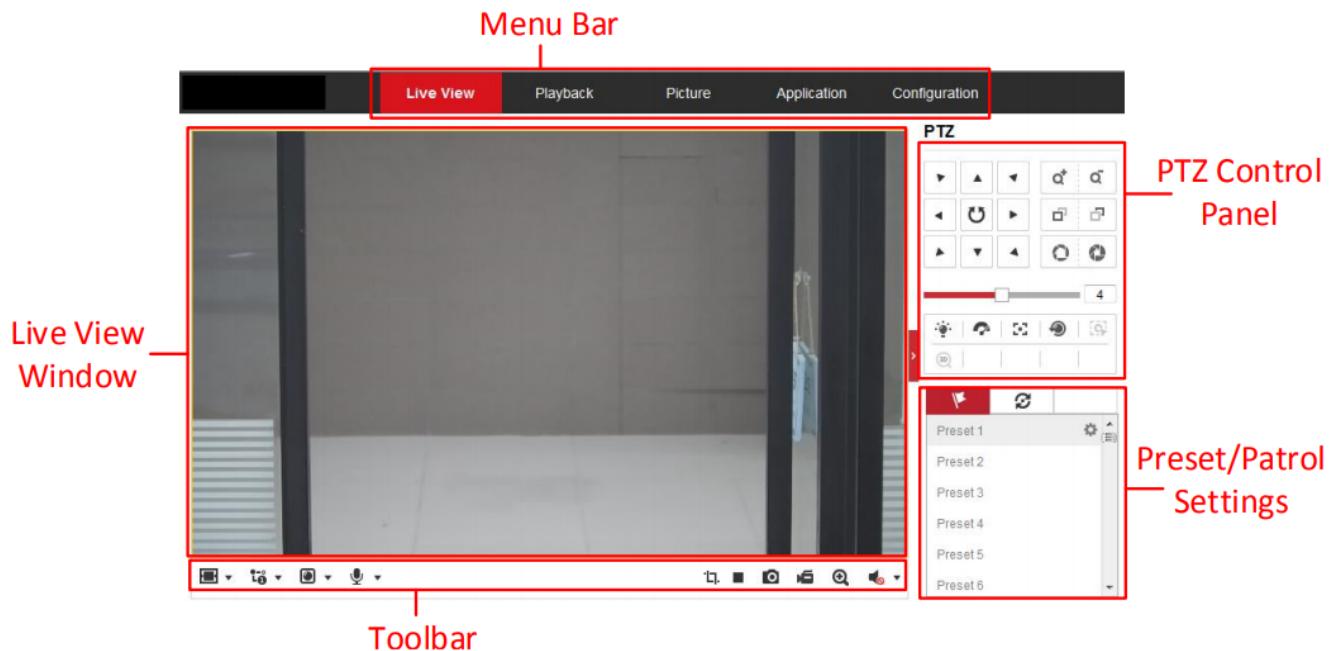


Figura 5.21: Imagen tomada de [107] que muestra la interfaz de la visualización del video en tiempo real desde la cámara de red en el cliente web de HIKVISION.

Una vez configurados estos cuatro parámetros de la cámara de red y habiendo logrado visualizar el video de la cámara en la NVIDIA Jetson en tiempo real, el siguiente paso es capturar dicho video, pues actualmente la NVIDIA Jetson solo es un medio para desplegarlo y visualizarlo únicamente.

Para esto, el video transmitido desde la cámara hacia la NVIDIA Jetson debe ser leído con ayuda del protocolo RTSP, para posteriormente decodificar su flujo por medio de la librería GStreamer, y así poder manipularlo, procesarlo y extraer sus frames en Python, apoyándose adicionalmente de la librería OpenCV (ver Figura 5.17). A continuación se describe dicho proceso.

5.9.2. Captura del flujo de video y extracción de frames

El **Protocolo de Transmisión en Tiempo Real (RTSP, Real Time Streaming Protocol)** es utilizado para obtener video en tiempo real desde la cámara de red en la NVIDIA Jetson, específicamente para ser procesado por medio de Python con OpenCV (ver Figura 5.17) [123] [124]. Es importante conocer la URI del protocolo relacionada a la cámara de red, que tiene el siguiente formato: **rtsp://user:password@ip:puerto**.

Para la cámara seleccionada, utilizando los parámetros por defecto, la URI sería:

rtsp://admin:XXXX@192.168.1.64:8000

JetPack 4.3 posee la **librería GStreamer**, que contiene el elemento **h264parse** necesario para decodificar el **flujo H.264 RTSP** del video de la cámara de red (ver Figura 5.17) [125] [124].

Además, JetPack 4.3 también posee la librería **OpenCV 4**, capaz de capturar vídeo desde cámaras externas sin necesidad de configurar o instalar paquetes adicionales [122].

El flujo de video ya decodificado es obtenido por medio de la URI del protocolo RTSP de la cámara de red y la librería GStreamer con ayuda de métodos de OpenCV (ver Figura 5.17); una vez capturado el flujo de video de la cámara de red, se va leyendo cada frame o imagen con color RGB y formato JPG dentro de un ciclo infinito, para así tenerlos finalmente listos para su preprocesamiento (ver sección [Preprocesamiento de la imagen](#)) [124].

El framerate máximo desplegado es el mismo que el nativo de la cámara de red: **30 fps** [105], aunque por medio de métodos de OpenCV se puede disminuir.

Para desplegar el video capturado desde la cámara de red en una ventana dentro del sistema operativo de la NVIDIA Jetson y ver que todo funcione debidamente, se puede hacer uso de métodos de OpenCV, además de ajustar la resolución o desplegarse en pantalla completa.

5.10. Preprocesamiento de la imagen

Esta sección está destinada al cumplimiento del bloque **3. Preprocesamiento de la imagen** del [Diagrama general del sistema](#), representado por la Figura 5.16.

Basándose en lo que indica la documentación de TensorRT [118], y sobretodo el tutorial oficial de NVIDIA que indica como ejecutar el modelo de la [Single Shot Detection](#) en esta librería [126], es necesario realizar cuatro operaciones principales de preprocesamiento en el frame obtenido por el diseño anterior (ver subsección [Captura del flujo de video y extracción de frames](#)):

1. **Redimensionar la imagen:** Originalmente el modelo de la [Single Shot Detection](#) únicamente acepta como datos de entrada imágenes de una **resolución de 300x300 píxeles** [126], pero posteriormente se hizo el ajuste para que esta admita imágenes de resoluciones dinámicas en TensorFlow [127]. Sin embargo, para ser utilizada en TensorRT es necesario cumplir esta condición, por lo que los frames provenientes de la cámara se deben redimensionar.
2. **Normalizar la imagen:** Es necesario convertir el frame a un arreglo NumPy y después normalizarlo (dividir entre 255, el valor máximo de un píxel en formato de colores RGB) ya que de lo contrario, los datos serían muy lejanos unos de otros y la [Single Shot Detection](#) no funcionaría.
3. **Subir ligeramente el brillo:** El tutorial de NVIDIA en [126] sugiere aumentar ligeramente el brillo de las imágenes para obtener mejores detecciones. Por este motivo, se decide hacerlo multiplicando el arreglo NumPy de la imagen por 2.
4. **Aplicar la transpuesta a la imagen:** La [Single Shot Detection](#) en TensorRT requiere que esta operación sea aplicada en la imagen de entrada como requisito, tal como indica el tutorial de NVIDIA [126].

Algunas otras de las condiciones de preprocesamiento que requiere la [Single Shot Detection](#) son que los arreglos de NumPy sean de tipo float32, que el color sea RGB (3 canales) y que su formato sea JPG; estas condiciones ya las cumple el frame "crudo" obtenido de la cámara [105]. La Figura 5.30 muestra parte de este preprocesamiento en la imagen antes de ser ingresada a la [Single Shot Detection](#).

Es importante mencionar que se recomienda mantener una copia del frame original de la cámara para aplicar la realidad aumentada sobre él y desplegarlo (ver sección [Generación de realidad aumentada](#)).

Posterior a la aplicación de este preprocesamiento, la imagen está lista para ingresar al modelo de [Single Shot Detection](#) y detectar rostros en ella. De esto se habla en la siguiente sección a continuación.

5.11. Detección de rostros en la imagen

Esta sección está destinada al cumplimiento del bloque **4. Detección de rostros en la imagen y 5. Extracción y preprocesamiento de los rostros** del [Diagrama general del sistema](#), representado por la Figura 5.16.

Como se abordó en el capítulo [Marco teórico](#), la red neuronal [Single Shot Detection](#) otorga grandes beneficios, como la capacidad de la detección de objetos a distintas escalas y resoluciones, además de realizarlo a una gran velocidad. Esto encaja perfectamente a las necesidades del proyecto, ya que se necesita de un tiempo de respuesta rápido y la capacidad de detectar rostros a distintas distancias de la cámara. Por ello se utiliza una [Single Shot Detection](#) usando como base una red neuronal [MobileNet](#) para tener un tiempo de procesamiento aún más bajo.

Es necesario realizar un nuevo entrenamiento personalizado para **adecuar la detección a solamente rostros humanos**, utilizando el API [Tensorflow Object Detection](#) [127] que ya contiene la red neuronal pre-entrenada para clasificar 80 tipos de clases, y apoyarse aplicando [Transferencia de aprendizaje](#). Por lo tanto, es turno de encontrar los datos que sean útiles para dicho entrenamiento.

5.11.1. Preprocesamiento del dataset para el entrenamiento de la SSD

Para entrenar la red neuronal [Single Shot Detection](#) se necesitan imágenes con las siguientes características, según los requerimientos del API [127]:

- Formato en JPG o JPEG.
- Color en RGB (hace referencia a los 3 colores básicos en forma de canales de la imagen: Red, Green y Blue).

Las etiquetas de cada imagen deben ser las siguientes (una etiqueta por rostro):

- Nombre del archivo.
- Clase (para fines del proyecto solo será "rostro").
- Coordenadas de la cara (el punto superior izquierdo y el inferior derecho. Ver la Figura 5.22).
- Ancho de la imagen en píxeles.
- Largo de la imagen en píxeles.

Se utilizan dos datasets diferentes. el primero de ellos es *Face Detection in Images* encontrado en [128] con 409 imágenes de las cuales:

- 26 son un formato distinto a JPG o JPEG.
- 1 no puede ser abierta correctamente.

El segundo es *Google Facial Expression Comparison Dataset* tomado de [129] del cual se utilizaron 8,335 imágenes, siendo 2 de ellas corruptas. Para este dataset fue necesario convertir las imágenes de formato PNG a JPG al ser una cantidad considerable. En la Tabla 5.5 se aprecia el número de imágenes ocupadas de cada dataset, dando un total de **8,715 imágenes**.

En la Figura 5.22 se representa un ejemplo de como se presentan las imágenes en los datasets: se encierra en un cuadro rojo el área de la cara.

Dataset	Núm. Imágenes ocupadas
Face Detection in Images [128]	382
Google Facial Expression Comparison Dataset [129]	8,333
Total	8,715

Tabla 5.5: Datasets ocupados para entrenar la SSD.



Figura 5.22: Ejemplo de una imagen tomada del dataset *Google Facial Expression Comparison Dataset* [129].

5.11.2. Configuración del API

Una vez teniendo los datasets completamente preprocesados, se mezclan para formar uno solo. Ahora se deben crear los archivos de entrada *TFRecords*, que es un formato de archivo usado por TensorFlow. Estos archivos contendrán los datos del dataset y serán dos: para entrenamiento y para prueba. El dataset se divide de la siguiente forma:

- Entrenamiento: 70 %
- Prueba: 30 %

Lo siguiente a hacer es la creación de un archivo *label map*. *Label Map* es un archivo de formato *pbtxt* que contiene las clases de objetos que identificará. En su contenido se coloca un identificador numérico de la clase y su etiqueta correspondiente. El formato es el siguiente:

```
item
{
  id:1,
  name:'Face'
```

Para aplicar [Transferencia de aprendizaje](#), primero se descargan los pesos/puntos de control del modelo previamente entrenado (tendrán el prefijo "checkpoint" como nombre), y después se configurarán los siguientes parámetros:

- La ruta del punto de control del modelo previamente entrenado (checkpoint).

- La ruta a los dos archivos *TFRecord*.
- Ruta al archivo *label_map.pbtxt*.
- Tamaño del lote de entrenamiento (batch)
- Cantidad de pasos de entrenamiento (épocas).
- Cantidad de tipos de clases.

5.11.3. Entrenamiento

Para el entrenamiento de la [Single Shot Detection](#) se comenzó utilizando solo el dataset *Face Detection in Images* [128], pues se tuvo la hipótesis de que este bastaría para obtener buenos resultados. La Figura 5.24 muestra un ejemplo de una prueba detección de rostros con un entrenamiento de **20,000 épocas**, en la cual se aprecia que hay una detección correcta y un falso positivo, es decir, se identifica el brazo de la persona como un rostro humano.

Por lo tanto, se continuó el entrenamiento hasta llegar a las **100,000 épocas** y debido a que las detecciones y el error no mejoraba, se decidió incluir el dataset *Google Facial Expression Comparison Dataset* [129]. Para no comenzar de cero el entrenamiento, se retoma el entrenamiento a partir de las 100,000 épocas y se entrena durante 30,000 épocas más. A lo largo de estas generaciones de entrenamiento se aprecia que se eliminaron los errores encontrados y el error disminuye drásticamente. La Figura 5.23 representa el entrenamiento de la red neuronal desde la época 1 hasta la 130,000.

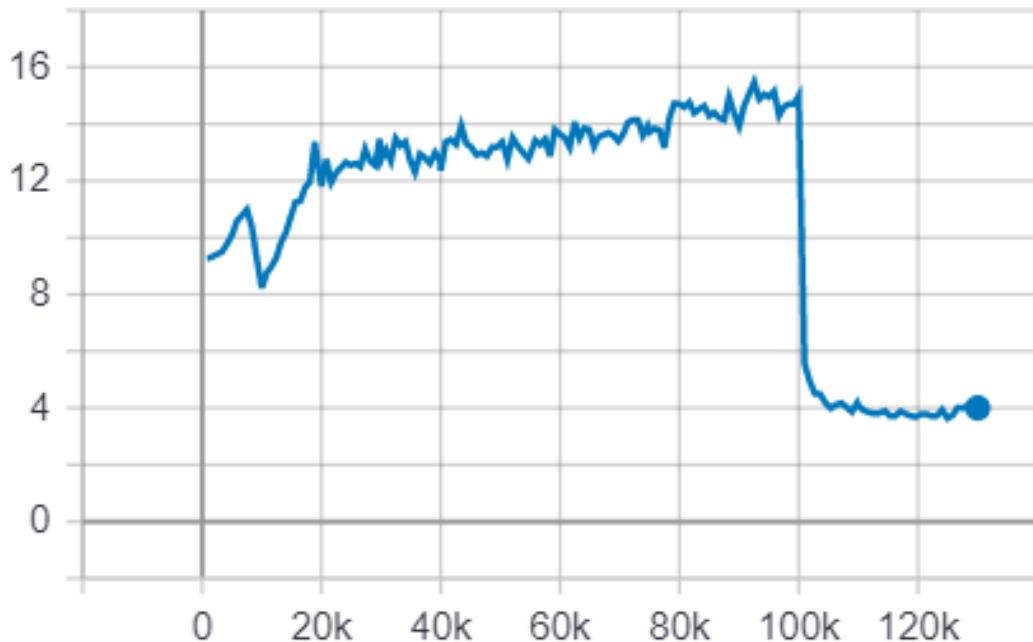


Figura 5.23: Función del error a lo largo del entrenamiento de la [Single Shot Detection](#).

En la Figura 5.25 se observa el problema de Figura 5.24 solucionado.

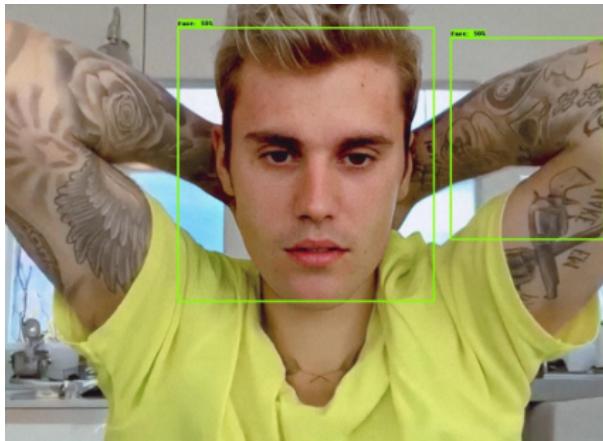


Figura 5.24: Prueba de detección con 20,000 épocas de entrenamiento. Imagen tomada de Google Imágenes.

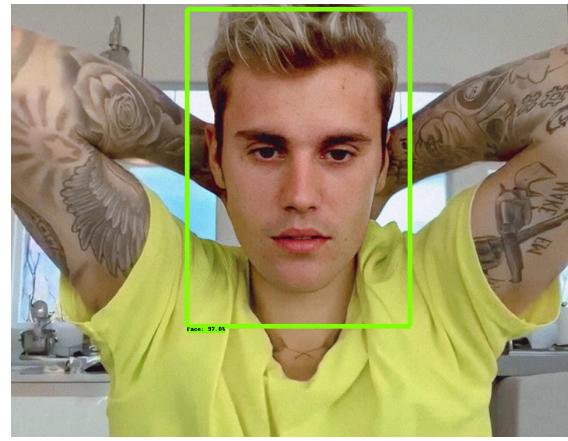


Figura 5.25: Prueba de detección con 130,000 épocas de entrenamiento. Imagen tomada de Google Imágenes.

5.11.4. Pruebas de detección de rostros

Se realizan pruebas en las cuales se evalúa que tan acertada es la red reconociendo los rostros en una imagen y encontrando las coordenadas correctas. Para esto se utiliza la métrica "Intersección sobre la Unión" (IoU), por sus siglas en inglés (Intersection over Union), se recomienda revisar la Figura 2.26.

Bajo esta métrica se consideran los siguientes tres posibles escenarios para construir la matriz de confusión:

1. **Verdadero positivo:** es el caso donde la red detecta un rostro y este verdaderamente es un rostro.
2. **Falso negativo:** ocurre cuando la red detecta un rostro pero en realidad no es un rostro.
3. **Falso positivo:** es el caso contrario contrario al falso negativo, se presenta cuando la red no detecta un rostro que debería de hacerlo.

En la Tabla 5.6 se muestran los resultados de la matriz de confusión antes mencionada.

	Positivo	Negativo
Verdadero	6818	-
Falso	830	2617

Tabla 5.6: Matriz de confusión de la detección de rostros.

La detección de rostros presenta fallas fallas a la hora de reconocer muchas caras dentro de la escena, esto es debido en gran parte porque la mayoría del dataset presenta únicamente a una persona por imagen. La Figura 5.26 refleja la detección correcta de todas las caras dentro de la imagen, mientras que la Figura 5.27 es un ejemplo de la detección parcial de los rostros.



Figura 5.26: Ejemplo de una detección correcta de rostros en la imagen.



Figura 5.27: Ejemplo de una detección incompleta de rostros en la imagen.

En ocasiones se logra una detección que se considera correcta, como en la Figura 5.28, pero debido a que el dataset muestra un etiquetado muy cerrado (recuadro negro de la Figura 5.28) y la red reconoce el rostro tomando coordenadas más extensas (recuadro azul de la Figura 5.28) se considera una detección errónea aunque realmente esta puede ser valida para el propósito del sistema.

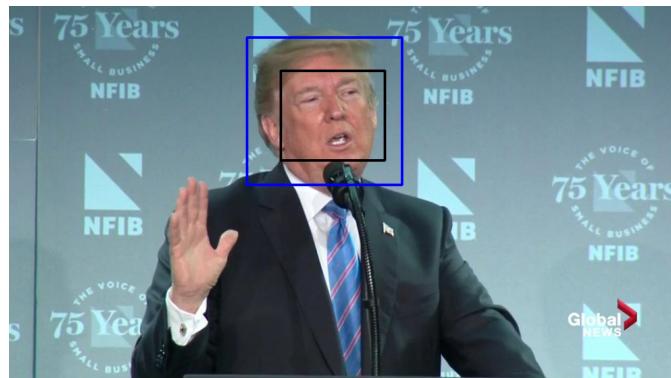


Figura 5.28: Ejemplo de una detección tomada como errónea en la prueba. El recuadro negro representa la etiqueta del dataset y el recuadro verde la detección de la red.

5.11.5. Implementación en TensorRT

Una vez terminado el entrenamiento de la [Single Shot Detection](#), el modelo es exportado, generando los siguientes archivos:

- checkpoint
- **frozen_inference_graph.pb**
- model.ckpt.data-00000-of-00001
- model.ckpt.index
- model.ckpt.meta
- pipeline.config
- saved_model/
- saved_model.pb
- variables

Según la documentación de TensorRT [118], para generar los motores optimizados de inferencia de algún modelo creado y entrenado por TensorFlow, debe de haber sido necesario utilizar la versión 1. Como se menciona en la sección de [Herramientas de software a utilizar](#), la [Single Shot Detection](#) se re-entrena en TensorFlow 2. Sin embargo, al momento de exportar el modelo también se genera un **frozen graph de TensorFlow 1**, por lo que este es seleccionado.

Como se muestra en la Figura 5.29, es necesario convertir el modelo frozen graph con formato protobuf (.pb) a un archivo de formato universal (.uff) [118]. Sin embargo, antes es necesario "reparar" el frozen graph del modelo, debido a que las capas de esta red neuronal poseen operaciones de TensorFlow que no son compatibles con el formato .uff que TensorRT necesita analizar [118] [126] [130].

Para lograr esto, primero se aplican una serie de métodos que solucionan incompatibilidades de operaciones en algunas capas del modelo, reemplazándolas por otras compatibles, o en el peor de los casos, eliminándolas y rehaciéndolas desde cero [126] [130]. Estas operaciones están disponibles y probadas en un script de configuración proporcionado por el ya mencionado tutorial oficial de NVIDIA en [126] y por otros no oficiales en [130] y [131], por lo que la funcionalidad del modelo no se ve alterada. Estos scripts están disponibles en el repositorio del sistema.

Después de corregir estos problemas de incompatibilidad entre operaciones, el modelo "arreglado" se convierte a formato .uff y es leído por TensorRT por medio de un analizador sintáctico que mapea el modelo a un archivo .bin (ver Figura 5.29), el cual corresponde a su motor optimizado de inferencia [118].

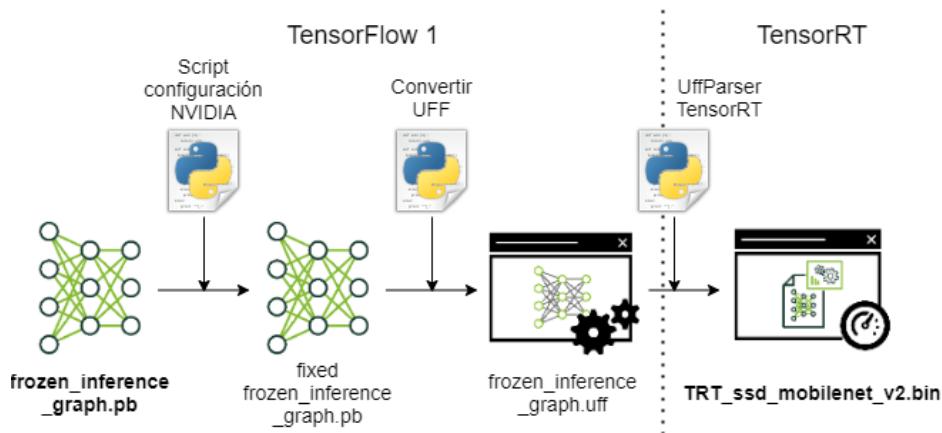


Figura 5.29: Diagrama que muestra la conversión del modelo [Single Shot Detection](#) de TensorFlow a TensorRT, basandose en la documentación oficial [118] y los tutoriales proporcionados en [126], [130] y [131].

Finalmente, solo queda leer este archivo .bin para que realice la detección de rostros en imágenes que recibe como entrada del video captado por la cámara de red preprocesadas (ver secciones [Captura de video a imagen](#) y

Preprocesamiento de la imagen). Al ejecutar una inferencia en un modelo de red neuronal utilizando TensorRT, esta envía los datos de entrada del CPU a la GPU de ida y vuelta por medio de buffers de comunicación de una dimensión [118]. De este modo, es necesario aplanar la imagen de tamaño 300x300x3 en un vector de tamaño 270000 para ser ingresado en estos buffers de comunicación. Posteriormente ya que los datos estén en la GPU, la imagen es reconstruida automáticamente por TensorRT e ingresa con sus dimensiones originales al modelo (antes de ser aplanada) [118]. La Figura 5.30 muestra el diagrama con las entradas y salidas del motor optimizado de inferencia del modelo de la [Single Shot Detection](#) cargado en la NVIDIA Jetson.

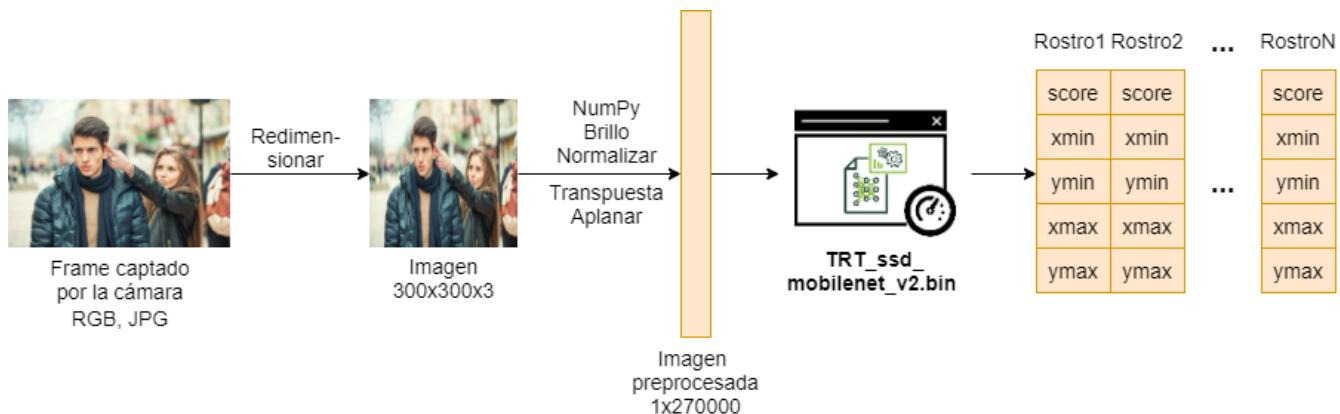


Figura 5.30: Diagrama que muestra el preprocesamiento del frame captado por la cámara (frame de ejemplo tomado de Google Imágenes) y la [Single Shot Detection](#) en TensorRT.

5.11.6. Recorte y procesamiento de los rostros detectados

Como se puede observar en la Figura 5.30, la salida del modelo de la [Single Shot Detection](#) son arreglos de distintas variables definidas por el API de TensorFlow [127] y por el script del tutorial de TensorRT NVIDIA [126].

Las variables cuyos arreglos son de interés para el objetivo de este diseño (cada posición en los arreglos corresponde a un posible rostro detectado) son: **score**, la precisión con la cual el modelo identifica un rostro; **xmin**, **ymin** corresponden a la coordenada superior izquierda del cuadro que posee el área de la cara (ver Figura 5.22), y **xmax**, **ymax** corresponden a la coordenada inferior derecha. Con ayuda de estas coordenadas, se recorta el rostro detectado de la imagen original cada vez que su precisión sea mayor al 80 % [127] [126] y cada rostro se guarda en una lista que será iterada.

Para cada rostro recortado es necesario aplicar su correspondiente preprocesamiento, el cual se ilustra en la Figura 5.31 y consiste de las siguientes operaciones:

- Aplicar formato selfie:** Tanto la [Red neuronal convolucional para Edad y Género](#) y la [Red neuronal convolucional para Big Five](#) [51] brindan mejores resultados si cada rostro posee un formato tipo selfie (ver subsección [Selfies](#) del capítulo del [Estado del Arte](#)), por lo que se opta por agregar bordes o margenes oscuros alrededor del rostro para que luzcan como tal.
- Redimensionar el rostro:** Ambas redes neuronales necesitan imágenes de una resolución de 208x208 píxeles, por lo que los rostros recortados en formato selfie se deben redimensionar.
- Normalizar el rostro:** Al igual que para la [Single Shot Detection](#), los rostros se convierten a arreglos NumPy de tipo float32, y después se deben normalizar para evitar resultados con un error muy alto. Como los rostros son recortados del frame original obtenido de la cámara (sin preprocesamiento), es necesario aplicar la normalización en ellos.

4. **Aplicar filtros de color:** La [Red neuronal convolucional para Edad y Género](#) requiere imágenes a color (RGB - 3 canales), y la [Red neuronal convolucional para Big Five](#) requiere imágenes en escala de grises (1 canal) [51], por lo que para la primera red la imagen entra tal cual esta, mientras que para la segunda red la imagen se pasa a escala de grises.

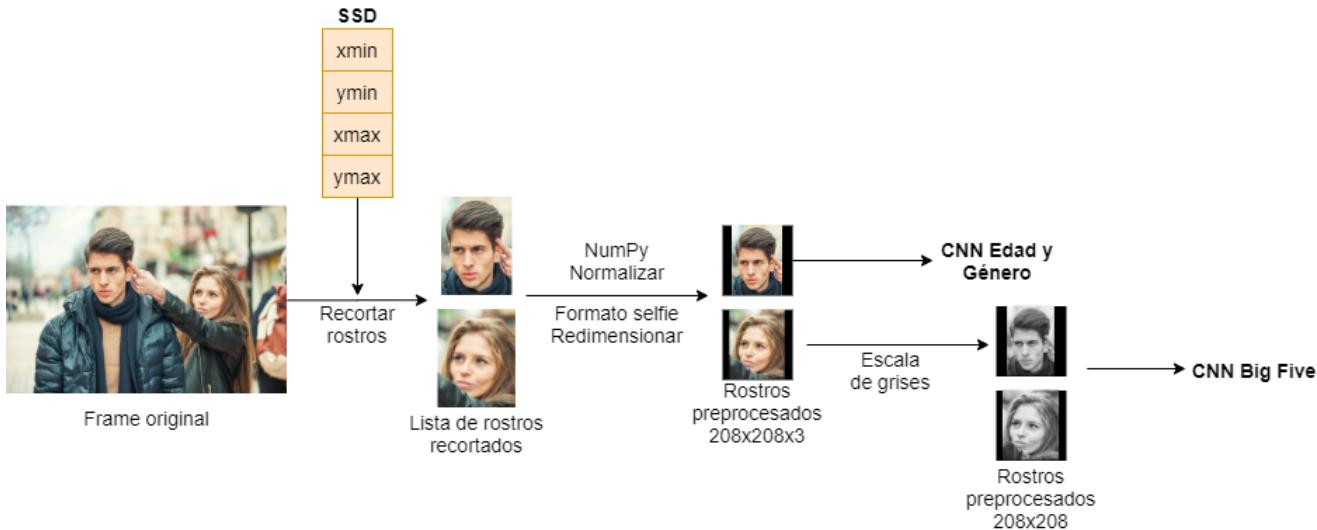


Figura 5.31: Diagrama que muestra la extracción de los rostros detectados por la [Single Shot Detection](#) de los frames capturados por la cámara de red, y su correspondiente procesamiento para ser ingresados a la [Red neuronal convolucional para Edad y Género](#) y la [Red neuronal convolucional para Big Five](#).

5.12. Red neuronal convolucional para Edad y Género

Esta sección está destinada al cumplimiento de la primera mitad del bloque **6. Propagación de cada rostro recortado en las redes neuronales** y del bloque **7. Obtención de la edad y el género** del [Diagrama general del sistema](#), representado por la Figura 5.16.

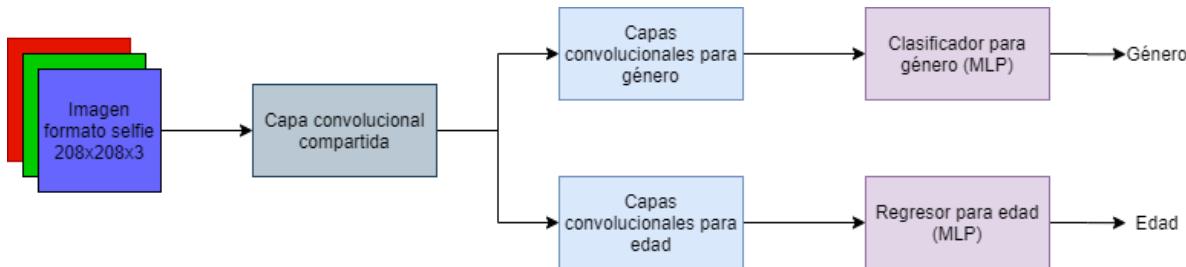


Figura 5.32: Arquitectura general de red neuronal convolucional para obtener edad y género a partir de una imagen en formato selfie de una persona.

La Figura 5.32 muestra un primer acercamiento a lo que se plantea, será la arquitectura encargada de obtener los atributos como la edad y el género a partir de la imagen de una persona en formato selfie. Se propone una arquitectura multitarea con el objetivo de usar solo una red neuronal y reducir la cantidad de recursos que se consumirán. La arquitectura estará dividida en cinco secciones:

1. **Capa convolucional compartida:** Este bloque hace referencia a la capa de convolución, espacial, que será compartida para las dos tareas.
2. **Capas Convolucionales para el género:** Capas convolucionales exclusivas para la tarea del género.
3. **Clasificador para género:** Para obtener el género, se propone un perceptrón multi capa configurado en modo de clasificador ya que solo se contemplan solo dos clases, mujer y hombre.
4. **Capas convolucionales para edad:** Capas convolucionales exclusivas para la tarea del edad.
5. **Regresor para edad:** A diferencia del género, para la edad se propone hacer uso de perceptrón multi capa en modo de regresor. De esta forma tendremos más flexibilidad al utilizarlo en el clasificador.

Se disponen de los siguientes 3 datasets para entrenar la red neuronal:

1. *Appa Real*: contiene 7,591 imágenes etiquetadas con el género y edad real [132].
2. *FairFace*: 97,698 imágenes etiquetadas con el género y edad categórica por décadas (10s, 20s,etc). Este dataset tiene la particularidad de ser balanceado en las razas de las personas [133].
3. *All Age Faces*: contiene 13,322 imágenes etiquetadas con el género y edad real. Esta conformado en su mayoría por personas asiáticas, además de tener un preprocesado en las imágenes donde se recorta y alinea el rostro de las personas [134].

5.12.1. Red neuronal convolucional para edad

Durante la [Revisión de trabajos de Edad y género](#) se observa que la tarea de clasificación según el género tiene resultados muy buenos en la mayoría de los trabajos, mientras que la estimación de la edad resulta un poco menos

alentadora. Por ello se decide proponer una arquitectura puramente para estimar la edad y luego convertirla en una multitarea. En la Figura 5.33 se presenta esta arquitectura, en la cual se cuentan con 3 capas convolucionales, cada una seguida inmediatamente de una Función de activación ReLU y una Capa de agrupación (Pooling). Al final se tiene El Perceptrón Multicapa configurado en modo regresor.

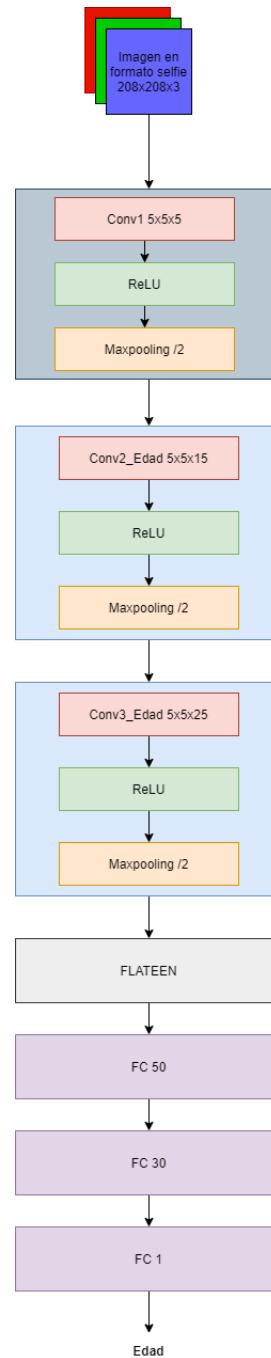


Figura 5.33: Primera propuesta de red neuronal convolucional para obtener la edad.

A la red ingresa una imagen de 208x208 píxeles con tres canales de profundidad (RGB). Los bloques en color azul claro representan capas convolucionales que se piensan exclusivos para la edad y en gris el bloque convolucional que se designará como compartido con el género. Dentro de cada bloque se colocó un texto de la forma *Conv#wxhxk* que significa lo siguiente:

- Conv#: Significa el número de la capa convolucional. Es solo un nombre para identificarlas. Si seguido del número se encuentra "Edad", esa capa es exclusiva para la estimación de la edad.
- w: Significa el ancho de cada filtro.
- h: Significa el alto de cada filtro.
- k: Significa el número de filtros que se extraerán.

Todas las capas ocupan una función de activación ReLU (consultar [Función de activación](#)) y se pueden visualizar en la figura en color verde claro.

Los bloques en color naranja representan [Capa de agrupación \(Pooling\)](#) máximas (Maxpooling) y el "/2" significa que el tamaño de la imagen será reducido a la mitad.

Por otro lado, los bloques en color morado significan capas completamente conectadas y el número seguido de "FC" representa la cantidad de neuronas que tendrá cada capa y los bloques "FLATTEN" significa que la matriz con los filtros será aplanada para poder ser ingresada a las capas completamente conectadas.

Para entrenar esta CNN se utiliza la combinación de los datasets Appa Real [132] y All Ages Faces[134] (el dataset FairFace [133] no se utiliza debido a que no se cuenta con un etiquetado de la edad real) con los cuales se tienen en total 20,931 imágenes y la distribución de las edades se aprecia en la Figura 5.34. Cabe mencionar que el dataset Appa Real [132] se preprocesó para tener las imágenes en formato selfie.

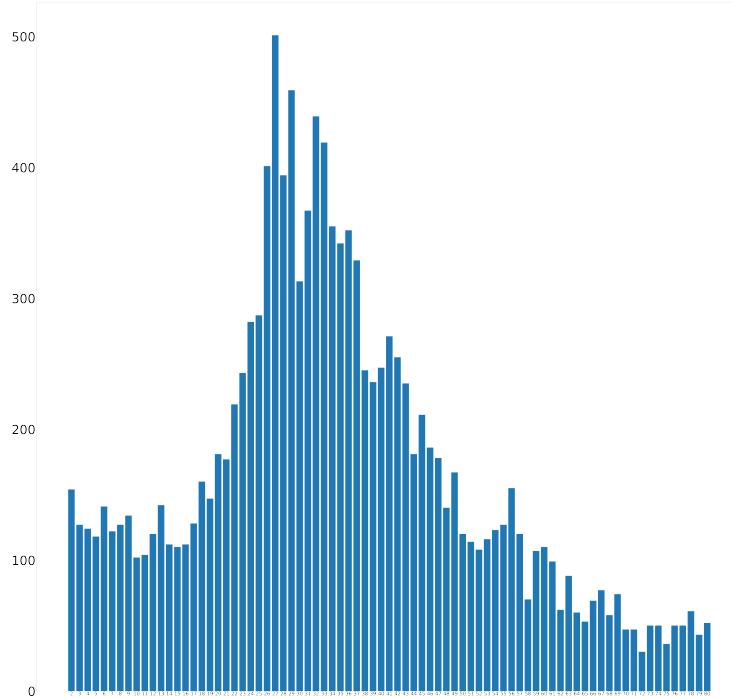


Figura 5.34: Distribución de las edad en la combinación de los datasets Appa Real [132] y All Ages Faces [134].

Hiper-parámetro	Valor
Épocas	2600
Factor de aprendizaje	1e-4
Tamaño de batch	4

Tabla 5.7: Hiper-parámetros utilizados para entrenar la CNN de edad.

En la Tabla 5.7 se encuentran los hiper-parámetros utilizados para el entrenamiento de la CNN. Las Figuras 5.35 y 5.36 representan los errores obtenidos en la fase de aprendizaje y de prueba durante estas 2,600 épocas, respectivamente.

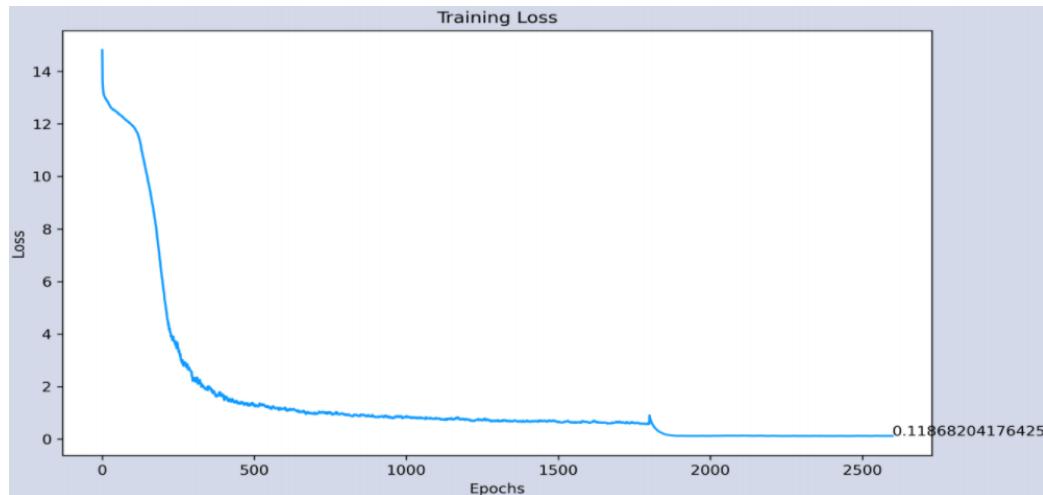


Figura 5.35: Gráfica del error en entrenamiento de la CNN de edad.

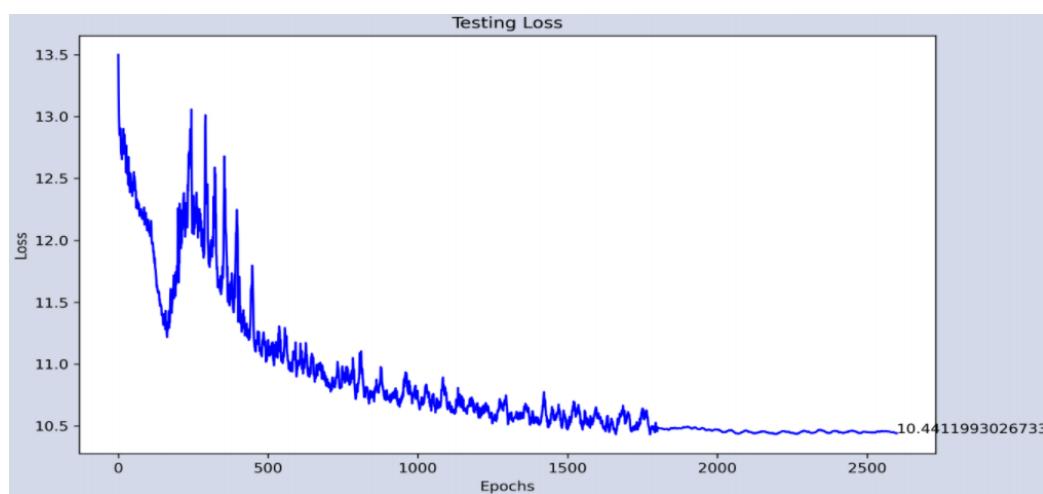


Figura 5.36: Gráfica del error en prueba de la CNN de edad.

5.12.2. Red neuronal convolucional mutitarea

Una vez que se tiene la Red neuronal convolucional para edad se crea a través de ella la red neuronal convolucional para género utilizando la técnica de Transferencia de aprendizaje en la cual se decide replicar la arquitectura de la Red neuronal convolucional para edad a partir de la segunda convolución e inicializar los pesos y bias con los obtenidos en la propia Red neuronal convolucional para edad. En la Figura 5.38 se puede apreciar el proceso, note que el bloque de convolución en color gris (conv1) es la capa compartida ya que se utiliza para ambas tareas (obtener edad y género) pero esta solo se entrena para la estimación de la edad y en el entrenamiento para la clasificación del género se congela, siendo entrenadas solo la parte del lado derecho de la arquitectura. En esta red es necesario aplicar regularización (capas Dropout) para su entrenamiento y se presentan en los bloques de color verde oscuro de la Figura 5.39, este bloque lleva seguido del nombre un 0.4 que representa el porcentaje de las neuronas a desactivar.

Para entrenar esta red se utiliza a combinación entre los datasets Appa Real [132] y FairFaces [133], teniendo un total de 103,789 imágenes, las cuales se dividen en los conjuntos de entrenamiento y prueba. En la Figura 5.37 se presenta la distribución de las clases, donde la clase 0 representa a mujer y 1 a hombre.

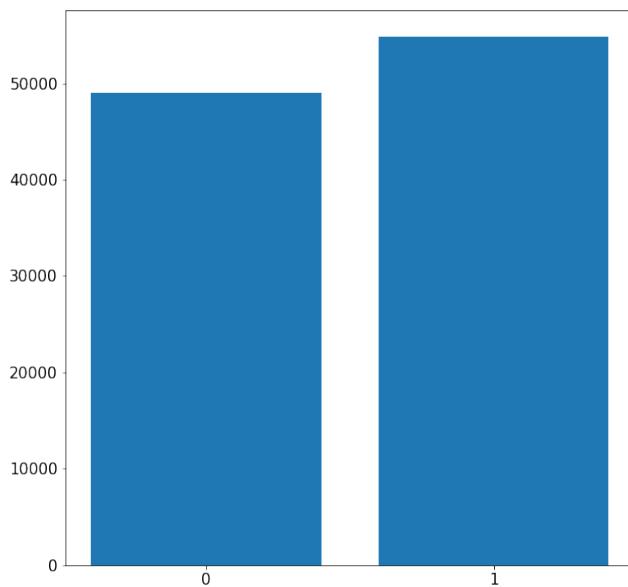


Figura 5.37: Distribución entre las clase 0 (mujer) y 1 (hombre) de la combinación entre los datasets Appa Real [132] y FairFaces [133].

Hiper-parámetro	Valor
Épocas	25
Factor de aprendizaje	1e-4
Tamaño de batch	4

Tabla 5.8: Hiper-parámetros utilizados para entrenar la parte de clasificación de género de la CNN multitarea.

Para entrenar la parte de la red que clasifica el género se ocuparon los hiper-parámetros descritos en la Tabla 5.8, obteniendo como resultado una exactitud del 80.33 %. Las Figuras 5.40 y 5.41 muestran las gráficas de la evolución de la exactitud en entrenamiento y prueba respectivamente, durante las 25 épocas.

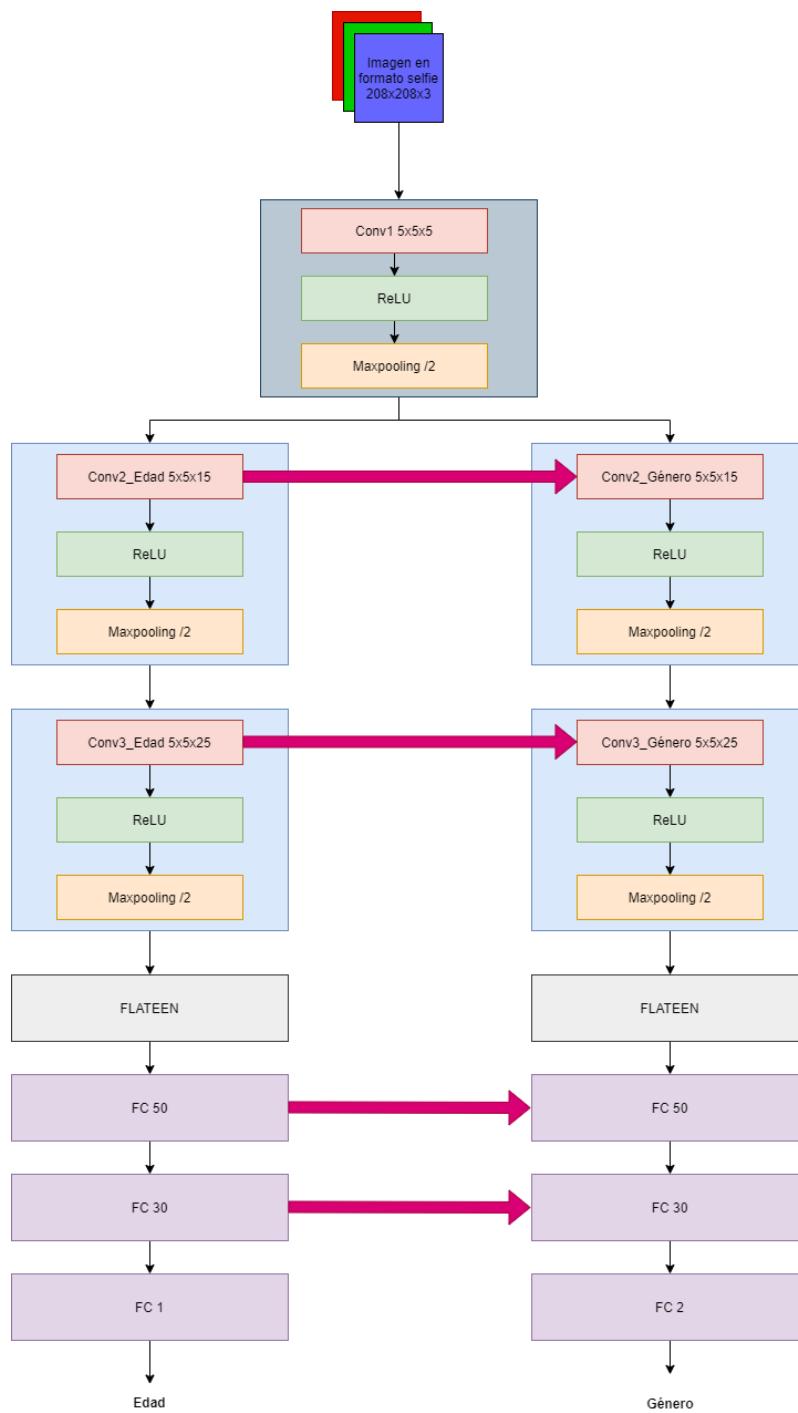


Figura 5.38: Transferencia de aprendizaje de la [Red neuronal convolucional para edad](#) para formar la arquitectura multitarea de edad y género.

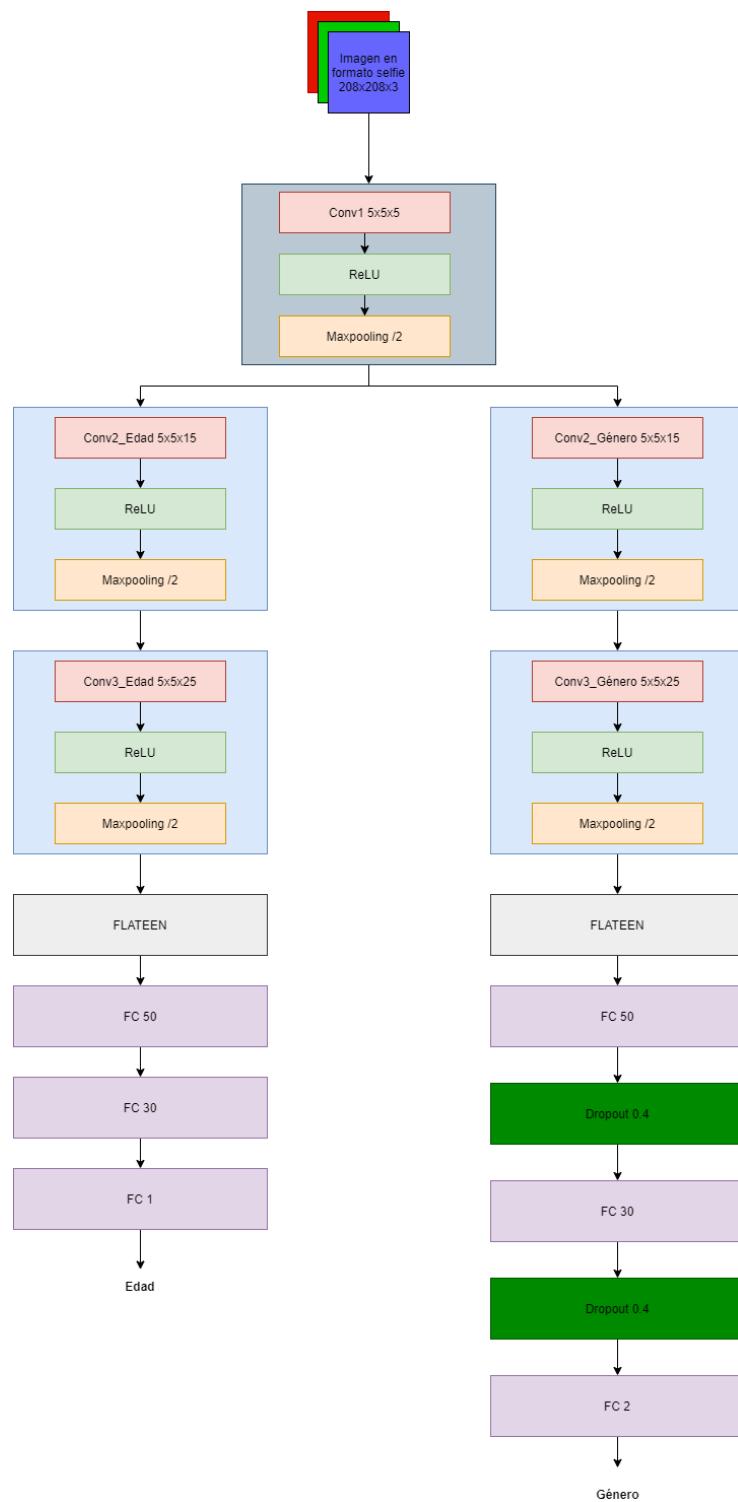


Figura 5.39: Arquitectura multitarea de edad y género.

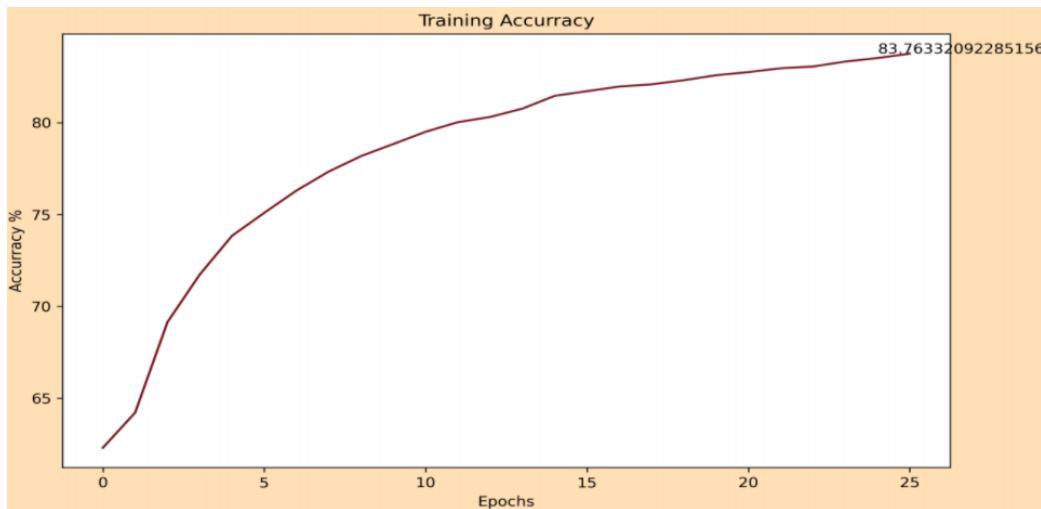


Figura 5.40: Evolución de la exactitud del conjunto de entrenamiento de la clasificación del género.

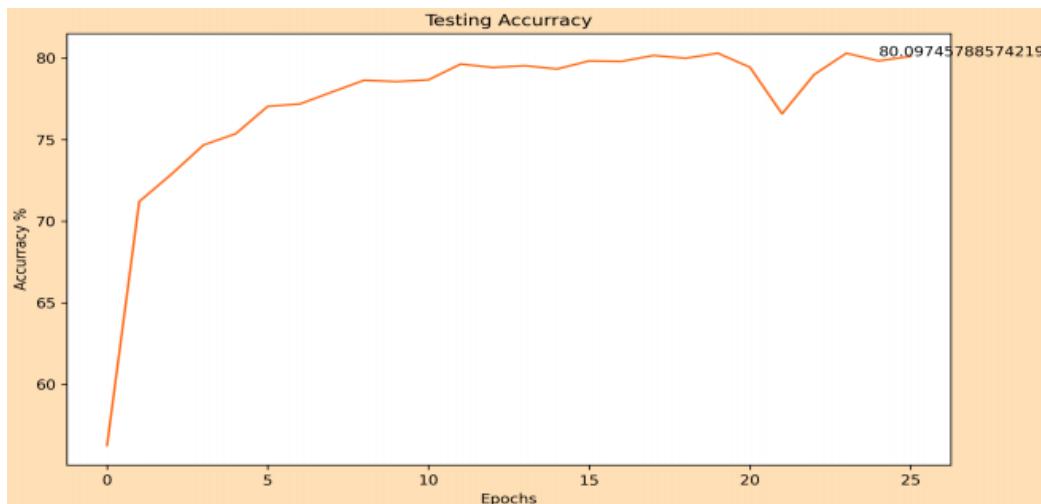


Figura 5.41: Evolución de la exactitud del conjunto de prueba de la clasificación del género.

Por último, dado a que es una clasificación también se evalúa usando una matriz de confusión (únicamente en el conjunto de prueba) para apreciar su comportamiento en los casos verdadero positivo, falso positivo, verdadero negativo y falso negativo, dando como resultados presentes en la Tabla 5.9

	Mujer (real)	Hombre (real)
Clasificado Mujer	5094	1455
Clasificado Hombre	1088	5295

Tabla 5.9: Matriz de confusión de la clasificación de género.

5.12.3. Implementación en TensorRT

Una vez terminado el entrenamiento de la red, el modelo es exportado, generando los siguientes archivos:

- saved_model.pb
- variables/
 - variables.data-00000-of-00001
 - variables.index

Como se menciona en la [Implementación en TensorRT](#) de la [Detección de rostros en la imagen](#), el modelo exportado con TensorFlow 2 debe convertirse a un **frozen graph de TensorFlow 1**. Luego, es necesario convertir el modelo frozen graph con formato protobuf (.pb) a un archivo de formato universal (.uff) [118] (ver Figura).

De forma similar a la [Single Shot Detection](#), esta red también debe ser "reparada" debido a que posee capas con operaciones de TensorFlow que no son compatibles con el formato .uff que TensorRT necesita analizar [118]. En cambio, a diferencia de esta última que fue ajustada por los scripts de configuración proporcionados por NVIDIA [126] y usuarios de su comunidad [130] [131], los métodos y correcciones que solucionan las incompatibilidades relacionadas a operaciones que no soporta TensorRT en algunas capas de este modelo fueron implementados de forma manual, de modo que la funcionalidad principal de la red no se vea alterada y al mismo tiempo el modelo pueda ser cargada por la librería. Este script con las operaciones aplicadas de forma manual esta disponible en el repositorio del sistema.

Después de resolver estas discrepancias entre operaciones, el modelo "arreglado" se convierte a formato .uff y es leído por TensorRT mediante su analizador sintáctico que mapea el modelo a un archivo .bin, el cual corresponde al motor optimizado de inferencia de esta red [118]. La Figura 5.42 presenta el proceso de conversión completo desde el modelo exportado con TensorFlow 2 hasta obtener el archivo .bin de la red que TensorRT ejecuta.

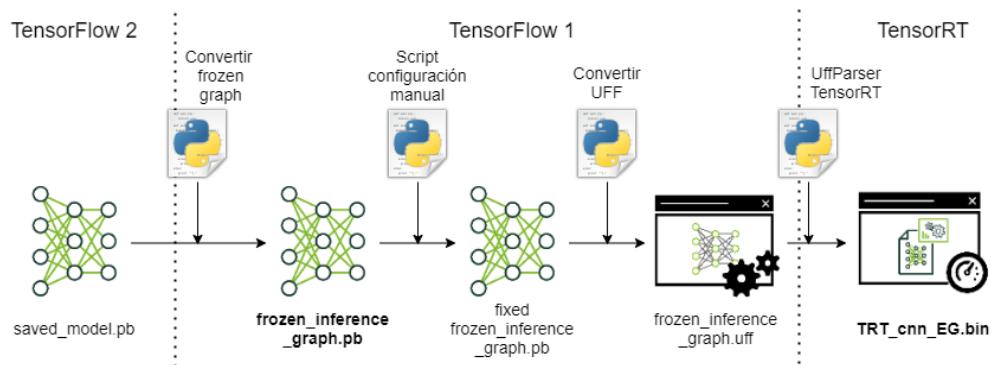


Figura 5.42: Diagrama que muestra la conversión de la red de Edad y Género de TensorFlow 2 a TensorRT, basándose en la documentación oficial [118] y en configuraciones manuales.

Finalmente, este archivo .bin se carga para que realice la obtención de Edad y Género en rostros recortados y procesados que recibe de la [Single Shot Detection](#) (ver subsección [Recorte y procesamiento de los rostros detectados](#)). Como se menciona y explica en la [Implementación en TensorRT](#) de la [Detección de rostros en la imagen](#), también es necesario aplanar el rostro de tamaño 208x208x3 en un vector de tamaño 129792 para ser ingresado en los buffers de comunicación que TensorRT necesita (ver Figura 5.43) [118].

Esta red neuronal es implementada con ayuda de hilos en Python y CUDA, cuya implementación compatible con TensorRT se obtiene del script oficial de NVIDIA en [135].

La Figura 5.43 muestra el diagrama con las entradas y salidas del motor optimizado de inferencia del modelo de la red de Edad y Género cargado en la NVIDIA Jetson. En esta misma se puede apreciar que el resultado correspondiente

a la edad se deja tal cual sale del modelo (como valor numérico flotante), mientras que el resultado del género al ser un vector de tamaño 2, en donde la primera posición corresponde a "Mujer" y la segunda a "Hombre", es necesario aplicarle una función de la siguiente forma:

- Si el valor obtenido en la primera posición es mayor al valor de la segunda, el modelo **detecta a una mujer**, y por lo tanto, esta función devuelve un **0**.
- De lo contrario, si el valor de la segunda posición es mayor al de la primera, **detecta a un hombre**; la función devuelve un **1**.

Esto se hace con el fin de almacenar un dato de tipo entero en la base de datos en vez de un arreglo.

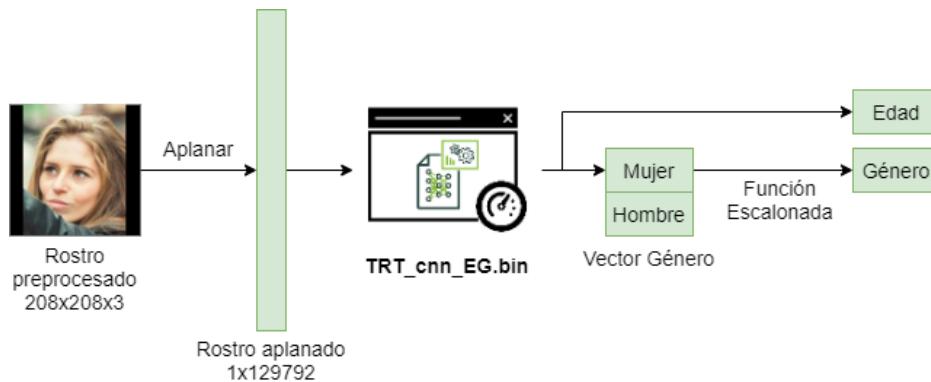


Figura 5.43: Diagrama que muestra el rostro de entrada aplastado, el modelo de la red de Edad y Género en TensorRT, y sus dos salidas: la edad obtenida del regresor, y el vector de género obtenido del clasificador.

5.13. Red neuronal convolucional para Big Five

Esta sección está destinada al cumplimiento de la segunda mitad del bloque **6. Propagación de cada rostro recortado en las redes neuronales** y del bloque **8. Obtención de la personalidad (Big Five)** del [Diagrama general del sistema](#), representado por la Figura 5.16.

Como ya se ha hablado en las secciones de [Personalidad: Big Five](#) en el capítulo del [Marco teórico](#), y de [Revisión de trabajos de Big Five](#) en el capítulo del [Estado del Arte](#), la estimación de la personalidad de una persona a partir de su rostro se medirá utilizando el modelo Big Five.

Adicionalmente, como se justificó en la [Discusión del capítulo de Estado del Arte](#), se tomará la arquitectura del Modelo de clasificación: CNN-4 del [trabajo de M. Moreno Sotelo](#) en [51] y [94], la cual es representada por la Figura 3.18.

5.13.1. Implementación en TensorRT

A diferencia de la [Single Shot Detection](#) y la [Red neuronal convolucional para Edad y Género](#), la implementación del [Modelo de clasificación: CNN-4](#) [51] [94] no requiere de scripts de configuración para "reparar" la red, debido a que esta no posee alguna operación de Keras (librería con la que fue construida y entrenada según [94]) que sea incompatible con TensorRT [118]. Originalmente la red consiste de dos archivos: **cnn4_train_2.json**, el cual posee la arquitectura completa de la red, y **cnn4_train_2.h5**, que son los pesos resultantes de su entrenamiento [94]. Entonces, es necesario leer la arquitectura de la red definida en el archivo JSON, y luego cargar sus pesos en ella.

El modelo en Keras debe convertirse a un **frozen graph de TensorFlow 1**, para después convertirlo directamente de formato protofub (.pb) a un archivo de formato universal (.uff), y finalmente ser leído por TensorRT utilizando su analizador sintáctico para mapear el modelo a un archivo .bin, correspondiente al motor optimizado de inferencia de esta red [118]. La Figura 5.44 presenta el proceso de conversión completo desde el modelo original en Keras hasta obtener el archivo .bin que TensorRT necesita para ejecutar la red.

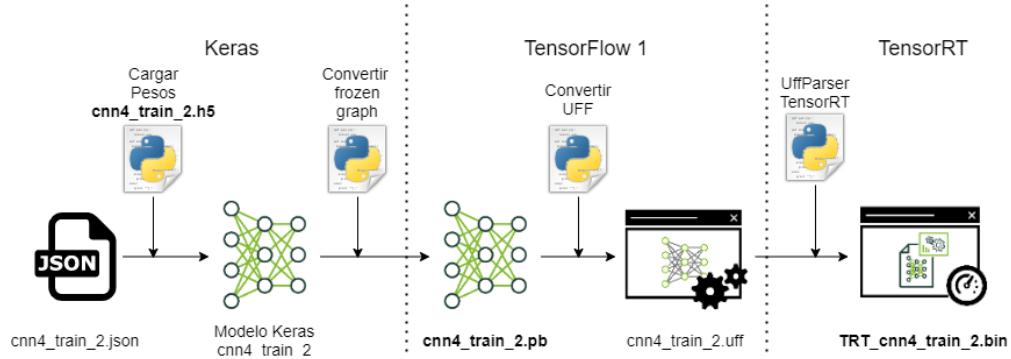


Figura 5.44: Diagrama que muestra la conversión del [Modelo de clasificación: CNN-4](#) [94] de la red de Big Five de Keras a TensorRT, basándose en la documentación oficial [118].

Finalmente, este archivo .bin se carga para que realice la obtención de Big Five en rostros recortados y procesados que recibe de la [Single Shot Detection](#) (ver subsección [Recorte y procesamiento de los rostros detectados](#)). Como se menciona y explica en la [Implementación en TensorRT](#) de la [Detección de rostros en la imagen](#), también es necesario aplanar el rostro de tamaño 208x208 en un vector de tamaño 43264 para ser ingresado en los buffers de comunicación que TensorRT necesita [118].

Similar a la [Red neuronal convolucional para Edad y Género](#), esta red neuronal también es implementada con ayuda de hilos en Python y CUDA, cuya implementación compatible con TensorRT se obtiene del script oficial de NVIDIA en [135].

La Figura 5.45 muestra el diagrama con las entradas y salidas del motor optimizado de inferencia del [Modelo de clasificación: CNN-4](#) de la red de Big Five [51] [94] cargado en la NVIDIA Jetson.

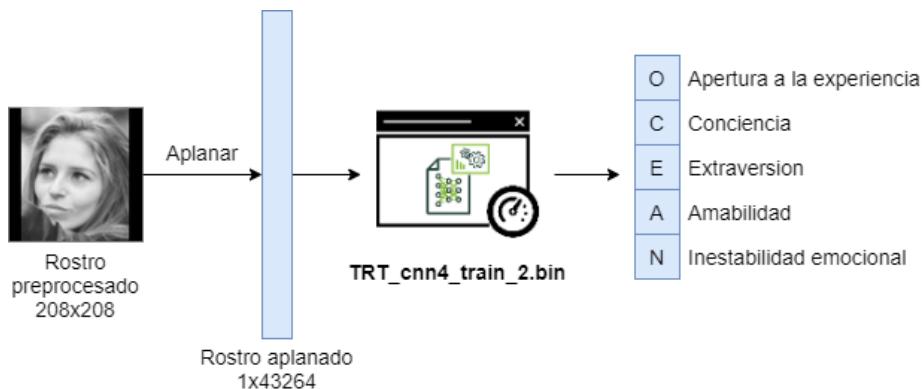


Figura 5.45: Diagrama que muestra el rostro de entrada aplanado, el [Modelo de clasificación: CNN-4](#) [94] de la red de [Personalidad: Big Five](#) en TensorRT, y el vector de salida de tamaño 5 en donde cada posición corresponde a una dimensión de personalidad como se indica.

5.14. Recomendador del sistema

Esta sección está destinada al cumplimiento de los bloques **10. Recomendación de la bebida**, **9. Obtención de la temperatura ambiental** y **12. Búsqueda de la bebida en el catálogo** del Diagrama general del sistema, representado por la Figura 5.16.

Tras la obtención del **Género**, **Edad**, **Personalidad: Big Five** y la temperatura ambiental, un **Recomendador** será el responsable de proporcionar al cliente una bebida apropiada. En este caso su funcionamiento estará basado en contenido debido a que este tipo de recomendadores funcionan muy bien para características objetivas, por lo que haciendo uso de los ingredientes del menú de **BubbleTown®** se tendrán suficientes datos para llevarlo a cabo. En contraparte, queda descartado un recomendador colaborativo debido a que este tipo de recomendadores funcionan mejor para aspectos que no son fácilmente medibles (subjetivos) y para la realización de un recomendador de este tipo se requieren de conjuntos de datos que, debido a las condiciones actuales, son muy difíciles de conseguir.

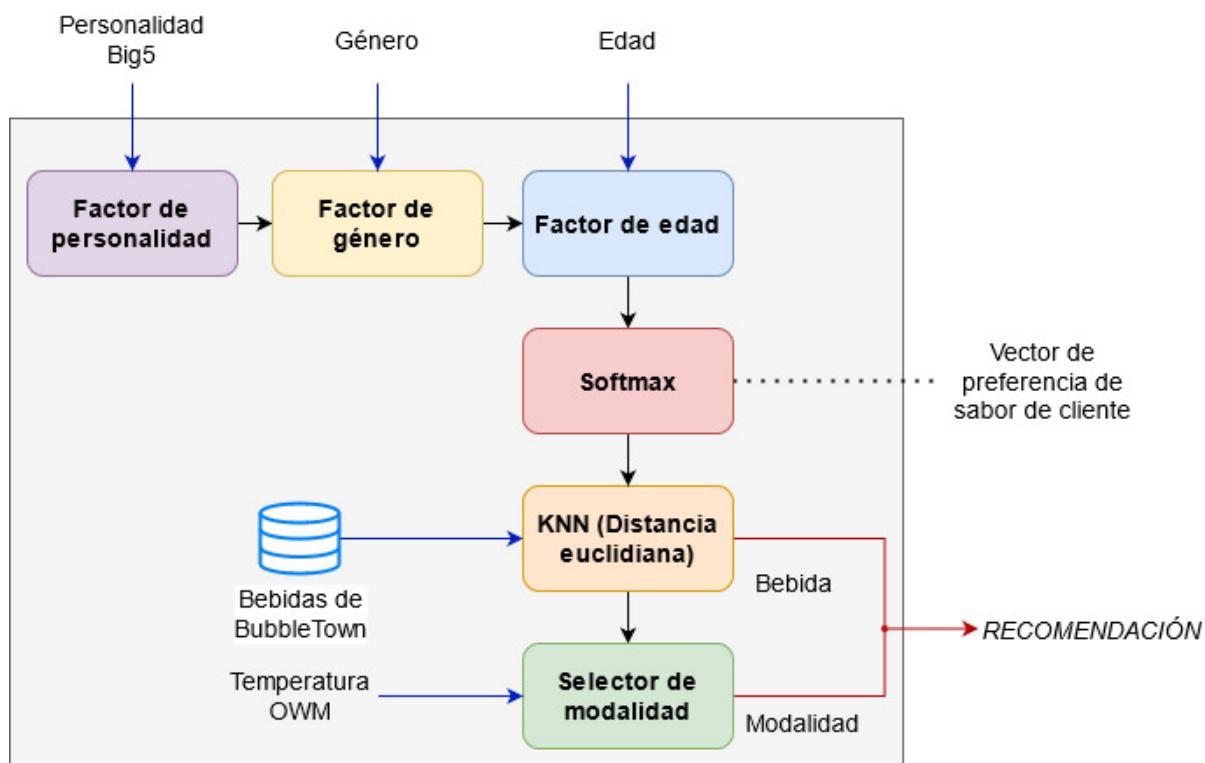


Figura 5.46: Diagrama del funcionamiento del recomendador.

En la Figura 5.46 se observa de forma general cómo es el funcionamiento del recomendador del sistema, lo cual se explica a continuación de forma más detallada. Es importante remarcar que para este proceso de recomendación se ha eliminado el sabor salado debido a que ninguna de las bebidas de **BubbleTown®** presentan este sabor.

- **Factor de personalidad:** La personalidad es uno de los factores que afecta en cuestión de preferencias de sabores así que en consecuencia, una vez que se obtiene el vector de la personalidad del cliente haciendo uso de la **Red neuronal convolucional para Big Five**, el vector será multiplicado por una matriz con los valores de la Tabla 5.10 para así obtener una matriz con la preferencia de sabor del cliente en función de su personalidad. Después se procederá a sumar todos los valores para cada sabor y así se obtendrá un vector de tamaño cinco, donde cada

valor del vector corresponderá a una clase de sabor.

Preferencia de sabor	Apertura	Conciencia	Extraversión	Amabilidad	Inestabilidad
Sabor dulce	0.7243	0.4357	0.6138	0.5614	0.4036
Sabor umami	0.2986	0.4900	0.5852	0.1886	0.1979
Sabor amargo	0.2786	0.1307	0.1600	0.2424	0.1350
Sabor ácido	0.0271	0.1264	0.2381	0.0252	0.1543
Sabor picante	0.4857	0.2421	0.5971	0.6424	0.5607

Tabla 5.10: Promedio de preferencia de sabor en función de la personalidad en valores de 0 a 1. Tabla obtenida con datos de [47].

- **Factor de género:** Como ya se mencionó en [Género](#), esta característica tiene influencia sobre la preferencia de sabor de una persona, por lo que con los resultados mostrados en diversos artículos y la capacidad del sistema de clasificar este rasgo, se puede hacer uso de esas preferencias para brindar una mejor recomendación. Para este objetivo se hará uso de la Tabla 5.11 que multiplicará al vector que se obtiene tras aplicar el factor de personalidad.

Preferencia de sabor	Mujer	Hombre
Sabor dulce	0.7789	0.7292
Sabor umami	0.4434	0.3752
Sabor amargo	0.2530	0.3596
Sabor ácido	0.3783	0.4472
Sabor picante	0.5431	0.8288

Tabla 5.11: Preferencia de sabor promedio en función del género (Tabla 2.9 con valores de 0 a 1). Tabla obtenida con datos de [47]

- **Factor de edad:** En [Edad](#) se explicó cómo a través de diversos estudios se ha demostrado que hay una relación entre la edad y las preferencias hacia un tipo de sabor. Tras la obtención de la edad estimada, el cliente será colocado en una de las cuatro clases de edad y, junto con la Tabla 5.12, se ajustará el vector de preferencia de sabor del cliente al ser multiplicado por dicha tabla. Con esto el tamaño del vector que se obtiene tras aplicar el factor de género no es modificado pero sí sus valores.

Una de las diferencias con la Tabla 2.6 es que aparece una nueva clase de "1 a 17 años" para la cual los valores de preferencia fueron calculados con el uso de la variación entre las clases posteriores para determinar qué tanto se aumenta o decremente la tendencia hacia cada clase de sabor. Igualmente se agregan el sabor *umami* y *picante*; sin embargo, en la Tabla ambos tienen valores de 0 (cero), lo que significa que la edad no influirá en estas clases de sabores.

- **Softmax:** Tras aplicar todos los factores que influyen en la preferencia de sabor se aplicará la función "Softmax" al vector para así obtener un **vector de preferencia de sabor de cliente**.
- **KNN (Distancia euclíadiana):** Ya que se tiene el **vector de preferencia de sabor de cliente** se hará uso de [KNN \(vecino más cercano\)](#) para obtener una bebida de la base de datos cuyo sabor sea el más similar a dicho vector.
- **Selector de modalidad:** Del paso anterior ya se tiene la bebida que será recomendada por el sistema. En este paso se seleccionará la modalidad de la bebida en función de la temperatura ambiental para lo cual, el sistema hará uso de la API [OpenWeatherMap](#) para obtener la temperatura en el punto de venta en ese momento y una vez obtenida, se seleccionará la modalidad válida más adecuada para dicha bebida.

Preferencia de sabor	1 a 17 años	18 a 36 años	37 a 50 años	más de 50 años
Sabor dulce	0.9520	0.8500	0.8100	0.7700
Sabor umami	0.0000	0.0000	0.0000	0.0000
Sabor amargo	0.2652	0.2400	0.1867	0.1700
Sabor ácido	0.1972	0.2033	0.2067	0.2233
Sabor picante	0.0000	0.0000	0.0000	0.0000

Tabla 5.12: Promedio de preferencia de sabor en función de grupo de edad (Tabla 2.6 en un rango de valores de 0 a 1). Tabla obtenida con datos de [40].

5.14.1. Vectorización de bebidas

Para el funcionamiento del recomendador se requirió generar un vector de sabor característico para cada bebida del menú, para lo que se hizo uso de *Coffee Taster's Flavor Wheel* debido a que si bien, hay ruedas de sabores específicas para té, esas ruedas habían sido construidas considerando características de aroma, textura y sabor; mientras que la rueda generada en [136] solo considera el sabor. Esta rueda puede observarse en la Figura 5.47.

Para la vectorización de las bebidas primero se generaron vectores para cada ingrediente que aparece en la rueda de sabor. Este proceso de vectorización fue identificando las clases de sabores dentro de la rueda, posteriormente se acomodaron los ingredientes en forma de lista, al ingrediente del centro para cada clase de sabor se le asignó el valor de 1 (uno) y a los ingredientes vecinos se le fue asignando un valor menor hasta llegar a 0 (cero). Este proceso fue realizado para cada sabor.

Con los sabores básicos vectorizados el siguiente paso fue hacer la suma de todos los ingredientes por bebida para posteriormente aplicar la función "softmax" al vector resultante de la suma de todos los ingredientes para cada bebida.

SCA

Coffee Taster's Flavor Wheel



Figura 5.47: Rueda de sabor del café. Imagen tomada de [136].

5.15. Base de datos para el sistema

Esta sección está destinada al cumplimiento de los bloques **11. Almacenamiento de datos** y **12. Búsqueda de la bebida en el catálogo** del [Diagrama general del sistema](#), representado por la Figura 5.16. El sistema deberá hacer uso de una base de datos para almacenar la información relevante como lo son el catálogo de bebidas, la lista de imágenes de publicidad estática o las recomendaciones que el sistema realiza para cada persona. Para ello, con el objetivo de aprovechar los beneficios de la flexibilidad de las bases de datos NoSQL ([Bases de datos NoSQL](#)) es que se decidió hacer uso de este tipo de bases para almacenar los datos deseados en este trabajo.

La base de datos está estructurada con cinco colecciones: "admins", "bebidas", "publicidad" y "recomendacion".

5.15.1. Colección de administradores

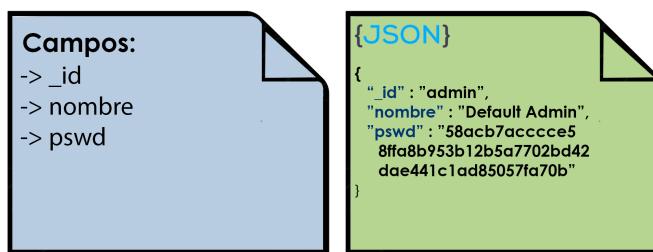


Figura 5.48: Atributos y ejemplo de documento para colección "admins"

Para los documentos que se almacenan en la colección "admins" se tienen los campos "`_id`", "`nomnbre`" y "`pswd`". Del lado izquierdo de la Figura 5.48 se pueden observar los campos, mientras que del lado derecho de la misma Figura se ve un ejemplo de la estructura del documento donde se pueden apreciar dos características importantes:

1. "`_id`": Es establecido por el administrador de la base de datos y es el valor usado para autenticar a los administradores.
2. "`nombre_id`".
3. "`pswdRequerimientos funcionales, lo que debe ser almacenado es valor resultante de calcular el hash.`

5.15.2. Colección de bebidas

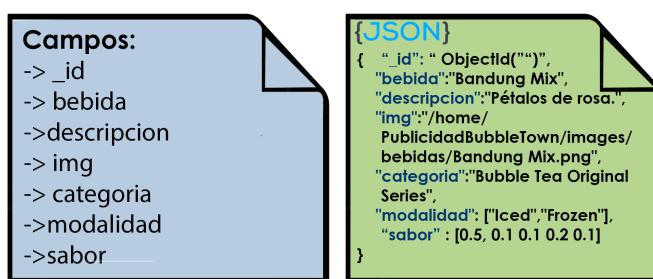


Figura 5.49: Atributos y ejemplo de documento para colección "bebidas"

Como se puede observar en la Figura 5.49, la estructura esperada para los documentos de la colección bebida son el “_id”, “bebida”, “descripcion”, “img”, “categoria”, “modalidad” y “sabor”.

1. El “_id” para este caso es generado por el gestor de forma automática
2. “bebida”: Nombre de la bebida dentro del catálogo de la empresa BubbleTown®
3. “descripcion”: Provee una pequeña descripción de la bebida en cuestión.
4. “Img”: Ruta donde se encuentra la imagen de la bebida en 2D.
5. “categoria”: Categoría a la que pertenece la bebida.
6. “modalidad”: disponibles para esa bebida se almacenan en formato de arreglo.
7. “bebida”: es un vector de tamaño 5 que contiene la representación de saber de cada bebida.

5.15.3. Colección de publicidad

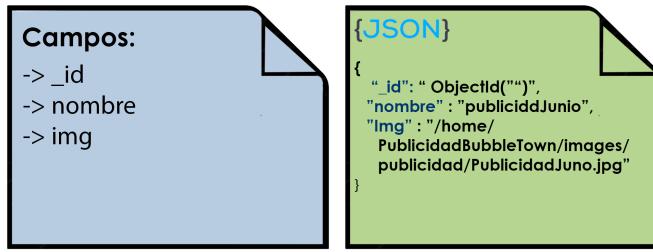


Figura 5.50: Atributos y ejemplo de documento para colección "publicidad"

En la Figura 5.50 se muestra un ejemplo de cómo se está almacenando la información que será usada para proyectar publicidad estática cuando el sistema no detecte un rostro humano.

1. “_id”: El “_id” es generado por el gestor de la base de datos
2. “nombre”: Campo para auxiliar a los administradores del sistema a identificar cada documento diferente.
3. “img”: Se almacena la ruta donde se encuentra la imagen a proyectar.

5.15.4. Colección de recomendaciones

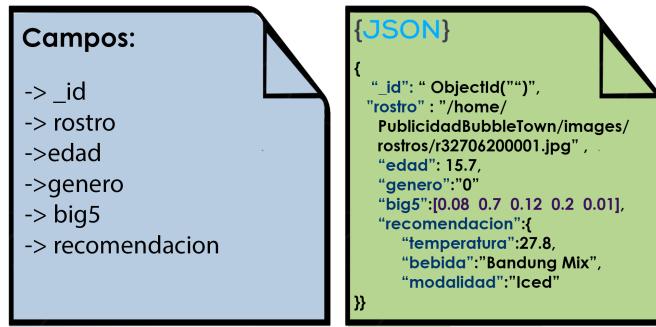


Figura 5.51: Atributos y ejemplo de documento para colección "recomendacion"

Los documentos en la colección "recomendacion" son los mas complejos (Figura 5.51) ya que tienen varios campos:

1. "_id": Es la "clave" que vuelve único a cada documento. Este valor es generado por el gestor de la BD.
2. "rostro": Este campo almacena la ruta donde se encuentra la información asociada al rostro usado para la recomendación. Es importante mencionar que los archivos no tienen un formato de imagen dado que el sistema debe generar un archivo cifrado de la información.
3. "edad": Se almacenará un valor numérico que corresponda a la edad estimada por la red neuronal.
4. "genero": En este campo se almacena la clase en la que la red neuronal clasificó el rostro asociado.
5. "big5": En este campo es almacenado un arreglo de 5 valores flotantes generado por la red neuronal de Big5.
6. "recomendacion": En este campo se guarda a su vez un documento que contiene la bebida que se recomendó, la modalidad y la temperatura ambiental cuando se realizó la recomendación.

5.16. Generación de realidad aumentada

Esta sección está destinada al cumplimiento de los bloques **13. Generación de realidad aumentada** y **14. Despliegue del resultado con realidad aumentada** del [Diagrama general del sistema](#), representado por la Figura 5.16.

Antes de presentar el diseño se especifica el resultado que se pretende. En la Figura 5.52 se puede apreciar un ejemplo de la realidad aumentada a realizar. En esta se observa la recomendación de la bebida dentro de un globo de pensamiento que surge a partir de la parte superior derecha del rostro detectado previamente. Adicionalmente se agrega un banner en la parte inferior de la imagen que especifica el nombre del producto y el logo distintivo de la marca en la doble superior izquierdo de la imagen. Todos los elementos a incrustar serán bidimensionales.



Figura 5.52: Ejemplo de la realidad aumentada a realizar. Tomada de Google Imágenes.

La Figura 5.53 representa el diseño del modulo de generación realidad aumentada a realizar, este tiene como entradas:

- Frame captado con la cámara.
- Rostros detectados.
- Publicidad de las bebidas a recomendar.

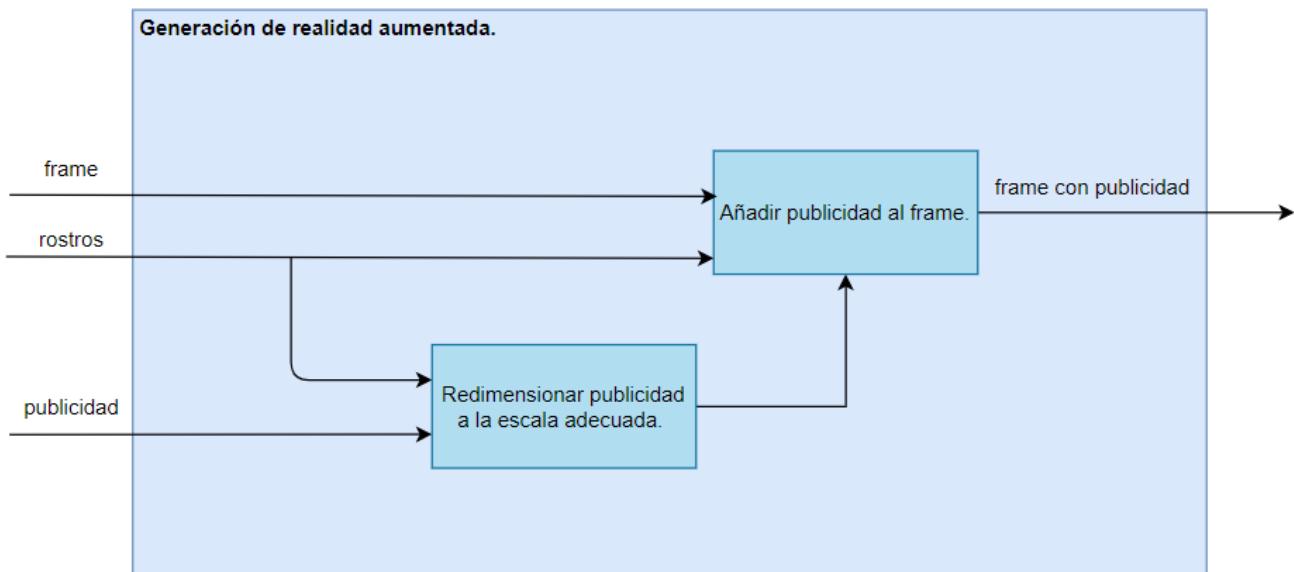


Figura 5.53: Diseño de realidad aumentada.

Dentro de este modulo se observan dos procesos internos. El primero es la redimensión de la publicidad a la escala adecuada de cada persona, debido a que todos los usuarios no se posicionarán siempre a la misma distancia. El segundo proceso es adjuntar la publicidad al frame obtenido previamente por la cámara. Este frame con publicidad se envía al bloque **14. Despliegue de resultado con realidad aumentada** del [Diagrama general del sistema](#), representado por la Figura 5.16.

5.17. Diseño de la interfaz gráfica

Pensando en que este sistema será usado por personas ajenas al área de la computación es necesario brindarles una interfaz sencilla que permita operarlo. Se presenta la necesidad de autenticar a las personas con acceso a los datos sensibles del sistema, por ello en la Figura 5.54 se muestra el diseño del *login* del sistema en la cual se solicita un *usuario* y *contraseña*.



Figura 5.54: IU1 Login de acceso.

Una vez que se da acceso, el administrador podrá realizar las acciones de añadir nueva publicidad estática, modificar los datos de las bebidas y hacer un respaldo de la información guardados previamente en la base de datos. En la Figura 5.55 se muestran estas opciones, además de un botón para cerrar la sesión.



Figura 5.55: IU2 Pantalla principal.

La opción de publicidad estática (botón izquierdo de color amarillo en la Figura 5.55) permite agregar o eliminar

una publicidad, además de mostrar las que ya existen, esto se refleja en la Figura 5.56.



Figura 5.56: IU3 Pantalla para gestión de publicidad.

Para añadir una publicidad se necesita establecer la dirección del archivo (este puede ser desde un archivo USB conectada a la NVIDIA Jetson), se da una previsualización en el lado derecho, ver la Figura 5.57. Por último se dan las opciones de "Guardar" y "Regresar".



Figura 5.57: IU4 Pantalla para agregar publicidad.

En la eliminación de una publicidad estática siempre se mostrará el aviso de la Figura 5.58 con el objetivo de que el administrador este completamente seguro de borrar la publicidad seleccionada.

Para realizar un respaldo (botón derecho de color amarillo de la Figura 5.55) se debe seleccionar la ruta en la que se hará el respaldo y la fecha a partir de la cual se tomarán los datos. La Figura 5.59 representa esta pantalla, además se agrega una advertencia sobre la distribución total o parcial de estos datos.

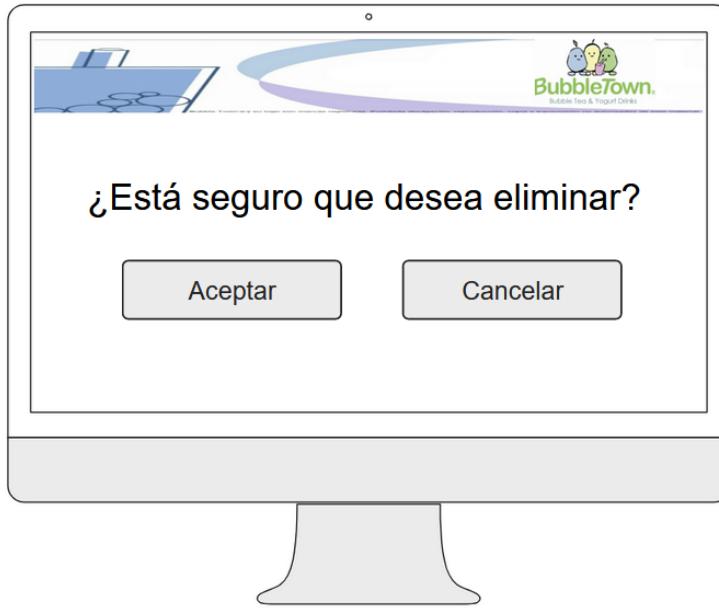


Figura 5.58: IU5 Pantalla de confirmación para eliminar.

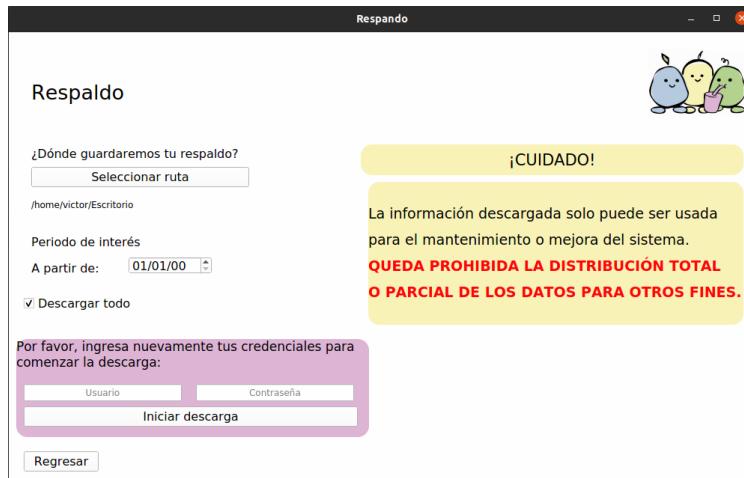


Figura 5.59: IU6 Pantalla para hacer respaldo.

En la Figura 5.60 permite gestionar las bebidas (botón de color verde de la Figura 5.55) con las opciones de *agregar*, *modificar* y *eliminar* una bebida. Además se muestra una visualización de la imagen de la bebida, se debe recordar que esta imagen es la que inserta en el globo de pensamiento de la Figura 5.52.

Para *agregar* y *modificar* (opciones de la Figura 5.60) las bebidas se da la interfaz de la Figura 5.61 donde se le puede poner un *nombre* y *descripción* a consideración del administrador. También se puede elegir la *categoría* y modalidad de la bebida. En el botón "Cargar imagen" se permite establecer la ruta de la imagen de la bebida a añadir o modificar.

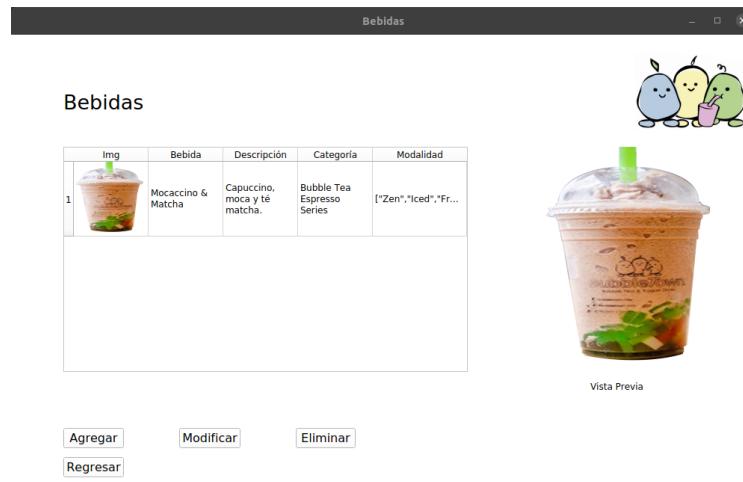


Figura 5.60: IU7 Pantalla de gestión de bebidas.

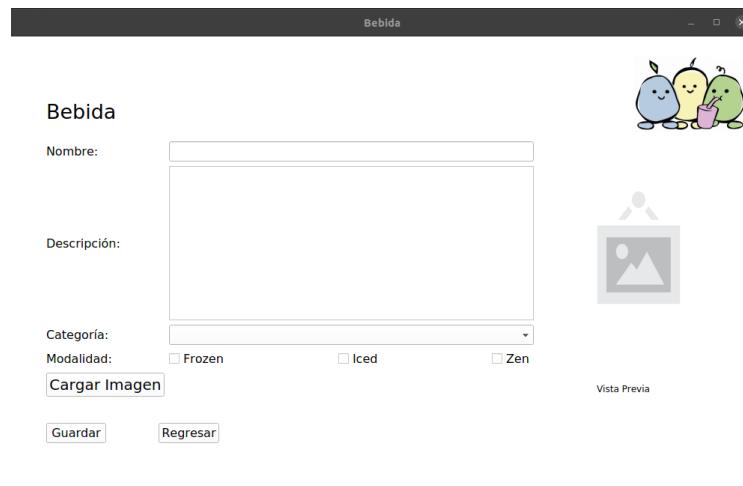


Figura 5.61: IU8 Pantalla para agregar o modificar bebida.

5.18. Atributos de calidad

Los atributos de la calidad representan el objetivo de todo diseño de software, entre los cuales se comprende la funcionalidad, la confiabilidad, la usabilidad, la eficiencia, la mantenibilidad y la portabilidad [137].

En la Tabla 5.13 se aprecia el análisis del conjunto de elementos de calidad.

Atributo	Cumple	¿Por qué?
Interoperabilidad	No	El sistema no está diseñado para que se intercambien datos con otros sistemas ya que busca poder trabajar de la forma más autónoma posible.
Seguridad de acceso	Si	Manejo de cuentas de administrador con contraseña

Atributo	Cumple	¿Por qué?
Cumplimiento funcional	- -	No aplica
Madurez	Si	No se detectan fallas
Tolerancia a fallos	Si	Se manejan fallas
Capacidad de recuperación de fallos	Si	El sistema guarda el usuario y contraseña introducida para recuperar la sesión a pesar de un fallo.
Cumplimiento de fiabilidad	- -	No aplica
Capacidad para ser entendido	Si	El software es específicamente para un uso particular
Capacidad para ser aprendido	No	El software resulta ser complejo
Capacidad para ser operado	Si	Debido a que requiere de pocos datos para operarse se vuelve rápido y fácil.
Capacidad de atracción	Si	Dado a que ocurre una interacción entretenida para el usuario.
Cumplimiento de usabilidad	- -	No aplica
Comportamiento temporal	No	No se muestra ninguna barra de progreso al hacer respaldos, al mostrar la publicidad no es necesario por el tiempo de respuesta del software.
Utilización de software	Si	Al hacer respaldos
Utilización de recursos	Si	Los respaldos y añadir publicidad son procesos que se ejecutan cuando la publicidad dirigida no está en uso
Cumplimiento de eficiencia	- -	No aplica
Capacidad para ser analizado.	Si	Debido a la alta documentación es posible determinar fallos o partes añadir
Capacidad para ser cambiado	No	Si se desea anexar una funcionalidad se tendría que re entrenar la red neuronal. Las funcionalidades actuales son las adecuadas para el cliente.
Estabilidad	Si	Es resistente a fallos inesperados.
Capacidad para ser probado	Si	Con los distintos ejemplos de prueba que se tienen para la red neuronal se permite hacer pruebas de nuevas funciones.
Adaptabilidad	Si	Es entrenado para adaptarse a distintos entorno sin afectar su funcionalidad ni eficiencia.
Instabilidad	Si	Puede ser instalado en distintos hardware que cumplan los requerimientos mínimos del sistema, siempre y cuando también se tengan las herramientas de software instaladas necesarias.
Coexistencia	No	Al requerir de altos requerimientos de software no se recomienda tener un uso compartido de recursos
Capacidad para ser reemplazado	Si	El sistema puede ser utilizado para el mismo propósito y en el mismo entorno que otro que haga su función
Portabilidad	- -	No aplica

Tabla 5.13: Atributos de calidad

5.19. Discusión del capítulo

Dado a que se utiliza un paradigma orientado a objetos, se ocupa UML para diseñar los [Diagramas de actividades](#) que planifican la ejecución de objetos; el [Diagrama de clases](#) define las clases y su relación entre ellas.

Se puede observar a lo largo del capítulo que el sistema tiene muchos módulos de alta complejidad que requieren de alto poder computacional para lograr los tiempos de procesamiento requeridos. Esto da lugar a seleccionar un hardware especializado que permitir ejecutar las tareas de forma rápida y eficiente conservando un espacio pequeño que no implique restarle espacio a la sucursal de [BubbleTown®](#).

Como se menciona en la sección de [Especificación del hardware](#), la NVIDIA Jetson cumple con gran parte de estos requerimientos y demuestra una gran velocidad en comparación con otros equipos similares, como la Raspberry Pi 3, según los resultados de la Tabla 5.11. La selección de la cámara marca HIKVISION fue con base a su buena calidad de video capturado y su compatibilidad con el resto del hardware, pues este dispositivo será el que proveerá la materia prima de todo el sistema: las imágenes; otro aspecto importante es su soporte de los protocolos TCP/IP, para comunicación, y RTSP, para transmisión de video en tiempo real. En esta sección también se determina el uso de un switch PoE para lograr tener un cableado más organizado y poder alimentar de electricidad a la cámara de red sin necesidad de tener más cables, además de contar con diversas conexiones extras.

En primer lugar se deben obtener los frames o imágenes el video proveniente de la cámara. Como se detalló en el diseño de [Captura de video a imagen](#), tras cumplir una serie de condiciones de cableado y conexión entre los dispositivos usados para este sistema, la obtención de estos frames es un procedimiento directo y flexible, gracias a las librerías integradas que ofrece el sistema operativo de la NVIDIA Jetson, en conjunto con los protocolos de comunicación compatibles.

Para la [Detección de rostros en la imagen](#) se utiliza una [Single Shot Detection](#) reentrenada aplicando [Transferencia de aprendizaje](#) para poder obtener solamente los rostros de las personas dentro de una escena. En las pruebas realizadas del entrenamiento se observa cómo a pesar de que en ciertos casos se evalúa como errónea la detección de la red, esta es adecuada para el propósito del sistema (observar el ejemplo de la Figura 5.28).

Por otro lado, el modelo generado de la [Single Shot Detection](#) posterior al reentrenamiento debe pasar por una serie de configuraciones para ser utilizado con la librería TensorRT en la NVIDIA Jetson, ya que como se habló en la sección de [Herramientas de software a utilizar](#) el diseño, construcción y entrenamiento de las [Redes neuronales](#) se realiza con ayuda de TensorFlow, pero esta herramienta posee algunas operaciones en las capas de las redes que no son compatibles con TensorRT, que es la librería seleccionada para la ejecución de todas las redes neuronales en vez de TensorFlow, a causa de que la misma documentación de la NVIDIA Jetson así lo recomienda para tener una mayor velocidad y rendimiento [109] [118]. Por último, una vez teniendo este modelo implementado, hay que recortar y extraer los rostros, además de aplicarles un nuevo preprocesamiento, que consiste en las siguientes operaciones: aplicación de formato selfie, redimensión y normalización.

La [Red neuronal convolucional para Edad y Género](#) es una red multitarea la cual lleva acabo la estimación de la edad y la clasificación del género, el modelo de esta red fue diseñada tomando en cuenta que no se necesitan resultados muy exactos pero sí rapidez, por ello se aprecia una arquitectura con pocas capas convolucionales y completamente conectadas con el menor de neuronas posibles, esto también ayuda al entrenamiento de la red correctamente (es una forma de combatir el sobre entrenamiento).

La implementación en TensorRT de este modelo fue similar a la de la [Single Shot Detection](#), ya que se tuvieron que solucionar errores de incompatibilidad del modelo generado después de su entrenamiento en TensorFlow. Los datos de entrada para esta red son las imágenes de los rostros recortadas preprocesadas como se indicó anteriormente (a color).

Por otro lado, la [Red neuronal convolucional para Big Five](#) es el [Modelo de clasificación: CNN-4](#) tomado del trabajo desarrollado por **M. Moreno Sotelo** [51] [94], tal y como se indicó en la [Discusión del capítulo](#) del capítulo del [Estado del Arte](#). La red permite conocer qué rasgo predomina en cada una de las cinco dimensiones de personalidad (ver sección [Personalidad: Big Five](#) del capítulo del [Marco teórico](#)) en la persona, y de esta manera utilizar esta información al momento de generar una recomendación de bebida.

La implementación en TensorRT de este modelo no fue tan compleja como en los modelos de las redes anteriores, ya que la arquitectura de la red en [94] no poseía operaciones incompatibles, pues es importante mencionar que no se realizó ningún ajuste o reentrenamiento en esta red, se tomó directamente tal y como estaba. Los datos de entrada para esta red son las imágenes de los rostros recortadas preprocesadas como se indicó anteriormente, pero además se les debe de aplicar un filtro de color para convertirlas a escala de grises.

Al abordar el tema del diseño del [Recomendador del sistema](#) se estableció que será un [Recomendación basada en contenido](#), esto porque como se comentó en el Marco Teórico (ver [Recomendador](#)) un recomendador que haga uso de [Filtrado colaborativo](#) requiere de datos más complejos de conseguir tras las medidas establecidas por el gobierno de la CDMX tras la pandemia a causa de la enfermedad COVID-19. Esto se debe a que los recomendadores que usan filtrado colaborativo requieren principalmente de la evaluación de los productos por varias personas y que, además, el usuario al que se le va a hacer la recomendación ya haya evaluado previamente otras bebidas. Con el uso de un [Recomendación basada en contenido](#) se vuelve más viable el proceso ya que solo se requieren características medibles de los productos y un perfil de preferencias del usuario; sin embargo, este perfil se crea haciendo uso de la teoría y los atributos obtenidos con ayuda de las redes neuronales.

Para el diseño e implementación de la [Base de datos para el sistema](#) se ha decidido hacer uso de una base de datos no relacional, siendo específicamente una base de datos en MongoDB. La decisión de usar este tipo de base de datos radica en dos cuestiones básicamente: la primera es que las bases de datos no relacionales concuerdan con la filosofía de las metodologías ágiles con respecto a que se pueden ir construyendo de forma sencilla, rápida y son altamente flexibles a modificaciones en caso de que se requieran durante el desarrollo; la segunda está relacionada con que MongoDB tiene como base el almacenamiento de los datos en forma de documentos JSON, características que se adapta adecuadamente a catálogos de productos. Con esto, aún cuando se tiene una estructura clara de los productos que se ingresan en la base de datos, el sistema es completamente capaz de adaptarse a nuevos productos que no necesariamente sean bebidas y que, por conseciente, tengan características diferentes.

Referente a la [Generación de realidad aumentada](#) se tiene una propuesta que cumple con el objetivo del proyecto; sin embargo, debido a que se cuentan con recursos limitados y hay tareas de mayor importancia que requieren poder de cómputo considerable como lo son las redes neuronales, las animaciones no son muy llamativas; no obstante, debido a la forma en la que se diseñó el sistema completo, las imágenes que usa el módulo de RA o incluso el módulo completo podría ser reemplazado fácilmente sin comprometer el funcionamiento general.

Con el [Diseño de la interfaz gráfica](#) se pretende darle a los administradores y trabajadores de [BubbleTown®](#) una forma fácil en la cual sin necesitar conocer ampliamente el sistema lo puedan iniciar, dejando en claro que solo al acceder como administrador se podrán hacer los cambios en la publicidad estática, en los datos de las bebidas y hacer respaldos de los datos recabados, esto para siempre mantener la seguridad e integridad de los datos importantes que se manejan.

CAPÍTULO 6

Pruebas y Resultados

6.1. Introducción al capítulo.

A lo largo de este capítulo se presentarán las diversas pruebas que se realizan al sistema para demostrar que el proyecto cumple con los [Requerimientos funcionales](#) y el funcionamiento que se ha establecido. Por lo anterior, para cada requerimiento funcional se presenta una tabla que detalla:

- Objetivo de la prueba,
- Prueba,
- Caso de prueba,
- Criterios de éxito,
- Interfaz asociada,
- Resultados,
- Consideraciones especiales en caso de ser necesarias.

La mayoría de las pruebas unitarias presentadas para verificar el correcto funcionamiento de los componentes serán realizadas con el apoyo del *framework unittest* de Python, que permite automatizarlas con código; sin embargo, también se colocan imágenes de los resultados de cada prueba con el objetivo de facilitar al lector la comprensión del funcionamiento del sistema ante los distintos casos.

6.2. Plan de pruebas

Nombre del sistema: Prototipo de sistema de publicidad dirigida a clientes usando visión por computadora y aprendizaje profundo	Versión: 1.0
Plan de pruebas	Fecha: 11/12/2020
Documento Plan de Pruebas	

Tabla 6.1: Especificaciones de las pruebas.

6.2.1. RF01 - Detección de rostros

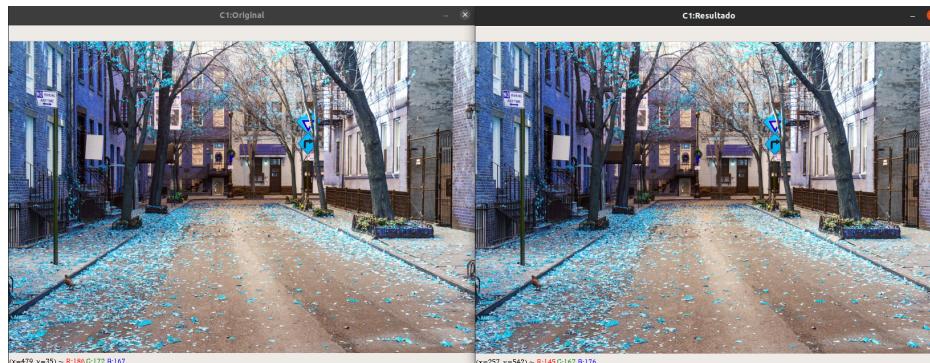
RF01 - Detección de rostros	
Objetivo de la prueba:	Verificar el correcto funcionamiento de la red encargada de detectar rostros (Single Shot Detection).
Prueba:	La red neuronal Single Shot Detection será alimentada con cuatro imágenes diferentes, que cumplan con los requisitos establecidos en la sección de Preprocesamiento de la imagen , para verificar su funcionamiento en diferentes condiciones.
Casos de prueba:	<ul style="list-style-type: none"> • C1: Introducir una imagen sin ningún rostro. • C2: Introducir una imagen con un solo rostro. • C3: Introducir una imagen con más de un rostro. • C4: Introducir una imagen con la cara de un animal.
Criterios de éxito:	<ul style="list-style-type: none"> • C1: La red Single Shot Detection no detecta ningún rostro. • C2: La red Single Shot Detection detecta solo un rostro. • C3: La red Single Shot Detection detecta todos los rostros de la imagen. • C4: La red Single Shot Detection no detecta ningún rostro.
Interfaz asociada:	No aplica.

Continua en la página siguiente.

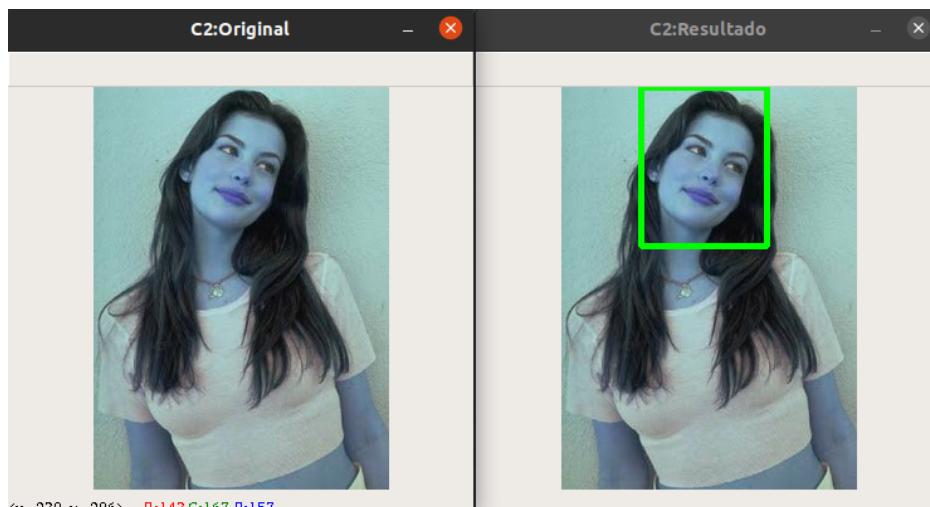
RF01 - Detección de rostros

Resultados:

- C1: Del lado izquierdo de la imagen se puede observar la imagen utilizada que, en este caso no contiene ningún rostro, mientras que del lado derecho se puede visualizar claramente como no se obtuvo ninguna detección.



- C2: Del lado izquierdo se puede apreciar que en la imagen solo se encuentra un rostro que, como se observa del lado derecho de la imagen, es correctamente detectado por la red.



Continua en la página siguiente.

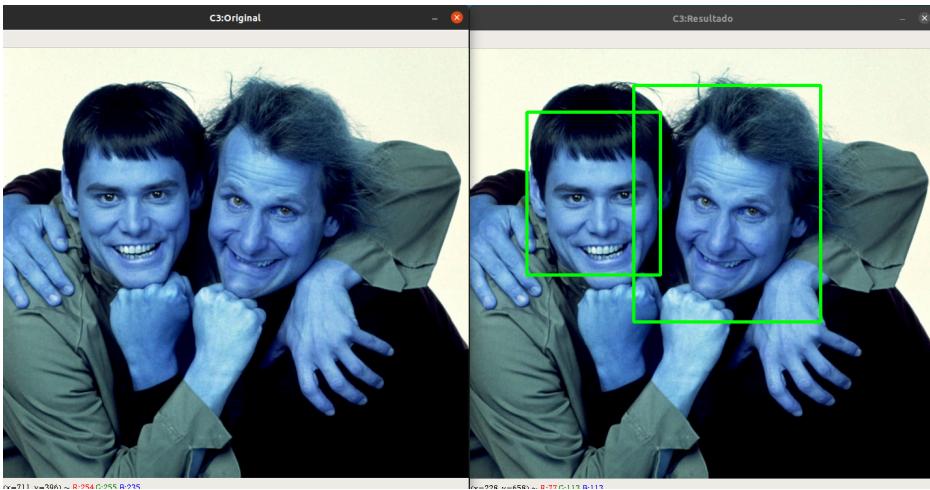
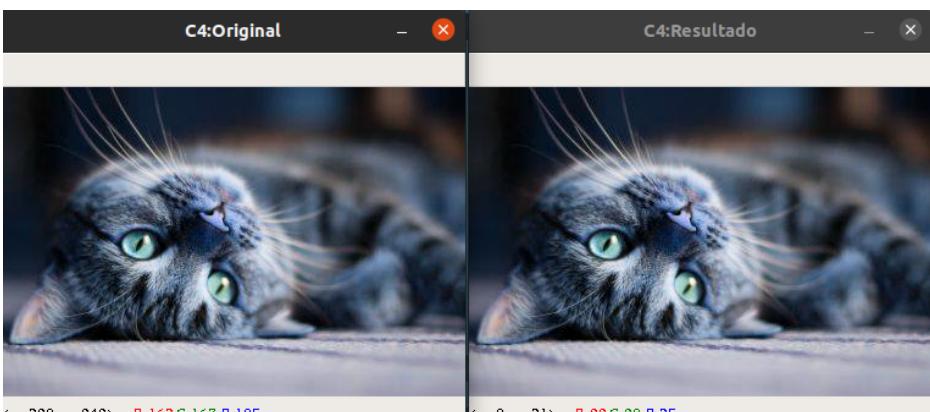
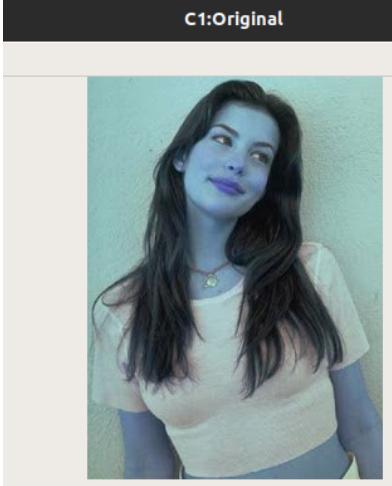
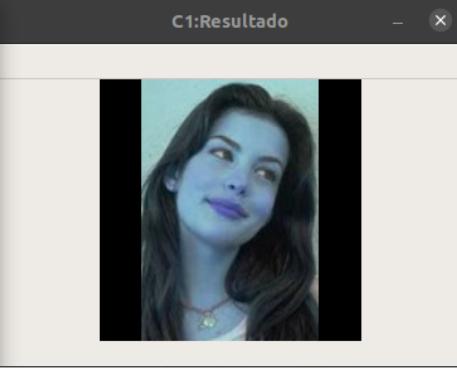
RF01 - Detección de rostros	
Resultados:	<ul style="list-style-type: none"> C3: La prueba para este caso concluye de forma satisfactoria debido a que la red detectó adecuadamente los dos rostros. 
	<ul style="list-style-type: none"> C4: Al introducir en la red la imagen de un gato la red no detectó ningún rostro, lo cual es correcto ya que la red fue desarrollada para detectar rostros (caras humanas), por lo que las caras que no son humanas no deben ser detectadas. 
Consideraciones especiales:	Las pruebas presentadas aquí son realizadas como demostración de los distintos casos que se le pueden presentar a la red neuronal; sin embargo, un análisis más adecuado sobre la evaluación de esta red pueden ser consultados en la sección Pruebas de detección de rostros .

Tabla 6.2: Pruebas para corroborar RF01.

6.2.2. RF02 - Preprocesamiento de rostros

RF02 - Preprocesamiento de rostros	
Objetivo de la prueba:	Comprobar que la red Single Shot Detection preprocesa los rostros detectados ya que estos alimentarán la Red neuronal convolucional para Edad y Género y la Red neuronal convolucional para Big Five para evitar que, en caso de que la red no preprocese adecuadamente, se generen excepciones que resultarían en una falla grave del sistema.
Prueba:	Alimentar la red Single Shot Detection con una imagen de resolución mayor a 208x208, que cumpla con los requisitos establecidos en la sección de Preprocesamiento de la imagen , y verificar que a la salida las dimensiones del rostro son de 208x208x3.
Casos de prueba:	<ul style="list-style-type: none"> • C1: Se alimenta la red Single Shot Detection con una imagen que contenga el rostro de una persona. • C2: Se alimenta la red Single Shot Detection con una imagen que contenga el rostro de dos personas.
Criterios de éxito:	<ul style="list-style-type: none"> • C1: El tamaño del rostro que es returnedo por la red debe ser forzosamente de dimensiones 208x208x3. • C2: El tamaño de los dos rostros que es returnedo por la red debe ser forzosamente de dimensiones 208x208x3.
Interfaz asociada:	No aplica.
Resultados:	<ul style="list-style-type: none"> • C1: La red logró preprocesar el rostro adecuadamente ya que no alteró la relación del tamaño del rostro. <div style="display: flex; justify-content: space-around;"> <div style="text-align: center;">  <p>C1:Original</p> <p>(x=233 y=293) ~ R:108 G:120 B:148</p> </div> <div style="text-align: center;">  <p>C1:Resultado</p> </div> </div>

Continua en la página siguiente.

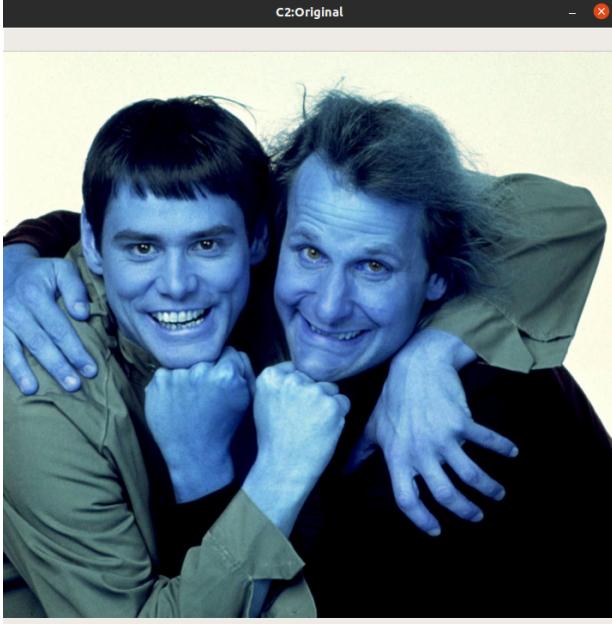
RF02 - Preprocesamiento de rostros	
Resultados:	<ul style="list-style-type: none"> • C2: A pesar de que los rostros se encuentran muy cercanos entre sí, la red fue capaz de preprocesarlos satisfactoriamente. <div style="display: flex; justify-content: space-around;"> <div style="text-align: center;">  <p>C2:Original</p> <p>(x=714 y=172) ~ R:248 G:248 B:230</p> </div> <div style="text-align: center;">  <p>C2:Resultado_2</p> </div> <div style="text-align: center;">  <p>C2:Resultado_1</p> </div> </div>
Consideraciones especiales:	<p>Se entiende por preprocesamiento a la tarea por la cual se redimensiona la imagen del rostro respetando la relación de alto y ancho del rostro en la imagen original. Para más información, consulte la sección de Recorte y procesamiento de los rostros detectados.</p>

Tabla 6.3: Pruebas para corroborar RF02.

6.2.3. RF03 - Estimación de edad y género

RF03 - Estimación de edad y género	
Objetivo de la prueba:	Comprobar que la red neuronal convolucional encargada de estimar la edad y género es capaz de obtener valores aceptables.
Prueba:	Alimentar la red neuronal con imágenes de personas distintas y evaluar los resultados.
Casos de prueba:	<ul style="list-style-type: none"> • C1: Se alimenta la red neuronal con imágenes de los tres autores de este trabajo: hombres de 22 años. • C2: Se alimenta la red neuronal con la imagen de un hombre de 13 años. • C3: Se alimenta la red neuronal con la imagen de un hombre de 27 años. • C4: Se alimenta la red neuronal con la imagen de una mujer de 34 años. • C5: Se alimenta la red neuronal con la imagen de una mujer de 50 años.
Criterios de éxito:	<ul style="list-style-type: none"> • C1: La red neuronal clasifica los rostros de los tres autores de este trabajo como hombres con una edad entre 21 y 33 años. • C2: La red neuronal clasifica el rostro como hombre con una edad entre 7 y 19 años. • C3: La red neuronal clasifica el rostro como hombre con una edad entre 21 y 33 años. • C4: La red neuronal clasifica el rostro como mujer con una edad entre 28 y 40 años. • C5: La red neuronal clasifica el rostro como mujer con una edad entre 44 y 56 años.
Interfaz asociada:	No aplica.
Resultados:	<ul style="list-style-type: none"> • C1: La red clasifica a los tres autores del presente trabajo como hombres con una edad de 25, 24 y 24 años respectivamente, por lo que se considera como una prueba exitosa. <div style="text-align: center; margin-top: 10px;">  </div>

Continua en la página siguiente.

RF03 - Estimación de edad y género	
Resultados:	<ul style="list-style-type: none"> C2: La red detecta un hombre de 18 años por lo que se considera que la prueba concluye de forma exitosa.  E: 18 G:1 <ul style="list-style-type: none"> C3: La red clasifica a un hombre con una edad de 21 años por lo que se considera como una prueba exitosa.  E: 21 G:1
	<ul style="list-style-type: none"> C4: Es reconocida como una mujer de 48 años. En este caso la edad estimada varía mucho con relación a la edad real. Por ese motivo se considera como fallida esta prueba.  E: 48 G:0 <ul style="list-style-type: none"> C5: Se detecta a un hombre de 27 años de edad, por lo que esta prueba es considerada como fallida ya que falló en la estimación de edad, así como en la clasificación del género.  E: 27 G:1
Consideraciones especiales:	<p>Esta prueba se realizó con el objetivo de mostrar al lector los posibles casos en los que la red multitarea puede verse involucrada. Sin embargo, en la sección de la Red neuronal convolucional para Edad y Género pueden consultarse de forma más detallada los resultados obtenidos para esta red.</p>

Tabla 6.4: Pruebas para corroborar RF03.

6.2.4. RF04 - Estimación de personalidad

RF04 - Estimación de personalidad	
Objetivo de la prueba:	Comprobar que la red neuronal convolucional encargada de estimar la personalidad es capaz de otorgar estimaciones aceptables para el objetivo de este sistema.
Prueba:	<p>Proporcionar a la red imágenes de personas en formato selfie de las que se conoce su personalidad para comprobar el funcionamiento de la red. En las imágenes resultantes se muestran los valores obtenidos de la red neuronal redondeados a 3 decimales del lado derecho y del lado izquierdo los datos de personalidad reales. Las letras presentadas son abreviaciones de las personalidades de la siguiente manera:</p> <ul style="list-style-type: none"> • O (Openness to Experiencie): Apertura a la Experiencia. • C (Conscientiousness): Conciencia. • E (Extraversion): Extraversión. • A (Agreeableness): Amabilidad. • N (Neuroticism): Inestabilidad Emocional.
Casos de prueba:	<ul style="list-style-type: none"> • C1: Se propaga a través de la red una imagen de una persona con valores altos en el rasgo de Apertura a la experiencia. • C2: Se propaga a través de la red una imagen de una persona con valores altos en el rasgo de Conciencia. • C3: Se propaga a través de la red una imagen de una persona de una raza distinta a la blanca y con valores altos en los rasgos de Conciencia y Amabilidad. • C4: Se propaga a través de la red una imagen de una persona con valores balanceados entre los rasgos de personalidad.
Criterios de éxito:	Tener un error de máximo de 0.22 (propuesto por M. Moreno Sotelo en [51]) en cada estimación de los 5 rasgos de personalidad del modelo Big Five. El criterio de éxito aplica en todos los casos (C1, C2, C3 y C4).
Interfaz asociada:	No aplica.

Continua en la página siguiente.

RF04 - Estimación de personalidad

Resultados:

- C1: La red proporciona la personalidad en el rango correcto del error permitido, ya que sólo hay un +0.06 de diferencia entre los valores de Apertura.
- C2: Aunque el rasgo de Conciencia (+0.15) no es de los más acertados, es admitido por la red dentro del rango correcto de error permitido.



- C3: Aunque los rasgos de Conciencia (+0.21) y Amabilidad (+0.21) no son de los más acertados, son admitidos por la red dentro del rango correcto de error permitido. La efectividad de la red con una persona de otra raza es acertada.
- C4: La red proporciona la personalidad en el rango correcto del error permitido. Esta se comporta bien en la mayoría de los rasgos, siendo su peor estimación la Apertura (-0.14) pero de igual forma entra en el rango de error propuesto.



Consideraciones especiales:

Esta red no fue diseñada ni entrenada durante el desarrollo de este sistema. En la sección de la [Red neuronal convolucional para Big Five](#) se explica de forma breve el fin de esta red, así como el proceso que se llevó a cabo para usarla en el sistema.

Tabla 6.5: Pruebas para corroborar RF04.

6.2.5. RF05 - Obtención de la temperatura

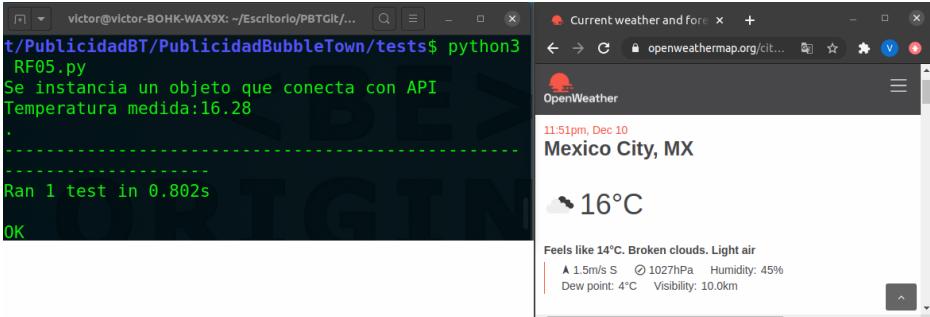
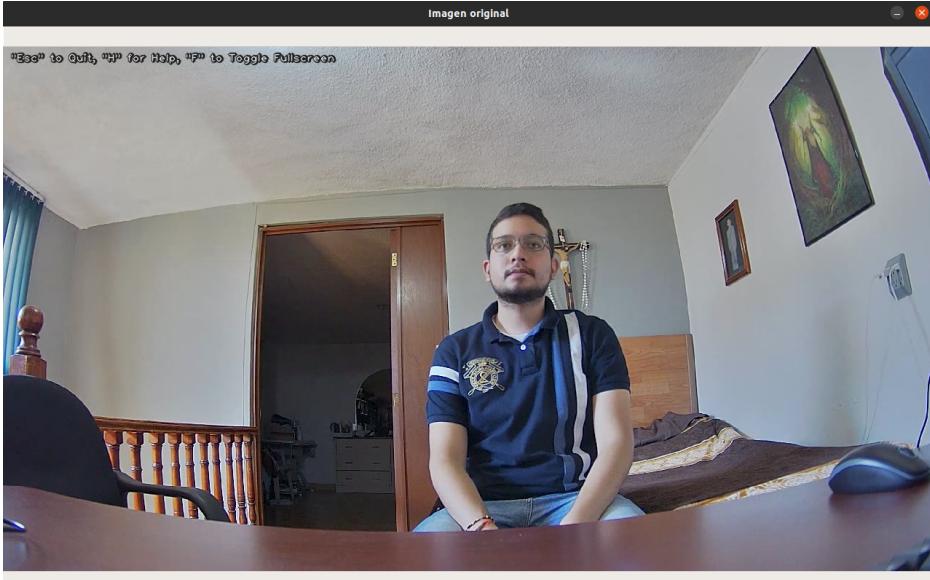
RF05 - Obtención de la temperatura	
Objetivo de la prueba:	El sistema requiere de la temperatura ambiental para recomendar una modalidad de bebida adecuada. Por ello con esta prueba se comprobará que el sistema pueda usar adecuadamente la API y se regresan los valores correctos correspondientes a la Ciudad de México.
Prueba:	Se instancia un objeto de la clase Temperatura para consultar la temperatura ambiental en grados celcius haciendo uso de la API y, posteriormente se verifica el valor a través de su página web.
Casos de prueba:	<ul style="list-style-type: none"> • C1: Se solicita la temperatura a la API.
Criterios de éxito:	<ul style="list-style-type: none"> • C1: El valor returnedo por la API coincide con el valor que se está reportando a través de su página web para la Ciudad de México.
Interfaz asociada:	No aplica.
Resultados:	<ul style="list-style-type: none"> • C1: La temperatura que proporcionó la API fue de 16º Celcius en el momento en que su página web reportaba 16º Celcius en la Ciudad de México. Con esta prueba se confirma el correcto funcionamiento de la API de temperatura. 
Consideraciones especiales:	Sin consideraciones.

Tabla 6.6: Pruebas para corroborar RF05.

6.2.6. RF07 - Generación de realidad aumentada

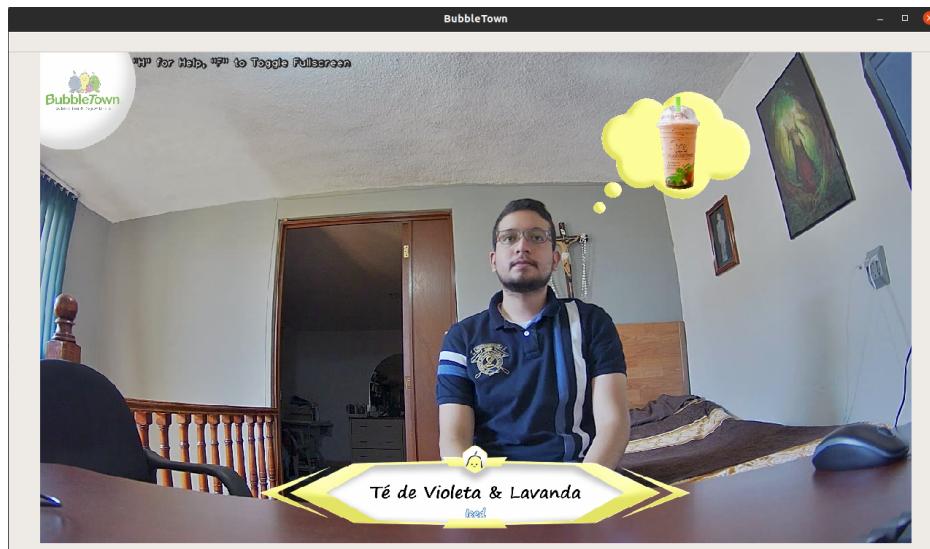
RF07 - Generación de realidad aumentada	
Objetivo de la prueba:	Mostrar que cuando la Single Shot Detection detecta un rostro, el módulo de publicidad dirigida funciona adecuadamente.
Prueba:	Alimentar al sistema con un video y observar el resultado en la salida del Visor.
Casos de prueba:	<ul style="list-style-type: none"> • C1: Se alimenta al sistema con el video de una persona.
Criterios de éxito:	<ul style="list-style-type: none"> • C1: El sistema debe proyectar en el Visor la imagen recibida de la cámara pero agregando un globo de pensamiento, el nombre de la bebida y la modalidad de la misma.
Interfaz asociada:	No aplica.
Resultados:	<ul style="list-style-type: none"> • C1: En la primera imagen se observa la captura recibida desde el video. <div style="text-align: center;">  </div>

Continua en la página siguiente.

RF07 - Generación de realidad aumentada

Resultados:

Mientras que en la segunda se puede observar el globo de pensamiento con una bebida dentro y en la parte inferior se recomienda la bebida "Té de Violeta Lavanda" en su modalidad "Iced". Con ello se demuestra que el módulo de realidad aumentada está funcionando adecuadamente.

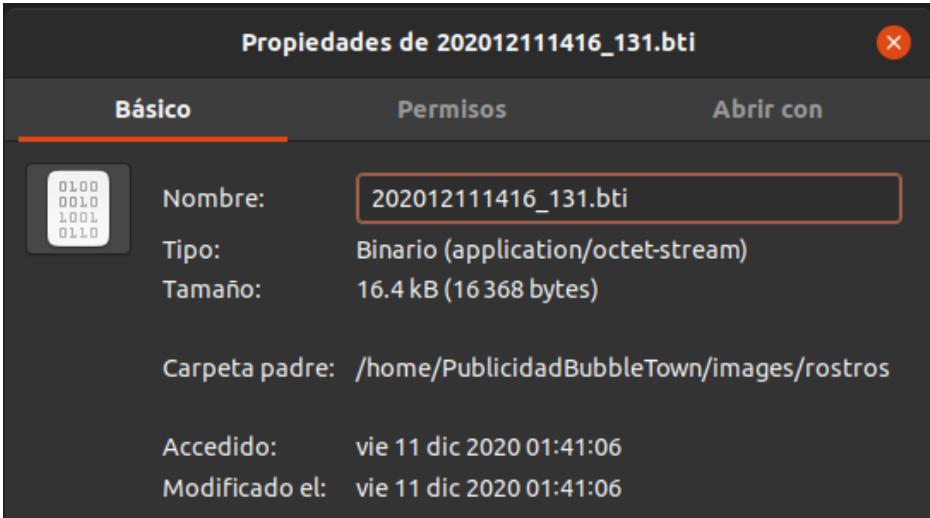


Consideraciones especiales:

Sin consideraciones.

Tabla 6.7: Pruebas para corroborar RF07.

6.2.7. RF08 - Almacenamiento de rostros

RF08 - Almacenamiento de rostros	
Objetivo de la prueba:	Se debe comprobar que el sistema almacenará en todo momento los rostros que se detectan de forma cifrada por motivos de seguridad.
Prueba:	Se crea una instancia de la clase Rostro, se invoca su método "create()" y se verifica que el elemento se insertó correctamente en la base de datos, se creó un archivo con extensión "bti" y es ilegible por medio de un visor de imágenes.
Casos de prueba:	<ul style="list-style-type: none"> • C1: Se instancian los valores del objeto que se quieren almacenar y se invoca el método "create()" que es encargado de insertar los valores en la base de datos y crear el archivo de la imagen.
Criterios de éxito:	<ul style="list-style-type: none"> • C1: El sistema almacena el rostro en la base de datos, se crea el archivo con extensión "bti" que es un archivo cifrado.
Interfaz asociada:	No aplica.
Resultados:	<ul style="list-style-type: none"> • C1: En la primera imagen se puede observar cómo el sistema genera archivos con extensión "bti". 

Continua en la página siguiente.

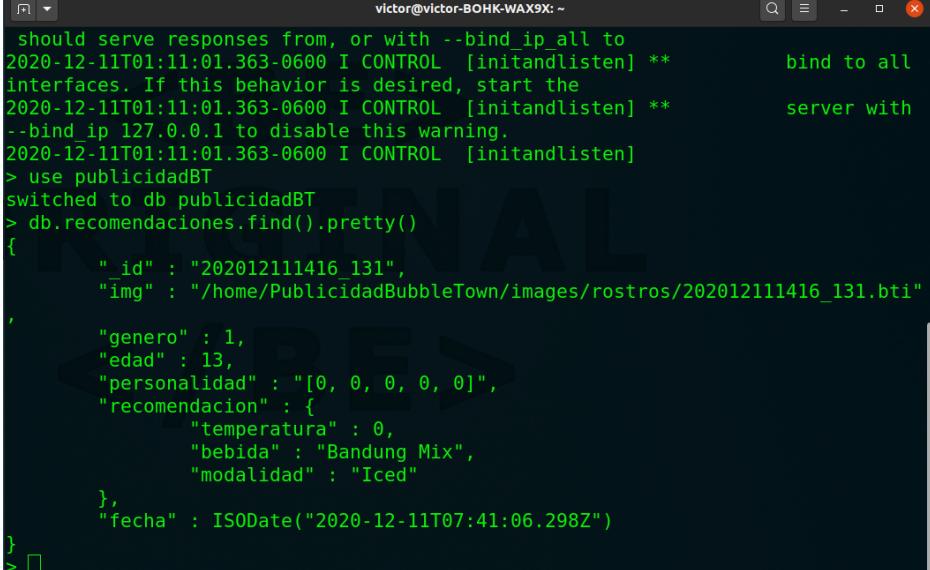
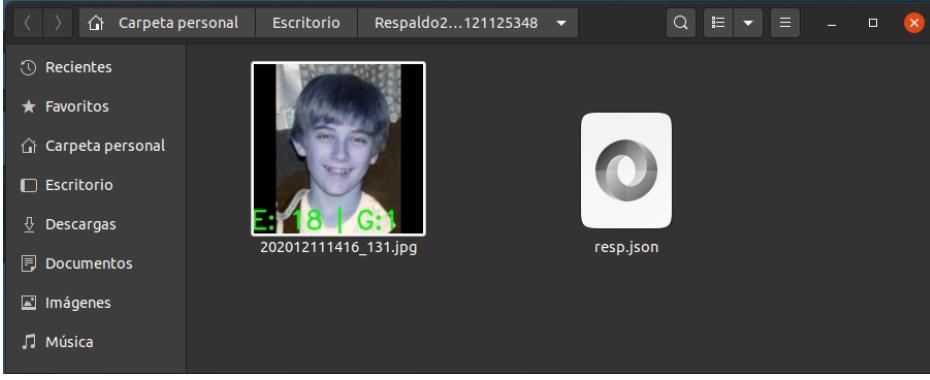
RF08 - Almacenamiento de rostros	
Resultados:	<p>Debido a que la información de las imágenes se encuentra cifrada, si se intenta abrir el archivo con algún visor de imágenes, se mostrará un error como se observa en la segunda figura.</p>  <p>Por último, en la tercera imagen se puede observar cómo es almacenada la recomendación en la base de datos.</p>  <pre> victor@victor-BOHK-WAX9X: ~ should serve responses from, or with --bind_ip_all to 2020-12-11T01:11:01.363-0600 I CONTROL [initandlisten] ** bind to all interfaces. If this behavior is desired, start the 2020-12-11T01:11:01.363-0600 I CONTROL [initandlisten] ** server with --bind_ip 127.0.0.1 to disable this warning. 2020-12-11T01:11:01.363-0600 I CONTROL [initandlisten] > use publicidadBT switched to db publicidadBT > db.recomendaciones.find().pretty() { "_id" : "202012111416_131", "img" : "/home/PublicidadBubbleTown/images/rostros/202012111416_131.bti" , "genero" : 1, "edad" : 13, "personalidad" : "[0, 0, 0, 0, 0]", "recomendacion" : { "temperatura" : 0, "bebida" : "Bandung Mix", "modalidad" : "Iced" }, "fecha" : ISODate("2020-12-11T07:41:06.298Z") } > [] </pre>
Consideraciones especiales:	Sin consideraciones.

Tabla 6.8: Pruebas para corroborar RF08.

6.2.8. RF09/RF10 - Almacenamiento y respaldo de recomendaciones

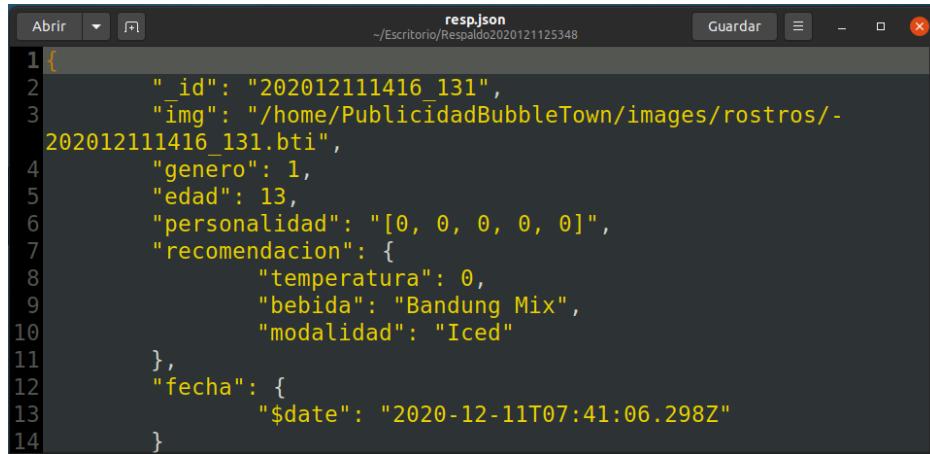
RF09/RF10 - Almacenamiento y respaldo de recomendaciones	
Objetivo de la prueba:	Asegurar que el procedimiento de respaldo de las recomendaciones funciona correctamente, generando una carpeta que contendrá las imágenes descifradas y un archivo JSON con todas las recomendaciones que se han realizado desde la fecha colocada.
Prueba:	Se creará una instancia de la clase Respaldo, que es la encargada de acceder a la base de datos y generar la copia de los archivos.
Casos de prueba:	<ul style="list-style-type: none"> • C1: Se invoca el método "generarRespaldo()" de una instancia de Respaldo indicando la ruta donde se almacenará y la fecha desde la que deberá generarse.
Criterios de éxito:	<ul style="list-style-type: none"> • C1: Se crea una carpeta que tendrá en su interior un archivo JSON con todos los datos de las recomendaciones, así como las imágenes descifradas.
Interfaz asociada:	No aplica.
Resultados:	<ul style="list-style-type: none"> • C1: La primer prueba muestra cómo se ejecutó de forma adecuada el código de la prueba programada. <pre>(base) victor@victor-BOHK-WAX9X:~/Escritorio/PBTGit/PublicidadBT/PublicidadBubbleTown/tests\$ python3 RF10.py C1: Se invoca el método "generarRespaldo" de una instancia de Respaldo. 1999-01-01 00:00:00 /home/victor/anaconda3/lib/python3.7/site-packages/Crypto/Random/_UserFriendlyRNG.py:77: DeprecationWarning: time.clock has been deprecated in Python 3.3 and will be removed from Python 3.8: use time.perf_counter or time.process_time instead t = time.clock() Padding len:7 2020-12-11T02:53:48.247-0600 connected to: localhost 2020-12-11T02:53:48.248-0600 exported 1 record . ----- Ran 1 test in 0.048s OK</pre>
	Luego, en la segunda imagen se puede observar la carpeta creada con la imagen descifrada y un archivo JSON.
	

Continua en la página siguiente.

RF09/RF10 - Almacenamiento y respaldo de recomendaciones

Resultados:

En la última imagen se muestra el archivo JSON con las recomendaciones.



```

1 {
2     "_id": "202012111416_131",
3     "img": "/home/PublicidadBubbleTown/images/rostros/-202012111416_131.bti",
4     "genero": 1,
5     "edad": 13,
6     "personalidad": "[0, 0, 0, 0, 0]",
7     "recomendacion": {
8         "temperatura": 0,
9         "bebida": "Bandung Mix",
10        "modalidad": "Iced"
11    },
12    "fecha": {
13        "$date": "2020-12-11T07:41:06.298Z"
14    }
}

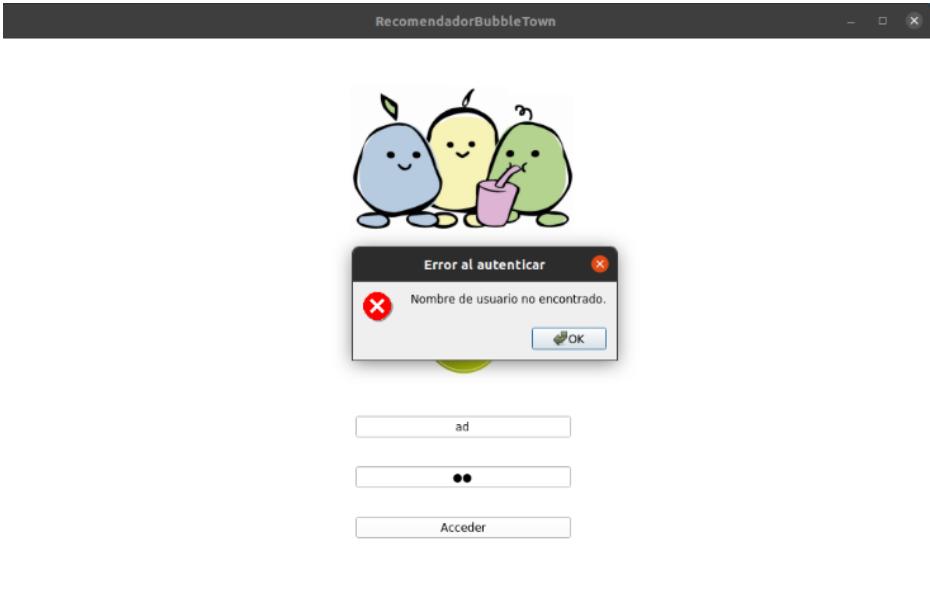
```

Consideraciones especiales:

Sin consideraciones.

Tabla 6.9: Pruebas para corroborar RF10.

6.2.9. RF11 - Autenticación de administrador

RF11 - Autenticación de administrador	
Objetivo de la prueba:	Verificar el proceso de autenticación de un usuario a partir de las credenciales proporcionadas.
Prueba:	Se ingresan credenciales no válidas, luego un usuario válido pero con contraseña incorrecta y al final con ambos datos válidos.
Casos de prueba:	<ul style="list-style-type: none"> • C1: Se introduce un nombre de administrador inexistente. • C2: Se introduce un nombre de administrador existente pero contraseña incorrecta. • C3: Se introduce un nombre de administrador y contraseña válidos
Criterios de éxito:	<ul style="list-style-type: none"> • C1: El nombre de administrador no coincide con ningún nombre válido, por lo que el sistema le muestra que su nombre de usuario no es válido. • C2: El sistema muestra un mensaje indicando que la contraseña es incorrecta. • C3: El proceso de autenticación concluye exitosamente, enviando al usuario al menú de opciones.
Interfaz asociada:	IU1 Login de acceso y IU2 Pantalla principal.
Resultados:	<ul style="list-style-type: none"> • C1: El sistema busca el nombre de usuario pero al no ser válido genera un error y muestra un mensaje indicando que no se encontró ese nombre de usuario. 

Continua en la página siguiente.

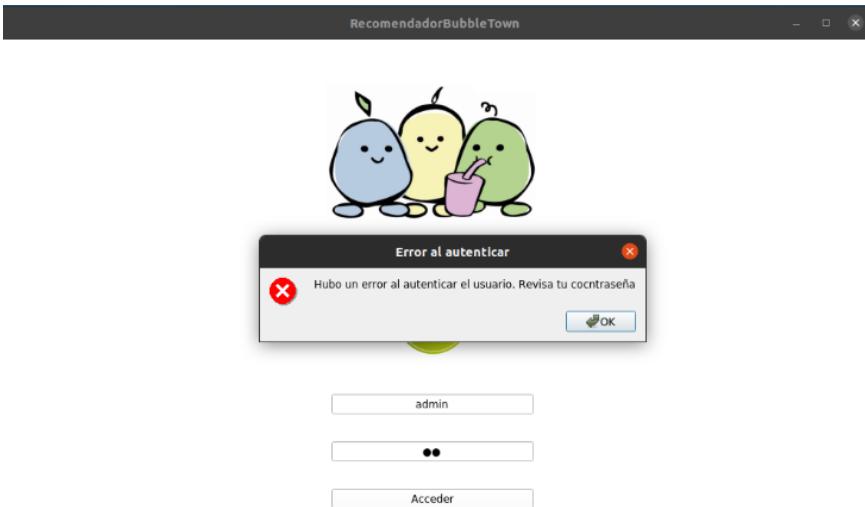
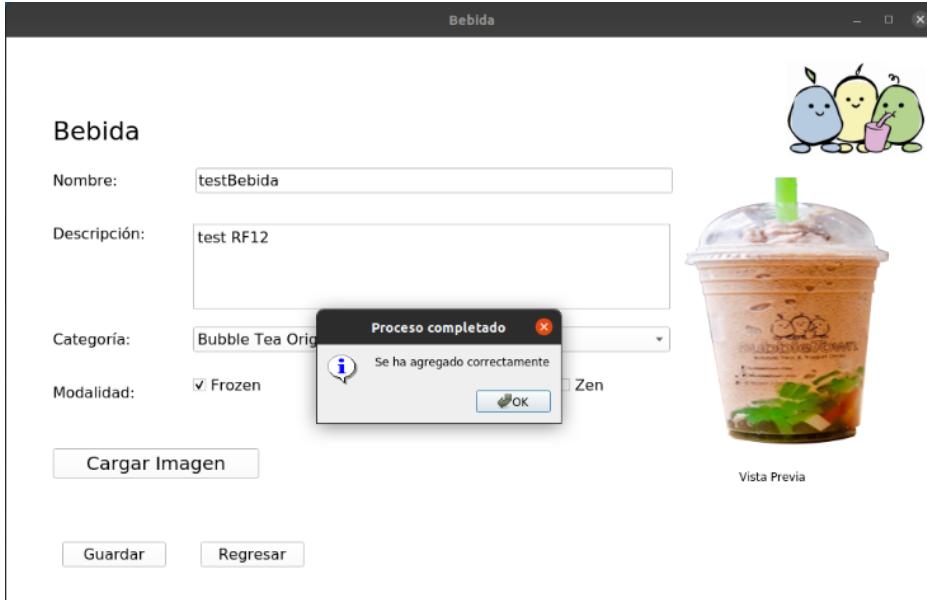
RF11 - Autenticación de administrador	
Resultados:	<ul style="list-style-type: none"> C2: El sistema reconoce el nombre de usuario como válido pero la contraseña no coincide, por lo que le indica al usuario que revise su contraseña. 
	<ul style="list-style-type: none"> C3: El usuario y contraseña son reconocidos como válidos por lo que le muestra al usuario la ventana del menú como se muestra en la imagen. 
Consideraciones especiales:	Sin consideraciones.

Tabla 6.10: Pruebas para corroborar RF11.

6.2.10. RF12 - Nueva bebida

RF12 - Nueva bebida	
Objetivo de la prueba:	Asegurar que el sistema es capaz de agregar una nueva bebida para mantener el catálogo actualizado.
Prueba:	El formulario correspondiente es completado y se encarga de instanciar los atributos de un objeto perteneciente a la clase "Bebida" para posteriormente invocar a su método "create()".
Casos de prueba:	<ul style="list-style-type: none"> • C1: Se introducen los datos con un nombre de bebida que no existe en el catálogo. • C2: Se introducen los datos con el nombre de una bebida ya existente en el catálogo.
Criterios de éxito:	<ul style="list-style-type: none"> • C1: Se agrega a la base de datos la nueva bebida. • C2: El sistema no permite la inserción de la nueva bebida porque encuentra una bebida que ya existía.
Interfaz asociada:	 IU8 Pantalla para agregar o modificar bebida.
Resultados:	<ul style="list-style-type: none"> • C1: La bebida "testBebida" es agregada satisfactoriamente al sistema debido a que no existe ninguna otra bebida con el mismo nombre. <div style="text-align: center;">  </div>

Continua en la página siguiente.

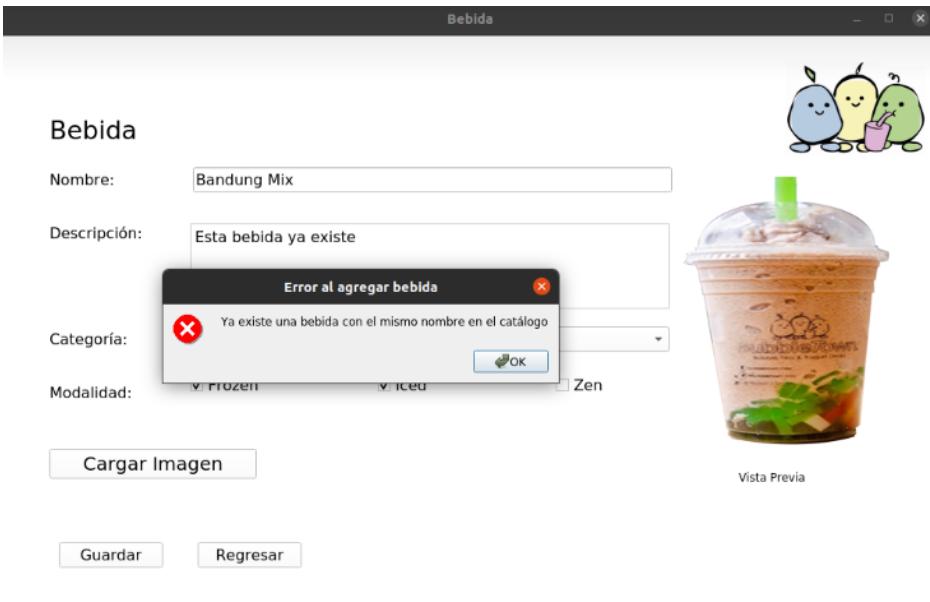
RF12 - Nueva bebida	
Resultados:	<ul style="list-style-type: none"> C2: Al intentar agregar una nueva bebida con el nombre "Bandung Mix" se genera una excepción que le indica al usuario que ya existe una bebida con el mismo nombre en el catálogo. 
Consideraciones especiales:	Sin consideraciones.

Tabla 6.11: Pruebas para corroborar RF12.

6.2.11. RF13 - Actualización de bebida

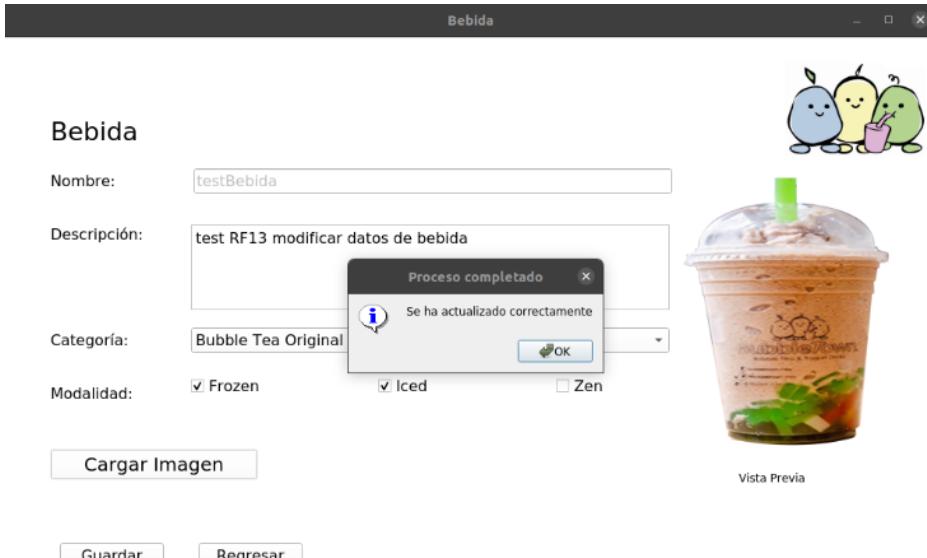
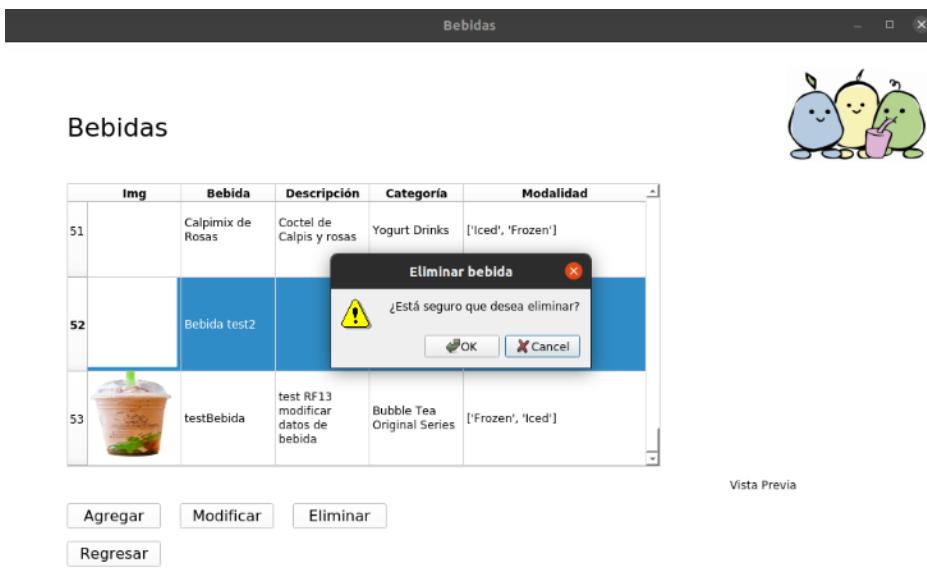
RF13 - Actualización de bebida	
Objetivo de la prueba:	Asegurar que el sistema es capaz de modificar la información de una bebida para mantener el catálogo actualizado.
Prueba:	Se seleccionará una bebida del menú y se dará clic en modificar. Dentro de la IU8 se cambiarán algunos valores y dará clic en "Guardar".
Casos de prueba:	<ul style="list-style-type: none"> • C1: Se cambian los valores almacenados de una bebida.
Criterios de éxito:	<ul style="list-style-type: none"> • C1: Los campos son actualizados correctamente.
Interfaz asociada:	 IU8 Pantalla para agregar o modificar bebida.
Resultados:	<ul style="list-style-type: none"> • C1: Se realiza la modificación de la descripción y al dar clic en "Guardar" el sistema indica que se ha actualizado correctamente por lo que la prueba concluyó adecuadamente. <div style="text-align: center;">  <p>The screenshot shows the 'Bebida' modification interface. It includes fields for Nombre (testBebida), Descripción (test RF13 modificar datos de bebida), Categoría (Bubble Tea Original), and Modalidad (Frozen, Iced, Zen). A confirmation dialog box titled 'Proceso completado' says 'Se ha actualizado correctamente' with an 'OK' button. To the right is a preview image of a bubble tea drink with a straw and a small cartoon illustration of three characters.</p> </div>
Consideraciones especiales:	Sin consideraciones.

Tabla 6.12: Pruebas para corroborar RF13.

6.2.12. RF14 - Eliminación de bebida

RF14 - Eliminación de bebida	
Objetivo de la prueba:	Asegurar que el sistema es capaz de eliminar una bebida cuando esta ya no se encuentre a la venta, con la finalidad de mantener el catálogo actualizado.
Prueba:	Al cargar la interfaz de usuario con la lista de las bebidas registradas, se seleccionará una bebida y se procederá a eliminarla.
Casos de prueba:	<ul style="list-style-type: none"> • C1: Se selecciona y se elimina una bebida.
Criterios de éxito:	<ul style="list-style-type: none"> • C1: La bebida ya no aparece en la lista debido a que ha sido eliminada correctamente.
Interfaz asociada:	 IU7 Pantalla de gestión de bebidas.
Resultados:	<ul style="list-style-type: none"> • C1: En la primera imagen se observa como tras seleccionar una bebida y dar clic en "Eliminar" el sistema pide al usuario que confirme la acción. <div style="text-align: center;">  <p>The screenshot shows a table titled 'Bebidas' with three rows of data. Row 51 contains 'Calpimix de Rosas'. Row 52 contains 'Bebida test2'. Row 53 contains 'testBebida'. A modal dialog box titled 'Eliminar bebida' is overlaid on the table, containing the message '¿Está seguro que desea eliminar?' with 'OK' and 'Cancel' buttons. The interface includes buttons for 'Agregar', 'Modificar', 'Eliminar', and 'Regresar' at the bottom.</p> </div>

Continua en la página siguiente.

RF14 - Eliminación de bebida

Resultados:

En caso de que el usuario confirme la acción se procederá a la eliminación y tras actualizarse la lista de bebidas, la bebida previamente seleccionada ya no existirá.

Bebidas





Img	Bebida	Descripción	Categoría	Modalidad
50	Calpimix de Menta	Coctel de Calpis y menta.	Yogurt Drinks	[‘Frozen’, ‘Iced’]
51	Calpimix de Rosas	Coctel de Calpis y rosas	Yogurt Drinks	[‘Iced’, ‘Frozen’]
52	testBebida	test RF13 modificar datos de bebida	Bubble Tea Original Series	[‘Frozen’, ‘Iced’]

Vista Previa

Consideraciones especiales:

Sin consideraciones.

Tabla 6.13: Pruebas para corroborar RF14.

198

Av. Juan de Dios Bátiz esq. Miguel Othón de Mendizábal S/N Col. Lindavista, GAM, D. F. 57296000 Ext. 52054

6.2.13. RF15 - Comprobación de bebida

RF15 - Comprobación de bebida	
Objetivo de la prueba:	Verificar que el sistema revisa la existencia de una bebida cuando se intenta crear una recomendación.
Prueba:	Se crearán instancias de la clase "Recomendacion" y se le colocarán nombres de bebidas. En caso de que la bebida exista no deben generarse errores, mientras que si la bebida no existe en la base de datos, entonces tendrá que lanzarse una excepción.
Casos de prueba:	<ul style="list-style-type: none"> ● C1: Se intenta crear una recomendación con una bebida existente en la base de datos. ● C2: Se intenta crear una recomendación con una bebida inexistente en la base de datos.
Criterios de éxito:	<ul style="list-style-type: none"> ● C1: El sistema permite la asignación de la bebida en la recomendación. ● C2: El sistema genera una excepción debido a la inexistencia de la bebida en la base de datos.
Interfaz asociada:	No aplica.
Resultados:	<p>Para comprobar la verificación se implementó una prueba automatizada que ejecuta ambas posibilidades y si en ambos casos cumple con las respuestas esperadas, entonces se considera que las pruebas fueron exitosas, tal cual como se muestra en la imagen.</p> <pre>(base) victor@victor-BOHK-WAX9X:~/Escritorio/PBTGit/PublicidadBT/PublicidadBubbleTown/tests\$ python3 RF15.py C1: Se intenta crear una recomendación con una bebida existente en la base de datos. .Se intenta crear una recomendación con una bebida inexistente en la base de datos. Se genera error Aun no existe una bebida con ese nombre . Ran 2 tests in 0.007s OK</pre> <ul style="list-style-type: none"> ● C1: Para el primer caso, el proceso de creación de una recomendación se llevó a cabo con normalidad. ● C2: En el segundo caso se generó un error que indicaba que aún no existía una bebida con ese nombre. Como este era el comportamiento esperado, la prueba de este caso también concluyó exitosamente.
Consideraciones especiales:	Sin consideraciones.

Tabla 6.14: Pruebas para corroborar RF15.

6.2.14. RF16 - Descifrado de rostros

RF16 - Descifrado de rostros	
Objetivo de la prueba:	Debido a que el descifrado se realiza de forma automática si el usuario se autentica correctamente, se debe comprobar que el proceso de Respaldo y por ende el descifrado de los rostros, no se puede iniciar si no se autentica correctamente un administrador.
Prueba:	Se intentará hacer el respaldo de las recomendaciones de la base de datos con credenciales erróneas y válidas en varias ocasiones.
Casos de prueba:	<ul style="list-style-type: none"> • C1: Se intenta realizar el respaldo con credenciales válidas • C2: Se intenta hacer el respaldo con credenciales erróneas. • C3: Se intenta hacer el respaldo por tres ocasiones seguidas con credenciales erróneas.
Criterios de éxito:	<ul style="list-style-type: none"> • C1: El proceso de respaldo y por consiguiente, el descifrado automático de las recomendaciones se concluye con éxito. • C2: El sistema notifica al usuario que las credenciales no son válidas. • C3: El sistema indica al usuario que han sido demasiados intentos y se cierra.
Interfaz asociada:	 IU6 Pantalla para hacer respaldo.
Resultados:	<ul style="list-style-type: none"> • C1: Se puede observar en la imagen inferior que efectivamente el sistema comenzó el respaldo (que internamente ejecuta el descifrado) sin ningún inconveniente. 

Continua en la página siguiente.

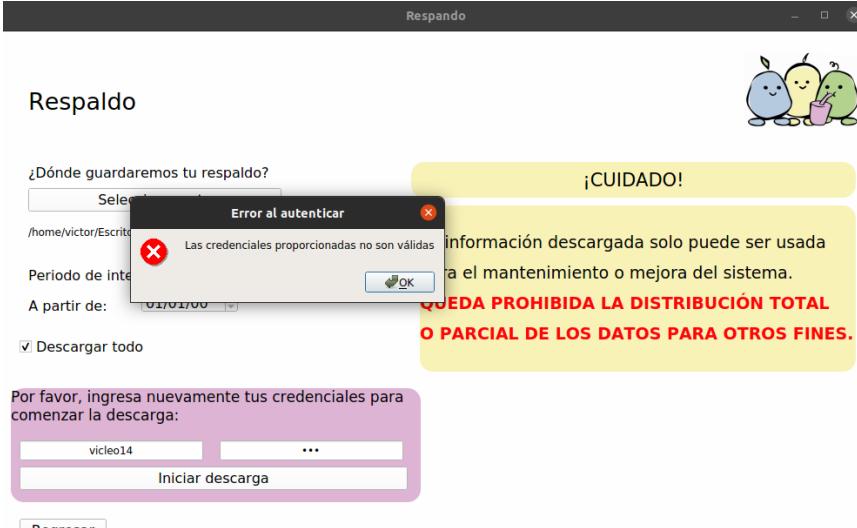
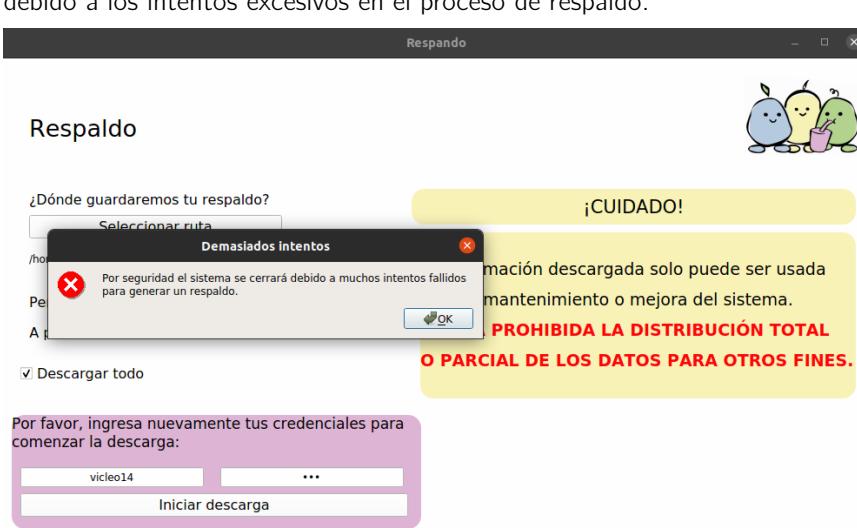
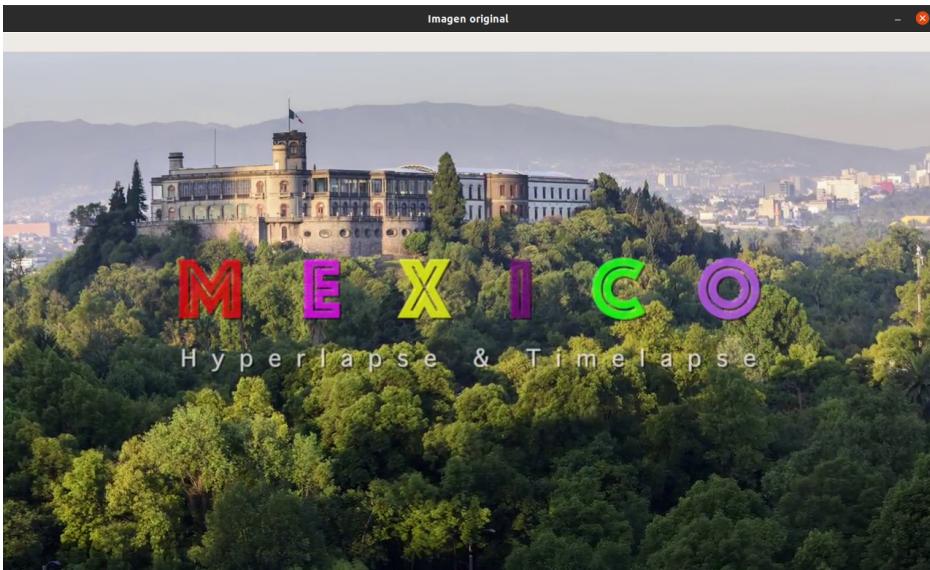
RF16 - Descifrado de rostros	
Resultados:	<ul style="list-style-type: none"> C2: En la imagen de abajo se puede observar que el sistema muestra un mensaje de error al usuario. 
	<ul style="list-style-type: none"> C3: En la imagen a continuación se muestra que el sistema muestra un mensaje debido a los intentos excesivos en el proceso de respaldo. 
Consideraciones especiales:	Sin consideraciones.

Tabla 6.15: Pruebas para corroborar RF16.

6.2.15. RF17 - Despliegue de publicidad estática

RF17 - Despliegue de publicidad estática	
Objetivo de la prueba:	Mostrar que cuando la SSD no detecta rostros, la publicidad mostrada en el visor es estática.
Prueba:	Alimentar al sistema con un video y observar el resultado en la salida del Visor.
Casos de prueba:	<ul style="list-style-type: none"> • C1: Se alimenta al sistema con un video donde no aparecen rostros.
Criterios de éxito:	<ul style="list-style-type: none"> • C1: El sistema debe proyectar en el Visor alguna de las imágenes almacenadas para publicidad estática.
Interfaz asociada:	No aplica.
Resultados:	<ul style="list-style-type: none"> • C1: En la primera imagen se observa la captura recibida desde el video. 

Continua en la página siguiente.

RF17 - Despliegue de publicidad estática

Resultados:	Mientras que en la segunda se puede observar la publicidad que se está mostrando a través del visor. Dado que no detectó rostros, usa publicidad estática previamente registrada.
Consideraciones especiales:	Sin consideraciones.

Tabla 6.16: Pruebas para corroborar RF17.

6.2.16. RF18 - Nueva publicidad estática

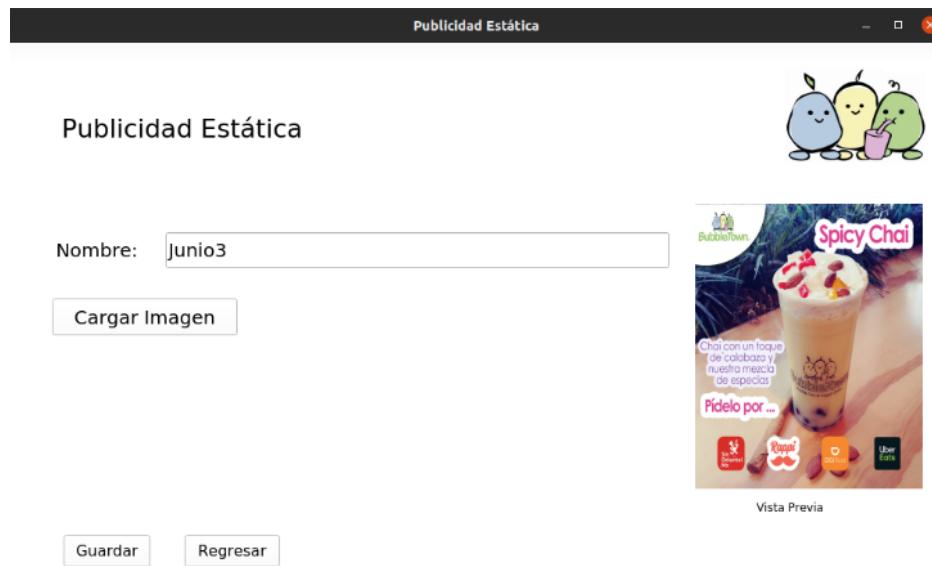
RF18 - Nueva publicidad estática					
Objetivo de la prueba:	Asegurar que el sistema es capaz de agregar nueva publicidad estática para mostrar en la pantalla.				
Prueba:	En el formulario correspondiente se agregarán los campos necesarios para agregar una nueva publicidad estática.				
Casos de prueba:	<ul style="list-style-type: none"> • C1: Se introduce el nombre referente a la publicidad y se carga una imagen. 				
Criterios de éxito:	<ul style="list-style-type: none"> • C1: La publicidad es agregada al sistema correctamente. 				
Interfaz asociada:	 IU3 Pantalla para gestión de publicidad y  IU4 Pantalla para agregar publicidad.				
Resultados:	<ul style="list-style-type: none"> • C1: En la primera imagen se puede observar como solo hay una imagen registrada. <div style="border: 1px solid #ccc; padding: 10px; width: fit-content; margin-left: auto; margin-right: auto;"> <p style="text-align: center;">Publicidad Estática</p> <div style="border: 1px solid #ccc; padding: 5px; margin-bottom: 10px;"> <table border="1" style="width: 100%; border-collapse: collapse;"> <thead> <tr> <th style="width: 10%;">Img</th> <th style="width: 90%;">Nombre descriptivo</th> </tr> </thead> <tbody> <tr> <td style="text-align: center;">  </td> <td style="text-align: center;">Junio2020</td> </tr> </tbody> </table> </div> <div style="display: flex; justify-content: space-between;"> <div style="flex: 1;">  </div> <div style="flex: 1;">  </div> </div> <div style="text-align: right; margin-top: 10px;"> Vista Previa </div> </div>	Img	Nombre descriptivo		Junio2020
Img	Nombre descriptivo				
	Junio2020				

Continua en la página siguiente.

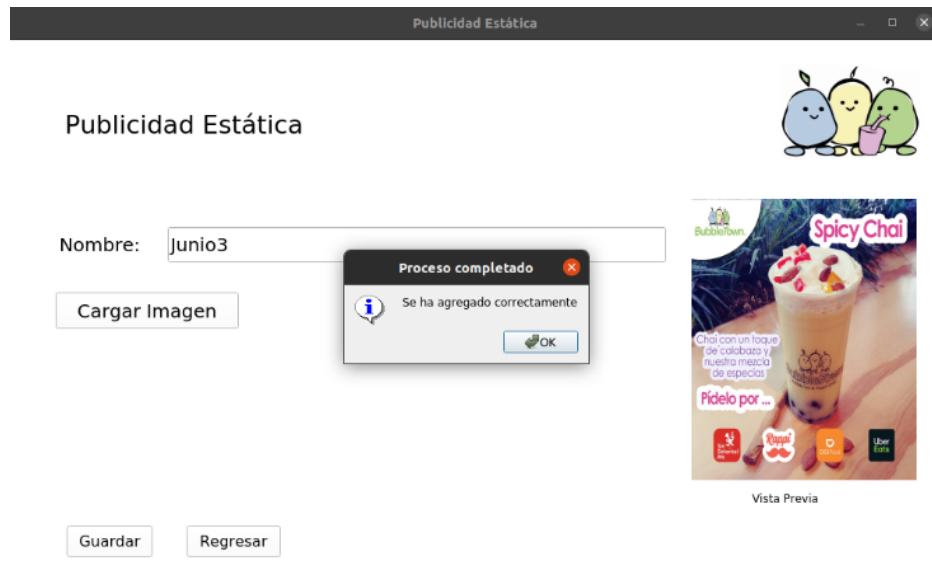
RF18 - Nueva publicidad estática

Resultados:

En la segunda imagen se observa el formulario donde se coloca el nombre de la publicidad junto con la imagen nueva cargada.



La tercera imagen muestra el mensaje generado cuando la nueva imagen de publicidad es agregada satisfactoriamente.



Continua en la página siguiente.

RF18 - Nueva publicidad estática

Resultados:

Por último, la cuarta imagen muestra en la lista la nueva publicidad.



Img	Nombre descriptivo
1	Junio2020
2	Junio3

Publicidad Estática

Vista Previa

Consideraciones especiales:

Sin consideraciones.

Tabla 6.17: Pruebas para corroborar RF18.

6.2.17. RF19 - Eliminación publicidad estática

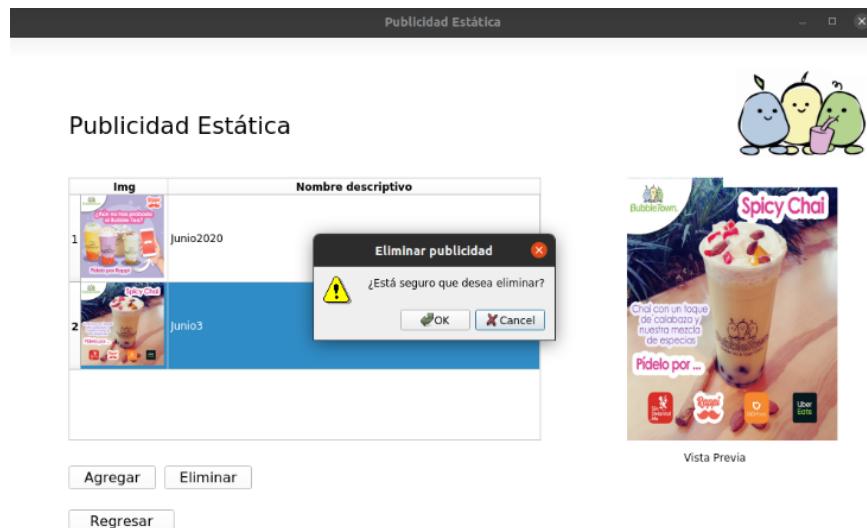
RF19 - Eliminación publicidad estática							
Objetivo de la prueba:	Asegurar que el sistema es capaz de eliminar una publicidad estática cuando ya no se quiera mostrar en la pantalla.						
Prueba:	Al cargar la interfaz de usuario con la lista de la publicidad registrada, se seleccionará una publicidad y se procederá a eliminarla.						
Casos de prueba:	<ul style="list-style-type: none"> • C1: Se selecciona y se elimina una publicidad. 						
Criterios de éxito:	<ul style="list-style-type: none"> • C1: La publicidad es eliminada correctamente, puesto que ya no aparece en la lista de publicidad registrada. 						
Interfaz asociada:	 IU3 Pantalla para gestión de publicidad y  IU5 Pantalla de confirmación para eliminar.						
Resultados:	<ul style="list-style-type: none"> • C1: En la primera imagen se puede observar como se encuentran dos imágenes registradas. <div style="border: 1px solid black; padding: 10px; width: fit-content; margin-left: auto; margin-right: auto;"> <p style="text-align: center;">Publicidad Estática</p> <table border="1" style="width: 100%; border-collapse: collapse; text-align: center;"> <thead> <tr> <th style="width: 5%;">Img</th> <th style="width: 45%;">Nombre descriptivo</th> </tr> </thead> <tbody> <tr> <td>1</td> <td>Junio2020</td> </tr> <tr> <td>2</td> <td>Junio3</td> </tr> </tbody> </table> <div style="display: flex; justify-content: space-around; margin-top: 10px;">   </div> <div style="text-align: center; margin-top: 10px;">  </div> <div style="text-align: right; margin-top: 20px;"> Vista Previa </div> </div> <div style="display: flex; justify-content: space-around; margin-top: 20px;"> <div style="text-align: center;">  </div> <div style="text-align: center;">  </div> </div>	Img	Nombre descriptivo	1	Junio2020	2	Junio3
Img	Nombre descriptivo						
1	Junio2020						
2	Junio3						

Continua en la página siguiente.

RF19 - Eliminación publicidad estática

Resultados:

En la segunda imagen se observa el mensaje de confirmación que muestra el sistema antes de proceder a borrar la publicidad.



La tercera imagen muestra en la lista solo una imagen para publicidad estática registrada, lo que indica que el proceso de borrado se llevó a cabo satisfactoriamente.



Consideraciones especiales:

Sin consideraciones.

Tabla 6.18: Pruebas para corroborar RF19.

6.3. Discusión del capítulo.

Durante la aplicación de las pruebas a los diferentes componentes de software, producto del presente trabajo, se observó que algunos casos las pruebas no fueron concluidas de forma satisfactoria bajo los criterios establecidos; sin embargo, al considerar la forma de funcionar del sistema completo puede que estos casos de prueba fallidos no sean críticos. Un ejemplo de ello es la prueba realizada para el requerimiento funcional [RF03 - Estimación de edad y género](#) que corresponde a la [Red neuronal convolucional para Edad y Género](#). En esta prueba una de las razones por las que se consideró como fallida fue debido a que la estimación de la edad (48 años) se alejó más de la tolerancia establecida de la edad real (34 años); no obstante, en términos prácticos la diferencia en los valores de las clases que se establecen en la Tabla 5.12 del diseño del [Recomendador](#) no varía significativamente.

También es importante mencionar que aunque se consideraron diversos escenarios a los que se enfrentaría el sistema al ser puesto en producción, no es posible considerar todos los casos a los que se enfrentará cuando sea colocado en la tienda de [BubbleTown®](#), ya que en ese caso el sistema no estará en condiciones controladas y posiblemente se presenten casos no considerados durante el desarrollo de este capítulo. Si bien uno de los objetivos era instalar el sistema en la tienda, debido al escenario en el que se desarrolló el proyecto (COVID-19) y las medidas sanitarias dispuestas por el gobierno de la CDMX, esto fue imposible de llevarse a cabo.

CAPÍTULO 7

Conclusiones generales

Generar publicidad dirigida, distinta y personalizada para recomendarle al cliente un producto de forma poco convencional no es tarea fácil. La explicación es sencilla: los gustos de los clientes pueden llegar a ser tan complejos que se convierten en preferencias únicas e irrepetibles; por eso, lograr generalizar todo el proceso dentro del sistema resulta altamente complejo en términos de desarrollo como efectividad.

La rapidez para este trabajo se convierte en una necesidad indispensable, incluso principal. La elección de la NVIDIA Jetson como unidad central se debe al alto procesamiento que se deberá realizar. Como se comenta en la [Especificación del hardware](#), este hardware especializado ha logrado altas velocidades en la [Detección de objetos](#) con la arquitectura [Single Shot Detection](#), sin olvidar que adicionalmente se deben ejecutar dos redes más y todo lo referente a la [Realidad Aumentada](#).

Según NVIDIA el poder computacional ofrecido por su minicomputadora Jetson Xavier ofrece altas expectativas; no obstante, tras el rendimiento presentado durante el proyecto da la impresión de no ser tan potente como NVIDIA presume. Cumple con las necesidades del proyecto, pero la velocidad que ofrece en la ejecución del mismo como todo un conjunto deja mucho que desear. Uno de estos efectos es el retraso que existe en el video captado y transmitido por la cámara, llegando a ser notable para los clientes, eliminando gradualmente ese "efecto espejo" que se busca, y que es mencionado en la sección de [Realidad Aumentada](#) como uno de los requisitos para un sistema que hace uso de esta tecnología.

No obstante, el dispositivo dedicado a inteligencia artificial de NVIDIA no es el culpable en su totalidad, debido a que el principal inconveniente es el lenguaje de programación seleccionado: Python, al ser uno de los lenguajes más lentos de todos a causa de que es interpretado en vez de compilado, con justa razón el rendimiento se verá comprometido. Sin embargo, gracias a la variedad de herramientas que tiene Python para *Deep Learning* permite desarrollar el proyecto sin comprometer la velocidad de desarrollo, además de que el tiempo de ejecución y procesamiento no es substancialmente alto. En algunos de estos es muy evidente el motivo de los retrasos ocasionados en el sistema, como lo son todos aquellos que involucran [Redes neuronales convolucionales](#) y [Realidad Aumentada](#) al generar una alta carga de procesamiento en la GPU y la CPU en la detección de parámetros y operaciones de imágenes. Otros en cambio no se habían llegado a tomar en cuenta que causaran este tipo de conflictos, como el de la [Captura de video a imagen](#) al tener un cableado físico y una red local, o la [Base de datos para el sistema](#), como consecuencia de la constante lectura y escritura de las llaves de encriptado.

Hablando de los módulos, queda claro que la materia prima del presente trabajo son las imágenes en donde aparecen los clientes, por eso el énfasis en poseer una cámara que logre un buen equilibrio costo-beneficio y protocolos de comunicación fáciles de manipular. Podría decirse que el preprocesamiento de las imágenes que serán ingresadas a los modelos de *deep learning* es uno de los principales aspectos que influirá para obtener buenos resultados, pues el tener un procesamiento adecuado en los datos permitirá al modelo obtener un resultado con una alta precisión usando menos recursos.

El reentrenamiento personalizado de la [Single Shot Detection](#) con ayuda de [Transferencia de aprendizaje](#) (para así poder llegar a buenos resultados sin necesidad de un entrenamiento exhaustivo) y sus resultados, así como las pruebas de ejecución de la [Red neuronal convolucional para Edad y Género](#) y la [Red neuronal convolucional para Big Five](#) confirman algo que se tiene bien claro en el mundo de la inteligencia artificial: ningún modelo tendrá una efectividad del 100 % en escenarios del mundo real. No hay modelo alguno que sea perfecto, por eso es que el campo del *deep learning* ha tenido a lo largo de la historia épocas de olvido y recientemente, gracias a computadoras más potentes, ha sido un tema de estudio de relevancia para el campo del cómputo, ya que justamente el obtener un modelo ideal es uno de los objetivos que estos campos buscan alcanzar.

Con todo esto, la detección de rostros tiene un desempeño aceptable para los fines del sistema, aunque podría ser mejor, pues hay que recordar que la casi nula existencia de conjuntos de datos disponibles para uso comercial o con licencias libres es la principal causa de no poder afinar o perfeccionar la [Single Shot Detection](#) para detectar rostros en condiciones muy difíciles, como lo son: personas con lentes (ya sean oscuros o transparentes), con sombreros, bufandas, con cabello recogido en el caso de las mujeres, con el rostro de perfil, y recientemente con cubre-bocas.

En el caso de la [Red neuronal convolucional para Edad y Género](#), para detectar la edad un regresor funcionó mejor para el propósito del sistema que un clasificador; como se aborda en el [Estado del Arte en la Revisión de trabajos de Edad y género](#), es más difícil predecir la edad que el género. La idea de primero construir una arquitectura para detectar la edad, y posteriormente usar transferencia de aprendizaje para construir otra más que detecte género, ofreció resultados con una precisión aceptable. Es importante aclarar que de entre los objetivos del presente trabajo, el proponer una arquitectura que detecte edad y género con precisiones superiores a las encontradas en la literatura a la fecha de la realización de este sistema, no es uno de ellos; no por eso se demeritó el trabajo y los resultados obtenidos con esta red, la única que fue hecha en su totalidad por el equipo (sin ocupar modelos de la literatura y aplicando reentrenamientos o transferencia de aprendizaje).

La librería TensorRT merece una mención especial: la arquitectura de la NVIDIA Jetson no soporta el uso de la librería TensorFlow, que es la más común para implementación de [Redes neuronales](#). La documentación oficial recomienda utilizar TensorRT en su lugar; sin embargo, surge un gran problema dado el poco soporte y compatibilidad entre ambas tecnologías, lo que obliga a manipular manualmente aquellos modelos cuando esto no debería hacerse (debido a que realizando alguna operación incorrecta o innecesaria, se puede alterar de forma crítica el funcionamiento o capacidades de aprendizaje reales de los modelos). No obstante, al final se logró ejecutar todas las [Redes neuronales](#) en esta biblioteca.

Hablando de la recomendación, se tiene la ventaja de la misma naturaleza y significado de la palabra: al final, simplemente es una sugerencia con base a ciertos parámetros identificados en una persona, y claramente quien tendrá la decisión de aceptarla o rechazarla será el mismo cliente. Más bien, la importancia reside en la "agresividad" en que ésta es sugerida, y al ser simplemente un gráfico que no compromete la decisión o intenciones del comprador, además de lo atractivo que puede resultar la [Realidad Aumentada](#) para un público no familiarizado con esta tecnología, se considera que es más probable que despierte interés en las bebidas de [BubbleTown®](#) en vez de algún rechazo o impacto negativo por violación de sus datos personales.

Como se describió en la sección [Recomendador](#) del [Marco teórico](#), los recomendadores requieren de perfiles de cada usuario para poder hacer una recomendación; sin embargo, el proyecto busca desarrollar un sistema que permita

a los usuarios usarlo sin complicaciones, por lo que auxiliándose de tecnologías como las redes neuronales, el sistema construye un perfil (en este caso la preferencia de sabores) con una sola imagen del rostro. Muy probablemente la aplicación de un cuestionario a cada cliente que desee comprar un producto permitiría obtener un perfil de preferencia más exacto pero afectaría considerablemente en la experiencia de usuario.

La presentación de la recomendación a un cliente resulta en una propuesta simple, pero atractiva y efectiva. Para la [Realidad Aumentada](#) BubbleTown no proporcionó un diseño específico oficial en la forma de presentar la publicidad dirigida; en su lugar solo estableció de qué forma este diseño debe ser presentado; el módulo implementado cumple en su mayoría con los cuatro requisitos que un sistema de realidad aumentada debe cumplir (ver sección [Revisión de trabajos de Realidad Aumentada](#)).

Como alternativa, el mismo sistema ofrece la ventaja de no generar publicidad dirigida si un cliente no se detiene enfrente del mismo a visualizar la publicidad, optando por alejarse; así el sistema continua funcionando sin comprometer la confidencialidad del cliente, convirtiéndose en un medio no invasivo.

Una parte complementaria a tener en cuenta de este sistema es la necesidad de salvaguardar los datos sensibles (rostros) de las personas, almacenados en la base de datos del mismo, para cumplir con las [Reglas del negocio](#) y no tener problemas legales. Para lograr esto, basta con proporcionar un aviso de privacidad junto con el sistema y manejar cifrado en los datos a almacenar, para evitar un mal uso de ellos.

Es muy relevante conocer la condición actual de la empresa, pues como se mencionó en los [Gestión de riesgos](#), hay una probabilidad alta de que [BubbleTown®](#) desaparezca, debido a la situación económica actual que vive el país. En dado caso que [BubbleTown®](#) cesara actividades, el sistema será instalado en el Centro de Investigación en Computación en cuanto las condiciones sanitarias y las medidas del gobierno de la CDMX así lo permitan.

Por último, a continuación se presenta el estado de los [Objetivos](#) propuestos al inicio del trabajo, en donde se discute de qué forma se logró o, en su caso, las razones por las que no se cumple con cada uno de ellos:

- Se logró generar publicidad dirigida a clientes utilizando como producto promocional bebidas de la empresa [BubbleTown®](#), haciendo uso de aprendizaje profundo y realidad aumentada, todo esto implementado en un prototipo de sistema compuesto por distintos componentes de hardware (ver sección [Especificación del hardware](#)), y con ayuda de diversos módulos y librerías en Python con funciones muy puntuales (ver [Diagrama general del sistema](#)).
- Por causas de fuerza mayor y totalmente ajenas a los integrantes y directores de este trabajo terminal, no se logró instalar los dispositivos del sistema en el establecimiento de [BubbleTown®](#).
- Se logró obtener no solo uno, sino cinco conjuntos de datos constituidos por imágenes del rostro de personas, de los cuales dos fueron empleados para el reentrenamiento y pruebas de la red [Single Shot Detection](#), es decir la [Detección de rostros en la imagen](#), y tres fueron empleados para el entrenamiento y pruebas de la [Red neuronal convolucional para Edad y Género](#) en TensorFlow. Estos datasets son de licencia libre y permiten el uso comercial (revisar las secciones indicadas en este punto para conocer más a detalle sobre estos).
- Se logró diseñar y entrenar una arquitectura convolucional multitarea, que además hizo uso de transferencia de aprendizaje durante su entrenamiento, para construir la [Red neuronal convolucional para Edad y Género](#) en TensorFlow (de autoría propia), que obtiene estos atributos a partir del rostro de una persona. La personalidad es otro de estos atributos obtenidos a partir del rostro, la cual se apoya del modelo [Personalidad: Big Five](#), pero a diferencia de la edad y género, la [Red neuronal convolucional para Big Five](#) corresponde al [Modelo de clasificación: CNN-4](#) del trabajo de Miguelangel Sotelo en [51]; no se realizó ninguna modificación en esta red, se implementó de forma directa e integró a nuestro sistema (no es de nuestra autoría). La implementación de estas redes se hicieron en TensorRT.

- Se logró generar recomendaciones de bebidas de [BubbleTown®](#) con base a los parámetros de edad, género, personalidad y temperatura ambiental (obtenida con ayuda de la API OWM, ver sección [Herramientas de software a utilizar](#)). En la sección de [Recomendador del sistema](#) se aborda esto con mejor detalle.
- Se logró implementar tecnologías y métodos de realidad aumentada para proyectar las recomendaciones obtenidas en una pantalla publicitaria, utilizando funcionalidades de la librería OpenCV como se aborda en el diseño de la [Generación de realidad aumentada](#). Los diseños de la [Realidad Aumentada](#) que poseen las recomendaciones de bebidas fueron incrustados en los frames originales captados por la cámara de red tras haber pasado por la extracción y detección de características.
- Se logró parcialmente hacer pruebas del sistema con imágenes de clientes potenciales de [BubbleTown®](#) para evaluar el funcionamiento del prototipo de sistema por medio de la satisfacción de los mismos con su recomendación obtenida, ya que fueron realizadas en ambientes controlados y en otros similares que emulan el transitar por fuera de un establecimiento; junto a esto, únicamente se utilizaron imágenes de los integrantes de este trabajo terminal.

7.1. Trabajo a futuro

La principal mejora que se propone es migrar del lenguaje Python a un lenguaje compilado que sea compatible con las tecnologías que se usan en este proyecto (por ejemplo, C++ aprovechando que ambos soportan el paradigma orientado a objetos), de preferencia el sistema entero o por lo menos aquellos módulos cuya velocidad de ejecución y respuesta debe ser muy alta, para así lograr una mayor y mejor interacción con el cliente, e incluso se podría llegar a lograr una ambiciosa respuesta en tiempo real; estos módulos son todos aquellos en donde hay [Redes neuronales convolucionales](#) involucradas; todos los que realizan procesamiento, captura, extracción, y demás operaciones en imágenes; la [Realidad Aumentada](#) y la lectura/escritura de archivos. Las librerías utilizadas en dichos módulos están disponibles en C/C++, por lo que no debería de haber problemas de compatibilidad ni de migración.

Otro aspecto que podría mejorar significativamente las recomendaciones de bebidas del sistema es la afinación de la precisión de los modelos de la [Single Shot Detection](#) y la [Red neuronal convolucional para Edad y Género](#) por medio de nuevos reentrenamientos, ya sea utilizando conjuntos de datos generados de forma manual, o bien utilizando algunos que no permitan su uso para fines comerciales pero que abundan en gran cantidad en Internet, aunque esto último impediría utilizar el sistema con alguna empresa real de cualquier giro. Un buen agregado a estos conjuntos de datos serían imágenes de personas con lentes (tanto oscuros como transparentes), bufandas, sombreros, cubre-bocas, y demás accesorios en la cabeza y cara comunes.

Para un nuevo reentrenamiento en la [Single Shot Detection](#) se recomienda utilizar en su totalidad las 77,222 imágenes en train del dataset *Google Facial Expression Comparison Dataset* [129], dado que en el entrenamiento realizado en este trabajo únicamente se utilizaron 8,333 imágenes del mismo.

También se plantea, si es posible, mejorar los resultados obtenidos en la [Red neuronal convolucional para Edad y Género](#) utilizando conjuntos de datos mejor balanceados entre clases y con mayor número de muestras.

Algo que se nota al momento de intercalar entre publicidad estática y dirigida es el cambio algo brusco que resulta cuando la [Single Shot Detection](#) está tratando de detectar un rostro en una posición complicada, por lo que se sugiere aplicar una transición de suavizado para que este intercambio sea más agradable a la vista.

Respecto a las recomendaciones de bebidas, es posible ampliar el catálogo, aunque implicaría mapear desde cero estos nuevos sabores para que el sistema logre relacionarlos con los parámetros obtenidos de los clientes. Con respecto a la vectorización de las bebidas actuales, el proceso fue realizado considerando los ingredientes; sin embargo, podría

reflejarse una mejora si el vector de cada bebida se realizara con ayuda de clientes, quienes podrían indicar qué sabores detecta en cada una de las bebidas. Igualmente, si se tiene la ayuda de los clientes de BubbleTown® en el mismo cuestionario se le podría pedir que califique el producto y junto con la recolección de su edad, género y personalidad, se darían las condiciones para generar un dataset que permita entrenar algún modelo de Machine Learning para obtener recomendaciones más adecuadas. Es importante notar que este último caso requiere de una cantidad de tiempo y recursos económicos elevados, así como de personas que fingirán como catadores para evaluar múltiples bebidas.

Los datos de edad, género y personalidad de una persona, junto a la temperatura ambiental y la bebida recomendada pueden ser utilizados para futuras recomendaciones al estar almacenados en la base de datos. Se pueden aplicar métodos de *data mining* para sugerir recomendaciones más precisas con base a información histórica de las preferencias en diversas personas. Adicionalmente, se puede integrar la información de las transacciones de bebidas vendidas a clientes.

En la **Realidad Aumentada** se podrían realizar diseños más atractivos con ayuda de un diseñador gráfico o algún especialista en publicidad y marketing. Incluso se puede pasar de una publicidad en 2D a 3D, aunque probablemente el procesamiento de la NVIDIA Jetson no sería suficiente.

Por último, se plantea el uso de los resultados obtenidos en este proyecto para otro tipo de rubros ajenos a los sistemas computacionales y la inteligencia artificial, como pueden ser la medicina, el marketing, la psicología, la seguridad pública, la gastronomía, la biología, etc., y de referencia para uso académico o científico.

APÉNDICE A

Manual de usuario

Hoja de Control

- **Nombre del Sistema:** Prototipo de sistema de publicidad dirigida a clientes usando visión por computadora y aprendizaje profundo
- **Autores:** Jaime Arturo Lara Cázares, Víctor Leonel Morales Flores, Enrique Ramos Diaz
- **Versión:** 1.0

Queda prohibido cualquier tipo de explotación y, en particular, la reproducción, distribución, comunicación pública y/o transformación, total o parcial, por cualquier medio, de este sistema sin el previo consentimiento expreso y por escrito.

Descripción del sistema

Objetivo

Generar publicidad dirigida y personalizada a clientes en el establecimiento de bebidas a base de té BubbleTown®.

Alcance

La empresa de bebidas BubbleTown® junto a sus clientes potenciales.

Funcionalidad

El sistema detecta rostros de personas con una cámara y a partir de este predice características como: edad, género y personalidad. Con ayuda de estas y la temperatu-

ra ambiental, ofrece una recomendación de alguna bebida de BubbleTown® desplegándola en un diseño interactivo y personalizado para cada cliente. Utiliza dispositivos de NVIDIA dedicados y especializados para este tipo de tareas de inteligencia artificial.



Requisitos de hardware

Mínimos

- NVIDIA Jetson Nano Developer Kit
- Sistema Operativo: NVIDIA Jetpack 4.3
- Espacio en disco: 4 GB
- Cámara compatible con el protocolo RTSP
- Switch PoE
- Pantalla LED 720p a 30fps

Recomendados

- NVIDIA Jetson AGX Xavier Developer Kit 32 GB
- Sistema Operativo: NVIDIA Jetpack 4.3
- Espacio en disco: 5 GB
- Cámara HIKVISION modelo DS-2CD2143G0-IS
- Switch PoE HIKVISION modelo DS-3E0105P-E
- Pantalla LED 50" 4K a 60fps

Instalación del hardware

El diagrama de la Figura A.1 muestra la conexión de todos los componentes de hardware. Adicional a los requisitos de hardware descritos en la sección anterior, se necesita lo siguiente:

- 3 Cables Ethernet RJ45
- 1 Cable HDMI
- Teclado y Mouse USB
- Adaptador de USB-Type A a USB-Type C (incluido con la NVIDIA Jetson AGX Xavier)

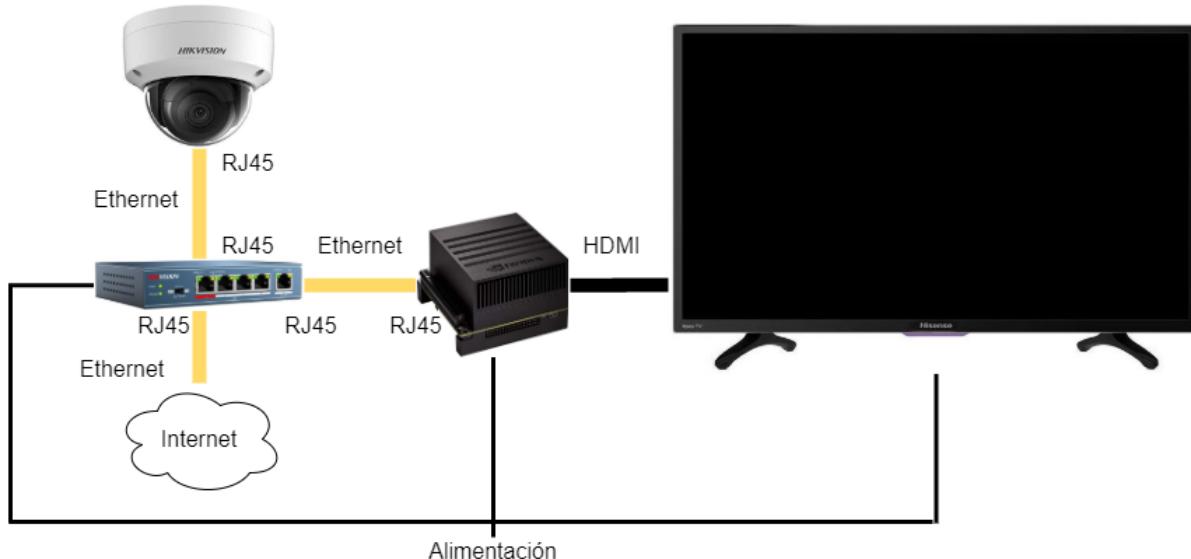


Figura A.1: Diagrama de conexión del hardware del sistema.

Requisitos de software

Las librerías y paquetes necesarios para la ejecución del sistema son los siguientes:

- **Python 3**
- **CUDA**
- **cuDNN**
- **OpenCV**
- **TensorRT**
- Pip 3
- NumPy
- MongoDB
- PyCryptodome
- PyOWM
- PyMongo
- PyCUDA
- PyQt5
- TensorFlow 1
- GraphSurgeon
- UFFConverter
- **GStreamer**

IMPORTANTE: Los paquetes en negritas vienen instalados de forma automática en el sistema operativo NVIDIA Jetpack 4.3, por lo que no se entra en detalles sobre su instalación.

Instalación de paquetes adicionales

Para aquellos módulos adicionales de Python que no vienen instalados de forma automática en Jetpack 4.3, la instalación para cada uno de estos se lista a continuación. Es recomendable instalar todos estos paquetes con permisos de superusuario:

```
#Actualizacion paquetes
apt-get update

#Paquetes de BD
apt install mongodb-server
apt install mongodb-clients

#Pip para instalar paquetes de Python
↳ necesarios
apt-get install python3-pip

#Paquetes de encriptado y desencriptado
pip3 install pycryptodome==3.9.7

#API Temperatura ambiental
pip3 install pyowm

#Paquetes necesarios para MongoDB
pip3 install pymongo==3.10.1

#Paquetes necesarios para las redes neuronales
apt-get install uff-converter-tf
pip3 install pillow
pip3 install numpy
apt-get install python3-libnvinfer-dev
pip3 install pycuda --user

apt-get install uff-converter-tf
pip3 install tensorflow-object-detection-api

#Paquetes por si es necesario generar
↳ nuevamente los modelos de TensorRT
apt-get install libhdf5-serial-dev hdf5-tools
↳ libhdf5-dev zlib1g-dev zip libjpeg8-dev
↳ liblapack-dev libblas-dev gfortran
pip3 install -U pip testresources
↳ setuptools==49.6.0
pip3 install -U numpy==1.16.1 future==0.18.2
↳ mock==3.0.5 h5py==2.10.0
↳ keras_preprocessing==1.1.1
↳ keras_applications==1.0.8 gast==0.2.2
↳ futures protobuf pybind11
pip3 install --pre --extra-index-url
↳ https://developer.download.nvidia.com/compute/
redist/jp/v43 'tensorflow<2'
pip3 install graphsurgeon

#Paquetes necesarios para la interfaz grafica
pip3 install --upgrade keyrings.alt
apt-get install qtcreator pyqt5-dev-tools
apt-get install python3-pyqt5
apt-get install qttools5-dev-tools
```

Descarga del repositorio

Dentro de una carpeta abra una terminal y descargue el repositorio con el siguiente comando:

```
git clone https://github.com/vicleo14/PublicidadBT.git
```

IMPORTANTE: Debe iniciar sesión con una cuenta de GitHub que tenga permisos para acceder al repositorio privado.

Instalación del sistema

1. Dentro de la carpeta creada por el repositorio, diríjase a la carpeta `/PublicidadBubbleTown` y compruebe que se encuentre el archivo llamado **instalacion.sh**.
2. Abra una terminal en la misma ruta donde se encuentran este archivo y ejecute los siguientes comandos (debe tener permisos de superusuario).

```
sudo chmod +x instalacion.sh
sudo sh instalacion.sh
```

3. Espere a que se terminen de realizar las operaciones necesarias.
4. Para comprobar que las instalaciones de MongoDB y de la base de datos se realizaron correctamente, ingrese en la terminal el comando:

```
mongo
```

5. Dentro del cliente de MongoDB ejecute el siguiente comando:

```
show databases
```

6. Le deberá mostrar una base de datos nombrada **publicidadBT**. Ya puede cerrar la terminal.
7. Ahora diríjase al directorio `/home/PublicidadBubbleTown/images` y asegúrese que se encuentren las siguientes 3 carpetas: `/bebidas`, `/publicidad` y `/rostros`.
8. Se deben asignar permisos a todos los usuarios para que puedan escribir en ellas.
9. Nuevamente abra una terminal en cualquier ruta y ejecute los siguientes comandos para asignar las direcciones IP correctas (debe tener permisos de superusuario).

```
sudo ifconfig eth0 169.254.47.88
sudo ifconfig dhclient eth0
```

10. La instalación se ha completado satisfactoriamente.

IMPORTANTE: Esta instalación está pensada únicamente para ejecutar el sistema en un dispositivo NVIDIA Jetson.

Ejecución del sistema

1. Asegúrese que la cámara de red y el switch PoE estén conectados y encendidos correctamente.
2. Dentro de la carpeta creada por el repositorio, diríjase a la carpeta `/PublicidadBubbleTown`, luego a la carpeta `/src` y compruebe que dentro de esta se encuentre el archivo llamado **main.py**.
3. Abra una terminal en esta ruta y ejecute el siguiente comando:

```
sudo python3 main.py
```

4. Debe de aparecer la ventana de [Login de acceso](#), que se muestra a continuación:



Figura A.2: Login de acceso

Navegación y uso del sistema

Generar y transmitir publicidad

1. En la ventana de [Login de acceso](#), de clic en el botón con el ícono verde de **Reproducir** del centro.
2. El sistema abrirá la [Ventana con el visor de publicidad](#) en donde se desplegarán ambos tipos de publicidad: estática y dirigida.
3. Para cambiar entre modo en ventana y modo en pantalla completa, oprima la tecla **F**.
4. Para cerrar esta ventana oprima la tecla **Q**. Regresará a la ventana de [Login de acceso](#).



Figura A.3: Ventana con el visor de publicidad

Autenticar administrador

1. En la ventana de [Login de acceso](#), ingrese su usuario y contraseña en los campos correspondientes.
2. De clic en el botón **Acceder**.
3. Si los datos de acceso son correctos, el sistema mostrará la [Ventana principal](#).
4. Si desea salir de esta ventana, de clic en el botón **Cerrar sesión**. El sistema volverá a la ventana de [Login de acceso](#).



Figura A.4: Ventana principal

Agregar publicidad estática

1. En la [Ventana principal](#), de clic en el botón **Publicidad estática**. El sistema abrirá la [Ventana para gestión de publicidad](#).
2. En la [Ventana para gestión de publicidad](#) de clic en el botón **Agregar**. El sistema mostrará la [Ventana para agregar publicidad](#).
3. Agregue un nombre de referencia en el campo correspondiente para ubicar la publicidad posteriormente.
4. De clic sobre el botón **Cargar imagen** y seleccione la ruta de la ubicación de la imagen que se desee agregar.
5. El sistema mostrará en la parte derecha de esta ventana una vista previa de la imagen seleccionada.
6. De clic en el botón **Guardar**. El sistema indicará que la nueva imagen fue cargada exitosamente.
7. De clic en el botón **Regresar** para volver a la [Ventana principal](#).



Figura A.5: Ventana para gestión de publicidad



Figura A.6: Ventana para agregar publicidad

Generar respaldo de la base de datos

1. En la [Ventana principal](#), de clic en el botón **Respaldo**. El sistema abrirá la [Ventana para hacer respaldo](#).
2. En la [Ventana para hacer respaldo](#) indique la ruta de la carpeta donde se almacenará el respaldo de los datos a través del botón **Seleccionar ruta**.
3. Indique la fecha a partir de la que quiere generar el respaldo por medio del selector **A partir de**.
4. Si desea hacer un respaldo de toda la información de la base de datos, marque la casilla **Descargar todo**.
5. Por motivos de seguridad, indique nuevamente las credenciales de un administrador. En caso de que intente generar un respaldo con credenciales erróneas en tres ocasiones, el sistema se cerrará de forma automática.
6. De clic en el botón **Iniciar descarga**. El sistema almacenará una copia de seguridad en la ruta indicada y volverá a la [Ventana principal](#).

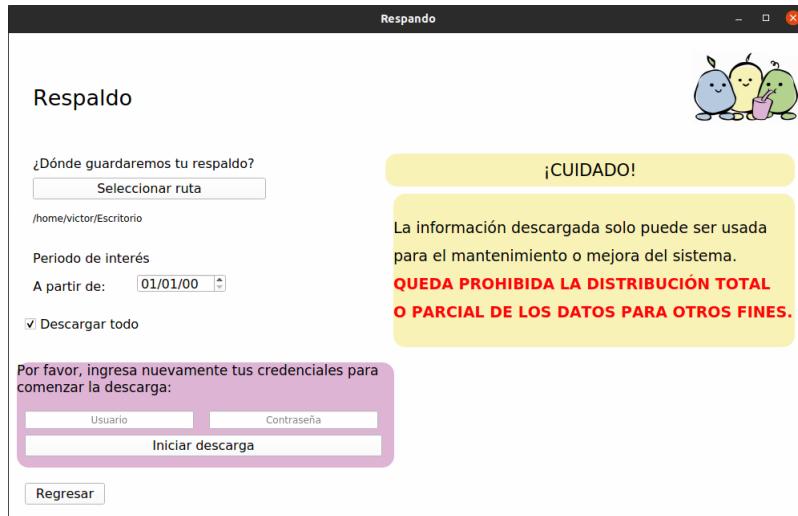


Figura A.7: Ventana para hacer respaldo

Agregar nueva bebida

1. En la [Ventana principal](#), de clic en el botón **Bebidas**. El sistema abrirá la [Ventana de gestión de bebidas](#).
2. La [Ventana de gestión de bebidas](#) desplegará en su parte izquierda una lista de las bebidas registradas en la base de datos con una pequeña imagen.
3. De clic en el botón **Agregar**. El sistema mostrará la [Ventana para agregar o modificar bebida](#).
4. Ingrese el nombre de la bebida en su campo correspondiente.
5. Ingrese una breve descripción de la bebida en su campo correspondiente.
6. Elija de la lista desplegable la categoría a la que pertenece la bebida.
7. Seleccione las casillas correspondientes de la o las modalidades en las que se encuentra disponible la bebida.
8. De clic en el botón **Cargar Imagen** y seleccione la ruta de la imagen de la bebida con las características establecidas.
9. El sistema mostrará en la parte derecha de esta ventana una vista previa de la imagen seleccionada.
10. De clic en el botón **Guardar**. El sistema indicará que la nueva bebida fue agregada exitosamente.
11. De clic en el botón **Regresar** para volver a la [Ventana principal](#).

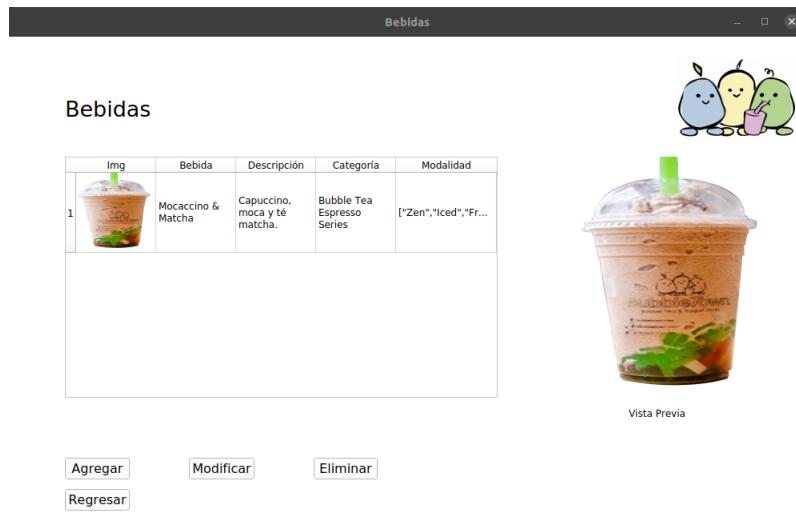


Figura A.8: Ventana de gestión de bebidas

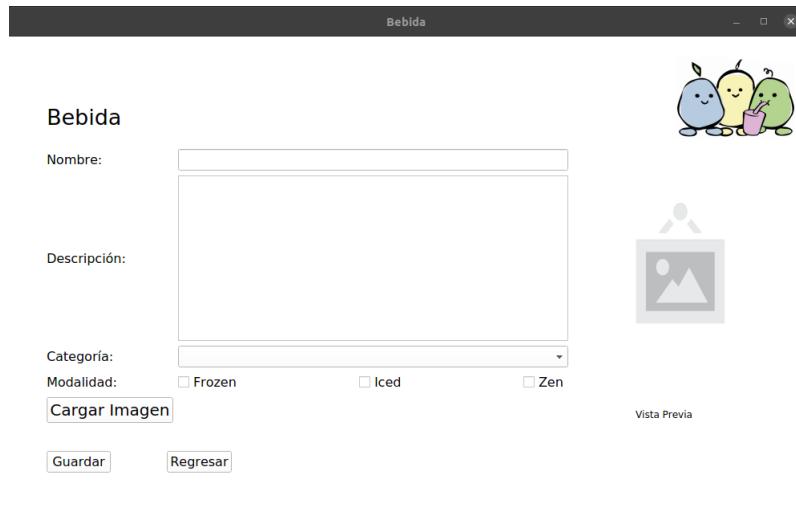


Figura A.9: Ventana para agregar o modificar bebida

Modificar bebida

1. En la [Ventana principal](#), de clic en el botón **Bebidas**. El sistema abrirá la [Ventana de gestión de bebidas](#).
2. La [Ventana de gestión de bebidas](#) desplegará en su parte izquierda una lista de las bebidas registradas con una pequeña imagen.
3. De clic en la bebida, de la lista de bebidas, cuya información deseé modificar.
4. El sistema habilitará los botones **Modificar** y **Eliminar** en la [Ventana de gestión de bebidas](#) y mostrará en la parte derecha de esta ventana una imagen ampliada de la bebida que se seleccionó.

5. De clic en el botón **Modificar**. El sistema mostrará la [Ventana para agregar o modificar bebida](#) con el campo del nombre de la bebida bloqueado para edición, y el resto de campos correspondientes con la información de la bebida seleccionada.
6. Modifique la información que desee, exceptuando el nombre de la bebida.
7. De clic en el botón **Guardar**. El sistema indicará que la nueva bebida fue actualizada exitosamente.
8. De clic en el botón **Regresar** para volver a la [Ventana principal](#).

Eliminar bebida

1. En la [Ventana principal](#), de clic en el botón **Bebidas**. El sistema abrirá la [Ventana de gestión de bebidas](#).
2. La [Ventana de gestión de bebidas](#) desplegará en su parte izquierda una lista de las bebidas registradas con una pequeña imagen.
3. De clic en la bebida, de la lista de bebidas, que deseé eliminar.
4. El sistema habilitará los botones **Modificar** y **Eliminar** en la [Ventana de gestión de bebidas](#) y mostrará en la parte derecha de esta ventana una imagen ampliada de la bebida que se seleccionó.
5. De clic en el botón **Eliminar**. El sistema mostrará la [Ventana de confirmación para eliminar](#).
6. De clic en el botón **Aceptar** para confirmar. El sistema indicará que la bebida fue eliminada exitosamente y volverá a la [Ventana principal](#).
7. Si oprime el botón **Cancelar** en su lugar, el sistema volverá a la [Ventana de gestión de bebidas](#).

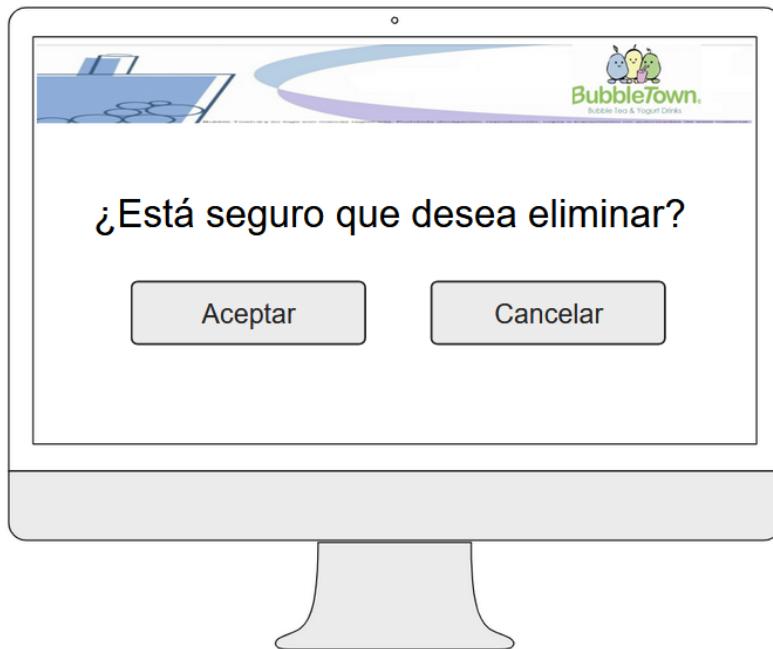


Figura A.10: Ventana de confirmación para eliminar

Eliminar publicidad estática

1. En la [Ventana principal](#), de clic en el botón **Publicidad estática**. El sistema abrirá la [Ventana para gestión de publicidad](#).
2. La [Ventana para gestión de publicidad](#) desplegará en su parte izquierda una lista de la publicidad registrada con una pequeña imagen.
3. De clic en la publicidad que deseé eliminar.
4. El sistema habilitará el botón **Eliminar** en la [Ventana para gestión de publicidad](#) y mostrará en la parte derecha de esta ventana una imagen ampliada de la publicidad que se seleccionó.
5. De clic en el botón **Eliminar**. El sistema mostrará la [Ventana de confirmación para eliminar](#).
6. De clic en el botón **Aceptar** para confirmar. El sistema indicará que la publicidad fue eliminada exitosamente y volverá a la [Ventana principal](#).
7. Si oprime el botón **Cancelar** en su lugar, el sistema volverá a la [Ventana para gestión de publicidad](#).

Errores y mensajes comunes

Mensajes en la terminal al iniciar el sistema

Descripción: Cuando se ejecuta el sistema, la terminal puede mostrar lo siguiente:

```
RTNETLINK answers: File exists
QStandardPaths: XDG_RUNTIME_DIR not set, defaulting to '/tmp/runtime-root'
```

A su vez, cuando el sistema abre la ventana en donde se despliega la publicidad estática y dirigida, también pueden aparecer una serie de mensajes como los siguientes:

```
(python3:13917): GStreamer-CRITICAL **: 18:14:25.186: gst_caps_is_empty: assertion 'GST_IS_CAPS
↳ (caps)' failed
(python3:13917): GStreamer-CRITICAL **: 18:14:25.186: gst_caps_truncate: assertion 'GST_IS_CAPS
↳ (caps)' failed
(python3:13917): GStreamer-CRITICAL **: 18:14:25.186: gst_caps_fixate: assertion 'GST_IS_CAPS (caps)'
↳ failed
(python3:13917): GStreamer-CRITICAL **: 18:14:25.186: gst_caps_get_structure: assertion 'GST_IS_CAPS
↳ (caps)' failed
(python3:13917): GStreamer-CRITICAL **: 18:14:25.186: gst_structure_get_string: assertion 'structure
↳ != NULL' failed
(python3:13917): GStreamer-CRITICAL **: 18:14:25.186: gst_mini_object_unref: assertion 'mini_object
↳ != NULL' failed
NvMMLiteOpen : Block : BlockType = 261
NVMEDIA: Reading vendor.tegra.display-size : status: 6
NvMMLiteBlockCreate : Block : BlockType = 261
Allocating new output: 1920x1088 (x 11), ThumbnailMode = 0
OPENMAX: HandleNewStreamFormat: 3605: Send OMX_EventPortSettingsChanged: nFrameWidth = 1920,
↳ nFrameHeight = 1080
[ WARN:0] global /home/nvidia/host/build_opencv/nv_opencv/modules/videoio/src/cap_gstreamer.cpp (896)
↳ open OpenCV | GStreamer warning: unable to query duration of stream
```

```
[ WARN:0] global /home/nvidia/host/build_opencv/nv_opencv/modules/videoio/src/cap_gstreamer.cpp (933)
→ open OpenCV | GStreamer warning: Cannot query video position: status=1, value=0, duration=-1
Gtk-Message: 18:14:25.337: Failed to load module "canberra-gtk-module"
```

Motivo: La aparición de estos mensajes es completamente normal, no reflejan errores y no afectan al funcionamiento del sistema.

Errores de TensorRT

Error: TensorRT no puede cargar los modelos de aprendizaje profundo.

Motivo: Tenga en cuenta que los archivos de los modelos que el sistema utiliza para detectar rostros, así como las predicciones de edad, género y personalidad fueron construidos para ser ejecutados exclusivamente en la NVIDIA Jetson AGX Xavier Developer Kit.

Solución: Si por alguna razón el sistema no logra cargar ni ejecutar estos modelos en este dispositivo o si se trata de otro modelo de la familia Jetson, es necesario volver a generarlos. Es importante leer los tutoriales en [126] y [130] antes de ejecutar esta solución (principalmente para hacer el ajuste en **graphsurgeon** que indica [126]).

1. Diríjase a la siguiente ruta `/PublicidadBubbleTown/src/rna/` y localice los siguientes 3 archivos: **DetectorRostros.py**, **CnnEG.py.py** y **CnnBig5.py**.
2. Dentro de cada uno de ellos hay un bloque de código comentado, delimitado por tres comillas ("") al inicio y al final, similar al siguiente:

```
...
model_path = os.path.abspath(os.path.dirname(os.path.realpath(__file__))) +
← '/../../bin/pretrained_model/frozen_inference_graph.pb'
model_output_name = ['NMS']
__PATH_UFF_SAVED_MODEL = os.path.abspath(os.path.dirname(os.path.realpath(__file__))) +
← '/../../bin/pretrained_model/frozen_graph_inference.uff'

# Compile model into TensorRT
if not os.path.isfile(model_TRTbin):
    dynamic_graph = model.add_plugin_to_SSD(gs.DynamicGraph(model_path))
    uff_model = uff.from_tensorflow(dynamic_graph.as_graph_def(), model_output_name,
← output_filename=__PATH_UFF_SAVED_MODEL)
    with trt.Builder(TRT_LOGGER) as builder, builder.create_network() as network,
← trt.UffParser() as parser:
        builder.max_workspace_size = 1 << 34
        builder.max_batch_size = 1
        builder.fp16_mode = True
        parser.register_input('Input', self.__model_dims)
        parser.register_output('MarkOutput_0')
        parser.parse(__PATH_UFF_SAVED_MODEL, network)
        engine = builder.build_cuda_engine(network)
        buf = engine.serialize()
        with open(model_TRTbin, 'wb') as f:
            f.write(buf)
...

```

Descomente este bloque de código en los 3 archivos.

3. También descomente las siguientes líneas que inician con un # al inicio en cada uno de estos:

```
#import uff
#import graphsurgeon as gs
#import util.ConfigEG as model
#import util.ConfigSSD as model
```

4. Ejecute nuevamente el sistema. La generación de los archivos que TensorRT necesita tardará aproximadamente de 10 a 20 minutos (este tiempo solo será por esta ocasión).
5. Una vez terminado, el sistema debería ejecutar las redes neuronales sin problemas. **Es importante que vuelva a comentar el bloque de código y las líneas que se indican en estos 3 archivos para reducir el tiempo de ejecución.**

El visor de publicidad permanece con una pantalla en negro

Error: Aparece el siguiente mensaje de error en la terminal:

ERROR GENERADO: 'NoneType' object has no attribute 'shape'

Motivo: El sistema no logra obtener las imágenes del video que la cámara esta capturando, debido a una mala configuración o conexión entre esta, el switch y la NVIDIA Jetson.

Solución:

- Asegúrese de tener estos tres dispositivos conectados de forma adecuada, además de revisar el estado de los cables y sus entradas.
- En un navegador web, intente acceder al cliente de la cámara de red ingresando la siguiente dirección en la barra de búsqueda: **169.254.47.87**. Debe de aparecer una interfaz como la siguiente:



- Si al tratar de acceder a la dirección anterior el navegador no carga esta interfaz y en su lugar el siguiente mensaje es mostrado:

```
No se puede acceder a http://169.254.47.87/.  

ERR_ADDRESS_UNREACHABLE
```

Pruebe a asignar nuevamente la dirección IP de la NVIDIA Jetson ejecutando los siguientes comandos en una terminal (debe tener permisos de superusuario):

```
sudo ifconfig eth0 169.254.47.88  

sudo ifconfig dhclient eth0
```

Error con la temperatura ambiental

Error: Aparece el siguiente mensaje de error en la terminal:

```
pyowm.exceptions.api_call_error.APIInvalidSSLCertificateError: Exception in calling OWM Weather API.  
Reason: HTTPConnectionPool(host='api.openweathermap.org', port=80): Max retries exceeded with url:  
  → /data/2.5/weather?APPID=eef109fddead9d738dd04b631ffa2c43&lon=-99.156818&lat=19.426633&lang=en  
  → (Caused by ConnectTimeoutError(<urllib3.connection.HTTPConnection object at 0x7f700995f8>,  
  → 'Connection to api.openweathermap.org timed out. (connect timeout=2)'))  
Caused by: None
```

Motivo: El sistema no tiene conexión a Internet.

Solución:

- Asegúrese de tener estos tres dispositivos conectados de forma adecuada, además de revisar el estado de los cables y sus entradas.
- Pruebe a asignar nuevamente la dirección IP de la NVIDIA Jetson ejecutando los siguientes comandos en una terminal (debe tener permisos de superusuario). Es importante que estos comandos se ejecuten en ese orden:

```
sudo ifconfig eth0 169.254.47.88  
sudo ifconfig dhclient eth0
```

Error de PyCUDA

Error: Aparece el siguiente mensaje de error en la terminal:

```
PyCUDA ERROR: The context stack was not empty upon module cleanup.  
-----  
A context was still active when the context stack was being  
cleaned up. At this point in our execution, CUDA may already  
have been deinitialized, so there is no way we can finish  
cleanly. The program will be aborted now.  
Use Context.pop() to avoid this problem.
```

Motivo: El sistema fue cerrado o su proceso fue terminado de forma abrupta (por medio de **Ctrl+C**). Sucede cuando la terminal que ejecuta al sistema es cerrada.

Solución:

- Cuando el visor que genera publicidad dirigida y estática este abierto, siempre cierre la ventana con la tecla **Q** del teclado (no use el botón **X**).
- Nunca cierre el sistema por medio de la terminal. En su lugar utilice el botón **X** en la ventana de la interfaz de **Login de acceso**.
- Utilice el botón **Regresar** para el resto de interfaces mientras se navega entre ellas (no use el botón **X**).

Errores de OpenCV

Error: Aparecen los siguientes mensajes de error en la terminal:

```
ERROR GENERADO: OpenCV(4.1.1)
↳ /home/nvidia/host/build_opencv/nv_opencv/modules/imgproc/src/resize.cpp:3720: error:
↳ (-215:Assertion failed) !_ssize.empty() in function 'resize'
ERROR GENERADO: OpenCV(4.1.1)
↳ /home/nvidia/host/build_opencv/nv_opencv/modules/imgproc/src/color.cpp:182: error:
↳ (-215:Assertion failed) !_src.empty() in function 'cvtColor'
```

Motivo: El sistema tiene problemas para detectar los rostros de las personas. No obstante, mientras las recomendaciones en la publicidad dirigida continúen generándose de forma normal, este fenómeno no afecta a la funcionalidad del sistema.

Possible solución: Para intentar reducir la aparición de estos mensajes, considere ajustar la cámara para tener una mejor visualización de imagen, además de procurar tener el rostro libre de objetos externos (como brazos, bufandas, lentes de sol, gorras, etc.). Recuerde que este no es un error crítico para el sistema, por lo que la anterior solución no asegura la eliminación definitiva del mismo.

Acceso al sistema como administrados

Error: El usuario y/o contraseña ingresados son incorrectos.

Motivo: Esto se debe a que los datos de acceso no pueden ser autenticados como administrador.

Solución: Póngase en contacto con un usuario que posea este rol.

Problemas al intentar añadir nuevas bebidas o publicidad

Error: No se puede cargar una imagen al intentar agregar una nueva bebida o publicidad estática.

Motivo: Esto se debe a que la imagen que se intenta cargar no cumple con las características necesarias para ser mostrada adecuadamente en la pantalla publicitaria.

Solución: Asegúrese de que el formato de la imagen sea JPG, que tenga 3 canales de color (RGB) y que su resolución esté entre 720p y 4K.

Contacto

- Jaime Arturo Lara Cázares: jlarac1300@alumno.ipn.mx
- Víctor Leonel Morales Flores: vmoralesf1300@alumno.ipn.mx
- Enrique Ramos Diaz: eramosd1300@alumno.ipn.mx

Bibliografía

- [1] A. Ferraz Martínez, *El lenguaje de la publicidad*, 8th ed. España: Arco Libros, 2004.
- [2] J. Otegui Castillo, "La realidad virtual y la realidad aumentada en el proceso de marketing," *Revista de Dirección y Administración de Empresas*, no. 24, p. 183, diciembre 2017.
- [3] A. Singhal, P. Sinha, and R. Pant, "Use of deep learning in modern recommendation system: A summary of recent works," *International Journal of Computer Applications (0975 – 8887)*, vol. 180, no. 7, pp. 17–22, diciembre 2017.
- [4] D. Marimón, T. Adamek, K. Göllner, and C. Domingo, "El futuro de la realidad aumentada móvil," *TELOS: Contenidos digitales para la nueva televisión*, vol. 84, pp. 10–11, septiembre 2010.
- [5] J. Sculley, *Moonshot!*, 1st ed. México: Planeta, 2016.
- [6] J. Klaric, *Véndele a la mente, no a la gente*. Business & Innovation Institute of America, 2015.
- [7] R. Manzano, T. Serra, and D. Gavilán Bouzas, "Marketing sensorial: nuevas claves para su gestión," *Marketing y Ventas*, vol. 103, pp. 28–33, 2011.
- [8] F. Coll Morales, "¿La publicidad intrusiva ciertamente funciona?" *Forbes México*, abril 2018.
- [9] C. Intel, "Digital Signage Solutions including Intel AIM Suite," <https://aimsuite.intel.com/sites/default/files/resources/Presentation%20-%20Intel%20OSC%20Digital%20Signage%20Webinar%20-%20Oct%2013%2C%202011.pdf>, 2011, [En línea; último acceso 1 septiembre de 2019].
- [10] J. M. Balderas Sánchez, "Aplicación móvil para recomendación de bebidas basada en redes neuronales profundas," Trabajo terminal, ESCOM-IPN, México, 2018.
- [11] P. Kotler and K. L. Keller, *Dirección de Marketing*, 14th ed. México: Pearson, 2012.
- [12] A. Bleier, A. De Keyser, and K. Verleye, *Customer Engagement Through Personalization and Customization*, agosto 2018, pp. 75–94.
- [13] K. Kwon and C. Kim, "How to Design Personalization in a Context of Customer Retention: Who Personalizes What and to What Extent?" *Electron. Commer. Rec. Appl.*, vol. 11, no. 2, p. 101–116, Mar. 2012.

- [14] A. Halimi, A. Chavosh, and S. Choshalyc, “The Influence of Relationship Marketing Tactics on Customer’s Loyalty in B2C Relationship – The Role of Communication and Personalization,” *European Journal of Economics, Finance and Administrative Sciences*, vol. 31, abril 2011.
- [15] N. Arora, X. Dreze, A. Ghose, J. Hess, R. Iyengar, B. Jing, Y. Joshi, V. Kumar, N. Lurie, S. Neslin, S. Sa-jeesh, M. Su, N. Syam, J. Thomas, and Z. Zhang, “Putting One-to-One Marketing to Work: Personalization, Customization and Choice,” 04 2008.
- [16] D. Ball, P. S. Coelho, and M. J. Vilares, “Service Personalization and Loyalty,” *Journal of Services Marketing*, 2006.
- [17] D. Pérez and I. Pérez, “La fidelización de los clientes. Técnicas CRM. Las nuevas aplicaciones del marketing moderno,” *Escuela de Negocios*, vol. 53, 2006.
- [18] I. Heimbach, D. S. Kostyra, and O. Hinz, “Marketing Automation,” *Business & Information Systems Engineering*, vol. 57, no. 2, pp. 129–133, 2015.
- [19] Wikipedia, “Reconocimiento de Objetos,” https://es.wikipedia.org/w/index.php?title=Reconocimiento_de_Objetos&oldid=120962962, 2019, [En línea; último acceso 17 abril de 2020].
- [20] E. R. Davies, *Computer and Machine Vision: Theory, Algorithms, Practicalities*. Academic Press, 2012.
- [21] S. S. Mohamed, N. M. Tahir, and R. Adnan, “Background Modelling And Background Subtraction Performance For Object Detection,” in *2010 6th International Colloquium on Signal Processing & its Applications*. IEEE, 2010, pp. 1–6.
- [22] J. R. Parker, *Algorithms For Image Processing And Computer Vision*. John Wiley & Sons, 2010.
- [23] S. Khan, H. Rahmani, S. A. A. Shah, and M. Bennamoun, “A Guide To Convolutional Neural Networks For Computer Vision,” *Synthesis Lectures on Computer Vision*, vol. 8, no. 1, pp. 1–207, 2018.
- [24] D. J. Matich, “Redes Neuronales: Conceptos básicos y aplicaciones,” *Universidad Tecnológica Nacional*, México, 2001.
- [25] T. Hagan Martin, B. Demuth Howard, H. Beale Mark *et al.*, “Neural network design,” *University of Colorado at Boulder*, 2002.
- [26] L. Torrey and J. Shavlik, “Transfer Learning,” in *Handbook of Research on Machine Learning Applications and Trends: Algorithms, Methods, and Techniques*. IGI Global, 2010, pp. 242–264.
- [27] I. Goodfellow, Y. Bengio, and A. Courville, *Deep Learning*. MIT press, 2016.
- [28] N. Srivastava, G. Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov, “Dropout: A Simple Way To Prevent Neural Networks From Overfitting,” *Journal Of Machine Learning Research*, vol. 15, no. 1, pp. 1929–1958, 2014.
- [29] K. Simonyan and A. Zisserman, “Very Deep Convolutional Networks For Large-scale Image Recognition,” *arXiv preprint arXiv:1409.1556*, 2014.
- [30] J. Hui, “SSD Object Detection: Single Shot MultiBox Detector for Real-time Processing,” https://medium.com/@jonathan_hui/ssd-object-detection-single-shot-multibox-detector-for-real-time-processing-9bd8deac0e06, marzo 2018, [En línea; último acceso 17 abril de 2020].
- [31] W. Liu, D. Anguelov, D. Erhan, C. Szegedy, S. Reed, C.-Y. Fu, and A. C. Berg, “SSD: Single Shot Multibox Detector,” in *European Conference On Computer Vision*. Springer, 2016, pp. 21–37.

-
- [32] A. Driaba, A. Gordeev, and V. Klyachin, “Recognition of Various Objects from a Certain Categorical Set in Real Time Using Deep Convolutional Neural Networks,” 2019.
 - [33] C.-F. Wang, “A Basic Introduction to Separable Convolutions,” <https://towardsdatascience.com/a-basic-introduction-to-separable-convolutions-b99ec3102728>, agosto 2018, [En línea; último acceso 17 abril de 2020].
 - [34] S.-H. Tsang, “Review: MobileNetV1 — Depthwise Separable Convolution (Light Weight Model),” <https://towardsdatascience.com/review-mobilenetv1-depthwise-separable-convolution-light-weight-model-a382df364b69>, octubre 2018, [En línea; último acceso 17 abril de 2020].
 - [35] K. Ricanek and B. Barbour, “What Are Soft Biometrics and How Can They Be Used?” *IEEE Computer*, vol. 44, pp. 106–108, septiembre 2011.
 - [36] M. Vasileiadis, G. Stavropoulos, and D. Tzovaras, “Facial Soft Biometrics Detection on Low Power Devices,” Master’s thesis, Information Technologies Institute, Centrre for Research and Technology Hellas.
 - [37] Y. Jiang, S. Li, P. Liu, and Q. Dai, “Multi-feature Deep Learning For Face Gender Recognition,” in *2014 IEEE 7th Joint International Information Technology and Artificial Intelligence Conference*. IEEE, 2014, pp. 507–511.
 - [38] G. García, “La Importancia de los Rangos Generacionales en la Estrategia Empresarial,” <https://www.biplaza.es/rangos-generacionales>, septiembre 2019, [En línea; último acceso 7 diciembre de 2020].
 - [39] AINIA, “Más del 55 por ciento de los Millennials prefiere el sabor dulce,” <https://www.ainia.es/noticias/prensa/mas-del-55-por-ciento-de-los-millennials-prefiere-el-sabor-dulce/?fbclid=IwAR3Wm2DDvxBTjzFg2P9FBGush3EcyNdMfkMq3JwypAyLQSyezrzqXSZV9cg>, [En línea, último acceso 19 de diciembre de 2020].
 - [40] R. Barragán, O. Coltell, O. Portolés, E. M. Asensio, J. V. Sorlí, C. Ortega-Azorín, J. I. González, C. Sáiz, R. Fernández-Carrión, J. M. Ordovas *et al.*, “Bitter, Sweet, Salty, Sour and Umami Taste Perception Decreases with Age: Sex-specific Analysis, Modulation by Genetic Variants and Taste-preference Associations in 18 to 80 year-old subjects,” *Nutrients*, vol. 10, no. 10, p. 1539, 2018.
 - [41] E. A. Gutiérrez, R. A. Rodríguez, L. F. Robles, G.-O. Olivera, D. García-Ordás, M. García-Ordás, and V. G. Castro, “Cómo Estimar la Edad de las Personas Utilizando Imágenes de su Rostro,” *Cuadernos de la Cátedra Telefónica*, no. 2, pp. 3–22, 2014.
 - [42] G. Rodríguez, “Estimacion de edad aparente utilizando informacion facial,” 2018.
 - [43] Y. Fu, G. Guo, S. Member, and T. S. Huang, “Age Synthesis and Estimation Via Faces,” noviembre 2010, [En línea; último acceso 7 diciembre de 2020].
 - [44] M. Yang, S. Zhu, F. Lv, and K. Yu, “Correspondence Driven Adaptation for Human Profile Recognitional,” in *CVPR 2011*. IEEE, 2011, pp. 505–512.
 - [45] G. Levi and T. Hassner, “Age and Gender Classification Using Convolutional Neural Networks,” in *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*, junio 2015.
 - [46] “Gender Differences In Food Choice And Dietary Intake In Modern Western Societies, author=Arganini, Claudia and Saba, Anna and Comitato, Raffaella and Virgili, Fabio and Turrini, Aida,” *Public Health-social and Behavioral Health*, vol. 4, pp. 83–102, 2012.
 - [47] C. J. Day, “An Exploration of the Relationships Between Personality, Eating Behaviour and Taste Preference,” Ph.D. dissertation, Sheffield Hallam University, 2009.

- [48] Y. Zhou, H. Ni, F. Ren, and X. Kang, “Face and Gender Recognition System Based on Convolutional Neural Networks,” in *2019 IEEE International Conference on Mechatronics and Automation (ICMA)*. IEEE, 2019, pp. 1091–1095.
- [49] C. R. Cloninger, T. R. Przybeck, and D. M. Svarkic, “The tridimensional personality questionnaire: Us normative data,” *Psychological reports*, vol. 69, no. 3, pp. 1047–1057, 1991.
- [50] C. R. Cloninger, “A systematic method for clinical description and classification of personality variants: A proposal,” *Archives of general psychiatry*, vol. 44, no. 6, pp. 573–588, 1987.
- [51] M. Moreno Sotelo, “Smart Prediction of Apparent Personality Traits through Big Five from Selfies,” Tesis Maestría, CIC-IPN, México, 2019.
- [52] O. Kampman, E. J. Barezi, D. Bertero, and P. Fung, “Investigating Audio, Video, and Text Fusion Methods for End-to-End Automatic Personality Prediction,” in *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*. Melbourne, Australia: Association for Computational Linguistics, julio 2018, pp. 606–611.
- [53] J. Gorbova, E. Avots, I. Lüsi, M. Fishel, S. Escalera, and G. Anbarjafari, “Integrating Vision and Language for First-Impression Personality Analysis,” *IEEE MultiMedia*, vol. 25, no. 2, pp. 24–33, 2018.
- [54] C. Ventura, D. Masip, and A. Lapedriza, “Interpreting CNN Models for Apparent Personality Trait Regression,” in *2017 IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, 2017, pp. 1705–1713.
- [55] V. Ponce-López, B. Chen, M. Oliu, C. Corneau, A. Clapés, I. Guyon, X. Baró, H. J. Escalante, and S. Escalera, “ChaLearn LAP 2016: First Round Challenge on First Impressions -Dataset and Results,” *European Conference on Computer Vision*, octubre 2016.
- [56] H. Suen, K. Hung, and C. Lin, “TensorFlow-Based Automatic Personality Recognition Used in Asynchronous Video Interviews,” *IEEE Access*, vol. 7, pp. 61 018–61 023, 2019.
- [57] J. C. S. Jacques Júnior, Y. Güclütürk, M. Pérez, U. Güçlü, C. Andújar, X. Baró, H. J. Escalante, I. Guyon, M. A. J. van Gerven, R. van Lier, and S. Escalera, “First Impressions: A Survey on Computer Vision-Based Apparent Personality Trait Analysis,” *CoRR*, vol. abs/1804.08046, 2018.
- [58] D. Das, L. Sahoo, and S. Datta, “A Survey on Recommendation System,” *International Journal of Computer Applications*, vol. 160, no. 7, 2017.
- [59] G. Takács, I. Pilászy, B. Németh, and D. Tikk, “Major Components of the Gravity Recommendation System,” *ACM Sigkdd Explorations Newsletter*, vol. 9, no. 2, pp. 80–83, 2007.
- [60] P. Brusilovsky, A. Kobsa, and W. Nejdl, *The adaptive web: methods and strategies of web personalization*. Springer Science & Business Media, 2007, vol. 4321.
- [61] V. Berlanga, M. J. Rubio Hurtado, and R. Vilà Baños, “Cómo aplicar árboles de decisión en SPSS,” *REIRE. Revista d’Innovació i Recerca en Educació*, vol. 6, no. 1, pp. 65–79, 2013.
- [62] K. M. Leung, “Naive Bayesian Classifier,” *Polytechnic University Department of Computer Science/Finance and Risk Engineering*, vol. 2007, pp. 123–156, 2007.
- [63] K. M. Al-Aidaroos, A. A. Bakar, and Z. Othman, “Naive Bayes Variants in Classification Learning,” in *2010 International Conference on Information Retrieval & Knowledge Management (CAMP)*. IEEE, 2010, pp. 276–281.

-
- [64] P. Brusilovsky, A. Kobsa, and W. Nejdl, *KNN*. Springer Science & Business Media, 2007, vol. 4321.
 - [65] R. A. Mur, “Clasificadores KNN-I.”
 - [66] L. Heras Lara and J. L. Villarreal Benítez, “Realidad Aumentada: Una Tecnología en Espera de Usuarios,” *Revista Digital Universitaria*, vol. 8, no. 6, pp. 2–9, 2007.
 - [67] O. González Reyes, J. C. Regalado Enríquez, J. D. K. Robles Pérez, and J. Zerón Ramírez, “Galería Inmersiva de Dinosaurios GID,” Trabajo Terminal, ESCOM-IPN, México, 2015.
 - [68] J. Carmigniani, B. Furht, M. Anisetti, P. Ceravolo, E. Damiani, and M. Ivkovic, “Augmented Reality Technologies, Systems and Applications,” *Multimedia Tools and Applications*, vol. 51, no. 1, pp. 341–377, 2011.
 - [69] R. T. Azuma, “A Survey of Augmented Reality,” *Presence: Teleoperators and Virtual Environments*, vol. 6, no. 4, pp. 355–385, 1997.
 - [70] A. W. Services, “¿Qué es NoSQL?” <https://aws.amazon.co/es/nosql/>, [En línea; último acceso 7 diciembre de 2020].
 - [71] C. Intel, “Inside AIM Suite: Pricing/Licensing,” <https://aimsuite.intel.com/inside-aim-suite/pricinglicensing>, [En línea; último acceso 1 septiembre de 2019].
 - [72] C. I. Orozco, F. Iglesias, M. E. Buemi, and J. J. Berlles, “Real-time Gender Recognition From Face Images Using Deep Convolutional Neural Network,” 2017.
 - [73] E. Learned-Miller, G. B. Huang, A. RoyChowdhury, H. Li, and G. Hua, “Labeled Faces in the Wild: A Survey,” in *Advances in Face Detection and Facial Image Analysis*. Springer, 2016, pp. 189–248.
 - [74] A. C. Gallagher and T. Chen, “Understanding Images of Groups of People,” in *2009 IEEE Conference on Computer Vision and Pattern Recognition*. IEEE, 2009, pp. 256–263.
 - [75] E. Fazl-Ersi, M. E. Mousa-Pasandi, R. Laganiere, and M. Awad, “Age and Gender Recognition Using Informative Features of Various Types,” in *2014 IEEE International Conference on Image Processing (ICIP)*. IEEE, 2014, pp. 5891–5895.
 - [76] J. Mansanet, A. Albiol, and R. Paredes, “Local Deep Neural Networks for Gender Recognition,” *Pattern Recognition Letters*, vol. 70, pp. 80–86, 2016.
 - [77] R. Ranjan, S. Sankaranarayanan, C. D. Castillo, and R. Chellappa, “An All-in-one Convolutional Neural Network For Face Analysis,” in *2017 12th IEEE International Conference on Automatic Face & Gesture Recognition (FG 2017)*. IEEE, 2017, pp. 17–24.
 - [78] Z. Liu, P. Luo, X. Wang, and X. Tang, “Deep Learning Face Attributes in the Wild,” in *Proceedings of the IEEE International Conference on Computer Vision*, 2015, pp. 3730–3738.
 - [79] S. Escalera, M. Torres Torres, B. Martinez, X. Baró, H. Jair Escalante, I. Guyon, G. Tzimiropoulos, C. Corneou, M. Oliu, M. Ali Bagheri *et al.*, “Chalearn Looking at People and Faces of the World: Face Analysis Workshop and Challenge 2016,” in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, 2016, pp. 1–8.
 - [80] S. Sankaranarayanan, A. Alavi, C. D. Castillo, and R. Chellappa, “Triplet Probabilistic Embedding for Face Verification and Clustering,” in *2016 IEEE 8th International Conference on Biometrics Theory, Applications and Systems (BTAS)*. IEEE, 2016, pp. 1–8.

- [81] N. Zhang, M. Paluri, M. Ranzato, T. Darrell, and L. Bourdev, “Panda: Pose Aligned Networks for Deep Attribute Modeling,” in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2014, pp. 1637–1644.
- [82] M. Ehrlich, T. J. Shields, T. Almaev, and M. R. Amer, “Facial Attributes Classification Using Multi-task Representation Learning,” in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, 2016, pp. 47–55.
- [83] R. Ranjan, V. M. Patel, and R. Chellappa, “Hyperface: A Deep Multi-task Learning Framework For Face Detection, Landmark Localization, Pose Estimation, and Gender Recognition,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 41, no. 1, pp. 121–135, 2017.
- [84] J. Wang, Y. Cheng, and R. Schmidt Feris, “Walk and Learn: Facial Attribute Representation Learning From Egocentric Video and Contextual Data,” in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2016, pp. 2295–2304.
- [85] S. Escalera, J. Fabian, P. Pardo, X. Baró, J. Gonzalez, H. J. Escalante, D. Misevic, U. Steiner, and I. Guyon, “Chalearn Looking at People 2015: Apparent Age and Cultural Event Recognition Datasets and Results,” in *Proceedings of the IEEE International Conference on Computer Vision Workshops*, 2015, pp. 1–9.
- [86] H. Han, C. Otto, and A. K. Jain, “Age Estimation From Face Images: Human vs. Machine Performance,” in *2013 International Conference on Biometrics (ICB)*. IEEE, 2013, pp. 1–8.
- [87] R. Ranjan, S. Zhou, J. Cheng Chen, A. Kumar, A. Alavi, V. M. Patel, and R. Chellappa, “Unconstrained Age Estimation With Deep Convolutional Neural Networks,” in *Proceedings of the IEEE International Conference on Computer Vision Workshops*, 2015, pp. 109–117.
- [88] J.-C. Chen, A. Kumar, R. Ranjan, V. M. Patel, A. Alavi, and R. Chellappa, “A Cascaded Convolutional Neural Network for Age Estimation of Unconstrained Faces,” in *2016 IEEE 8th International Conference on Biometrics Theory, Applications and Systems (BTAS)*. IEEE, 2016, pp. 1–8.
- [89] R. Rothe, R. Timofte, and L. Van Gool, “Dex: Deep Expectation of Apparent Age from a Single Image,” in *Proceedings of the IEEE International Conference on Computer Vision Workshops*, 2015, pp. 10–15.
- [90] X. Liu, S. Li, M. Kan, J. Zhang, S. Wu, W. Liu, H. Han, S. Shan, and X. Chen, “Agenet: Deeply Learned Regressor and Classifier for Robust Apparent Age Estimation,” in *Proceedings of the IEEE International Conference on Computer Vision Workshops*, 2015, pp. 16–24.
- [91] J. Xing, K. Li, W. Hu, C. Yuan, and H. Ling, “Diagnosing Deep Learning Models for High Accuracy Age Estimation from a Single Image,” *Pattern Recognition*, vol. 66, pp. 106–116, 2017.
- [92] Y. Liu, A. Wai Kin Kong, and C. Keong Goh, “A constrained deep neural network for ordinal regression,” in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2018, pp. 831–839.
- [93] A. Krizhevsky, I. Sutskever, and G. E. Hinton, “Imagenet classification with deep convolutional neural networks,” in *Advances in Neural Information Processing Systems*, 2012, pp. 1097–1105.
- [94] M. Moreno Sotelo, “personality,” <https://github.com/miguelmore/personality>, [En línea; último acceso 22 diciembre de 2020].
- [95] M. Mirror, “Magic Mirror Technologies and Latest Development,” <https://www.magicmirror.me/Products/Ph otobooth- Technologies>, [En línea; último acceso 1 septiembre de 2019].

-
- [96] T. W. Mirrors, “Smart Mirror Product Information (For 2019),” <https://www.twowaymirrors.com/smart-mirror/>, [En línea; último acceso 1 septiembre de 2019].
 - [97] A. Arcangel, “Snapchat Filters: How Do They Work?” <https://dev.to/aubreyarcangel/snapchat-filters-how-do-theywork-4c83>, [En línea; último acceso 1 septiembre de 2019].
 - [98] J. Francia, “¿Qué es Scrum?” <https://www.scrum.org/resources/blog/que-es-scrum>, [En línea; último acceso 1 septiembre de 2019].
 - [99] O. O. Soto Romero and G. H. A. Salinas, “Scrum, ¿Un Paradigma de Administración de Proyectos que Cumple lo que Promete?” *Software Gurú, Conocimiento en Práctica*, vol. 5, pp. 1–35, agosto 2009.
 - [100] ClickUp, “ClickUp: All the features, without all the mess,” <https://clickup.com/features>, [En línea; último acceso 1 septiembre de 2019].
 - [101] R. S. Pressman, *Ingeniería del Software: Un enfoque práctico*. McGraw-Hill Interamericana, 2005.
 - [102] K. E. Kendall, *Análisis y Diseño de Sistemas*. Pearson Educación, 2005.
 - [103] I. Sommerville and M. Alfonso Galipienso, “Ingeniería de Software,” Ciudad de México, México, 2011.
 - [104] G. Booch, J. Rumbaugh, I. Jacobson, J. J. G. Molina, and J. S. Martínez, *El lenguaje unificado de modelado: guía del usuario*. Addison-Wesley, 2006.
 - [105] Hikvision, “Datasheet DS-2CD2143G0-I(S) 4 MP IR Fixed Dome Network Camera,” <http://www.cctvcentersl.es/upload/Catalogos/DS-2CD2143G0-I.pdf>, [En línea; último acceso 28 marzo de 2020].
 - [106] ——, “Datasheet DS-3E0105P-E 4-Port 100 Mbps Unmanaged PoE Switch,” https://us.hikvision.com/sites/default/files/acc_ds-3e0105p-e_040119na.pdf, [En línea; último acceso 28 marzo de 2020].
 - [107] ——, “Manual de Usuario UD14456B Network Camera,” https://www.hikvision.com/mtsc/uploads/product/accessory/UD14456B_Baseline_User_Manual_of_Network_Camera_V5.6.0_20190430.pdf, [En línea; último acceso 12 abril de 2020].
 - [108] NVIDIA, “Jetson AGX Xavier Developer Kit,” <https://developer.nvidia.com/embedded/jetson-agx-xavier-developer-kit>, [En línea; último acceso 8 diciembre de 2020].
 - [109] ——, “Jetson AGX Xavier and the New Era Of Autonomous Machines,” http://info.nvidia.com/rs/156-OFN-742/images/Jetson_AGX_Xavier_New_Era_Autonomous_Machines.pdf, [En línea; último acceso 28 marzo de 2020].
 - [110] D. Franklin, “Jetson Nano Brings AI Computing to Everyone,” <https://devblogs.nvidia.com/jetson-nano-ai-computing/>, marzo 2019, [En línea; último acceso 28 marzo de 2020].
 - [111] NVIDIA, “Jetpack,” <https://developer.nvidia.com/embedded/jetpack>, [En línea; último acceso 29 marzo de 2020].
 - [112] ——, “Install Jetson Software with SDK Manager,” <https://docs.nvidia.com/sdk-manager/install-with-sdkmanager/index.html>, [En línea; último acceso 29 marzo de 2020].
 - [113] ——, “NVIDIA SDK Manager,” <https://developer.nvidia.com/nvidia-sdk-manager>, [En línea; último acceso 29 marzo de 2020].
 - [114] TensorFlow, “¿Por qué TensorFlow?” <https://www.tensorflow.org/about>, [En línea; último acceso 18 noviembre de 2020].

-
- [115] Google, “Te damos la bienvenida a Colaboratory,” <https://colab.research.google.com/notebooks/intro.ipynb>, [En línea; último acceso 19 noviembre de 2020].
 - [116] MongoDB, “What is MongoDB?” <https://www.mongodb.com/es/what-is-mongodb>, [En línea, último acceso 10 de diciembre de 2020].
 - [117] OpenCV, “About,” <https://opencv.org/about/>, [En línea; último acceso 18 noviembre de 2020].
 - [118] NVIDIA, “NVIDIA TensorRT Documentation,” <https://docs.nvidia.com/deeplearning/tensorrt/developer-guide/index.html>, [En línea; último acceso 19 noviembre de 2020].
 - [119] ——, “NVIDIA TensorRT,” <https://developer.nvidia.com/tensorrt>, [En línea; último acceso 18 noviembre de 2020].
 - [120] NumPy, “About Us,” <https://numpy.org/about/>, [En línea; último acceso 18 noviembre de 2020].
 - [121] OpenWeather, “OpenWeather Mobile App,” <https://openweathermap.org/>, [En línea, último acceso 10 de diciembre de 2020].
 - [122] NVIDIA, “Guía de Usuario Jetson AGX Xavier Developer Kit,” https://developer.download.nvidia.com/embedded/L4T/r32-3-1_Release_v1.0/jetson_agx_xavier_developer_kit_user_guide.pdf?Expires=1608401800&Signature=sLtpQSpznSAz2VMMo_Efxj76pc4UmKinJlgJ-GlceqPAUf9XGDt_mE-QxcRaHaE00AzUmn40l8QFLjJ05TNvfisBHdALMI95mRBLqVBmirY_1C0r_Ew7r6eClgHO7C-TyILxe_uMDpM2B5ofXLTOccNn9sKB1cPmH49HzLzkXUGwJhsjfPJFy0pWWmEmyrY, [En línea; último acceso 12 abril de 2020].
 - [123] jwielandt and NVIDIA, “Connecting IP Cameras to Nvidia Jetson AGX Xavier,” <https://forums.developer.nvidia.com/t/connecting-ip-cameras-to-nvidia-jetson-agx-xavier/82954>, [En línea; último acceso 12 abril de 2020].
 - [124] J. Jung, “How to Capture and Display Camera Video with Python on Jetson TX2,” <https://jkjung-avt.github.io/tx2-camera-with-python/>, [En línea; último acceso 12 abril de 2020].
 - [125] NVIDIA, “Guía de Usuario Accelerated GStreamer for NVIDIA Jetson AGX Xavier devices,” https://developer.download.nvidia.com/embedded/L4T/r32_Release_v1.0/Docs/Accelerated_GStreamer_User_Guide.pdf?Expires=1608401800&Signature=g-BRbe7mL6knMCqxD5Y2vM2eik9nx5-oljqOG2Iaf8LwTGZaimYBBWpa6IU6eFZTN_q1tBskXngsb1zg6rxstYD5bvlyeLUSugY-IWr1gGMkDN2Xm3NgmfihTsS4PqhGsbsiYmjNzr05RwRpJeOV-bs2hS6ZHKO fzwlPZgqxFtXyPPJNks, [En línea; último acceso 12 abril de 2020].
 - [126] AastaNV and NVIDIA, “TensorRT Python Sample for Object Detection,” https://github.com/AastaNV/TRT_object_detection, [En línea; último acceso 19 noviembre de 2020].
 - [127] J. Huang, V. Rathod, D. Chow, C. Sun, M. Zhu, A. Fathi, and Z. Lu, “Tensorflow Object Detection Api,” https://github.com/tensorflow/models/tree/master/research/object_detection, [En línea; último acceso 7 diciembre de 2020].
 - [128] D. Mishra and DataTurks, “Face Detection in Images Dataset,” <https://www.kaggle.com/datarurks/face-detection-in-images>, 2018, [En línea; último acceso 7 diciembre de 2020].
 - [129] R. Vemulapalli and A. Agarwala, “A Compact Embedding for Facial Expression Similarity,” in *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, junio 2019.
 - [130] J. Bédorf, “Deploying SSD MobileNet V2 on the NVIDIA Jetson and Nano Platforms,” <https://www.minds.ai/post/deploying-ssd-mobilenet-v2-on-the-nvidia-jetson-and-nano-platforms>, diciembre 2019, [En línea; último acceso 19 noviembre de 2020].

-
- [131] J. Jung, “tensorrt_demos/ssd/build_engine.py,” https://github.com/jkjung-avt/tensorrt_demos/blob/ssd_mobilenet_v3/ssd/build_engine.py#L286, enero 2020, [En línea; último acceso 28 diciembre de 2020].
 - [132] A. Clapés, O. Bilici, D. Temirova, E. Avots, G. Anbarjafari, and S. Escalera, “From Apparent to Real Age: Gender, Age, Ethnic, Makeup, and Expression Bias Analysis in Real Age Estimation,” in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, 2018, pp. 2373–2382.
 - [133] K. Kärkkäinen and J. Joo, “FairFace: Face Attribute Dataset for Balanced Race, Gender, and Age,” *arXiv preprint arXiv:1908.04913*, 2019.
 - [134] J. Cheng, Y. Li, J. Wang, L. Yu, and S. Wang, “Exploiting Effective Facial Patches for Robust Gender Recognition,” *Tsinghua Science and Technology*, vol. 24, no. 3, pp. 333–345, 2019.
 - [135] AastaLLL and NVIDIA, “How to use TensorRT by the multi-threading package of python,” <https://forums.developer.nvidia.com/t/how-to-use-tensorrt-by-the-multi-threading-package-of-python/123085/8>, mayo 2020, [En línea; último acceso 23 diciembre de 2020].
 - [136] M. Spencer, E. Sage, M. Velez, and J.-X. Guinard, “Using Single Free Sorting and Multivariate Exploratory Methods to Design a New Coffee Taster’s Flavor Wheel,” *Journal of Food Science*, vol. 81, no. 12, pp. S2997–S3005, 2016.
 - [137] R. Pressman, *Ingeniería del Software (Un Enfoque Práctico)*, 7th ed. España: Mc-GrawHill, 2010, vol. 1.