



UNIVERSIDAD DE LAS PALMAS DE GRAN CANARIA
Escuela de Ingeniería Informática



Re-identificación de personas en redes de sensores RGBD

Proyecto Fin de Carrera

Alberto Manuel Mireles Suárez

Tutor: Modesto Castrillón Santana

Tutor: Javier Lorenzo Navarro

Tutor: Daniel Hernández Sosa

Las Palmas de Gran Canaria, Mayo de 2015

Agradecimientos

Me gustaría mostrar mi agradecimiento a todas las personas que, de un modo u otro, me han ayudado a lo largo de estos últimos años en la carrera.

En primer lugar quiero darle las gracias a mis tutores: Modesto, Javier y Daniel, por brindarme su apoyo y conocimientos, sin los cuales no habría sido capaz de completar este proyecto. A mis profesores y todos aquellos que formaron parte de esta etapa formativa también les doy las gracias.

Debo mencionar también a los asistentes a las reuniones periódicas con los tutores: Rubén, Luis, Pedro, Eduardo y Dani en las que íbamos mostrando nuestros avances y las dificultades que surgían. Muchas gracias por todas esas pequeñas ideas y sugerencias que ayudaron a sortear los obstáculos que aparecieron por el camino.

Lo mejor que me llevo de estos años en la universidad son los amigos que he hecho, con los que he vivido muchos momentos buenos y algunos que otros difíciles. Quiero nombrar a Dani, Aitor, Eduardo, Mayca y David por haberme ayudado siempre, ser unos muy buenos amigos y haber tenido tanta paciencia conmigo durante las muchas prácticas que realizamos. También quiero mencionar a Atte, quien siempre me apoyó y ayudó cuando lo necesitaba. A ellos y al resto de mis compañeros, muchas gracias.

Por último, quiero agradecer a mis padres y mi hermana por todo el apoyo que me han dado. Gracias por estar ahí cuando os he necesitado y haber tenido tanta paciencia conmigo.

Índice general

Índice de figuras	III
Lista de algoritmos	v
1. Introducción	1
1.1. Objetivos del proyecto	1
1.2. Sistemas de identificación	2
1.2.1. Detección de presencia	3
1.2.2. Rasgos biométricos	3
1.3. Reconocimiento de personas	3
2. Herramientas de desarrollo	7
2.1. Cámara Kinect	7
2.2. Software utilizado	9
2.2.1. Biblioteca OpenCV	9
2.2.2. Biblioteca PCL	10
2.2.3. MATLAB	11
2.3. Entorno de desarrollo	12
3. Planificación	13
3.1. Metodología	13
3.2. Temporización	13
3.3. Coste de la instalación	15
4. Escenario experimental	17
4.1. Posicionamiento de la cámara	18
4.2. Etiquetado	19
4.3. Calibración	21
5. Desarrollo e implementación	23
5.1. Generación de nube de puntos	24

5.2.	Detección de personas	25
5.2.1.	Identificación del fondo	25
5.2.2.	Sustracción del fondo	28
5.3.	Segmentación de individuos	29
5.4.	Seguimiento de personas	34
5.5.	Caracterización de la identidad	41
5.5.1.	Características físicas	42
5.5.2.	Color	46
5.5.3.	Modelado	49
5.6.	Almacenamiento de las características	52
5.7.	Múltiples cámaras	53
5.8.	Búsqueda de correspondencias	54
6.	Resultados de las pruebas	61
6.1.	Espacio de color RGB normalizado	61
6.2.	Espacio de color HSV	68
6.3.	Análisis de resultados	71
7.	Trabajo futuro y conclusiones	73
7.1.	Mejoras del sistema	73
7.1.1.	Dispositivo de captura	73
7.1.2.	Detección y segmentación	74
7.1.3.	Caracterización	74
7.2.	Posibles aplicaciones del sistema	75
7.3.	Conclusiones	76
	Glosario	79
	Bibliografía	81

Índice de figuras

1.1. Sistema automático de control de aduanas SmartGate.	4
1.2. Comprobación de la identidad mediante huella dactilar en un punto de actualización del DNIe	5
1.3. Muestra de las limitaciones de reconocimiento del usuario en Xbox One.	6
2.1. Cámara Kinect de Microsoft	8
2.2. Rango de visión de la cámara Kinect.	8
2.3. Logotipo de la biblioteca OpenCV.	10
2.4. Logotipo de la biblioteca Point Cloud Library.	10
2.5. Logotipo de MATLAB.	11
3.1. Diagrama representativo de la temporización del proyecto. . .	14
3.2. Diagrama representativo de la temporización de la fase de desarrollo.	14
4.1. Geometría del posicionamiento de la cámara Kinect.	19
4.2. Ejemplo del visor mostrando los datos etiquetados para una persona.	20
5.1. Fases que componen el sistema.	23
5.2. Captura de la escena que representa el fondo que debe ser eliminado.	25
5.3. Resultados de la detección de planos empleando la biblioteca PCL.	26
5.4. Nube de puntos con ruido que representa el modelo del fondo detectado.	27
5.5. Fondo modificado al abrir una puerta en un momento posterior al modelado.	28
5.6. Ejemplo de sustracción de fondo	31
5.7. Ejemplo que resalta el área de interés en el que se encuentran individuos diferentes que pueden ser segmentados	33

5.8.	Ejemplo de segmentación de múltiples individuos	34
5.9.	Ejemplo en el que se muestra el correcto seguimiento de las personas.	37
5.10.	Ejemplo de detección de una fusión de personas	39
5.11.	Todas las nubes parciales que representan a una persona, cada una en un color.	43
5.12.	Histogramas de la profundidad a la que se encuentran los puntos.	44
5.13.	Ejemplo de las cuatro secciones a escala	46
5.14.	Ejemplo de los colores promedio de las cuatro secciones para una persona.	47
5.15.	Ejemplo de alineación de 5 nubes parciales de la misma persona.	51
5.16.	Resultado de generar el modelado de una persona a partir de múltiples nubes parciales.	52
5.17.	Estructuras de datos empleadas en el módulo de comparación.	56
6.1.	Primera prueba comparando el grupo de muestras con el grupo de la galería	62
6.2.	Resultados tras la eliminación de las muestras las personas que no se encuentran en la galería	63
6.3.	Resultados tras la eliminación de las muestras las personas del sistema formadas por fusiones.	64
6.4.	Resultados de la primera prueba mediante medidas de distancia variables.	66
6.5.	Resultados de la primera prueba ajustando los pesos del pelo y la cabeza.	67
6.6.	Resultados de la prueba en la que se modifica el umbral de altura.	68
6.7.	Resultados de la primera prueba empleando el espacio de color HSV y H[0-180].	69
6.8.	Resultados del espacio de color HSV para H[0-180] y S[0-50].	70
6.9.	Resultados del espacio de color HSV para H[0-100] y S[0-100].	71

Lista de algoritmos

1. Extracción de grupos euclídeo 30
2. Seguimiento de la trayectoria de una persona 36

Capítulo 1

Introducción

En los últimos años, el interés en detectar y analizar el comportamiento de personas en diferentes contextos ha aumentado. Esto unido a que recientemente se han logrado grandes avances en los sistemas de captura de imágenes, reduciendo a su vez los costes, ha provocado la expansión de estos sistemas lo que resulta en la obtención de muchas capturas que necesitan ser procesadas.

En el presente proyecto se plantea emplear sensores RGBD en entornos interiores en los que pueden variar las condiciones lumínicas, con el fin de atenuar los problemas que sufren los sistemas basados exclusivamente en información visual. Los sensores adoptarán una configuración cenital lo cual permite, tal y como se detallará en el capítulo 4, preservar la privacidad y facilitar la detección y seguimiento de las personas presentes en el rango de visión del sensor. Los sujetos detectados serán descritos en base a sus características físicas y posteriormente re-identificados al ser detectados de nuevo.

1.1. Objetivos del proyecto

En este proyecto se pretende implementar un sistema que permita la re-identificación de personas en un entorno de redes de cámaras. Con la re-identificación se pretende identificar a una misma persona entre las imágenes grabadas con la misma o diferentes cámaras dentro de una red sin conocer la identidad real del individuo. De esta manera seremos capaces de seguir a un mismo individuo a lo largo del área de detección que forman las cámaras, sin que éstas solapen sus campos de visión. Como veremos en el capítulo 4, el uso de herramientas de visión implica realizar un estudio previo del escenario en

el que se trabajará con el fin de obtener un conjunto experimental de datos válido.

Por otro lado, y adelantándonos al capítulo 5, se pretende que el sistema realice de forma independiente las diferentes tareas en las que se puede descomponer el objetivo, siendo éstas en líneas generales:

- *Modelado del fondo.* En primer lugar, será necesario establecer el fondo de la imagen con el fin de distinguir más adelante las personas.
- *Detección de personas.* Esta es una etapa crítica del proyecto ya que de ella depende el correcto funcionamiento del sistema. Se deberá detectar en cada fotograma de la secuencia obtenida y segmentar las personas que pueden encontrarse en el mismo seleccionando regiones de interés.
- *Seguimiento y conversión a nubes de puntos.* No es suficiente con detectar *blobs* ya que no proporcionan suficiente información por sí mismos. Por ello, se ha de poder seguir los múltiples *blobs* que representan a la misma persona avanzando por la zona que se está grabando y convertir la información que se encuentra en una imagen RGB y otra de profundidad a un medio que represente mejor la realidad, concretamente, una nube de puntos tridimensional.
- *Análisis del individuo.* Se procederá a tratar toda la información parcial obtenida de la fase anterior y se obtendrán una serie de valores que caracterizarán a la persona.
- *Comparación.* Basándonos en los descriptores obtenidos se buscarán coincidencias entre las diferentes personas minimizando la diferencia empleando una serie de medidas de distancia.

1.2. Sistemas de identificación

En la actualidad existen diferentes soluciones disponibles con el fin de detectar y caracterizar individuos. Las aplicaciones de estos sistemas son variadas, en base a las capacidades de los mismos, por lo que pueden clasificarse en sistemas de detección de presencia y sistemas que caracterizan la identidad de una persona, pudiendo estos últimos integrar un sistema que permita detectar cuándo una persona entra en la zona de detección. Esto se debe a que en primer lugar, antes de caracterizar la identidad o reconocer a una persona es necesario saber si en la imagen aparece una, por lo que es imperativo detectar la presencia de un individuo.

1.2.1. Detección de presencia

Actualmente, se dispone de múltiples métodos y dispositivos que permiten detectar la presencia de personas con fines diversos, desde facilitar una tarea a las personas hasta los sistemas de seguridad. Los sensores más utilizados para estos casos pueden dividirse en dos categorías principales: los sensores activos, los cuales generan algún tipo de señal o luz y detectan si existe algún cambio en ella, y los pasivos, que se basan en las ondas infrarrojas con el fin de detectar la temperatura del cuerpo humano. Como ejemplos del primer tipo estarían los sensores que controlan puertas o sistemas de iluminación automáticos; mientras que el segundo se emplea en sistemas de seguridad como las alarmas. Como se observa, todos ellos detectan la presencia de las personas tal como lo cual se desencadena una determinada acción.

1.2.2. Rasgos biométricos

Existen diferentes métodos y sistemas que permiten reconocer a una persona basándose en un conjunto de características que la hacen única.

En primer lugar, se encuentran las que podríamos considerar las características biométricas que se emplean a día de hoy ya que son consideradas las que mejores resultados ofrecen. Estas son la huella dactilar o, tal y como propusieron los doctores Carleton Simon e Isadore Goldstein en [Simon and Goldstein, 1935], emplear la retina con el fin de identificar a una persona de forma inequívoca. Sin embargo, aunque son las características más estables y que más detalles tienen, también son más complicadas de obtener sin que la persona se aproxime a un sensor el cual escanee su huella, tal y como se aprecia en la figura 1.2, o la retina.

Por otro lado, nos encontramos con las características físicas más visibles, como son la cara, la altura o la forma del cuerpo. Al contrario que las anteriores, estos rasgos varían con el tiempo, pero pueden ser obtenidos a distancia, mediante sensores ópticos como las cámaras de vigilancia u otro tipo de sensores que incorporen, por ejemplo, información sobre la profundidad.

1.3. Reconocimiento de personas

El reconocimiento de personas, también llamado reconocimiento biométrico, es un problema actual que interesa resolver debido a su utilidad en múltiples ámbitos, tanto los relacionados con la seguridad como otras

aplicaciones menos críticas en un entorno más personal.

Existe un gran interés en investigar y lograr sistemas de reconocimiento de personas con altas tasas de acierto, ya que son un medio que permite automatizar algo que hasta ahora sólo los humanos podemos hacer con gran precisión. A su vez es algo de lo que depende la seguridad, ya sea de un país, una empresa o individuo, ya que reconocer a alguien es algo que se realiza de forma cotidiana. Desde pagar en un comercio con medios electrónicos, acudir a un banco a hacer alguna transacción, acceder al puesto de trabajo, entrar en zonas de acceso restringido o incluso viajar, todo ello requiere que nos identifiquemos.

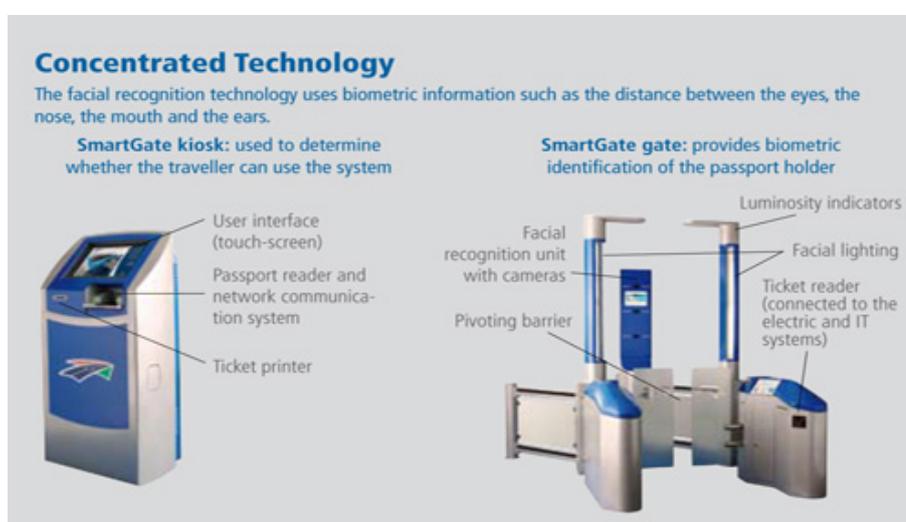


Figura 1.1: Sistema automático de control de aduanas SmartGate. [SAFRAN Morpho, 2015]

En la actualidad algunos de estos sistemas ya se han implantado, como por ejemplo el que aparece en la figura 1.1, *SmartGate*, el cual según [Australian Customs, 2015], es un sistema automatizado de control de pasaportes en funcionamiento en Australia y que, para los portadores de un número limitado de pasaportes electrónicos, permite pasar por la aduana del aeropuerto empleando un sistema de reconocimiento para comparar el individuo con los datos que figuran en el pasaporte. Un sistema similar a este es el IRIS (Iris Recognition Immigration System), el cual según [Daugman, 2009] es usado para el control de fronteras en el Reino Unido con la diferencia de que emplea la retina para reconocer a los individuos.

Otro sistema más sencillo y que puede considerarse bastante extendido es la identificación por huella dactilar. Es un método que se considera fiable y válido como prueba en los juzgados, ya que suele determinar la presencia de un individuo en una escena. Por otro lado, es un sistema empleado por las Fuerzas y Cuerpos de Seguridad del Estado para confirmar la identidad de un individuo ya que, desde el momento en el que, al menos en España, se obtiene un documento de identidad, las huellas dactilares son escaneadas.



Figura 1.2: Comprobación de la identidad mediante huella dactilar en un punto de actualización del DNIe. [Viafirma Developers, 2015]

Finalmente, también existen sistemas de reconocimiento que hacen uso de las técnicas anteriormente mencionadas en un entorno más personal que van desde el reconocimiento del usuario que sostiene un móvil, o identificar al usuario que enciende una consola que disponga de una cámara, como puede ser la Xbox 360 con el sensor Kinect. En este último caso, se almacenan datos sobre la fisonomía del jugador, aunque principalmente se basan en la cara, lo cual hace que sea difícil la re-identificación si el usuario cambia ligeramente su aspecto, como podría ser quitarse o ponerse unas gafas, tal y como se muestra en la figura 1.3.

A pesar de todos estos avances, el empleo de sistemas de re-identificación por lo general están limitados por las estrictas normativas en cuanto a la privacidad de los datos. En este aspecto, los datos biométricos como la huella dactilar, la imagen de la retina o el ADN son considerados información



Figura 1.3: Muestra de las limitaciones de reconocimiento del usuario en Xbox One. [Ackerman, 2013]

sensible que debe ser protegida. Es por tanto debido a la compleja regulación que se aplica a esta información, por lo que no se emplea de manera habitual en los entornos públicos. En [Grupo de Trabajo del Artículo 29, 2012], podemos encontrar un informe en el que se tratan los riesgos de los sistemas biométricos más habituales en el que se concluye que cierta información, como el ADN, se consideran un dato sensible y sobre el que se deberán aplicar garantías especiales aparte de los principios generales de protección de datos.

Capítulo 2

Herramientas de desarrollo

Este capítulo se centra en describir las bases tecnológicas del proyecto, presentando las diferentes herramientas, tanto hardware como software, que se han empleado. El soporte hardware es la cámara Kinect, cuyas características se detallan en la sección 2.1. Por otro lado, el soporte software del que se ha hecho uso es el descrito en la sección 2.2.

2.1. Cámara Kinect

Como se ha comentado anteriormente, el sensor de profundidad utilizado es la Kinect de Microsoft. Se escogió este, debido a lo extendido que está su uso y que existe una comunidad activa desarrollando nuevas aplicaciones que hacen uso del mismo. A su vez, la principal razón por la que se decidió emplear este tipo de cámaras, que detectan la profundidad y el color es que se puede hacer uso de la información de profundidad, la cual no es tan sensible a los cambios de iluminación, pudiendo operar incluso en ambientes oscuros. De depender sólo de imágenes RGB se tendrían dificultades en ciertos entornos en la que la iluminación no es suficiente para obtener datos fiables.

Como se puede apreciar en la figura 2.1, el dispositivo tiene una cámara RGB, un sensor de profundidad y un micrófono multi-array bidireccional que, en conjunto, capturan imágenes y movimientos de los cuerpos en 3D, además de ofrecer reconocimiento facial y comandos de voz.

Tal y como se detalla en [Mathe et al., 2012], el sensor de Kinect obtiene imágenes de video mediante un sensor CMOS de color a una frecuencia de 30 Hz con formato RGB de 32 bits y una resolución VGA de 640x480 píxeles. El canal de video monocromo CMOS es de 16 bits con una resolución QVGA



Figura 2.1: Cámara Kinect de Microsoft. Imagen tomada de [Wikipedia, 2010].

de 320x240 píxeles y hasta 65.536 niveles de sensibilidad.

Para poder calcular la distancia entre un cuerpo y el sensor, se emite un haz láser infrarrojo que proyecta un patrón de puntos sobre los cuerpos cuya distancia quiere ser determinada. Una cámara infrarroja capta este patrón y, mediante hardware, calcula la profundidad de cada punto. El rango de profundidad del sensor está entre 0,4 y 4 metros.

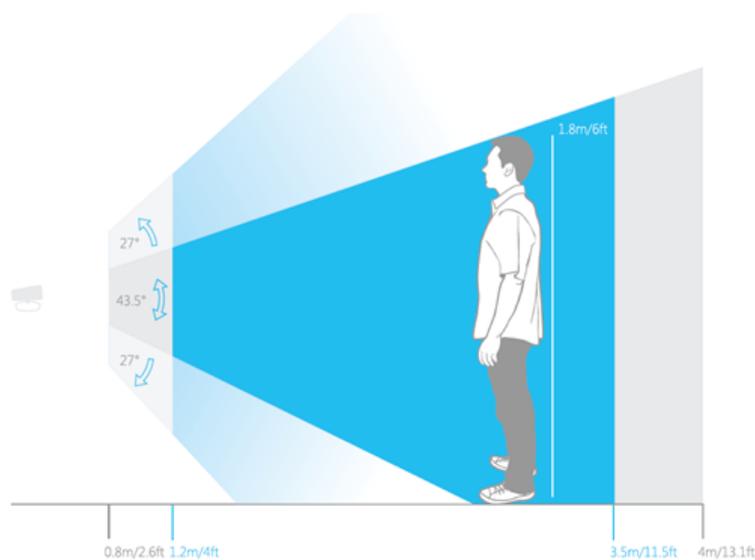


Figura 2.2: Rango de visión de la cámara Kinect. [Kinect for Windows, 2012]

El ángulo de visión (FOV) es de 58° horizontales y 45° verticales. Por otro lado, la cámara dispone de un pivote que permite orientar en elevación, hacia arriba o hacia abajo incrementando por tanto del FOV hasta en 27°.

Sin embargo, las características oficiales del sensor, tal y como se muestra en la figura 2.2 corresponden a los valores óptimos de visión según el fabricante.

Durante el desarrollo de este proyecto, Microsoft lanzó al mercado una nueva versión de la cámara Kinect, la Kinect para Xbox One. A pesar de emplear una tecnología similar al anterior sensor ofreciendo, entre otros aspectos, una mayor resolución de imagen, no se planteó utilizarla debido a la fase en la que encontraba el proyecto.

2.2. Software utilizado

Teniendo en cuenta el área en el que se enmarca este proyecto, es necesario hacer uso de bibliotecas de software que faciliten el tratamiento de imágenes.

Se ha decidido emplear la biblioteca OpenCV (Open Computer Vision, [OpenCV, 2015]) la cual se describirá con más detalle en la sección 2.2.1, para el manejo de los datos de entrada, es decir, las imágenes RGB y de profundidad.

Con el fin de asemejar las imágenes obtenidas al mundo real se hace uso de la biblioteca PCL (Point Cloud Library) [PCL, 2011d], la cual, según se ampliará en la sección 2.2.2, representa las imágenes como nubes de puntos tridimensionales, algo muy útil a la hora de fusionar información de color en dos dimensiones con información de profundidad.

Finalmente, tras procesar los datos de entrada se obtienen valores numéricos que caracterizan la identidad, por lo que se hace uso de la herramienta de *software* matemático MATLAB (MATrix LABoratory) para analizar los resultados de las comparaciones. Se detallará más sobre esta herramienta en la sección 2.2.3.

2.2.1. Biblioteca OpenCV

OpenCV es una biblioteca de código libre de visión artificial que proporciona múltiples funciones para el tratamiento de imágenes. Está escrita en código C y C++ y su código está muy optimizado aprovechando las características de bajo nivel del lenguaje, lo cual le permite ofrecer un rendimiento suficiente para ser empleada en aplicaciones de tiempo real.



Figura 2.3: Logotipo de la biblioteca OpenCV.

Se distribuye bajo licencia BSD por lo que es posible usarla con fines comerciales o de investigación, dentro de los límites de la misma. Esto ha propiciado que sea ampliamente utilizada en diferentes entornos como la vigilancia o los sistemas reconocedores de objetos.

En el presente proyecto se han empleado una serie de funciones de OpenCV para realizar las operaciones necesarias sobre las imágenes con el fin de detectar personas, seguirlas a lo largo de la trayectoria y la comparación de los histogramas de color, aspectos del proyecto que se tratarán más adelante.

2.2.2. Biblioteca PCL

PCL es una biblioteca de relativamente reciente creación, en relación a la anteriormente mencionada. Se encuentra completamente escrita en C++ y en sus inicios apostó por el aprovechamiento de los avances en los conjuntos de instrucciones de los procesadores. A su vez, ofrece soporte para OpenMP (Open Multi-Processing, [OpenMP, 2015]) e Intel® TBB (Intel® Threading Building Blocks, [Intel, 2015]) para explotar la paralelización en los procesadores con múltiples núcleos.



Figura 2.4: Logotipo de la biblioteca Point Cloud Library.

La creación de esta biblioteca fue motivada por la bajada de precios de los sensores 3D como la cámara Kinect. Esta proveía información de una escena tridimensional y era necesario establecer mecanismos para procesar

de manera eficiente las nubes de puntos [Bogdan Rusu and Cousins, 2013]. Se podría considerar que se trata de la evolución del concepto de la biblioteca OpenCV la cual se emplea para el tratamiento de datos en dos dimensiones puesto que tienen en común múltiples funciones.

En la aplicación que se ha desarrollado se ha empleado esta biblioteca no sólo por la idoneidad de la misma para el manejo de nubes de puntos, sino que además, representa la información de una manera más cercana al concepto humano de una escena tridimensional. Es por ello por lo que resulta más sencillo tratar de resolver los diferentes problemas que se han abordado a lo largo del proyecto, tal y como veremos más adelante.

2.2.3. MATLAB

MATLAB es un lenguaje de alto nivel diseñado para realizar cálculos técnicos. Integra el cálculo, la visualización y la programación en un entorno sencillo de utilizar donde los problemas y las soluciones se expresan con notación matemática. Se trata de un sistema basado en intérprete y orientado al desarrollo rápido de prototipos que no precisa de declaración o dimensionamiento previo de las variables, lo cual permite resolver muchos problemas matemáticos de forma cómoda, especialmente aquellos que involucren vectores y matrices.



Figura 2.5: Logotipo de MATLAB.

La principal ventaja de MATLAB es la disponibilidad de funciones de áreas específicas llamadas *toolboxes*. Se trata de grupos de funciones y comandos que pueden emplearse para resolver problemas de diferentes ámbitos como puede ser el procesamiento digital de señales, los sistemas de control o las redes neuronales.

2.3. Entorno de desarrollo

A la hora de desarrollar el sistema, se quiso optar por una solución portable, con el fin de poder emplear diferentes equipos. Por otro lado, se decidió emplear un entorno de escritorio basado en Linux, ya que existe mayor documentación de las bibliotecas que se iban a utilizar para esta plataforma. Por ello el desarrollo se llevó a cabo en una máquina virtual, la cual se ejecutaba con *VirtualBox* [Oracle, 2003], en la que se instaló el sistema operativo Ubuntu 12.04 LTS, el cual no se modificó durante todo el desarrollo, una vez fueron instaladas las bibliotecas y se comprobó su correcto funcionamiento. La ventaja que aportaba el uso de una máquina virtual es que era posible replicarla fácilmente y por tanto se obtenía un entorno de trabajo estable. La máquina virtual se configuró de manera que hiciera uso de dos procesadores y un máximo de 2048MB de memoria RAM.

Cabe destacar que al ser PCL una biblioteca aún en desarrollo, si es comparada con OpenCV, fue necesario hacer múltiples cambios ya que al seguir los pasos indicados en la guía de referencia de PCL, el funcionamiento de esta no era estable y se generaban errores de dependencias al ejecutar ciertas funciones básicas. Tras un análisis y búsqueda de información, se descubrió que la versión que se estaba usando no disponía de un fichero `pcl-geometry.pc` el cual era indispensable para ejecutar ciertos métodos. Por otro lado, también fue necesario realizar modificaciones en el fichero `flann.pc`, el cual también está incorporado en la biblioteca OpenCV, sin embargo, ambos son diferentes y PCL no es capaz de funcionar correctamente haciendo uso del que proporciona OpenCV, por ello se editó este fichero con el fin de asegurar que se use la versión de PCL.

A su vez se empleó un sistema de control de versiones para almacenar y poder trabajar desde las diferentes máquinas virtuales que se encontraban en diferentes equipos. Se decidió usar el sistema Git [Git, 2008], ya que ya había trabajado anteriormente con dicho sistema y el repositorio se almacenó en Bitbucket, similar a otros como Github, sin embargo, este permite tener repositorios privados en cuentas gratuitas y además permite grupos gratuitos de hasta 5 usuarios.

Capítulo 3

Planificación

3.1. Metodología

Para el desarrollo del proyecto desde el punto de vista de la ingeniería del software se decidió usar una metodología evolutiva en espiral tal y como se define en [Boehm, 1988]. Esta metodología es la que se consideró la más indicada para desarrollar el sistema ya que, en cada iteración, iba evolucionando para ir adquiriendo nuevas funcionalidades. Tras cada iteración, se analizaban los objetivos a realizar en la siguiente, en base a los resultados que se obtenían tras finalizar la anterior.

A su vez, conforme se iba avanzando cada iteración fue necesario ir revisando los objetivos o los métodos para llevarlos a cabo, lo cual implicaba una fase de búsqueda dentro de cada iteración. Por ello esta metodología se adaptaba a las necesidades del proyecto.

3.2. Temporización

Tal y como se ha indicado anteriormente, este proyecto se compone de diferentes partes. Para la realización del mismo se ha tenido que dedicar una cantidad de tiempo determinada a cada uno de los componentes que forman el proyecto.

En la figura 3.1 se puede apreciar el tiempo que se ha dedicado de manera global a cada una de las fases, destacando la etapa del desarrollo del sistema, la cual requirió el 51 % del tiempo total del proyecto. Cabe destacar que si bien representa una gran parte, tal y como se indica en la sección 5, durante el desarrollo fue necesario realizar búsquedas bibliográficas con el fin de ir

resolviendo los problemas que fueron surgiendo.

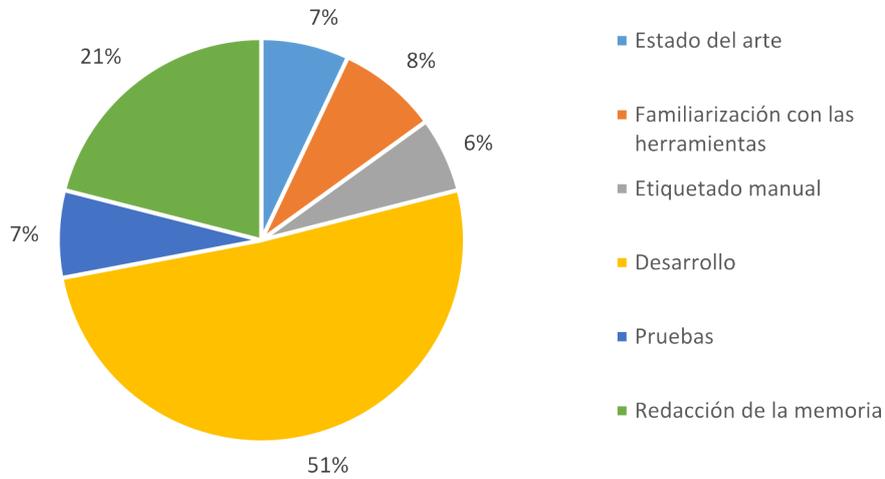


Figura 3.1: Diagrama representativo de la temporización del proyecto.

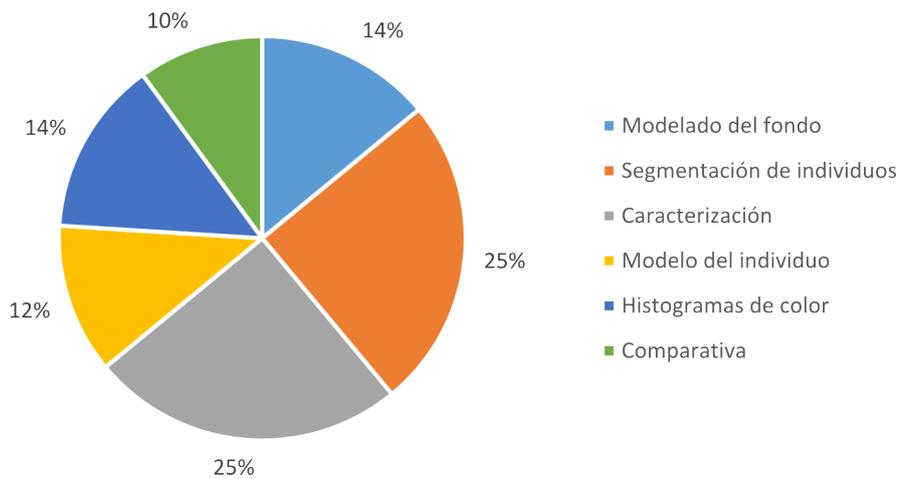


Figura 3.2: Diagrama representativo de la temporización de la fase de desarrollo.

A su vez, la fase del desarrollo puede dividirse en los diferentes módulos o fases que se diseñaron para el sistema. Si analizamos el tiempo dedicado a cada parte del desarrollo obtenemos el diagrama de la figura 3.2, en el que se aprecia que las partes en las que fue necesario invertir más tiempo fueron la segmentación de los individuos y la caracterización de los mismos.

3.3. Coste de la instalación

Para este proyecto no se han tenido que instalar los equipos ni el material que sería necesario para que el mismo funcione ya que para la etapa de la captura, la cual implica una mayor planificación y desembolso, se ha hecho uso de una base de datos de imágenes. Aún así, es de vital importancia conocer el coste, el cual vendrá dado por el número de sensores que se deseen instalar.

En primer lugar tenemos el sensor empleado, la cámara Kinect, la cual a día de hoy ya ha sido sustituida por el fabricante, Microsoft, por el sensor Kinect para Windows v2, el cual posee una mayor resolución y calidad de imagen. Este sensor puede conseguirse por 199,99 euros en la página del fabricante [Microsoft, 2014], si bien este precio puede variar dependiendo del vendedor. Al comienzo del desarrollo del proyecto, la cámara Kinect que se encontraba en el mercado tenía un coste aproximado de 120 euros.

A continuación hay que tener en cuenta que cada sensor debe estar conectado a un equipo. Un posible escenario sería que hubiera un ordenador conectado a cada uno, el cual iría almacenando las imágenes. Puesto que la tarea de almacenar imágenes no requiere de gran capacidad de cómputo se podría hacer uso de un equipo de bajo coste como la *Raspberry Pi* [Upton and Halfacree, 2012], la cual está bastante extendida y posee una comunidad bastante activa. Por tanto, se podría emplear la placa *Raspberry Pi 2 Model B 1GB* la cual tiene un precio aproximado de 40 euros. A este equipo sería necesario dotarlo de un espacio de almacenamiento suficiente, en base a la cantidad de minutos de grabación que se quiera almacenar en cada punto, teniendo en cuenta las posibles limitaciones debido a la LOPD (Ley Orgánica de Protección de Datos), por lo que sería necesario añadirle el coste de un medio de almacenamiento externo al equipo.

Finalmente sería necesario un equipo de escritorio en el que ejecutar el sistema, basándose en las imágenes obtenidas mediante los sensores. Es importante tener en cuenta que, además del coste de los equipos, puede ser necesario realizar una instalación eléctrica en los lugares en los que se quiera instalar cada sensor, y puede ocurrir que no se dispongan de tomas de corriente en zonas elevadas de la pared o en los techos.

Capítulo 4

Escenario experimental

Uno de los primeros aspectos a tener en cuenta antes de comenzar con el desarrollo es la obtención de todo lo necesario. En este proyecto, se depende de una fuente de imágenes que representan los datos de entrada del sistema. Por ello se decidió emplear la base de datos *DEPTHVISDOOR* [Castrillón Santana, 2014], realizada por los tutores de este proyecto, la cual se ajustaba a las necesidades del sistema. Esto facilitó las primeras fases del desarrollo puesto que ya se disponía de la base con la que poder comenzar.

Las imágenes de la base de datos se obtuvieron en el año 2013 durante varios días, en el Edificio de Ingenierías de la Universidad de las Palmas de Gran Canaria. Se aprovecharon las horas docentes para obtener imágenes de la entrada y la salida de clase de los alumnos y algún profesor. En los siguientes apartados se describen las características de las imágenes y las tareas que se tuvieron que realizar a continuación con el fin de obtener la información necesaria de cada persona para, en etapas posteriores, poder validar los resultados de la re-identificación.

Es importante destacar en este punto que la base de datos está formada por imágenes obtenidas a diferentes horas, por lo que las condiciones de luz son distintas en cada una de las capturas y además, durante la grabación se producen cambios intencionados en la iluminación de la estancia mediante el encendido y el apagado de las luces. Con esto se intenta representar la variación de las condiciones a las que normalmente se ven sometidas las cámaras.

4.1. Posicionamiento de la cámara

En términos generales, las cámaras de vigilancia o cámaras RGBD suelen colocarse en posiciones en las que queden frente a los usuarios a su misma altura, o a una altura elevada, de manera que siempre pueda verse el rostro de las personas que van pasando delante del sensor. Como ejemplos tenemos que la cámara Kinect, para su uso en consolas, se debe posicionar delante del jugador, mientras que las cámaras de vigilancia suelen estar en zonas altas, pero frente a una entrada para poder distinguir a las personas. Sin embargo, esta configuración puede derivar en oclusiones de algunos individuos, si detrás de una persona pasa otra, por lo que no se podría detectar o identificar correctamente. Para intentar solucionar este problema, las empresas especializadas en videovigilancia han desarrollado nuevas cámaras omnidireccionales para el techo que ofrecen una visibilidad de 360 grados.

Otro aspecto a tener en cuenta es la legislación referente a la privacidad de los datos, en este caso, las imágenes de una persona. Por ello, a la hora de escoger la posición de la cámara se ha tenido en cuenta este aspecto, con el fin de intentar preservar la privacidad de las personas evitando capturar el rostro de las mismas. A pesar de esto, el uso de sensores de profundidad nos proporciona más información sobre la persona, de manera que compensa la falta de datos del rostro.

Teniendo en cuenta lo mencionado, se decidió colocar la cámara en el techo, en posición cenital tal y como se muestra en la figura 4.1. Según [Lorenzo Navarro et al., 2013a], es necesario analizar el escenario con el fin de establecer la posición exacta del sensor. Puesto que el ángulo de visión vertical de la cámara Kinect es de 43° , según Microsoft, la distancia visible para la cámara cuando una persona se desplaza en una dirección depende de la posición del sensor. La distancia r puede ser calculada conforme a la siguiente expresión,

$$r = 2 \tan\left(\frac{43^\circ}{2}\right)(T - h)$$

Si tomamos la altura media de una persona como $h = 1,75m$, situando la Kinect a $T = 3m$ sobre el suelo, obtenemos $r = 0,98m$. Para una velocidad andando de $1,4m/s$, el sensor será capaz de capturar una media de 18 imágenes (si consideramos un ratio de 25 fps) de cada persona que pase por la zona, cantidad que puede considerarse suficiente para modelar a un individuo.

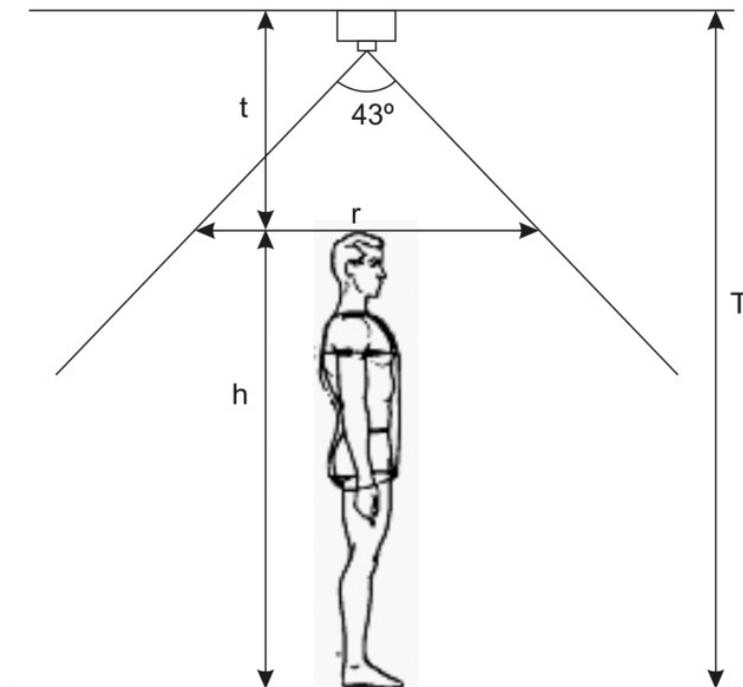


Figura 4.1: Geometría del posicionamiento de la cámara Kinect.
 [Lorenzo Navarro et al., 2013a]

4.2. Etiquetado

Una vez obtenidas las imágenes, tenemos el conjunto de datos de entrada necesario para hacer funcionar el sistema. Sin embargo, es necesario comprobar la fiabilidad de los resultados obtenidos. Por ello los autores de la base de datos tuvieron que realizar un etiquetado manual de las imágenes, para saber con certeza la identidad de cada individuo que aparecía en la grabación. Partiendo de lo expuesto en [Castrillón Santana, 2014], para cada individuo se especifican los siguientes elementos:

- Identidad
- Sexo
- Color de pelo y longitud
- Color de las prendas del torso y estilo

- Tipo de prenda de las piernas y color
- Presencia de mochila
- Presencia de bolso
- Presencia de algo en las manos
- Tipo de zapatos
- Indicación de entrada o salida de la habitación
- Primer y último fotograma del individuo
- Localización aproximada de la cabeza

La extracción de tanta información se debe a que pueda reutilizarse para otros propósitos, por lo que optaron por definir la mayor cantidad de información posible al etiquetar las imágenes. A su vez, la librería incluye un visor para mostrar las características mencionadas anteriormente junto con el fotograma correspondiente. En la figura 4.2 se muestra un ejemplo de lo que muestra el visor.



```
Command Window
female_sandaliasblancas_3 OUT 3/121
20130426-08-01-28-432 - 20130426-08-01-30-148
Coleta female
Hair
  Tail Straight Black
Torso
  Color Open_Jacket_or_similar Flat, 2 colores: dark blue
Legs
  Long, 1 colores: dark
Shoes
  Open, 1 colores: white
No_backpack No_backpack Something, 0 colores:
PRESS A MOUSE BUTION TO JUMP TO THE NEXT
PRESS ANY A KEY FOR THE NEXT TRAJECTORY FRAME
```

(a) Fotograma resaltando el individuo etiquetado.

(b) Datos definidos para el individuo.

Figura 4.2: Ejemplo del visor mostrando los datos etiquetados para una persona.

Una vez obtenidos los datos sobre la identidad, que es de la que se hará uso en este proyecto, se dispone de la información necesaria para, en etapas posteriores, comparar los resultados obtenidos por el sistema de re-identificación con la realidad, obteniendo así una comparación coherente con

la identidad real de la persona. Tal y como se analizará con más detalle en el apartado 5.8, será necesario realizar un segundo etiquetado de los datos que manejará el sistema para poder hacer la comparación.

4.3. Calibración

Desafortunadamente, las imágenes que se obtienen de la cámara Kinect no tienen el mismo tamaño. El sensor de profundidad proporciona datos sobre un área menor que el área de la imagen de color. Por ello, es necesario centrar las imágenes para que al seleccionar un píxel en ambas imágenes, éste se corresponda con la misma posición en ambas. Las imágenes con las que se ha trabajado para realizar este proyecto ya estaban calibradas puesto que quienes facilitaron las imágenes conocían esta característica de los datos obtenidos, por lo que no fue necesario calcular, y luego corregir la diferencia.

Puesto que las imágenes de profundidad son menores en tamaño, el área de detección se ve reducida, ya que es un requisito tener los datos de color y profundidad para poder realizar la re-identificación. Debido a esto, existe una pequeña zona en la que se desprecian los datos de color al no existir correspondencia en la imagen de profundidad.

En la sección 5.1, se tratará también la calibración de la cámara, ya que al tener datos de profundidad, no disponemos de posiciones absolutas en la escena que discurre ante la cámara. Por ello, en el proceso de conversión de los datos de color y profundidad a una nube de puntos que sí representa una escena 3D, será necesario aplicar una matriz de calibración que permita pasar a coordenadas en un espacio 3D, el valor de la profundidad junto con las coordenadas X e Y en la imagen 2D.

Capítulo 5

Desarrollo e implementación

En este capítulo se tratarán los aspectos del desarrollo y la implementación del sistema. Se ha decidido separar el capítulo en las tareas o fases que pueden considerarse independientes entre sí, lo cual permite entender en líneas generales los pasos que se han seguido para obtener un reconocedor funcional. A su vez, tal y como se detallará en el apartado 5.5.3, se expondrán algunas funcionalidades interesantes que fueron desarrolladas y operan correctamente pero que finalmente no se incluyeron en la versión final debido a que lo que aportaban a la mejora de la tasa de acierto no era significativo, teniendo en cuenta, principalmente, su complejidad computacional.

En la figura 5.1 se muestran las diferentes fases que forman el sistema. A lo largo de este capítulo se irá detallando el funcionamiento de cada uno de ellas según el orden en el que se ejecutan, el cual coincide con el de la figura 5.1.

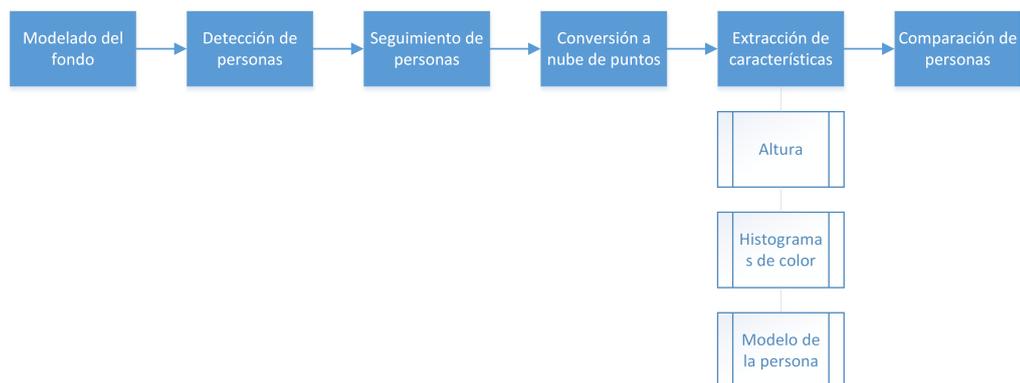


Figura 5.1: Fases que componen el sistema.

5.1. Generación de nube de puntos

A lo largo del desarrollo del sistema, la conversión de los datos a una nube de puntos se ha hecho en diferentes etapas, con el fin de hacer uso de la potencia de la biblioteca PCL a la hora de tratar los datos. Puesto que se nombrará en diferentes secciones, se explicará en primer lugar en qué consiste el proceso de generar una nube de puntos a partir de la imagen de profundidad y la de color, proporcionadas por el sensor.

La técnica aplicada se basa en lo expuesto en [ROS, 2014] y [PCL, 2014] para obtener la transformación que se ha de realizar para ajustar los datos de profundidad para cada punto X e Y a una escena en 3D. Esto se obtiene partiendo de la longitud focal del sensor y de su matriz de proyección, habiendo usado los valores que se proponen, los cuales dan buenos resultados. Tal y como se propone en [PCL, 2014], se debe calcular un nuevo valor de X e Y para cada punto, por lo que se desplaza cada punto de profundidad junto con los datos de color a las coordenadas transformadas con el fin de que se corresponda con la escena en 3D. Por lo tanto, para un punto P en RGBD obtenemos el equivalente P' en PCL de la siguiente manera:

$$\begin{aligned} P'_X &= (P_X - M(0, 2)) * P_D * FI_X \\ P'_Y &= (P_Y - M(1, 2)) * P_D * FI_Y \\ P'_Z &= P_D \end{aligned} \tag{5.1}$$

siendo P_X y P_Y las componentes X e Y del punto RGBD, M la matriz de calibración del sensor, P_D la profundidad en el punto RGBD y FI_X y FI_Y las longitudes focales inversas en cada componente del sensor.

El primer paso es instanciar una nueva nube de puntos que será del tamaño de la imagen de profundidad, por lo que en primer lugar se declaran sus dimensiones y se define su tamaño en base a ellas. A continuación se recorre la estructura tratando de asignar los valores correspondientes a las componentes de color RGB y las coordenadas XYZ, empleando la transformación descrita en 5.1. La ausencia de datos se señalará mediante la instrucción `std::numeric_limits<float>::quiet_NaN()`, la cual establecerá dicho punto con un valor nulo. Una vez realizada esta operación para todos los puntos, se volverán a reafirmar las dimensiones de la nube para poder acceder a sus datos como si fuera una matriz de dos dimensiones.

5.2. Detección de personas

Una de las primeras etapas a la hora de desarrollar el sistema es la detección de las personas. Esto se debe a que no siempre estarán pasando personas bajo el sensor, por lo que se pueden obviar todos los datos relacionados con imágenes en las que no aparecen personas. Por ello, en primer lugar, partiendo de las imágenes de profundidad y mediante el uso de la biblioteca OpenCV, se debe detectar e identificar el fondo de la escena, tal y como se explicará a continuación. Seguidamente, en el apartado 5.2.2 se detallará el proceso por el cual se logra separar a las personas, eliminando el fondo.

5.2.1. Identificación del fondo



Figura 5.2: Captura de la escena que representa el fondo que debe ser eliminado.

Tal y como se ha indicado, el primer paso para detectar las personas en el sistema es identificar el escenario que aparece en las capturas como el de la figura 5.2. Partiendo del hecho que la cámara no se mueve ni se inclina una vez iniciada la grabación, hay partes de la escena que son estáticas, como pueden ser el suelo, las paredes u otros objetos, que deben ser ignorados, ya que a priori conocemos que no son personas y por lo general no tienden a moverse. Esto puede conseguirse aplicando una sustracción de fondo a todas las imágenes, para lo cual es necesario establecer lo que se considera el fondo

o suelo.

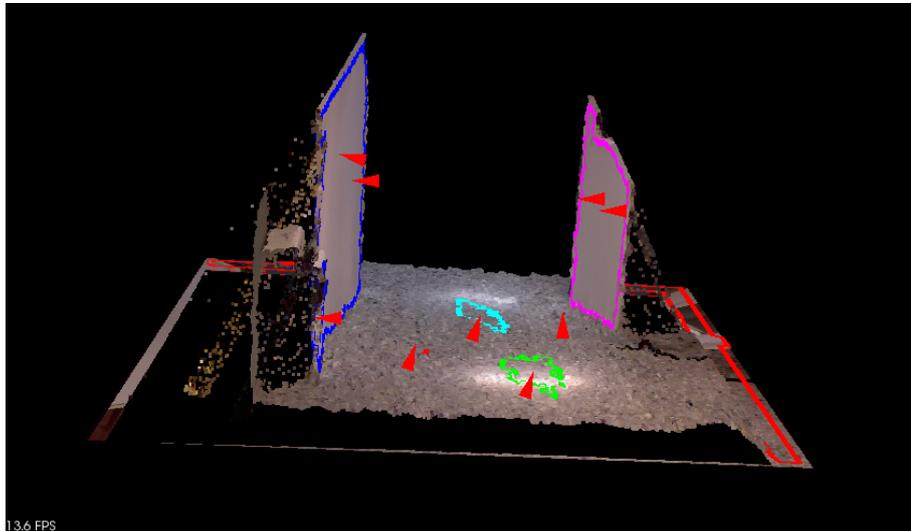


Figura 5.3: Resultados de la detección de planos empleando la biblioteca PCL.

Una primera aproximación para resolver este problema fue aprovechar la potencia de la biblioteca PCL, trabajando directamente con una nube de puntos obtenida del primer fotograma que se graba, en el que no debe haber nadie. Se pretendía unificar la detección de fondo con la eliminación, buscando los planos en la nube de puntos, es decir, las superficies que forman planos. El suelo es liso, por lo que en la escena 3D formará un plano, al igual que las posibles paredes, puertas u objetos inmóviles. Esto se hizo empleando los métodos disponibles en la biblioteca de búsqueda de planos a partir de puntos. La figura 5.3 muestra los resultados obtenidos. Como se observa, se han detectado correctamente los planos correspondientes a las puertas, señalados en color azul y malva. Sin embargo, la detección de planos no es capaz de reconocer el que forma la superficie del suelo, salvo algunas áreas. Esto se debe a la irregularidad del suelo, consecuencia del ruido que proviene del sensor, algo que no es posible solucionar, por lo que se optó por buscar otra vía.

Se decidió emplear el método de modelado del fondo indicado en [Lorenzo Navarro et al., 2013b] en el que el fondo se calcula a partir de los fotogramas iniciales. La nube de puntos del fondo \bar{p}_c , se calcula mediante el promedio de las k primeras nubes de puntos como:

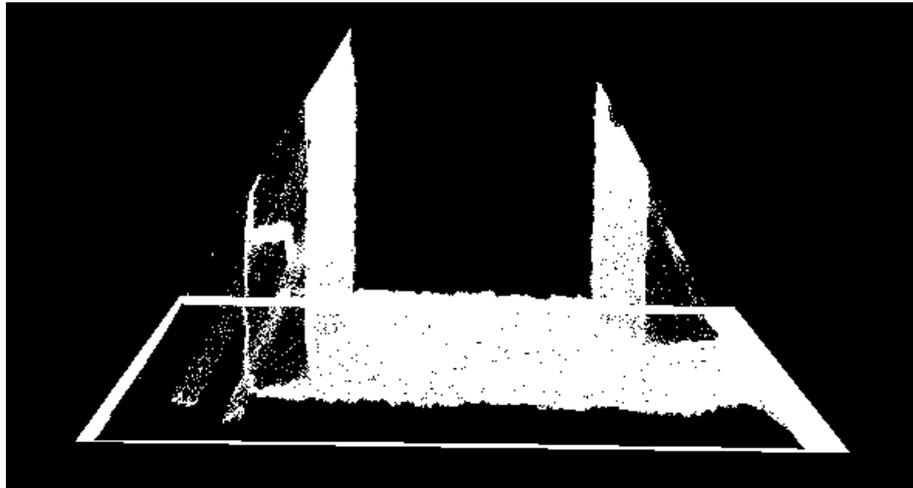


Figura 5.4: Nube de puntos con ruido que representa el modelo del fondo detectado.

$$\bar{pc}(i, j) = \frac{\sum_{l=1}^k pc^l(i, j)}{k}$$

donde $pc^l(i, j)$ son las coordenadas (x, y, z) asociadas al píxel (i, j) de las nubes de puntos empleadas para modelar el fondo. Por supuesto, durante el proceso de modelado del fondo no debe haber tránsito de personas bajo el área del sensor ya que, de lo contrario, el fondo se detectaría de forma errónea, debido a las irregularidades que se formarían (picos de profundidad en el suelo que debería ser liso). Finalmente tras este proceso se obtiene el modelo de la figura 5.4.

Durante el procesamiento de diferentes grupos de imágenes, capturadas en diferentes días, se puede apreciar el comportamiento del sistema cuando se produce un cambio inesperado en el fondo detectado. En el caso expuesto en la figura 5.5, se realizó de forma correcta la detección del fondo, en el que había una puerta cerrada. Sin embargo, alguien procede a abrir la puerta, con el fin de que pasen personas, lo cual desencadena un fallo en el sistema de detección, ya que considerará que la puerta es una persona. Este tipo de fallos pueden ser solucionados empleando el sistema de detección de cabezas del proyecto [Quesada Díaz, 2015].

Finalmente, tras el proceso se obtiene una matriz del tamaño de la imagen de profundidad que representa el fondo, de manera que en pasos posteriores podamos compararla con las imágenes en las que aparecen personas.



Figura 5.5: Fondo modificado al abrir una puerta en un momento posterior al modelado.

5.2.2. Sustracción del fondo

Tal y como se ha explicado anteriormente, debemos diferenciar en las imágenes de entrada la parte que se considera que es el suelo, de las personas. Para ello se realizará una comparación, basada en realizar una sustracción del fondo de la imagen de profundidad.

En primer lugar se intentó aprovechar los métodos existentes en la biblioteca PCL con el fin de eliminar los elementos estáticos. Para ello se usó la función `getPointCloudDifference`, la cual, para dos nubes de puntos devuelve otra que contiene sólo las diferencias entre las dos primeras. Sin embargo, al ejecutarlo no se producían cambios en la nube resultante, es decir, no se realizaba la eliminación del fondo y se observó que computacionalmente, el proceso era bastante costoso, ya que se empleaban árboles de búsqueda para realizar dicha operación. Tras examinar en detalle la biblioteca, se llegó a la conclusión que no era posible usar dicho método ya que, la nube de puntos generada a partir de una conversión desde la imagen de entrada no era compatible con la nube que se generó a partir de la imagen de fondo. Esto se debe a la organización interna de los datos de la nube, que se realiza en base a los datos que almacena, perdiendo la información de la posición X e Y de los puntos. Al tener cada punto una posición relativa a la nube, puesto que las dos nubes no provienen de las mismas fuentes, resultan datos de posición diferentes, de ahí que el método sea incapaz de encontrar puntos en común. A esto hay que sumar el hecho que los datos de profundidad son muy ruidosos y pueden no coincidir siempre en ambas nubes.

Tras el intento de emplear funciones de PCL para realizar la sustracción de fondo, se decidió emplear el método de sustracción de fondo propuesto en [Lorenzo Navarro et al., 2013b] el cual se basa en el concepto que los puntos más cercanos tendrán una componente z mayor. Por tanto, se considera que un píxel no forma parte del fondo si la diferencia de la distancia con el mismo píxel del fondo es mayor a un umbral. La imagen que representará el primer plano se calcula partiendo de un umbral de distancia aplicado al valor del modelo del fondo (x, y, z) , tomando como píxeles del primer plano aquellos cuya diferencia en la componente z de la nube actual con el modelo del fondo sea mayor que un umbral que está establecido en 10 centímetros.

Para un píxel, en una nube de puntos $pc(i, j)$, su correspondiente píxel en la imagen del primer plano, fg , se calcula de la siguiente manera:

$$fg(i, j) = \begin{cases} pc(i, j) & \text{si } pc(i, j) > bg(i, j) + \tau \\ 0 & \text{en otro caso} \end{cases}$$

De esta manera, tras finalizar este proceso obtendremos la nueva matriz con la parte de la escena de interés para realizar la posterior segmentación de los individuos, tal y como se explicará en la sección 5.3.

5.3. Segmentación de individuos

Una vez se ha eliminado la información irrelevante, en este caso, el fondo, nos queda sólo la parte que nos interesa, es decir, las personas u objetos que pasan por el campo de visión del sensor. Si bien era posible empezar con la segmentación directamente desde las imágenes obtenidas, se intentó aplicar una etapa de filtrado previa. Esto se debe a que el sensor introduce mucho ruido, por lo que se obtienen múltiples valores espúreos o puntos que no tenían información de color. Puesto que estos valores sólo podían detectarse al combinar la información de profundidad y la de color, al formar la nube de puntos se decidió posponer la aplicación de filtros a una etapa en la que se tuviera la nube de puntos. El proceso de filtrado que se siguió puede verse en detalle en la sección 5.1.

El siguiente paso en el proceso consiste en separar las diferentes personas que pueden aparecer en una misma captura. Para ello, se empezó por buscar métodos en la biblioteca PCL, y se encontró en la documentación un sistema de extracción de grupos que se implementó, partiendo de [PCL, 2015a]. El proceso resultante, denominado extracción de grupos euclídeo se resume en

Algoritmo 1 Extracción de grupos euclídeo

```
Crear un árbol KD que represente los datos de la nube de puntos de entrada
P
Instanciar una lista vacía de grupos C
Instanciar una cola de puntos que deben comprobarse Q
para todo punto  $P_i$  en P hacer
  Añadir  $P_i$  a la cola actual Q
  mientras Q no esté vacía hacer
    Buscar el conjunto  $P_{ik}$  de puntos vecinos a  $P_i$  en una esfera con radio
     $r < d_{th}$ 
    para todo cada vecino  $P_{ki}$  en  $P_{ki}$  hacer
      si  $P_{ki}$  no ha sido procesado entonces
        Añadir  $P_{ki}$  a Q
      fin si
    fin para
    si todos los puntos de Q se han procesado entonces
      Añadir Q a la lista de grupos C
      Asignar una lista vacía a Q
    fin si
  fin mientras
fin para
```

el algoritmo 1.

Tras las primeras pruebas, se observa que el resultado de la segmentación no es el esperado. Los puntos correspondientes a las cabezas forman grupos independientes, separados del resto del cuerpo. Un motivo de esto es que en la nube, la cabeza no está unida al resto del cuerpo debido a la oclusión que causa la cabeza sobre el área que se encuentra bajo esta. Para solucionar este problema se decidió analizar algo más en profundidad la función en la biblioteca PCL para buscar alguna posible solución puesto que la documentación de la función no entraba en detalles sobre el funcionamiento. Se observó que se empleaba RANSAC (Random Sample Consensus), un método iterativo para calcular los parámetros de un modelo matemático de un conjunto de datos observados. A la implementación de este algoritmo en la biblioteca PCL, se le indica mediante un parámetro la forma en la que buscar los vecinos de los puntos, lo cual limita la búsqueda. En este caso, se hacía uso de "PLANE MODEL", es decir, intentaba buscar vecinos que tendieran a formar planos y al existir una zona sin puntos entre la cabeza y el resto del cuerpo, el algoritmo no podía aproximar un plano que uniera las dos

zonas. Para solventar esto se buscaron otras opciones dentro del algoritmo y se encontró una manera de indicar diferentes modelos, entre los cuales se encontraba: CYLINDER. Se decidió escoger este modelo ya que, al estar la cámara sobre las personas, se puede considerar que el área en la que se encuentra una persona, desde este punto de vista, es un cilindro. Para poder hacer uso de este nuevo modelo en el algoritmo fue a su vez necesario ajustar diferentes parámetros, entre los que se encuentra la distancia máxima entre vecinos y el área mínima que deben formar la unión de diferentes puntos, ya que en una primera ejecución no se obtenían buenos resultados al realizar la segmentación.

Una vez realizadas distintas pruebas ajustando los diferentes parámetros se llega a la conclusión que todo depende del umbral de distancia entre puntos para considerarlo vecino. De tener un umbral muy bajo se formarían muchos grupos pequeños, sin embargo, al aumentarlo para intentar que junte el grupo de la cabeza con el resto del cuerpo ocurre que se forman grupos de puntos que pertenecen a diferentes personas. Por tanto se decidió que no era viable hacer uso de esta solución para segmentar las personas ya que no se adaptaba bien al problema y la función que realiza la operación no puede ajustarse para resolverlo. Por ello se optó por buscar otro método.

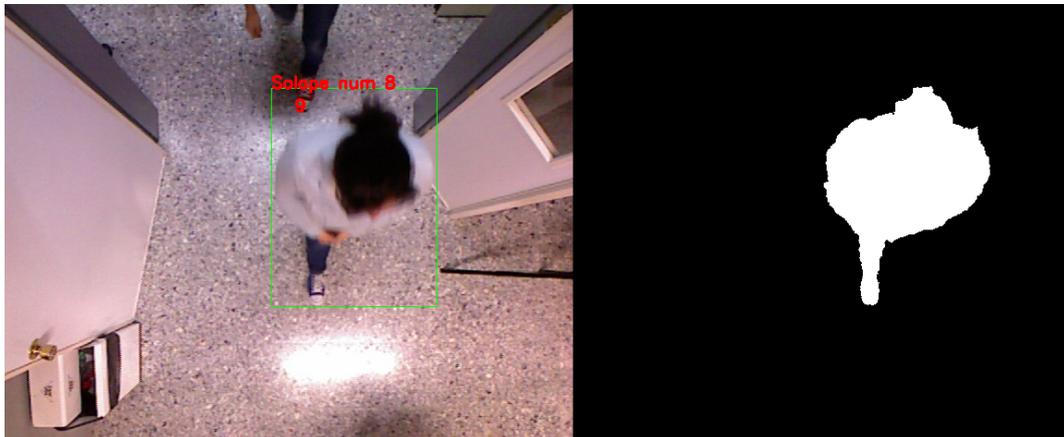


Figura 5.6: A la izquierda, imagen en la que se resalta una persona detectada. A la derecha, la máscara que representa el área de interés sin el fondo de la persona, tras realizar la operación de sustracción de fondo.

Tras intentar realizar una segmentación de individuos partiendo de métodos para datos tridimensionales, es decir, la nube de puntos, se

retrocedió en este caso y se decidió segmentar empleando datos en 2 dimensiones, en este caso, la matriz de profundidad sin el fondo, con la esperanza de que se simplifique el problema. Si analizamos la matriz de profundidad veremos que contiene valores de profundidad junto con valores con ruido o puntos en los que no existen datos de profundidad. Se decide pues, crear una máscara partiendo de la matriz de profundidad, en la que, en un principio, todos los valores de la misma están a cero. A continuación se compara la imagen de entrada con la imagen del fondo. Para cada punto en ambas imágenes, si la distancia al punto es al menos 5 centímetros más cercana en la imagen de entrada con respecto a la imagen del fondo, suponemos que no forma parte del fondo, y para ese píxel, en la nueva máscara se pone el valor 1, de manera que luego podremos aplicar dicha máscara a la imagen de entrada y obtendremos directamente los píxeles que nos interesan de la misma. Una vez obtenida la máscara que nos separa las partes de la imagen que nos interesa tratar, podemos hacer uso de la biblioteca OpenCV, la cual dispone de muchas herramientas para el tratamiento de imágenes. Como se observa en la figura 5.6, la cual representa la máscara, tenemos diferentes áreas que representan diferentes personas, por lo que podemos buscar los contornos que forman dichas áreas mediante la función `findContours`, lo que nos devolvería una lista de contornos, es decir, una lista en la que cada elemento representa el área en el que se encuentra una persona. A su vez, tenemos información sobre el área que ocupa, y podemos obtener el área de interés que indica el área que encierra todos los puntos en el que se encuentra dicha persona, obteniendo como resultado las áreas que se muestran en la figura 5.7. Es importante indicar que al emplear este método cualquier área será considerada como una única persona, esto implica que, si dos personas que están separadas, al pasar bajo el área del sensor llegan a juntarse, se obtendría un solo área, por lo que en este paso se detectaría como una sola persona. En la sección 5.4 se explicará en profundidad cómo se solventa en la mayor parte de los casos este problema.

Tal y como se ha comentado con anterioridad, las imágenes tienen ruido, el cual hay que tener en cuenta. A su vez, también puede ocurrir que una persona esté entrando en el radio de detección de la cámara y sólo se disponga de una muy pequeña parte de su cuerpo reflejada en la imagen. Debido a esto se debe aplicar un umbral de tamaño para evitar tratar estas áreas que son muy pequeñas y no aportan información útil alguna. Por tanto, cualquier región segmentada que tenga un tamaño menor que el 1% del área de la imagen será descartada. A su vez, para evitar posibles errores de segmentación que puedan ser arrastrados a etapas posteriores, cualquier



Figura 5.7: Ejemplo que resalta el área de interés en el que se encuentran individuos diferentes que pueden ser segmentados

área que ocupe más del 45 % de la imagen también será descartada. De esta manera, los posibles valores aislados con ruido dispersos en la imagen, de encontrarse fuera del área que representa una persona, no serán tenidos en cuenta.

En este punto, es cuando se genera, a partir de cada área segmentada, una nube de puntos. Para ello, primero se genera, junto con la máscara, la nube de puntos con la escena completa eliminando el fondo. El proceso de creación de una nube de puntos a partir de las imágenes de profundidad y color se ha explicado anteriormente en la sección 5.1. Para generar la nube con la persona segmentada se recorre el área de interés de los contornos que se encuentran dentro del rango de tamaño anterior, y para cada punto de la máscara hallada anteriormente con valor 1, copiaremos el punto equivalente a esa posición de la nube de puntos de la escena, el cual tiene toda la información de posición y color, a una nueva nube de puntos que representará al individuo en cuestión. Se repetirá esta operación para cada punto del área encerrada por el contorno, obteniéndose tantas nubes de individuos como contornos, tal y como se aprecia en la figura 5.8.

El siguiente paso será organizar estas nubes de puntos ya que cada individuo se considera único y deben ser separados. A continuación, en el siguiente fotograma se deberá realizar el mismo proceso pero se tendrán que

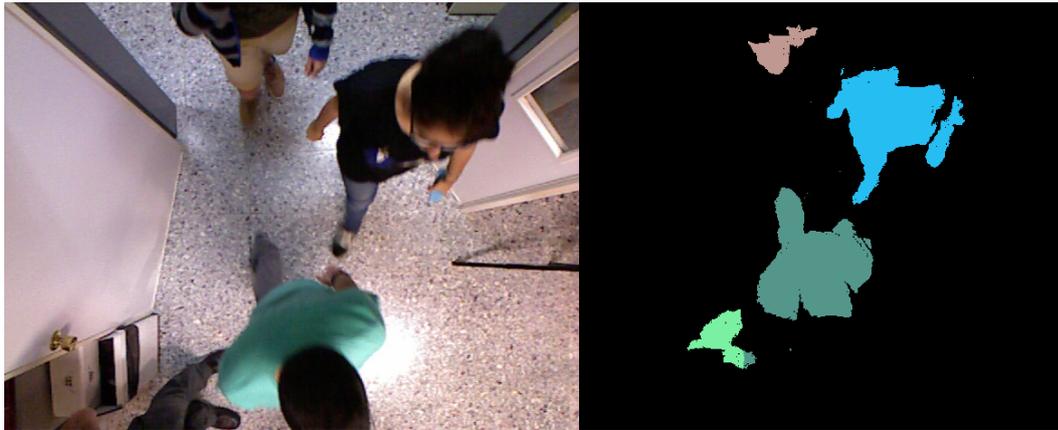


Figura 5.8: A la izquierda, imagen RGB capturada por la cámara. A la derecha, nube de puntos en la que cada color representa la figura de cada individuo diferente, tras finalizar el proceso de segmentación.

ir asignando las nuevas nubes de puntos al individuo que le correspondan, evitando confundir los diferentes individuos. En la siguiente sección se explicará como se solventó esta situación.

5.4. Seguimiento de personas

En el apartado anterior se explicó cómo se logra obtener, para cada imagen, la nube que representa a cada individuo. El siguiente paso se basa en ser capaz de seguir a cada individuo correctamente a lo largo de la trayectoria que describe cuando camina bajo la cámara. Es decir, cuando una persona pasa bajo el sensor, tendremos varias imágenes consecutivas en las que aparecerá dicho individuo, pero también pueden estar pasando más personas a la vez, por lo que, aparte de seguirlo es necesario buscar un método para diferenciar cada persona que va pasando antes de identificarla siquiera.

En primer lugar se tuvo que idear la manera en la que ir almacenando la información. Teniendo en cuenta que una nube parcial que represente a un individuo puede tener entre 30.000 y 60.000 puntos, se descartó la idea de mantener tanta información en memoria, ya que una grabación de 2.200 fotogramas de una cámara generaba aproximadamente 1,5 GBytes al generar las nubes de puntos. Se decidió por tanto que, independientemente de la solución algorítmica que se planteara, las nubes correspondientes a cada persona se almacenarían en disco, y otro módulo sería el encargado de realizar los siguientes pasos. Con este cambio se evitaría mantener tanta

información en memoria y permitiría dividir el sistema en diferentes módulos que realizarían distintas tareas, lo cual facilitaría en un futuro la ejecución distribuida de cada etapa del sistema. De esta manera se tendría un módulo encargado de la recopilación de imágenes, otro encargado de la sustracción del fondo y la segmentación, otro que extraería la características de cada individuo y finalmente otro encargado de la re-identificación.

Tras aclarar el almacenamiento de la información, se decide que lo más sencillo es que el sistema cree una carpeta por individuo que traza una trayectoria en un lugar del disco, por lo que al finalizar el proceso se obtendrán tantas carpetas como trayectorias bajo la cámara, de manera que luego, dentro de cada carpeta se encuentren todas las imágenes, en este caso nubes. Se puede considerar como las instantáneas tomadas al individuo mientras discurre por la zona que se está grabando. Cada carpeta estará numerada y además, todas las nubes de cada personas estarán numeradas en orden, con el fin de facilitar el acceso ordenado a las mismas. Estas nubes, se llamarán nubes parciales, esto es debido a que son partes de la nube entera que representa la escena y a su vez, aporta información parcial sobre la persona, puesto que una sola nube no es suficiente para identificar a un individuo.

Una vez definido el esquema que seguirán los datos en el disco, se planteó la siguiente estructura de datos mediante la cual almacenar la información que se genera en tiempo de ejecución sobre las personas que pasan bajo el sensor.

```
struct persona{  
    char path[128];  
    cv::Mat mask;  
    short ult_aparicion;  
    short num_blobs;  
    int area;  
    short num_persona;  
};
```

Por tanto, para cada sujeto que pase bajo la cámara almacenaremos la siguiente información:

- La ruta a la carpeta asignada para almacenar las nubes parciales.
- La máscara del área que ocupa la persona.
- Un contador para llevar un control sobre la última aparición.

- Un contador del número de nubes parciales.
- El tamaño del área que ocupa.
- Un número identificativo.

Nótese que con última aparición nos referimos a la última vez que aparecía el sujeto con respecto a la trayectoria que estaba siguiendo. Esto es necesario para ir eliminando de memoria las personas que ya han cruzado la zona y han salido del área de detección.

Para seguir la trayectoria de cada persona y asignar la nube parcial a la persona correcta, se parte de la máscara de profundidad creada en la fase de segmentación. Lo que se hará para determinar las siguientes nubes parciales de una persona que va avanzando es buscar intersecciones entre la máscara actual con la última máscara de todas las personas que se tienen en memoria, y se considera que aún están en la imagen. En el algoritmo 2 se describe en líneas generales el proceso para realizar el seguimiento de las personas en cada fotograma.

Algoritmo 2 Seguimiento de la trayectoria de una persona

```

si lista no está vacía entonces
  Verificar si hay personas que salieron
fin si
para todo personas segmentadas hacer
  si lista vacía entonces
    Crear nueva persona con datos actuales
  si no
    para todo personas de la lista hacer
      si área de intersección entre persona de la lista y la segmentada es
      mayor que el umbral entonces
        Actualizar máscara de la lista con la nueva y actualizar contadores
      fin si
    fin para
  si no se ha actualizado ninguna persona de la lista entonces
    Crear nueva persona
  fin si
fin para

```

Al finalizar este proceso se obtienen todas las nubes parciales de los sujetos organizadas en carpetas. Aparte de las nubes parciales, también se almacena

en cada carpeta un fichero de texto con las rutas a cada una de las nubes parciales de la persona y, a su vez, se almacena en un fichero de texto las rutas a cada uno de estos ficheros. De esta manera tenemos un fichero para poder acceder a los datos de cada persona, y de cada persona tenemos un fichero para acceder a sus nubes. Así, para la siguiente fase, la extracción de las características de cada individuo, tan solo tenemos que indicar la ruta del fichero que almacena las rutas al fichero donde se encuentra la información de todas las personas que se han segmentado.

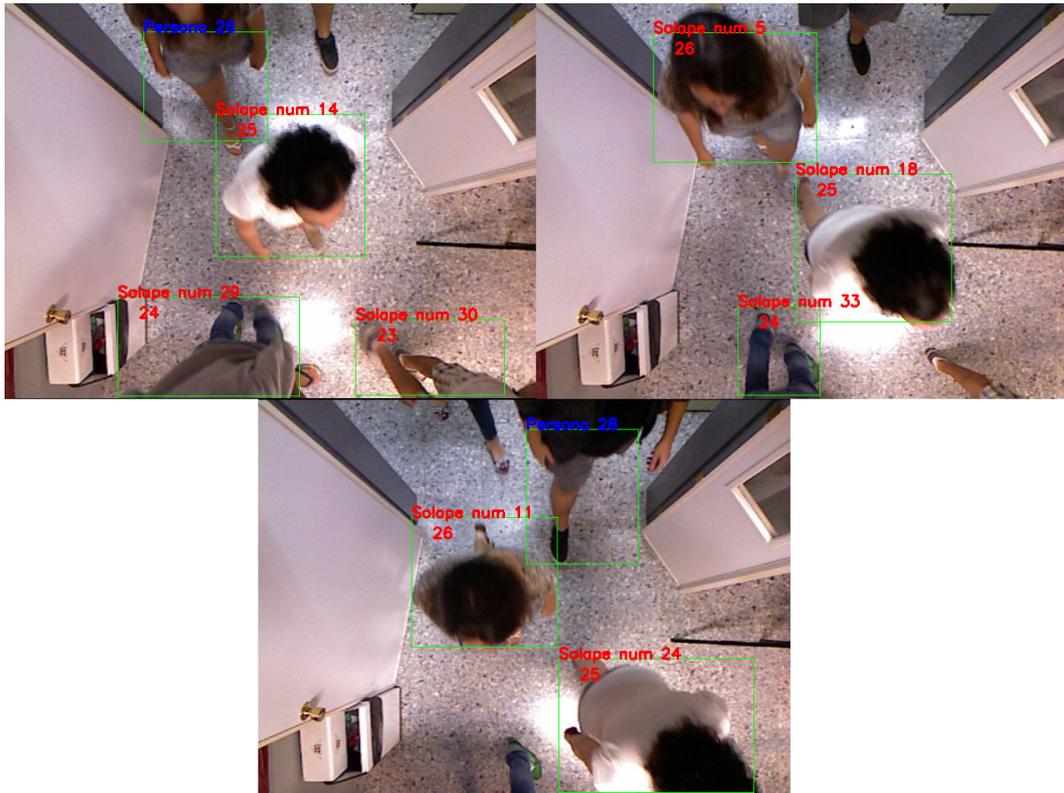


Figura 5.9: Ejemplo en el que se muestra el correcto seguimiento de las personas.

Durante las pruebas del módulo de seguimiento se detectó que, en ciertas ocasiones, podía ocurrir que se perdiera la trayectoria de la persona cuando se encuentra a mitad de su paso bajo el sensor, debido a que el área que forma la intersección no supera el umbral. Se averiguó que ocurría principalmente en momentos en los que discurrían muchas personas a la vez o si coincidía que la persona avanzaba más rápido de lo esperado en momentos en los que la tasa de imágenes por segundo caía, por lo que la distancia que caminó

la persona entre una imagen y otra es muy grande como para obtener un área de intersección. Por tanto, al no ser capaz de asignarle esa nube a ninguna persona existente, el sistema considera que es una nueva persona. En la figura 5.7, podemos observar cómo una persona es detectada como una nueva, al estar marcado el identificador de persona en azul, cuando existen imágenes previas de esa persona. En este caso, conocemos que una persona no aparece en la zona central espontáneamente, sino que debe pasar primero por los bordes de detección, por lo que llegamos a la conclusión que no es una persona nueva, sino una que se encuentra ya en la imagen. Habría pues que reconsiderar el método empleado para seguir las personas con el fin de contemplar estos casos, provocados principalmente por los datos de entrada, sin embargo se decidió continuar sin solucionarlo y plantearlo como una posible mejora ya que no comprometía el objetivo principal de re-identificar personas.

Como se ha indicado, con este paso obtenemos la información organizada y separada a partir de la que se tendrá que extraer las características. Esto se ha hecho así ya que el sistema debe ser capaz de funcionar como mínimo con dos fuentes de entrada, es decir, necesita de dos conjuntos de datos para poder comparar. Al separar el sistema en diferentes aplicaciones independientes se puede lograr tener un proceso en ejecución para cada fuente de datos, que además pueden ejecutarse en diferentes máquinas, para un procesamiento posterior en la misma u otra máquina. Esto es importante debido al elevado coste computacional que tienen por lo general los sistemas que manejan gran cantidad de datos, ya que hay que tener en cuenta que este sistema maneja fácilmente varios Gigabytes de información para grabaciones de pocos minutos.

A la hora de analizar el funcionamiento de la segmentación de personas se observó que, tal y como se aprecia en la figura 5.10, en múltiples ocasiones, las personas se juntaban para poder pasar por la puerta a la vez. Este caso no estaba contemplado, formándose un conjunto solapado conteniendo personas previamente segmentadas, lo cual derivaba en un fallo de detección, ya que además, al separarse, sólo se asignaban las imágenes a una de las personas y el resto se detectaban como nuevos individuos. Esto se denominó la fusión de varios sujetos durante el paso bajo el sensor y se consideró algo imperativo como mínimo detectar estos casos, ya que aportaría información errónea.

Con el fin de detectar los casos en los que dos o más personas se fusionaban, es decir, se juntaban tanto como para formar un solo blob en



Figura 5.10: Ejemplo en el que se muestra el momento en el que varias personas se juntan para poder pasar y se detectará como una fusión.

la imagen, se decidió primero observar las imágenes capturadas con el fin de analizar cómo ocurrían estos casos e intentar encontrar algún patrón. Tras observar detenidamente las imágenes, se llegó a la conclusión de que, al producirse una fusión, el área del blob crecía, y en ocasiones llegaba a ser mucho mayor que el doble que en la captura anterior. Por tanto, se tomó como medio de detección de la fusión de personas el área que ocupa cada persona en base a la máscara anterior. Es decir, al buscar intersecciones del blob actual con las personas que están en memoria, se compara el área ocupada por la persona en el fotograma anterior con el actual para, en el caso de haber aumentado al doble o más de su tamaño anterior, ser considerado como una fusión, y tratar este caso aparte. Con esto se logró detectar todas las fusiones, sin embargo, se producían falsos positivos en los extremos de las imágenes cuando, al entrar una persona sólo se encuentra en el rango de la cámara el pie o una pequeña parte de la pierna. En estos casos, en la siguiente imagen, al pasar a verse el resto del cuerpo, se produce un aumento del área que ocupa la persona, detectándose como una fusión. Con el fin de evitar estos casos que se producían con mucha frecuencia se decidió incorporar una condición adicional al considerar como fusiones sólo la unión de dos o más personas que

en una imagen anterior estaban bien definidas. Si bien esto podría implicar una limitación a la hora de detectar fusiones, hemos de tener en cuenta que en los extremos no se tendría información suficiente sobre las personas de forma individual como para garantizar una correcta detección y extracción de la información. A su vez, tal y como se comentará más adelante, las nubes de los extremos no contienen tanta información útil como el resto.

Una vez detectados los casos en los que se producen fusiones de personas, es necesario distinguirlas y seguirlas individualmente para que, una vez separadas, la detección y el seguimiento continúe normalmente, de manera que no se produzca un fallo de detección al separarse y las personas no se asignen correctamente a las mismas antes de producirse esta situación. Con el fin de separar las múltiples personas que pueden encontrarse en un blob grande se analizó la situación que producía este caso.

Antes de una fusión existen como mínimo dos personas que ya han sido detectadas y segmentadas. Por tanto, al producirse una fusión, el área debería poder solaparse con todas las personas que se han juntado del fotograma anterior. Esto llevó a implementar, antes de seguir a las personas en los casos de fusión, una detección de falsos positivos ya que, de detectar una fusión debido a un aumento súbito del área que ocupa una persona, si en dicha área sólo puede solaparse con una persona de las que aparecen en el fotograma anterior, implica que no se trata de una fusión, tal y como se ha definido, por lo que se tratará como un caso normal. Aprovechando esta comprobación, se decidió emplear este mismo método para seguir las personas en blobs fusionados. Sin embargo, al contrario que en el caso anterior, en vez de almacenar como máscara del fotograma anterior el blob completo fusionado, se almacena sólo el área que intersecta entre el blob fusionado y el fotograma anterior. Por un lado esto limita el seguimiento, ya que se dispondrá de un área menor para realizar el siguiente solape, pero es necesario para garantizar que no existen correspondencias con el área ocupada por otras personas, debido a no poder discernir en el área ocupada por las personas, que parte le corresponde a cada una, es decir, subdividir el blob en blobs individuales.

Si bien con el método expuesto se logró realizar el correcto seguimiento de las personas aún produciéndose fusiones, se llegó a la conclusión que de las nubes parciales en las que habían varias personas juntas no se podrían extraer características debido a no ser capaz el sistema de separarlas. Se realizó una búsqueda con el fin de encontrar métodos que pudieran ser añadidos al mismo y mejorar la segmentación en estos casos. Existe un proyecto de

fin de carrera, [Quesada Díaz, 2015], en el que se detectan las cabezas y de esa manera se podía asegurar el número de personas que había en una imagen, sin embargo, aún estaba en desarrollo durante la realización de este proyecto y requería cambiar múltiples partes de las fases anteriores, por lo que se descartó. Por otro lado, se encontró un artículo, [Zhang et al., 2012], en el que se planteaba un método para resolver segmentaciones basado en el planteamiento del algoritmo *Watershed* [Roerdink and Meijster, 2001] de manera no supervisada y de forma más eficiente, mediante la simulación de inundación del área y analizando las zonas en las que se acumularía agua. Se decidió, debido a que por lo general se obtiene información previa o posterior de las personas fusionadas, no implementar ningún tipo de segmentación para blobs fusionados, al valorar la complejidad de resolverlo en comparación a la mejora que se obtendría.

Finalmente, las nubes parciales resultantes de fusiones se almacenarán como si fuera un nuevo individuo, sabiendo que está compuesto por múltiples personas, ya que ocurren casos en los que se produce una identificación correcta de una de las personas que se encuentra en uno de estos conjuntos. Por otro lado, estas nubes que representan los sujetos fusionados no se asignarán a las personas que han sido segmentadas correctamente, con el fin de tener la información con el menor ruido o elementos que puedan interferir a la hora de extraer las características.

En el siguiente apartado se tratarán en detalle los aspectos que se han tenido en cuenta a la hora de determinar lo que caracterizará a una persona en el sistema, partiendo de las nubes parciales que han sido obtenidas, las cuales, se recuerda, se han volcado en disco mediante una estructura organizada en carpetas.

5.5. Caracterización de la identidad

En esta sección se explicarán las diferentes aproximaciones al problema de caracterizar la identidad de un individuo con la información obtenida en las fases anteriores. En este punto conviene recordar que se dispone de múltiples nubes parciales para cada persona que camina bajo el sensor, pudiendo pasar varias veces la misma persona y siendo detectada como personas diferentes cada vez ya que en las fases anteriores no existe ningún método de identificación, sólo se realiza un seguimiento de las trayectorias.

Tal y como se ha visto en la sección 1.2.2, existen múltiples maneras

para caracterizar la identidad de una persona, y por tanto, ser capaz de reconocerla. A día de hoy ya existen múltiples sistemas que emplean este tipo de información para realizar una identificación, tal y como se ha mostrado en la sección 1.3. Sin embargo, para este proyecto se decidió imponer una restricción importante con respecto a dichos sistemas, no se podrá captar la parte frontal un individuo, por lo que no se dispondrá de la imagen de la cara de las personas con el fin de preservar su intimidad y evitar, en la medida de lo posible almacenar información sensible.

A continuación se detallarán las diferentes características de una persona que se extrajeron de las nubes de puntos, si bien, tal y como se comentará más adelante, no todas se incluyeron en el sistema final.

5.5.1. Características físicas

Las personas por lo general se diferencian unas de otras por sus rasgos o características físicas. Como se ha indicado en el apartado anterior, no se dispone de la información de la cara de los individuos, por lo que se han de extraer otro tipo de características. Partiendo de lo expuesto en [Lorenzo Navarro et al., 2013a] se pueden considerar los datos como altura, el área, el volumen proyectado o la velocidad como características físicas de la persona que pueden ser de utilidad a la hora de distinguirla. Si bien se expondrán de forma independiente las características extraídas, muchas de ellas se extraen de forma conjunta al analizar las nubes parciales, siendo algunas medidas procesadas después para tener en cuenta todos los valores obtenidos y dar un resultado más fiable.

Altura

En primer lugar se intentó extraer la información sobre la altura de la persona ya que, en comparación con otras características, puede considerarse como la más básica y que debe incorporarse obligatoriamente. Además de caracterizar la identidad, es un indicador que permite descartar personas que tengan una diferencia de altura muy grande puesto que, por lo general, una persona no sufre cambios muy apreciables en la altura, si no tenemos en cuenta posibles cambios en el calzado o en el peinado.

Puesto que partimos, para cada persona, de múltiples nubes parciales, disponemos de muchas alturas diferentes para un mismo individuo que sabemos a priori que es el mismo. Esto se debe, como se aprecia en la figura 5.11, al rango de visión de la cámara ya que, en los extremos, que están

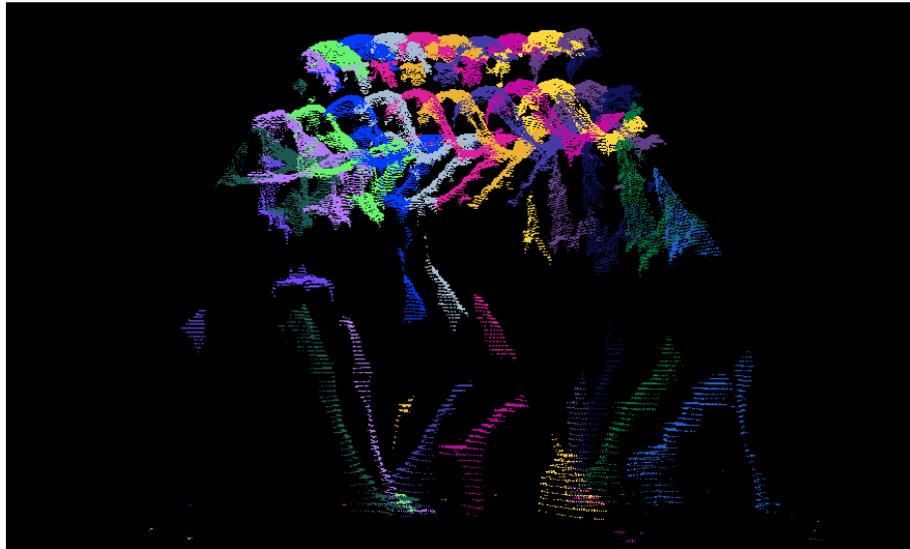
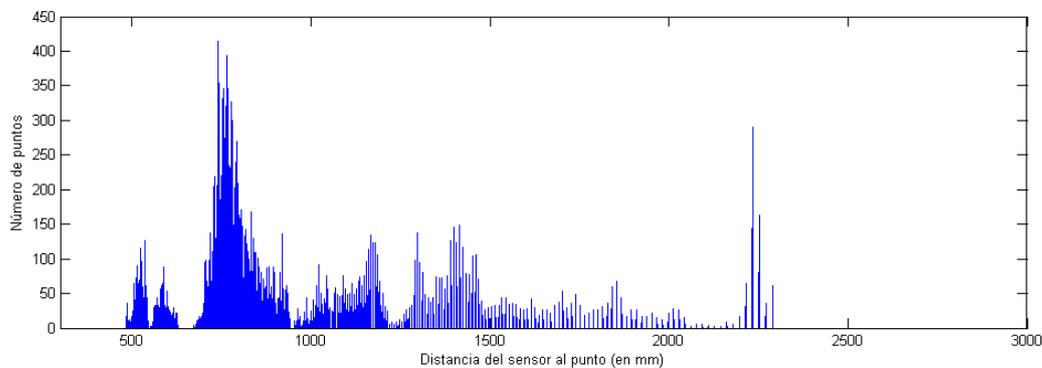


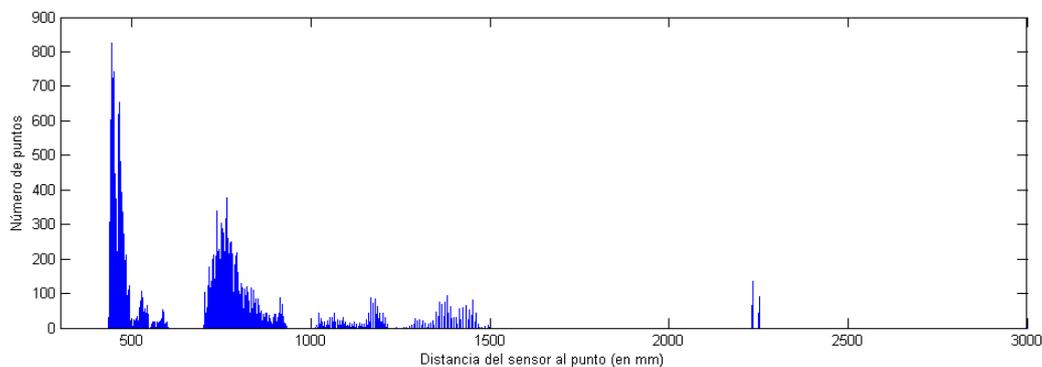
Figura 5.11: Todas las nubes parciales que representan a una persona, cada una en un color.

representadas como las primeras y las últimas nubes de la persona, entre otras, no suele aparecer la cabeza ya que al entrar y al salir la persona del rango de detección, sólo se aprecian sus pies y piernas. Por tanto, estas nubes no son representativas de la altura de la persona. Para resolver este problema se planteó ir recorriendo todas las nubes disponibles sobre la persona e ir obteniendo la altura a partir del punto más alto que corresponde a la persona, e ir guardando las tres medidas mayores con el fin de poder realizar un promedio. Tras realizar esta operación, los resultados obtenidos no eran los esperados, ya que se obtenían alturas mayores a dos metros y medio, por lo que eran erróneas al no corresponder esta altura a ninguna de las personas que figuran en las imágenes. Se decidió pues, analizar en profundidad los datos de la nube de puntos realizando un histograma de las alturas de cada punto para valores de altura comprendidos entre 300 y 3000 milímetros, obteniendo los resultados que pueden verse en la figura 5.12. A su vez, esto sirvió para analizar las zonas del cuerpo que eran más visibles para el sensor en base a la posición del individuo. Por ejemplo, para la figura 5.12(a) tenemos el histograma de altura cuando el individuo se encuentra cerca del extremo del rango de detección el cual nos indica que hay un elevado número de puntos para parte de la cabeza, el torso y las piernas, teniendo menos información sobre el pelo. Por otro lado, la figura 5.12(b), representando un individuo bajo el sensor, nos revela que hay mucha información sobre el pelo y los hombros, teniendo mucha menos información del torso y las piernas en comparación

con el caso anterior.



(a) Histograma de profundidad de persona cerca del extremo de detección del sensor.



(b) Histograma de profundidad de persona bajo el sensor.

Figura 5.12: Histogramas de la profundidad a la que se encuentran los puntos.

Analizando los resultados del histograma de alturas se observó que seguían apareciendo valores espúreos con alturas que no se corresponden con la altura de la persona. Para poder obtener un valor fiable de la altura se decidió realizar un histograma de alturas para cada nube parcial, lo cual permite obtener una representación de las diferentes alturas que aparecen en la nube. Se observó que los valores espúreos son valores aislados, y aunque a veces aparecen algunos con la misma altura, no tienden a haber muchos en una misma altura, o en alturas próximas. Por otro lado, se observó que para alturas muy cercanas a la altura real de la persona existía una gran acumulación de puntos, lo cual se corresponde con los puntos que representan la parte alta de la cabeza, que están aproximadamente a la misma

altura. Tras descubrir esta particularidad del problema, fue sencillo encontrar una solución al mismo. Se optó por recorrer los histogramas, desde alturas mayores a menores, buscando acumulaciones de puntos, por lo que, de llegar a una altura determinada, se examinan las 20 menores posiciones del vector, las cuales representan un rango de 2 centímetros, buscando acumulaciones de puntos. De encontrarse más de 20 puntos en dicho rango, se tomará esa como la altura de la persona en dicha nube parcial.

Finalmente, tras obtener para cada nube parcial la altura, nos quedamos con las tres alturas mayores para luego realizar el promedio. Este promedio será la altura que se considera que tiene la persona y se almacenará para, en pasos posteriores, recuperarla con el fin de realizar la re-identificación.

Área

Otra información que puede ser de interés es el área que ocupa la persona en la escena, ya que permitiría distinguir entre personas más o menos voluminosas. Con el fin de obtener esta información se plantearon diferentes métodos. En primera instancia se intentó contar el número de puntos de la persona para luego obtener el promedio, sin embargo, debido a que en cada imagen se puede disponer de una parte del cuerpo diferente a las que aparecen en otra imágenes, sumado al hecho de que las nubes en los extremos de la trayectoria sólo dan información de las piernas, no se consideró como un buen método para obtener el área.

Otra aproximación fue limitar la selección de puntos a la hora de considerarlos para calcular el área de solamente los puntos de la cabeza, por lo que se podría aproximar el área de esa zona. En este caso tenemos el mismo problema anterior con respecto a la acumulación de los puntos para obtener una media, salvo por el hecho de que la cabeza no sufre tantos problemas derivados de las detecciones en los extremos y las oclusiones como en brazos y piernas. Tras analizar el área de la cabeza obtenida en diferentes casos de personas iguales se apreciaban diferencias elevadas. Por ello, se descartó este método de obtención del área.

Finalmente, se optó por obtener un área menor, seleccionando la parte alta de la cabeza y calculando el área de dicha zona. A su vez, para ello se decidió no realizar un promedio, debido a las grandes diferencias que se encuentran entre distintas nubes, así que se obtuvo el área únicamente de la nube parcial de cada persona en la que la altura de la misma sea mayor. Para elegir qué puntos tomar en consideración se decidió contar aquellos que se

encontraban en el rango de altura comprendido entre la altura de la persona y dos centímetros menos. De esta manera se obtiene una muestra representativa del área superior de la cabeza. Al comparar las áreas obtenidas obtuvimos el mismo resultados que en los anteriores casos, existía mucha diferencia en el valor de las áreas para la misma persona, por tanto, se decidió no tener en cuenta el área para realizar la re-identificación.

5.5.2. Color

Una característica importante, que puede ayudar a diferenciar a una persona es el color. Cuando hablamos de color, en este caso nos referimos al color del pelo, la cara, el torso y las piernas de la persona. De esa manera obtenemos cierta información que a priori tiende a mantenerse constante a lo largo de varios días, como el color del pelo y el rostro, mientras que los colores asociados al torso y las piernas varían en función de la ropa que lleve la persona, por lo que suele variar en diferentes días, incluso en el mismo día, en el caso de llevar puesto un abrigo y quitárselo. Con el fin de analizar en primera instancia los valores de color que se obtenían de una persona se dividió en cuatro secciones y así aislar los colores de las zonas nombradas anteriormente, para hacer las diferentes secciones.

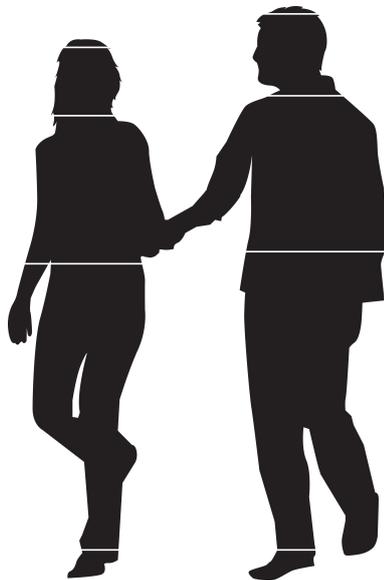


Figura 5.13: Ejemplo de las cuatro secciones a escala modificada a partir de [Todo Vector, 2014].

Se decidió establecer el tamaño de las cuatro secciones de manera que

fuera algo más representativo de la forma de la persona y se tuviera en cuenta la altura de la misma. Para ello, se decidió dividir la persona en siete partes en base a su altura a partir de las cuales se calcularán las secciones que se han definido. Esto proviene de las diferentes normas que establecen las proporciones ideales del cuerpo humano. Nos hemos basado en el canon antropométrico, concretamente en el estándar, tal y como se define en [Horcajada González, 2011], también llamado canon de Langer, en el que la altura de una persona se considera que es 7 veces y media la altura de la cabeza. Para la primera sección, la cual representa el pelo, se toman los puntos desde la altura de la persona hasta dos centímetros más abajo. Para la sección del resto la cabeza se examinan los puntos desde la anterior hasta un séptimo de la altura. El torso se tomará desde el punto anterior hasta tres séptimos de la altura, perteneciendo el resto a las piernas. Todo esto puede apreciarse en la figura 5.13.

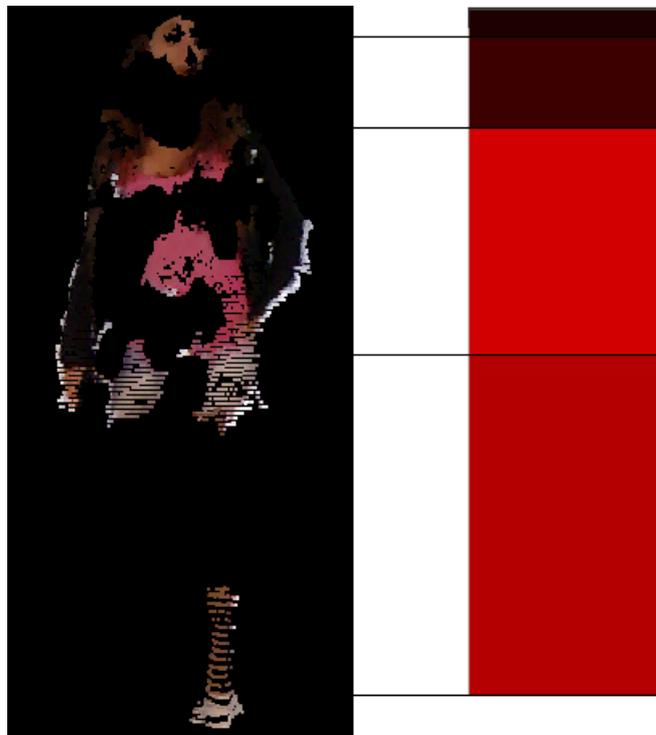


Figura 5.14: Ejemplo de los colores promedio de las cuatro secciones para una persona.

Con el fin de obtener los colores representativos de las cuatro zonas se obtuvo un rectángulo de cuatro colores, resultantes de aplicar un promedio sobre cada componente RGB de los colores obtenidos para cada zona. Se

observó que los colores del pelo tendían a ser muy parecidos para diferentes personas, ya que en muchos casos, los colores oscuros se codificaban con valores próximos al negro. Por otro lado, se observó que el ruido existente hacía variar mucho los valores y para la misma persona en diferentes imágenes se obtenían variaciones del color. Para paliar los efectos de las variaciones de color, muchas veces debido a la variación de la iluminación en diferentes zonas de la imagen se optó por simplificar los valores de color con el fin de buscar los colores predominantes en cada persona, obteniendo un resultado similar al mostrado en la figura 5.14, en la que además podemos comprobar que promediar los colores no proporciona un color representativo de los colores predominantes en la imagen.

Se implementó en primer lugar una pequeña función la cual, para un valor de una componente RGB, devolvía un nuevo valor correspondiente a un nuevo espacio de color en el que se reducía el rango de colores a la mitad del RGB. Fue así como se pudo observar rápidamente que se obtenía una mejora en cuanto al ruido y los colores se asemejaban más al color predominante de cada persona. Por ello, se siguió trabajando en la misma línea, teniendo en cuenta que era indispensable reducir el espacio de color. Por otro lado, si bien esto nos proporcionaba un color aproximado al de la persona, pudiendo entenderlo perfectamente un humano, un color promedio no es adecuado a la hora de comparar dos muestras de colores, teniendo en cuenta además que las muestras provienen de realizar una media de múltiples colores. Para este tipo de problemas se emplean histogramas, de esta manera, en vez de un solo valor para cada componente, obtenemos una curva que indica claramente los colores que predominan en la imagen.

Por tanto, se decidió calcular cuatro histogramas, uno para cada sección de la persona, de manera que, para cada punto y componentes se halla la posición que le corresponde en el histograma en base al número de bins, en este caso 128, es decir, se reduce a la mitad el rango del espacio de color, y se cuenta para cada posición el número de apariciones. Finalmente, obtenemos cuatro matrices de $[128 \times 3]$, para cada nube parcial. Puesto que se pretende analizar todas las nubes parciales, se realiza este proceso para todas ellas, guardando en una lista los histogramas para posteriormente almacenar toda esa información, lo cual se detallará en la sección 5.6.

Recordando la figura 5.11 en la que mostrábamos cómo las nubes parciales de los extremos tendían a proporcionar menos información sobre la parte alta de la persona, nos damos cuenta que es necesario tener en cuenta este hecho

a la hora de extraer el color de la persona. Asimismo, como se ha explicado anteriormente, la cabeza oculta el resto del cuerpo cuando la persona se encuentra más cerca del sensor, por lo que el color de la ropa no siempre puede obtenerse de esas nubes. Dicho esto cabe destacar que, si bien los extremos no proporcionan información sobre el torso y la cabeza, aportan información sobre las piernas, por lo que no deberían descartarse sin más.

Volviendo a la figura 5.11, se observa cómo, durante una gran parte del recorrido de la persona, la altura tiende a mantenerse estable, y que el recorrido es considerable. Durante la observación de múltiples nubes parciales se llegó a la conclusión de que, durante el recorrido, en las nubes parciales del principio y el final en las que se aprecia la figura completa de la persona se dispone de bastante información sobre el torso y las piernas, aparte de la cabeza. Por tanto se decidió no tener en cuenta las nubes en las que, la altura máxima fuera menor al 90 % de la altura de la persona en cuestión, datos que ya se conocen. De esta manera, nos aseguramos que tanto el histograma de la primera nube tras este proceso, como la última, tienen información fiable de todas las secciones.

5.5.3. Modelado

Aparte de los métodos mencionados anteriormente para obtener información que permita caracterizar a una persona, se propuso explorar la alternativa de, a partir de un modelo tridimensional completo de la persona, obtener con mayor precisión datos sobre el volumen de la misma, e incluso llegar a describir la forma del cuerpo. Para ello sería necesario obtener la representación de la persona con una malla en el entorno tridimensional para luego extraer información. Sin embargo, con el sensor sólo obtenemos la información vista desde la planta, por lo que una imagen no tiene la información suficiente como para representar la persona completamente.

Tal y como se ha explicado en otros apartados, para cada persona disponemos de una serie de nubes parciales en las que aparecen diferentes partes de la persona a lo largo de la trayectoria que describe bajo el sensor. Por lo tanto, tenemos información suficiente en las múltiples nubes parciales como para representar casi todo el cuerpo de una persona, dependiendo de la trayectoria trazada, su velocidad y si se ha juntado con otra persona durante su paso dentro de la zona de detección. Se propuso pues, intentar unir las múltiples nubes parciales, formando luego una única nube, compuesta por la unión de todas las obtenidas durante la trayectoria, en la que tendríamos

todos los puntos que forman la figura en tres dimensiones de la persona. Por desgracia, hemos de tener en cuenta que la posición de la persona varía a lo largo de las diferentes nubes parciales, por lo que no están alineadas las unas con las otras. A pesar de ello, sabemos que siempre, de una nube a otra, se ha producido un movimiento en una determinada dirección, y si bien en nubes diferentes se muestran partes diferentes, siempre se tienen puntos en común en nubes consecutivas. Por ello, podemos deducir que, si logramos calcular la transformación que han sufrido los puntos en común de la nube x_i con la nube x_{i+1} , conoceremos la transformación que se ha de aplicar a todos los puntos de la nube x_i para que se alinee con la nube x_{i+1} , incluyendo los puntos que no tengan en común y que aporten más información a la siguiente nube parcial.

En [PCL, 2015b], se propone el uso del algoritmo iterativo del punto más cercano para, de forma incremental, ir uniendo una serie de nube de puntos de dos en dos. El objetivo es transformar todas las nubes parciales con el fin de llevarlas a la posición de la primera. Para ello, se calcula la mejor transformación entre cada par de nubes consecutivas, acumulando cada transformación conforme se van procesando más nubes parciales. Se optó por añadirlo al sistema ya que, de esta manera, se podría conseguir en una sola nube toda la información posible sobre la persona. Sin embargo, para ello fue necesario obviar los datos de color, ya que los métodos proporcionados por la biblioteca no soportaban el tipo de punto que se empleaba, teniendo pues que convertir las nubes a otras sin los datos de color.

Las primeras pruebas se realizaron con pequeñas muestras, de cinco a ocho nubes parciales. Los resultados obtenidos fueron bastante satisfactorios ya que, en efecto, se lograba alinear con una precisión razonable cada par de nubes parciales. El inconveniente de este método era la lentitud con la que se ejecutaba, ya que se realizan múltiples iteraciones con el fin de obtener una mejor alineación. Si bien es cierto que es posible limitar el número de iteraciones, el proceso sigue suponiendo un elevado coste computacional para alinear solamente dos nubes parciales, por lo que el problema no sería abarcable en un tiempo razonable para todas las personas. Por otro lado, tenemos que tener en cuenta el ruido que tienen las imágenes, el cual provoca que el modelo de la persona no sea muy preciso y, a su vez, se acumulan los puntos aislados que pueda tener cada nube parcial. Finalmente, algo que no se tuvo en cuenta al comenzar a explorar esta opción fue que, durante el paso de las personas bajo la cámara, éstas van moviendo sus extremidades, es decir, los brazos y las piernas y el algoritmo empleado sólo puede aplicar una

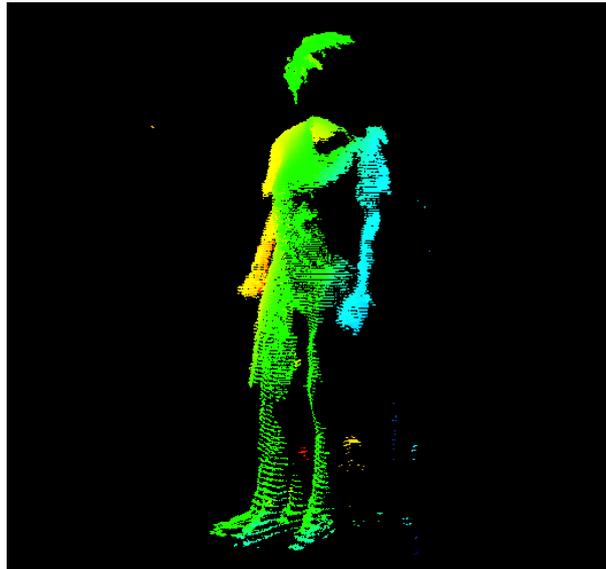


Figura 5.15: Ejemplo de alineación de 5 nubes parciales de la misma persona.

transformación a todo el conjunto de nubes, lo cual implica que los brazos y las piernas no se alinearán y en el modelo resultante, obteniéndose pues, un modelo en el que aparecen las piernas y los brazos en diferentes posiciones. Tal y como se muestra en la figura 5.15, se obtiene un resultado aceptable para el torso, sin embargo las piernas y el brazo derecho no están muy bien definidos.

A pesar de todo lo expuesto, se intentó aplicar filtros con el fin de reducir el número de puntos de la nube resultante, la cual se componía de la suma de todos los puntos de las nubes parciales que la componen, por lo que reducir su número era imperativo si se quiere obtener una simplificación e intentar disminuir el ruido. En primer lugar se examinó el filtro `PassThrough`, disponible en [PCL, 2011b], sin embargo tras analizarlo no servía para nuestros propósitos, por lo que tras seguir buscando en la biblioteca PCL se encontró el filtro `StatisticalOutlierRemoval`, tal y como se indica en [PCL, 2011e], el cual se emplea para eliminar puntos debidos al ruido de la captura de la imagen y se decidió aplicarlo. Una vez hecho esto, se intentó generar, a partir de los puntos, una malla que representara el modelo de la persona ya que de esta manera se simplificaría bastante el problema y sería más sencillo calcular el volumen ocupado de la malla. Para generar una malla a partir de una nube de puntos se usó lo explicado en [PCL, 2011a] y se añadió tras la generación de la nube resultante de la unión de las parciales.

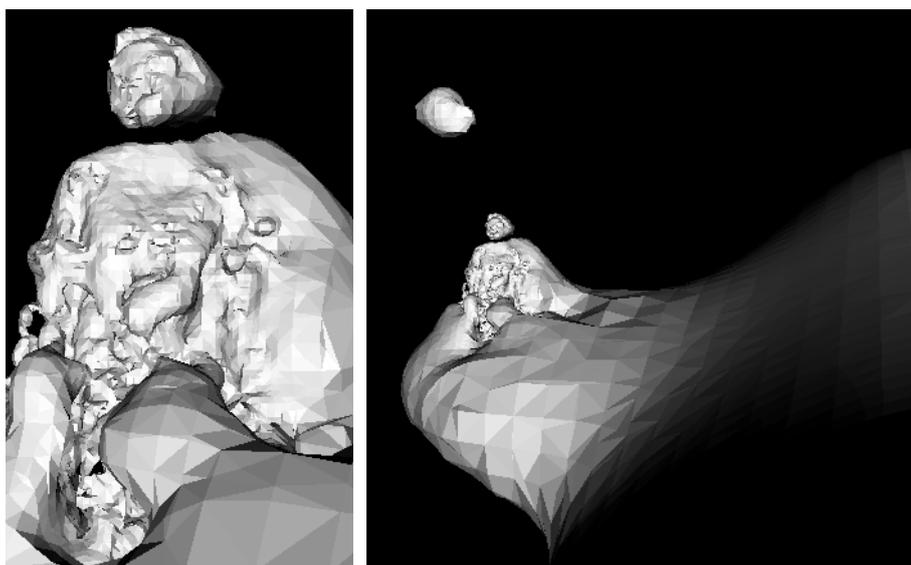


Figura 5.16: Resultado de generar el modelado de una persona a partir de múltiples nubes parciales.

Desgraciadamente, el resultado que se obtenía no era satisfactorio ya que, en un principio no se obtenía ninguna malla y luego, tras ajustar los parámetros de la función, la imagen obtenida no era una representación de la persona. Como se puede apreciar en la figura 5.16, se obtiene una figura similar a un busto, el cual está unido a una gran superficie, por lo que no se genera una malla con sólo la figura de la persona.

Debido a lo expuesto, si bien el generar un modelo de una persona sería muy positivo y aportaría bastante información útil, debido a los problemas que se encontraron se decidió no incorporarlo al sistema final, aunque se mantiene como un módulo independiente que podría funcionar correctamente de tener imágenes con menos ruido y si sólo tenemos en cuenta el torso de la persona, el cual tiende a moverse menos que las extremidades en las imágenes obtenidas.

5.6. Almacenamiento de las características

Anteriormente se han explicado las diferentes características que se intentaron extraer de las personas, quedándonos finalmente con los datos de la altura y el color de la persona, debido a lo poco útil o los problemas que daban otras características que se consideraron. Por tanto, se ha de extraer,

procesar y almacenar la información que describirá a una persona de alguna manera.

En primer lugar, tras obtener la altura y el área, se procede a realizar los histogramas de las nubes parciales cuya altura supera el 90% de la altura de la persona, lo cual llamaremos histogramas válidos. Recordemos que se realizan cuatro histogramas en base a las cuatro secciones definidas anteriormente. Cada histograma es almacenado en disco en un fichero de tipo XML, en el formato de ficheros de OpenCV, con el fin de simplificar la lectura y escritura de los mismos. En este fichero se almacena el área de la nube parcial, que se representa como el área que ocupa la zona que consideramos como el pelo, la altura máxima de la persona en dicha nube y los cuatro histogramas de manera separada. Estos archivos que contienen los histogramas se almacenarán de forma consecutiva en base al orden de las imágenes de entrada.

Una vez almacenados los histogramas parciales de la persona, se generará otro fichero de tipo XML. En él se almacenará el número de la persona asignado para dicha cámara, la altura, el área y el número de histogramas válidos que tiene la persona. De esta manera se conoce el número máximo de histogramas que existen, los cuales se encuentran almacenados en la misma ruta que las nubes parciales y el fichero con los datos de la persona. En este punto, una vez procesadas todas las personas detectadas, se tienen almacenadas todas las características como valores numéricos, reduciendo, de manera muy considerable el tamaño de la información asociada a cada persona si lo comparamos con la fase en la que se disponía de las nubes de puntos.

5.7. Múltiples cámaras

Hasta ahora no se ha tratado el tema del funcionamiento de la captura de datos desde múltiples fuentes ya que tan sólo se ha explicado individualmente el funcionamiento de cada parte. Si bien es cierto que no ha sido necesario aún, es imprescindible para poder realizar comparaciones, tal y como se detallará en la sección 5.8, el obtener datos desde diferentes cámaras, o la misma en diferentes momentos.

Tal y como se ha explicado anteriormente, la información que se obtiene es un volumen considerable de datos, por lo que se ha separado el sistema

en diferentes módulos o programas que pueden ejecutarse de manera independiente, siempre que se les proporcione la entrada de datos adecuada a cada uno de ellos, razón por la que se vuelca en disco toda la información en las etapas intermedias. Por tanto, cada ejecución independiente se considera una cámara diferente. Es decir, para cada cámara se ejecutarán los pasos expuestos en las secciones anteriores y la información resultante será la que se procese y se considere como una cámara. En la fase de comparación, explicada en la sección 5.8, será donde se carguen al mismo tiempo los datos de todas las cámaras en el mismo proceso, con el fin de buscar las similitudes entre las personas de las diferentes cámaras.

La ventaja que aporta plantear así el sistema es que las cámaras no tienen que estar conectadas al mismo tiempo ni en el mismo equipo. Sólo es necesario recoger los datos y se podrán analizar posteriormente en cualquier equipo. Esto también permite que el proceso de conversión de los datos y la extracción de las características, que son los procesos más costosos computacionalmente y los que ocupan más espacio en memoria, se hagan de manera distribuida en el equipo que controla cada cámara, para luego sólo tener que transferir la información necesaria para la re-identificación al equipo que se encargue de buscar coincidencias.

5.8. Búsqueda de correspondencias

Una vez se dispone de la información necesaria para cada persona y al menos dos fuentes diferentes de datos, que puede traducirse en dos cámaras o en una cámara en diferentes momentos, llega el momento de buscar coincidencias. Siguiendo el modelo empleado en el resto de fases del proyecto, se decide crear un módulo independiente que realice esta tarea, y que, al invocarlo, se le pase por parámetros la ruta de información de cada cámara, de manera que se instanciarán tantas cámaras o fuentes de datos como parámetros.

Antes de explicar el funcionamiento, debemos tener en cuenta la información de la que se dispone en este punto. Para cada persona se conoce su altura, su área y los histogramas. Recordemos que las nubes de puntos contenían mucha información, lo cual ocupaba demasiada memoria. A lo largo de las diferentes etapas esta información se ha ido resumiendo para, en este punto tener, para cada persona, si despreciamos el tamaño ocupado por la estructura de los datos, dos valores reales. Si suponemos que una persona puede tener hasta 25 histogramas obtenemos que el tamaño ocupado por

cada persona puede ser de un máximo de:

$$(3 * 128 * 32bits) * 4 * 25 + 64bits = 150Kbytes$$

, teniendo en cuenta que para cada histograma tenemos 3 canales con 128 bins de valores reales, para cada una de las 4 partes en las que se ha dividido a la persona. Se puede apreciar que la reducción de la información es considerable. Esto se tuvo en cuenta a la hora de plantear la carga de los datos en este módulo de comparación ya que, con este tamaño, nos podemos permitir cargar todas las personas de una cámara en memoria, siempre que se trate de un número razonable de personas ya que un equipo moderno tiene memoria suficiente para manejar esta cantidad de datos.

A continuación se planteó definir una estructura para cargar en memoria todos los datos que se encontraban en disco, y se empleó la estructura representada en la figura 5.17. Como puede observarse, se instanciarán tantas cámaras como sea necesario, que serán identificadas por un valor entero y, usando las listas proporcionadas por C++, se define la lista de personas que han sido detectadas por dicha cámara. A continuación tenemos que cada persona almacenará la ruta en la que se encuentran sus histogramas, un identificador de la persona añadido manualmente, con el fin de, posteriormente poder calcular la tasa de aciertos o fallos del sistema, el número de histogramas que se dispone para dicha persona, los datos de altura y el área, y finalmente la lista de los histogramas que estarán ordenados. Cada tipo `Histograma` contendrá la altura máxima de la persona para dicho histograma y los histogramas de cada sección. De esta manera, una vez cargados todos los datos desde el disco, se puede acceder a toda la información disponible para cada cámara, a través de la estructura planteada, ya que contendrá todo lo necesario para la comparación con otras personas. Esto nos permite tener los datos organizados en listas y en clases separadas, e ir liberando datos que dejen de ser útiles conforme se vaya ejecutando el programa, en caso de ser necesario.

En la fase de comparación, se definirán y cargarán los datos de todas las cámaras pasadas como argumentos al programa en base a la estructura que se acaba de explicar. En este punto hemos de aclarar que la comparación se realizará de par en par de cámaras, es decir, no se buscará simultáneamente en todas las cámaras. Se decidió hacerlo de esta manera para encuadrar este en el tipo de problema de re-identificación denominado *multi-versus-multi shot*. Por lo tanto, de cara a la ejecución, para cada persona de la cámara 1 del par de cámaras, tendremos que buscar entre un conjunto de personas de

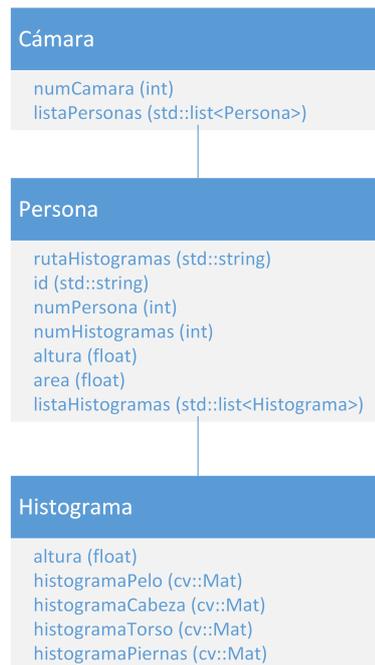


Figura 5.17: Estructuras de datos empleadas en el módulo de comparación.

la cámara 2. Esto se llama en el campo de la re-identificación la muestra y la galería respectivamente. Lo que se espera obtener es, para cada muestra una lista ordenada de personas de la galería en base a su similitud. Por ello, debemos condensar aún más la información para expresar la similitud de una persona en base a otra mediante un valor. Es decir, la salida que se obtendría sería una tabla en la que se tendrán tantas filas como muestras y tantas columnas como personas hay en la galería, en la que aparecerán los diferentes valores que miden la similitud o distancia entre cada persona. Sin embargo, este tipo de salida no es la adecuada para analizar la tasa de aciertos del sistema, por lo que se decidió generar una matriz binaria del mismo tamaño que la tabla mencionada. Para cada fila, tendremos ordenadas las personas de la misma manera que la tabla, sin embargo, en el caso de coincidir con la identidad, sea cual sea la posición, se pondrá como valor en ese punto un 1, y si no, un 0. Esto se puede hacer ya que durante las grabaciones se anotaron las identidades de las personas y se añadió dicha información con el fin de automatizar la generación de los resultados, los cuales se explicarán en la siguiente sección.

Tal y como se ha comentado en apartados anteriores, ocurren muchos

casos de fusiones, algunos incluso, a pesar de detectarse, no pueden ser manejados correctamente, por lo que se decidió mantener algunas personas fusionadas como un individuo único de cara al sistema. Esto es así, ya que, tal y como se analizará en detalle más adelante, es posible la re-identificación de un sujeto en un grupo de personas fusionadas. Se tuvo que analizar individualmente todas las personas separadas por el sistema para encontrar estos casos de fusiones y etiquetar correctamente las que aparecen para luego, al comprobar la identidad a la hora de generar la matriz binaria, permitir que el individuo que se busca se encuentre entre un conjunto de personas y se vea reflejado en dicha matriz.

Recordemos pues que nos basaremos en la altura y los histogramas disponibles de cada sujeto para obtener una medida de distancia que poder usar. En primer lugar, debemos resolver la comparación de histogramas de cada parte. Se decidió hacer uso de los histogramas de tres instantes de tiempo para cada persona. Desde el punto de vista de la lista, estos serían el primero, el central y el último. El primero representa el individuo entrando, por lo que se pueden apreciar los colores de la parte frontal, el central es aquel con altura máxima, es decir, el punto en el que pasa más cerca del sensor, para poder apreciar bien el pelo y la cabeza, y el último es en el que va saliendo del área de detección, por lo que se dispone de la información de color de la parte trasera de la persona. En todos estos histogramas hay suficiente información representativa ya que, recordemos que provienen de la fase anterior y está representado como mínimo el 90 % del cuerpo.

Para comparar los diferentes histogramas se hizo uso de la función `compareHist` de OpenCV, ya que los histogramas, a pesar de haber sido calculados manualmente en vez de con funciones de la biblioteca, son compatibles con los de OpenCV y además, se puede indicar a la función mediante un parámetro que emplee diferentes medidas de distancia para calcular su diferencia, siendo las que se encuentran disponibles: la correlación, Chi-Cuadrado, intersección y Bhattacharyya:

- Intersección [Zhang and Canosa, 2014]: Distancia en la que se comparan los elementos del mismo conjunto realizando la intersección de los dos histogramas a comparar.

$$intersec(H, H') = \sum_{i=1}^n \min(H_i, H'_i)$$

- Correlación [Marín Reyes, 2015]: Medida de distancia que compara los mismos elementos de dos conjuntos la cual refiere a una relación

estadística que implica dependencia.

$$\text{correlation}(H, H') = \frac{\sum_{i=1}^n (H_i - \bar{H})(H'_i - \bar{H}')}{\sqrt{\sum_{i=1}^n (H_i - \bar{H})^2 \sum_{i=1}^n (H'_i - \bar{H}')^2}}$$

- Chi-Cuadrado [Zhang and Canosa, 2014]: Distancia que compara los elementos del mismo conjunto de origen estadístico en la que se tiene en cuenta que la diferencia entre un conjunto pequeño puede ser más importante que la diferencia de otros conjuntos más grandes.

$$\chi^2(H, H') = \sum_{i=1}^n \frac{2(H_i - H'_i)^2}{H_i + H'_i}$$

- Bhattacharyya [Naik et al., 2009]: Medida de distancia que compara los mismos elementos de dos conjuntos, en la que se mide la similitud de dos distribuciones de probabilidad.

$$\text{Bhattacharyya}(H, H') = 1 - \sqrt{\sum_{i=1}^n \frac{\sqrt{H_i H'_i}}{\sqrt{\sum_{i=1}^n H_i \sum_{i=1}^n H'_i}}}$$

Tras hacer pruebas con los diferentes métodos de distancia, se observó que el método de Bhattacharyya daba los mejores resultados. Además, según se recoge en [Marín Reyes, 2015], de los métodos de los que disponemos se trata de uno de los recomendados en términos de re-identificación en el caso de histogramas. Cabe destacar que se hacen comparaciones de cada parte del cuerpo de forma individual, es decir, teniendo en cuenta que se extraen datos de tres instantes, se comparan con uno a uno con los de dos personas, cada histograma de cada parte del cuerpo con el correspondiente de la otra persona. Sin embargo, pueden ocurrir casos en los que no existan puntos suficientes para una sección y que la comparación de histogramas indique que los colores son independientes, algo que ocurre principalmente en los extremos. Esto se denominan valores perdidos, ya que no se disponía de datos suficientes, algo que suele ocurrir en análisis estadísticos. Como se indica en [Navarro Pastor, 1998], esto no debe ser un problema para continuar con el análisis ya que se suele asignar un valor a cada dato perdido y continuar con el análisis. Existen muchos métodos diferentes para imputar los valores e incluso se propone el uso de redes neuronales para dicho fin. Esto suponía desviarse del objetivo principal del proyecto, por lo que se planteó inferir el valor a partir de la comparación de otro de los histogramas con la misma sección, algo que mejoró la tasa de aciertos, por lo que se solucionó el problema que

provocaban los valores perdidos.

Una vez obtenidos los resultados de la comparación de histogramas se puede continuar con el proceso de comparación en base al resto de datos. Tal y como se ha comentado anteriormente, tenemos la altura y el área, despreciando este último dato debido a que, tras las pruebas realizadas, no es representativo de la persona. Por tanto, sólo se tendrá en cuenta la altura, aparte de los histogramas de color. En primer lugar, se intentó podar los resultados ya que a priori sabemos que, por lo general, si la altura entre dos personas difiere mucho, no son la misma persona, por lo que se estableció que si la diferencia de altura entre la muestra y el candidato de la galería era mayor de dos centímetros, se descartaría el candidato y en caso contrario se establecería un valor de distancia en base a un cálculo ponderado entre la diferencia de altura y el color. Sin embargo este tipo de resultados no era el deseado ya que esto no permitía obtener los resultados conforme al estándar en este tipo de problemas, en los que todo depende del valor de la diferencia y no se descarta ningún candidato. Por ello se decidió asignar pesos a todos los datos comparativos para obtener un único valor de diferencia, asignando un mayor peso a la altura que a los colores, sin embargo, se obtenía una tasa de acierto menor al 30 % debido a que, aún a pesar de tener alturas diferentes, tenían colores muy similares y tenía valores de diferencia muy similares al compararlo con la persona que en efecto era la misma. Analizando las medidas de diferencia para esta aproximación se constató que no era la adecuada.

Se decidió entonces tener en cuenta lo que se sabe a priori del problema, y es que una persona no suele cambiar espontáneamente de altura, por lo que se debe priorizar en cierto modo las personas que tienen alturas más cercanas, ya que existen más probabilidades de que sea un buen candidato. Para intentar simular esto, se decidió establecer una asignación de pesos y cálculo de distancia variable, en base a la altura, simplificándolo a dos tipos de cálculos y pesos, uno para candidatos con alturas muy cercanas, y otro para el resto. El primer caso, para un candidato de altura cercana tenemos que la distancia $Dist$ se calcula de la siguiente manera:

$$Dist_1 = (Dist_{ITorso} + Dist_{IPiernas} + Dist_{FTorso} + Dist_{FPiernas} + Dist_{CTorso}) \\ * 0,14 + Dist_{CPelo} * 0,08 + Dist_{CCabeza} * 0,2$$

en la que $Dist_{ITorso}$ representa la distancia entre los histogramas del inicio del torso, $Dist_{IPiernas}$ la distancia entre los de las piernas en el inicio, $Dist_{FTorso}$ la distancia entre los histogramas finales del torso, $Dist_{FPiernas}$ la distancia

entre los últimos de las piernas, $Dist_{CTorso}$ la distancia entre los histogramas del punto central de mayor altura del torso, $Dist_{CPelo}$, la distancia entre los del pelo del punto central y $Dist_{CCabeza}$, la distancia entre los histogramas de la sección de la cabeza del punto central. Como se aprecia mediante los pesos, se obtiene un resultado entre 0 y 1, indicando mediante 0 que se trata de la misma persona y 1 que las personas son totalmente diferentes.

Por otro lado, si el candidato tiene una diferencia en altura mayor a un umbral, aplicaríamos otro criterio para calcular la distancia, en el que sí tenemos en cuenta la diferencia de altura. Dif_{Altura} , en el cálculo. En este caso la distancia sería la siguiente:

$$Dist_2 = (Dist_{ITorso} + Dist_{IPiernas} + Dist_{FTorso} + Dist_{FPiernas} + Dist_{CTorso}) * 0,1 + Dist_{CPelo} * 0,05 + Dist_{CCabeza} * 0,15 + Dif_{Altura} * 0,3$$

De esta manera, a pesar de existir una diferencia en la altura, se tiene en cuenta a la hora de hallar la distancia al asignarle un peso. Este método de cálculo se empleó tras probar múltiples aproximaciones y analizando los valores de distancia que se obtenían, tanto el valor final como la similitud entre cada histograma. Fue de esa manera por la que se asignaron los pesos que figuran para cada parte. Se debe a que se observó que existía un patrón y que los colores obtenidos en ciertas zonas eran más fiables para la re-identificación que otros, como por ejemplo el color de la cabeza. Sin embargo, algo que sorprendió fue que el color del pelo no arrojara muy buenos resultados ya que, no existía tanta variedad en las muestras y muchos individuos tenían colores similares, los cuales, al pasar por el proceso de reducción del color, se volvían muy similares.

Tal y como se ha indicado, el método de comparación empleado no se escogió de manera arbitraria, sino que se hicieron múltiples pruebas en las que se fueron descubriendo las zonas que mejores valores de distancia daban para la misma persona, es decir, se basó en el método de prueba y error para ir mejorando el proceso hasta conseguir un método que diera buenos resultados en las muestras que se disponían. A continuación se detallarán los resultados que se obtuvieron.

Capítulo 6

Resultados de las pruebas

En este capítulo se detallarán los diferentes resultados que se obtuvieron una vez se completaron todas las fases del sistema y se probaron todas en conjunto. Esto llevó a utilizar diferentes espacios de color a la hora de realizar los histogramas, y a los cambios mencionados en la etapa de búsqueda de correspondencias sobre los diferentes métodos para calcular las distancias. Para mostrar los resultados obtenidos, representados mediante la tasa de aciertos del sistema, se realizó una pequeña función en MATLAB con el fin de generar unos resultados más legibles, empleando gráficas. Para la realización de las pruebas se analizaron imágenes en las que se recogen 121 trayectorias de personas bajo el sensor, produciéndose el paso de las mismas personas en diferentes ocasiones. Las imágenes se encuentran divididas en dos conjuntos, tomando uno como muestra, representando una grabación de tres minutos de duración y otro como galería de cuatro minutos y medio de duración.

6.1. Espacio de color RGB normalizado

En primer lugar se optó por cambiar el espacio de color RGB al RGB normalizado ya que, en este espacio de color, la luminancia no se encuentra como una componente independiente. Con el fin de minimizar variaciones de color debidas a cambios de la iluminación, se decidió realizar esta conversión, la cual se calcula de la siguiente manera:

$$r = \frac{R}{R + G + B} \quad g = \frac{G}{R + G + B} \quad b = \frac{B}{R + G + B}$$

La primera prueba que se ejecutó fue comparar todas las personas que se encontraban en el grupo de las muestras con todos los del grupo de la galería,

obteniendo el resultado que se muestra en la figura 6.1. En estas primeras pruebas se hizo uso de la primera aproximación para el cálculo de la distancia entre personas mencionada en la sección 5.8.

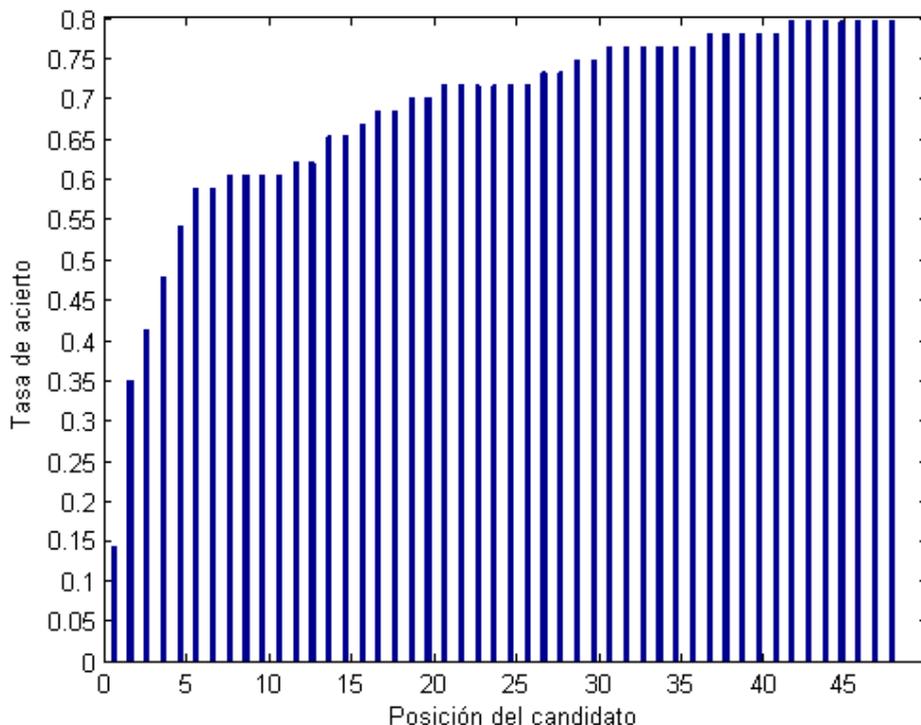


Figura 6.1: Primera prueba comparando el grupo de muestras con el grupo de la galería

El resultado se muestra mediante una gráfica CMC (Cumulative Matching Curve) y se interpreta de la siguiente manera: el eje de ordenadas representa la tasa de acierto y el eje de abscisas representa el orden de los candidatos de la galería para cada muestra. De esta manera tenemos la tasa de acierto para los n candidatos de la galería acumuladas. Por lo que en la posición 1 tenemos la tasa de acierto cuando la muestra corresponde con el primer candidato, en la posición 2 tenemos la tasa de acierto cuando la muestra se corresponde con el primer o segundo candidato, y así para cada posición del eje de abscisas.

Como puede apreciarse en la figura 6.1, el resultado obtenido no es nada satisfactorio ya que se consiguen unas tasas de acierto más bajas de lo esperado. Se intentó, por tanto, averiguar el motivo por el cual se obtenían

estos datos, con el fin de obtener una mejora en los resultados.

Analizando las imágenes se averiguó que muchas personas que se encontraban en el grupo de muestras no se encontraban en la galería, por lo que, tal y como se aprecia en la figura 6.1, la tasa de aciertos no se aproxima al 100%. A su vez, esto impide dar validez al resultado obtenido en la anterior ejecución. Por ello se decidió crear un subconjunto de las personas pertenecientes a las muestras que a su vez estén en la galería. De esta manera no se falsearán las tasas de acierto debido a que personas que no se encuentran en la galería no se tendrán en cuenta. En la figura 6.2 se muestran los resultados para la prueba con las condiciones mencionadas.

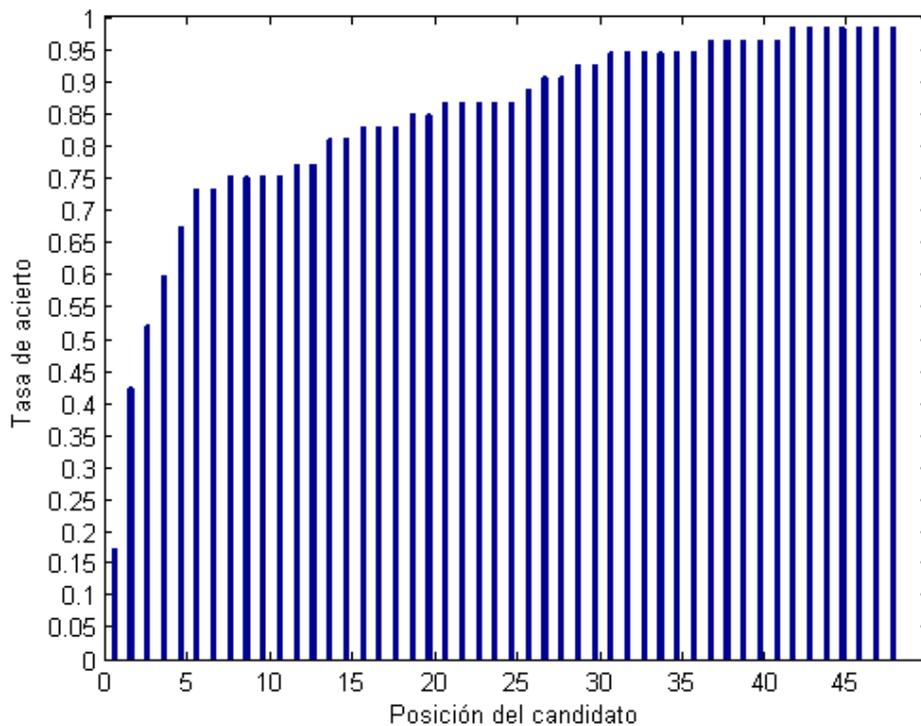


Figura 6.2: Resultados tras la eliminación de las muestras las personas que no se encuentran en la galería

Para esta prueba podemos observar cómo la tasa de acierto alcanza la unidad una vez tenida en cuenta la totalidad de la galería. Sin embargo, la tasa de acierto sólo ha mejorado muy ligeramente y aún dista mucho de lo que se quiere conseguir con el proyecto. En este punto cabe destacar que, durante la realización de esta prueba, se observó que la tasa de acierto superaba

la unidad, debido a que en ciertas ocasiones, una persona de las muestras se encontraba varias veces en la galería, es decir, que pasó en múltiples ocasiones bajo el sensor. Por ello, se decidió tan sólo tener en cuenta la primera concordancia en la lista de candidatos para el cálculo de la tasa de acierto.

A continuación se decidió analizar con más detalle las imágenes de las personas de las muestras. Como se ha mencionado en apartados anteriores, no todos los casos de fusión de personas se lograron detectar correctamente, por lo que algunas personas definidas en el sistema son fruto de la fusión de varias. Por ello se decidió eliminar de las muestras, dichos individuos, para que se pueda caracterizar correctamente la persona que se buscará en la galería. En este último conjunto sí se mantienen las personas formadas por la fusión de varias de ellas para averiguar si la detección de un individuo en un grupo es satisfactoria.

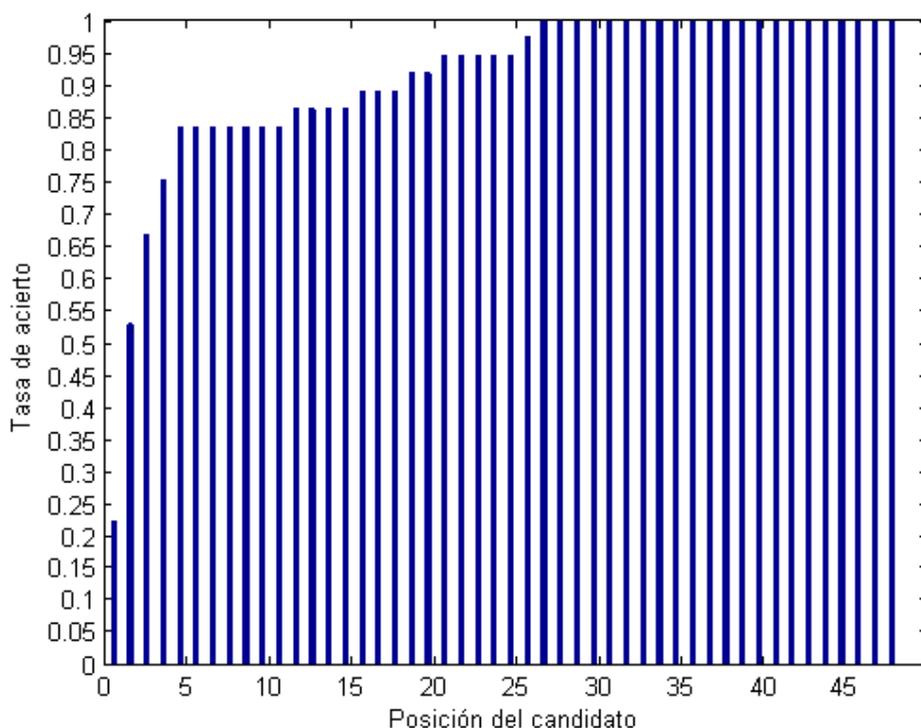


Figura 6.3: Resultados tras la eliminación de las muestras las personas del sistema formadas por fusiones.

Los resultados obtenidos al realizar los cambios mencionados son los

representados en la figura 6.3. Como puede apreciarse, se ha producido una pequeña mejora a nivel general en las tasas de acierto para los primeros candidatos. Si bien aún la tasa de acierto para el primer candidato es muy baja, la tendencia de la mejora de los resultados indica que los cambios que se han hecho con el fin de ajustar los datos que se usarán para continuar ejecutando pruebas hacen más fiable el sistema de reidentificación.

El siguiente paso, tal y como se explicó en la sección 5.8, fue el ajuste de la medida de distancia. Con esto nos referimos a la aplicación del cálculo variable de la distancia con la asignación de unos pesos determinados. A continuación se detallarán las diferentes pruebas que se efectuaron para lograr finalmente el sistema de medida de distancia explicado en la sección 5.8.

Tras implementar el sistema de pesos variable en función de la altura se estableció el mismo peso a todas las secciones de los histogramas, y la decisión de la aplicación de un método de distancia u otro se basó en si la diferencia de altura entre la muestra y el candidato era mayor a un centímetro. Al ejecutar la comparación con estos parámetros se obtuvo el resultado de la figura 6.4. Los resultados muestran una mejora considerable, teniendo en cuenta que la tasa de acierto del primer candidato se ha duplicado con respecto a la prueba anterior y un ligero aumento para el resto de los primeros candidatos.

Debido a la mejora que aportó este cambio, se decidió seguir realizando pruebas empleando el cálculo variable de distancias modificando diferentes parámetros como son la diferencia de altura a la hora de decidir el cálculo de distancia a aplicar o los pesos de las diferentes secciones en las que están divididos los histogramas de color.

El siguiente cambio que se planteó fue ajustar los pesos del pelo y la cabeza, es decir, emplear las medidas de distancia tal y como figuran en la sección 5.8, en los que se disminuyó el peso de los histogramas del pelo y se aumentó el de la cabeza. Esto se hizo ya que, analizando los valores de distancia entre los múltiples histogramas, se apreció la tendencia de que el pelo tendía a ser similar para personas diferentes, y con la cabeza ocurría lo contrario. Por ello, se quiso constatar si este comportamiento que se apreciaba en los datos podía traducirse en un ajuste en los pesos que nos generara mejores resultados.

En consecuencia, la siguiente prueba se basó en establecer un peso de 0,08 a la diferencia del pelo y uno de 0,2 a la del resto de la cabeza para candidatos de altura similar, manteniendo a su vez la diferencia de altura que establecía el método de distancia a aplicar. Mediante estos parámetros

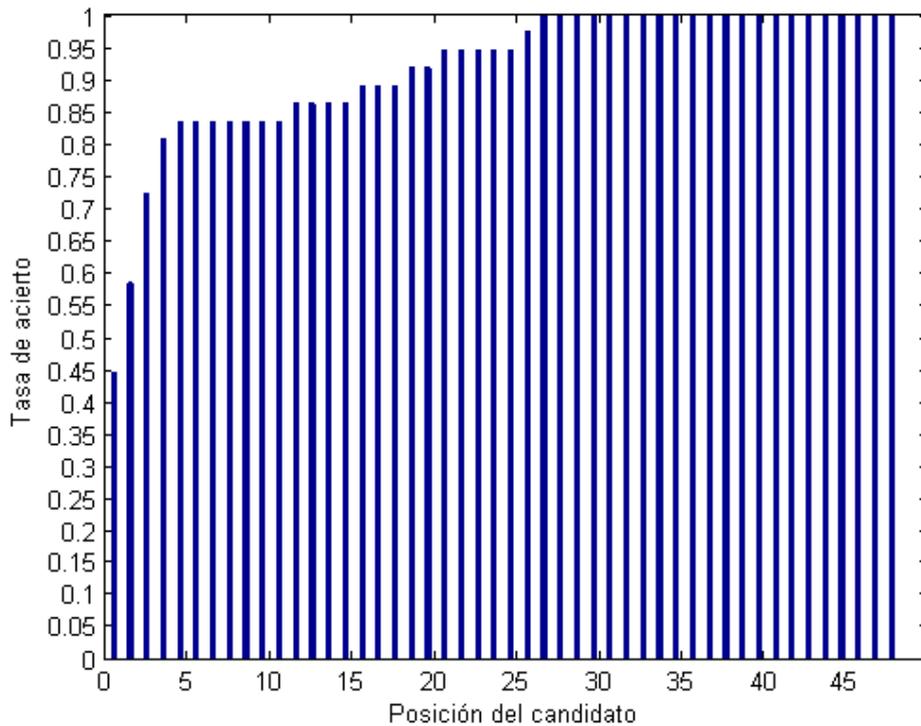


Figura 6.4: Resultados de la primera prueba mediante medidas de distancia variables.

se obtuvo el resultado de la figura 6.5.

Como se puede apreciar, mediante la aplicación de estos cambios se ha vuelto a mejorar la tasa de acierto para los primeros candidatos y se ha superado muy ligeramente el 50% de acierto para el primer candidato de la galería. A pesar de ello, se continuó realizando pruebas modificando más parámetros para intentar aumentar la tasa de acierto del primer candidato ya que, si bien conforme se examinen más candidatos la tasa de acierto aumenta considerablemente, el escenario ideal sería obtener una tasa elevada para los primeros candidatos.

La siguiente vía que se exploró fue la de modificar el valor umbral de altura para la toma de decisión del cálculo de la distancia de cada candidato. Por ello se decidió aumentar este valor para analizar el comportamiento del sistema, y averiguar si la modificación de este parámetro influye de manera positiva en los resultados. Tras realizar la prueba se obtuvo la gráfica de la figura 6.6.

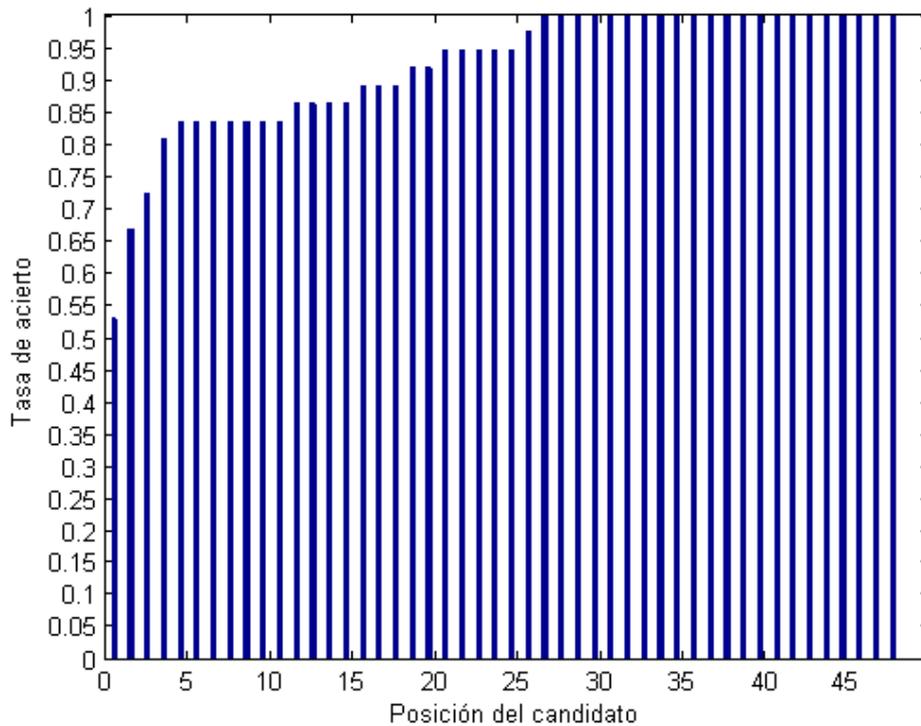


Figura 6.5: Resultados de la primera prueba ajustando los pesos del pelo y la cabeza.

Como es posible apreciar, no se produce ningún cambio significativo en la tasa para los dos primeros candidatos, sin embargo se aprecia una ligera disminución para el tercero. Se consideró que esta vía no aportaba mejoras al sistema, por lo que, partiendo de lo expresado en [Marín Reyes, 2015] se decidió emplear otro espacio de color. El objetivo es que el sistema no se vea tan afectado por las condiciones de luz de la imagen, si bien para el caso de RGB, empleando el método de normalizado se consigue atenuar los cambios que producen estas condiciones. A continuación se detallarán los pasos que se siguieron para cambiar a otro espacio de color y las pruebas que se efectuaron.

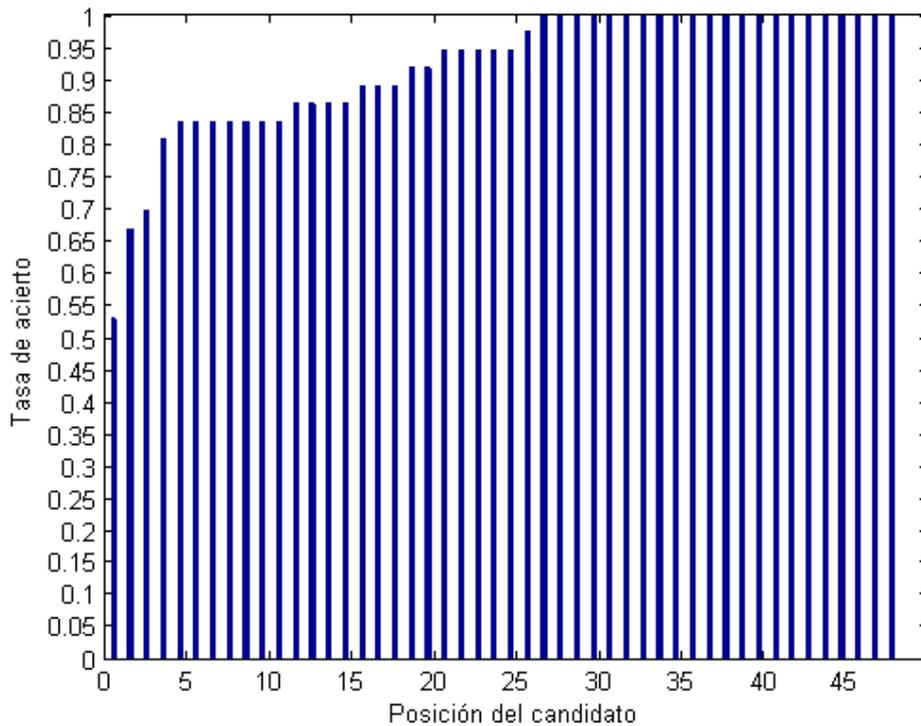


Figura 6.6: Resultados de la prueba en la que se modifica el umbral de altura.

6.2. Espacio de color HSV

Partiendo de las conclusiones de [Marín Reyes, 2015], se decidió emplear el espacio de color HSV. En este esquema, una de sus componentes, el valor, representa la luminancia, por lo que, de producirse alteraciones en la iluminación de la escena, sólo esta componente se verá afectada. Por tanto, se puede ignorar el valor del color, y tan solo tener en cuenta las componentes H y S, el matiz y la saturación a la hora de hacer los histogramas, los cuales son invariantes a las condiciones lumínicas. Para realizar la conversión desde los valores RGB se decidió hacer uso de la función disponible en la biblioteca PCL, `PointXYZRGBtoXYZHSV`, la cual para un punto dado en RGB, genera uno equivalente en HSV, pudiendo a continuación acceder a las componentes H, S y V para generar los histogramas correspondientes.

La primera prueba que se realizó empleando este espacio de color se hizo modificando el rango de valores del matiz del color, es decir, la componente

H, a la mitad pasando de 360 a 180 posibles valores, manteniendo el rango de la saturación e ignorando el valor del color. Empleando estos valores para realizar los histogramas se obtuvo el resultado que se muestra en la figura 6.7.

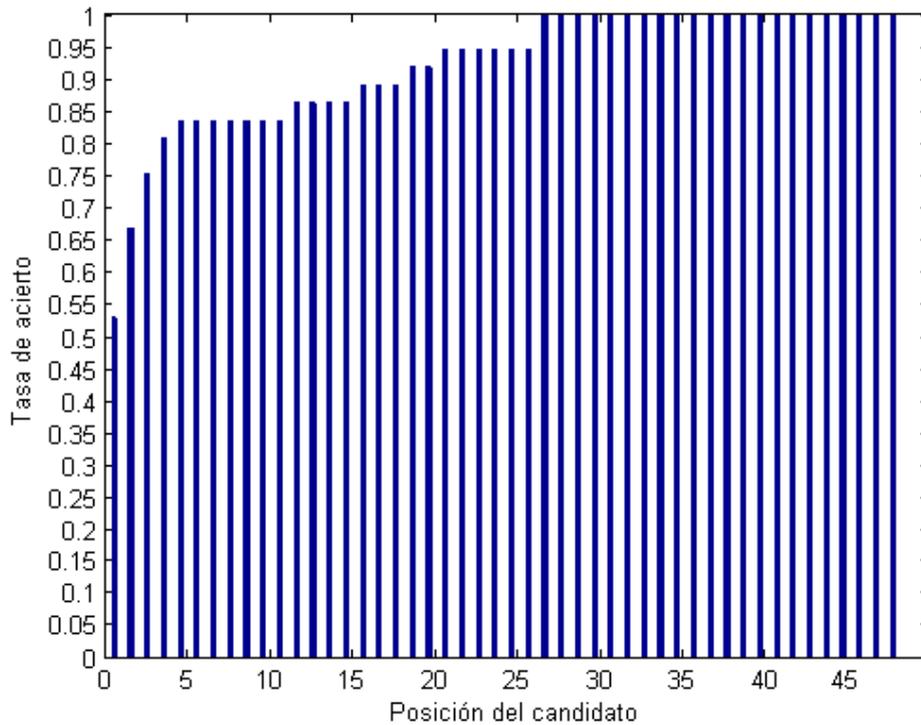


Figura 6.7: Resultados de la primera prueba empleando el espacio de color HSV y $H[0-180]$.

Tal y como se puede apreciar, no se han producido cambios significativos en la tasa de aciertos para los dos primeros candidatos; sin embargo, para los tres primeros candidatos se aprecia una mejora en los aciertos del sistema. Por otro lado, los resultados conforme se exploran más candidatos son similares a los obtenidos en el espacio de color RGB.

A continuación se decidió modificar el valor de la componente S, la saturación, y reducirlo a la mitad, de manera que se tendrían 50 posibles valores, en vez de 100, con el fin de analizar si el seguir reduciendo el espectro de colores tiene efectos positivos en la fiabilidad de los resultados. Tras este cambio, se obtuvo el resultado mostrado en la figura 6.8.

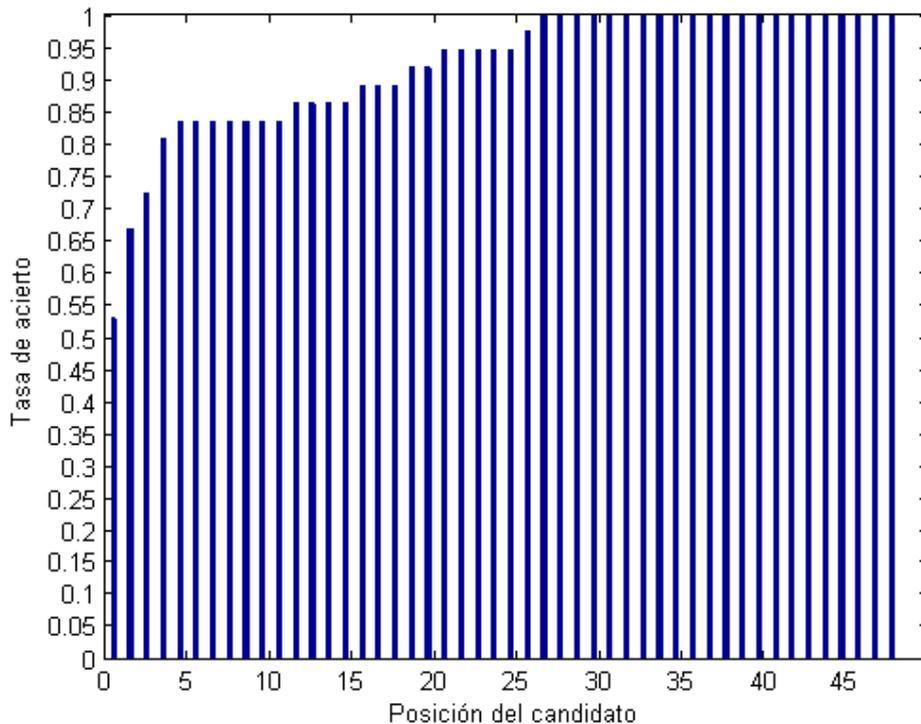


Figura 6.8: Resultados del espacio de color HSV para $H[0-180]$ y $S[0-50]$.

En este caso, al igual que para la prueba anterior, no se produjo un cambio en los dos primeros candidatos, pero sí se produce un ligero empeoramiento de la tasa de acierto para los tres primeros. Esto puede indicar que reducir la componente S puede reducir la fiabilidad de los resultados que genera el sistema. Por ello, se decide reducir la componente H aún más y mantener el rango original para la componente S. Habiendo disminuido a 100 posibles valores el matiz se obtuvo el resultado de la figura 6.9.

Al examinar el resultado de esta prueba, se puede apreciar que se ha obtenido el mismo resultado para los histogramas con un rango de la componente H de 180 valores, el mostrado en la figura 6.7, y para un rango de 100, es decir, el de la figura 6.9. Al contrario de lo esperado, no se ha producido ninguna mejora.

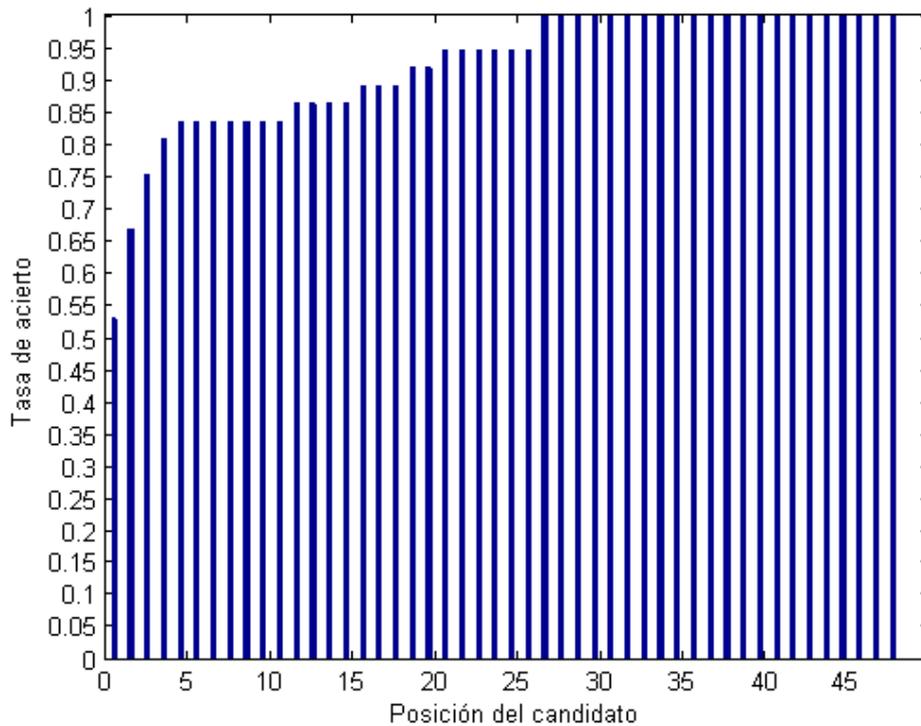


Figura 6.9: Resultados del espacio de color HSV para H[0-100] y S[0-100].

6.3. Análisis de resultados

Tal y como se ha podido constatar en la sección anterior, a pesar de realizar múltiples pruebas del sistema ajustando diferentes parámetros, no se ha logrado obtener una buena tasa de acierto para el primer candidato, obteniendo un 53% en el mejor caso. Sin embargo se ha obtenido una tasa del 75% en el mejor caso para la aparición de la persona buscada entre los tres primeros candidatos, llegando hasta un 84% para los cinco primeros candidatos con más parecido al individuo buscado, según el sistema.

Para poder interpretar estos resultados hemos de tener en cuenta que muchos de los datos que se habían planteado extraer de los individuos, como el área o el volumen de la misma, no han dado buenos resultados a la hora de realizar la comparación, teniendo pues que descartar esta información. Esto ha provocado que el sistema dependa en exceso de la información de color, la cual sufre muchos cambios y provoca errores en la re-identificación en el

caso en el que el individuo lleve alguna prenda de diferente color entre cada cámara. Por otro lado, hemos de tener en cuenta el elevado ruido que poseen las imágenes, lo cual deriva en una posible falta de precisión o fallos en la detección de la profundidad y en el color.

A pesar de haber encontrado estos contratiempos, se han intentado identificar las diferentes etapas que son susceptibles de ser mejoradas, tal y como se detalla en la sección 7.1 donde a su vez se proponen soluciones, métodos alternativos y otras posibles opciones para caracterizar a cada individuo que podrían hacer mejorar de forma significativa el sistema, proporcionando resultados más fiables.

Capítulo 7

Trabajo futuro y conclusiones

Este proyecto se ha realizado con el fin de obtener un sistema que permita re-identificar personas en base a sus características físicas. A lo largo del desarrollo se han encontrado algunos problemas que no han podido resolverse completamente, sin embargo, se han intentado abarcar y se han propuesto soluciones a muchos de ellos. Estas soluciones o mejoras podrían ser tomadas en consideración con el fin de aumentar la tasa de acierto en la re-identificación, o aplicar el sistema desarrollado a ciertas tareas que en un principio no se habían planteado, pero que, de llevarse a cabo ciertas modificaciones el sistema podría tener múltiples aplicaciones.

7.1. Mejoras del sistema

Tal y como se ha indicado anteriormente, durante algunas de las fases del sistema se han encontrado imprevistos que impidieron utilizar ciertos datos u obtener cierta información útil, que en un principio se planteó como indispensable a la hora de caracterizar a una persona. A continuación se plantearán las posibles mejoras en las diferentes partes del proyecto que, podrían aplicarse de continuar con el mismo.

7.1.1. Dispositivo de captura

En primer lugar, como bien se ha expresado en diversas ocasiones, el dispositivo de captura, la cámara Kinect no posee una resolución muy grande, ni un nivel de detalle o precisión suficiente. Esto se aprecia en la cantidad de ruido de los datos que esta proporciona. Al comienzo del desarrollo del proyecto, se decidió usar este sensor, aún sabiendo sus limitaciones ya que, en ese momento, era de las mejores opciones que podían considerarse y existía

una extensa documentación sobre su uso. Sin embargo, más adelante se lanzó al mercado una nueva Kinect, la cual tenía una mayor resolución y precisión en comparación con su predecesora, llegando incluso a estimar el pulso de una persona mediante los cambios que se producen en la piel.

Una posible mejora sería emplear sensores con mayor resolución como el mencionado, el cual aún no dispone de controladores oficiales para su uso desde un ordenador, ya que no existiría tanto ruido en las imágenes y no se tendrían tantos puntos en la imagen de color sin datos de profundidad, tal y como ocurre con la cámara utilizada.

7.1.2. Detección y segmentación

En la fase de detección y segmentación es posible aplicar algunas mejoras con el fin de evitar falsas detecciones, tal y como se ha comentado. Se podría incluir el sistema de detección de cabezas de [Quesada Díaz, 2015] para evitar que cambios en el entorno entorpezcan la detección. A su vez, la detección de cabezas sería de utilidad para aumentar la fiabilidad la detección de personas fusionadas, es decir, dos personas que pasan juntas, pero que en imágenes anteriores aparecían separadas. Recordemos que este comportamiento estaba controlado en este proyecto, sin embargo, no se diferencia el área que pertenece a cada persona en el área fusionada, lo cual facilitaría el seguimiento de la persona.

7.1.3. Caracterización

A lo largo del desarrollo se ha constatado lo complicado que ha resultado obtener características físicas de la persona, debido principalmente a la falta de precisión del sensor, lo cual se traduce en valores muy diferentes para diferentes mediciones de la misma persona. Por ello, este apartado sería muy importante al objeto de que el sistema funcione bien re-identificando a personas durante diferentes días, debido a la alta dependencia de los datos de color, tal y como se ha desarrollado en el proyecto. Esta falta de datos sería uno de los puntos que se podría mejorar si nos basamos en [Quesada Díaz, 2015], donde se detecta el sexo de la persona y se obtienen valores de circularidad de la cabeza. Por otro lado tenemos la textura que representa el modelo de la persona, el cual no se generaba correctamente debido principalmente al ruido del sensor. Sería interesante añadir esta información como características físicas con el fin de analizar si se produce un aumento considerable en la tasa de acierto, ya que lo esperable es que mejore en cierta medida.

7.2. Posibles aplicaciones del sistema

Si bien el proyecto tiene como objetivo lograr un sistema funcional, no se ha tenido en cuenta que sea un producto que pueda usarse de forma directa. Se trata de intentar realizar un sistema que podría tener múltiples usos en diferentes áreas.

En primer lugar podemos pensar en una gran superficie que desee analizar las zonas por las que pasan los clientes y de qué manera. De tener un número considerable de sensores instalados, sería posible identificar a cada individuo que pasa bajo el área de detección, pudiendo así saber, tras analizar toda la información, las zonas por las que ha pasado, lo cual podría traducirse en las zonas que más le atraen. Esto puede ser de utilidad en estudios de mercado, ya que se podrían analizar las zonas que son más o menos atractivas para los clientes, y en el caso de mejorar el sistema para incluir datos como el sexo, se podría hacer un análisis en mayor profundidad teniendo en cuenta el sexo de los clientes.

En ámbitos de seguridad, este sistema puede emplearse para seguir a una persona a lo largo de una zona con múltiples cámaras, en el caso que se desee saber los lugares en los que ha estado. Esto sería aún más útil de lograr hacer que el sistema se ejecute en tiempo real, algo que se podría llegar a hacer, teniendo en cuenta el tiempo de procesamiento de los datos. Como se ha indicado, cada módulo del sistema es independiente y puede tenerse un equipo ejecutando las primeras fases y enviando los datos característicos de cada persona a un sistema central, el cual reciba los datos de las cámaras. Aún así, sigue suponiendo una elevada carga computacional a tener en cuenta.

Relacionado con lo anterior, en el caso de funcionar en tiempo real, el sistema podría usarse con otros fines ya que, hasta ahora, nos hemos basado en analizar el comportamiento de las personas una vez han pasado por las cámaras. Sin embargo, podría emplearse para buscar algún patrón que pudiera aparecer en el futuro. Para ejemplificar esta situación supongamos un centro comercial con múltiples cámaras, donde se da la circunstancia de que un menor se separa de sus padres y se pierde. El sistema podría usarse, tras ciertas modificaciones, para que, tras introducir los datos básicos de la persona que se ha perdido como la altura aproximada o el color de su ropa, detecte cuando alguien con dichas características pasa bajo alguna de las cámaras lanzando un aviso, por lo que se podría localizar fácilmente a la persona perdida.

7.3. Conclusiones

Con este proyecto se ha querido proponer y diseñar un sistema que permita re-identificar a personas a partir de las imágenes capturadas por un sensor externo sin obtener imágenes del rostro de las mismas. Como se ha podido comprobar, se ha logrado obtener un sistema funcional, sin embargo las tasas de acierto no son lo suficientemente elevadas para lograr que el sistema funcione de manera totalmente autónoma, por lo que ofrece una serie de candidatos entre los que se puede encontrar una persona que coincida con la que se busca.

Tras el desarrollo, se ha podido comprobar que el tratamiento de imágenes puede ser una tarea compleja a la hora de procesarlas y extraer de ellas una determinada información. Esto se debe a múltiples razones entre las que podemos destacar que el sensor que captura las imágenes introduce mucho ruido, generando valores perdidos o información errónea en ciertos puntos. Si a esto le sumamos el hecho de que el escenario puede sufrir alteraciones y pueden ocurrir casos que no están contemplados, como por ejemplo que dos personas decidan pasar con sus cabezas juntas, comprendemos que es necesario que el sistema tenga una cierta tolerancia a fallos.

Así pues, se ha propuesto y logrado un sistema alternativo a los que pueden encontrarse actualmente en el mercado, si bien este tiene un fin diferente y que intenta preservar la intimidad de las personas que son grabadas, ya que a priori se intenta no obtener datos sobre el rostro. Aún así, se condensa la información sobre cada individuo de manera que llegado a un punto del proceso dejamos de trabajar con las imágenes obtenidas; se construye un fichero de texto con datos que describan las características que se consideran necesarias para la posterior re-identificación, desconociendo la identidad real de la persona, la cual, a efectos del sistema será un número.

Por otro lado, se han investigado y propuesto posibles mejoras o métodos alternativos para realizar algunas de las operaciones llevadas a cabo por el sistema, con el fin de lograr mejores resultados o aportarle otras funcionalidades o utilidades al mismo. De esta manera se aclaran las líneas futuras que podrían seguirse para mejorar el sistema y llegar a hacerlo un producto comercializable.

Finalmente, como proyecto académico se puede considerar que se han cumplido las expectativas de un Proyecto de Fin de Carrera en el que se ha puesto en práctica toda la experiencia previa de la que se disponía. La

realización del mismo ha resultado una experiencia enriquecedora en cuanto a los conocimientos que he adquirido y a la experiencia de realizar un proyecto de una cierta envergadura como este, en el que se aprecia la importancia de la constancia en el trabajo y la capacidad de resolución de los problemas que han surgido durante el desarrollo. De esta manera se ha podido comprobar de primera mano cómo se va desarrollando un proyecto desde el momento en el que es una idea, para luego ir tomando forma e ir adaptándose a las circunstancias o limitaciones que van surgiendo durante las diferentes fases del desarrollo.

Glosario

bin En la realización de histogramas se denomina al intervalo de valores al se reduce el intervalo original. 48

blob Término usado en re-identificación para nombrar el área ocupada en la imagen por un individuo, pudiendo ser representado como una máscara binaria. 2, 38–41

canon antropométrico Se trata de un sistema de medidas de proporción del cuerpo humano, obtenido mediante la comparación de una parte de la población, perteneciendo pues al ámbito científico definido en el siglo XIX. 47

CMC (Cumulative Matching Curve) Curva que representa la probabilidad de identificación para una lista de candidatos de tamaño 1 a N. 62

galería Término usado en re-identificación para el conjunto de imágenes de individuos entre los que se buscará un candidato. Proviene del término inglés "gallery". 56, 59, 61–64, 66

LOPD Ley Orgánica de Protección de Datos. 15

muestra Término usado en re-identificación para el individuo base que se pretende reidentificar entre una serie de individuos denominada galería. 56, 59, 61–65

multi-versus-multi shot [Fusiello et al., 2012] Tipo de escenario en re-identificación en el que la galería y los probes de cada persona tienen ambos múltiples imágenes. También se denomina como problema MvsM. 55

RGBD Siglas en inglés de Rojo, Verde, Azul y Profundidad. Referido a la obtención de un conjunto de datos de una imagen en color RGB y la

obtención de un valor de profundidad para cada punto de dicha imagen.
1, 18, 24

XML Siglas en inglés de eXtensible Markup Language ('lenguaje de marcas extensible'), es un lenguaje de marcas desarrollado por el World Wide Web Consortium (W3C) utilizado para almacenar datos en forma legible. 53

Bibliografía

- [Ackerman, 2013] Ackerman, D. (2013). Xbox one and playstation 4: Facial recognition shootout. URL: <http://www.cnet.com/news/xbox-one-and-playstation-4-facial-recognition-shootout/>.
- [Australian Customs, 2015] Australian Customs, a. B. P. S. (2015). Smartgate. URL: <http://www.customs.gov.au/smartgate/>.
- [Boehm, 1988] Boehm, B. (1988). *A spiral model of software development and enhancement*. Computer, 21(5), 61-72. doi:10.1109/2.59.
- [Bogdan Rusu and Cousins, 2013] Bogdan Rusu, R. and Cousins, S. (2013). *3D is here: Point Cloud Library (PCL)*. IEEE International Conference on Robotics and Automation (ICRA).
- [Castrillón Santana, 2014] Castrillón Santana, M. (2014). Depthvisdoor database. URL: <http://berlioz.dis.ulpgc.es/roc-siani/descargas/depthvisdoor-database-1>.
- [Daugman, 2009] Daugman, J. (2009). *Iris Recognition at Airports and Border-Crossings*. Encyclopedia of Biometrics, pp 819-825.
- [Fusiello et al., 2012] Fusiello, A., Murino, V., and Cucchiara, R. (2012). *Computer Vision – ECCV 2012. Workshops and Demonstrations : Florence, Italy, October 7-13, 2012, Proceedings, Part I*. Springer.
- [Git, 2008] Git (2008). Git (abril 2015). URL: <https://git-scm.com/>.
- [Grupo de Trabajo del Artículo 29, 2012] Grupo de Trabajo del Artículo 29 (2012). *Dictamen 3/2012 sobre la evolución de las tecnologías biométricas*. Comisión Europea.
- [Horcajada González, 2011] Horcajada González, R. (2011). *Apuntes generales de anatomía morfológica aplicada: Cánones y proporciones*. Universidad Complutense de Madrid.

- [Intel, 2015] Intel (2015). Threading building blocks (marzo 2015). URL: <https://www.threadingbuildingblocks.org/>.
- [Kinect for Windows, 2012] Kinect for Windows, D. (2012). Skeletal tracking. URL: <https://msdn.microsoft.com/en-us/library/hh973074.aspx>.
- [Kramer et al., 2012] Kramer, J., Burrus, N., Herrera C., D., Echtler, F., and Parker, M. (2012). *Hacking the Kinect*. Apress.
- [Lorenzo Navarro et al., 2013a] Lorenzo Navarro, J., Castrillón Santana, M., and Hernández Sosa, D. (2013a). On the use of simple geometric descriptors provided by rgb-d sensors for re-identification. *Sensors*.
- [Lorenzo Navarro et al., 2013b] Lorenzo Navarro, J., Castrillón Santana, M., and Hernández Sosa, D. (2013b). People semantic description and re-identification from point cloud geometry.
- [Marín Reyes, 2015] Marín Reyes, P. A. (2015). *Estudio comparativo de medidas de distancia para histogramas en problemas de reidentificación*. ULPGC.
- [Mathe et al., 2012] Mathe, L., Samban, D., and Gómez, G. (2012). *Estudio del funcionamiento del sensor Kinect y aplicaciones para bioingeniería*.
- [Microsoft, 2014] Microsoft (2014). Sensor kinect para windows v2. URL: http://www.microsoftstore.com/store/mseea/es_ES/pdp/Sensor-Kinect-para-Windows-v2/productID.299242600.
- [Naik et al., 2009] Naik, N., Patil, S., and Joshi, M. (2009). *A Scale Adaptive Tracker Using Hybrid Color Histogram Matching Scheme*. Emerging Trends in Engineering and Technology (ICETET), 2009 2nd International Conference on: 279-284.
- [Navarro Pastor, 1998] Navarro Pastor, J. B. (1998). *Aplicación de redes neuronales al tratamiento de datos incompletos*. Universitat Autònoma de Barcelona.
- [OpenCV, 2015] OpenCV (2015). Opencv — opencv (marzo 2015). URL: <http://opencv.org/>.
- [OpenMP, 2015] OpenMP (2015). Openmp.org (marzo 2015). URL: <http://openmp.org/>.

- [Oracle, 2003] Oracle (2003). Oracle vm virtualbox. URL: <https://www.virtualbox.org>.
- [PCL, 2011a] PCL (2011a). Fast triangulation of unordered point clouds. URL: http://pointclouds.org/documentation/tutorials/greedy_projection.php.
- [PCL, 2011b] PCL (2011b). Filtering a pointcloud using a passthrough filter. URL: <http://pointclouds.org/documentation/tutorials/passthrough.php>.
- [PCL, 2011c] PCL (2011c). How to use random sample consensus model. URL: http://pointclouds.org/documentation/tutorials/random_sample_consensus.php.
- [PCL, 2011d] PCL (2011d). Pcl - point cloud library (pcl). URL: <http://pointclouds.org>.
- [PCL, 2011e] PCL (2011e). Removing outliers using a statisticaloutlierremoval filter. URL: http://pointclouds.org/documentation/tutorials/statistical_outlier.php.
- [PCL, 2014] PCL (2014). Conversion from cv::mat to pcl::pointcloud (2014). URL: <http://www.pcl-users.org/Conversion-from-cv-Mat-to-pcl-PointCloud-td4023936.html>.
- [PCL, 2015a] PCL (2015a). Euclidean cluster extraction (marzo 2015). URL: http://www.pointclouds.org/documentation/tutorials/cluster_extraction.php.
- [PCL, 2015b] PCL (2015b). How to incrementally register pairs of clouds (mayo 2015). URL: http://pointclouds.org/documentation/tutorials/pairwise_incremental_registration.php.
- [Quesada Díaz, 2015] Quesada Díaz, E. (2015). *Estudio sobre el uso de sensores RGB-D para el conteo de personas y su caracterización*. ULPGC.
- [Roerdink and Meijster, 2001] Roerdink, J. B. and Meijster, A. (2001). *The Watershed Transform: Definitions, Algorithms and Parallelization Strategies*. Institute for Mathematics and Computing Science University of Groningen.
- [ROS, 2014] ROS (2014). Kinect calibration - technical (2014). URL: http://wiki.ros.org/kinect_calibration/technical.

- [SAFRAN Morpho, 2015] SAFRAN Morpho (2015). Border crossings: Morpho has developed a biometric solution for australia. URL: <http://www.morpho.com/references/border-control/border-crossings-morpho-has-developed-a-biometric-solution-for-australia>.
- [Simon and Goldstein, 1935] Simon, C. and Goldstein, I. (1935). *A New Scientific Method of Identification*. New York State Journal of Medicine, Vol. 35, No. 18.
- [Todo Vector, 2014] Todo Vector (2014). Imágenes de personas caminando. URL: <http://todovector.com/vector/personas/parejas/caminando/>.
- [Upton and Halfacree, 2012] Upton, E. and Halfacree, G. (2012). *Meet the Raspberry Pi*. John Wiley and Sons.
- [Viafirma Developers, 2015] Viafirma Developers (2015). No recuerdo el pin de mi dne ¿cómo puedo cambiarlo o solicitar uno nuevo? URL: <https://developers.viafirma.com/no-recuerdo-el-pin-de-mi-dnie-como-puedo-cambiarlo-o-solicitar-uno-nuevo>.
- [Wikipedia, 2010] Wikipedia (2010). Kinect. URL: <https://en.wikipedia.org/wiki/Kinect>.
- [Zhang and Canosa, 2014] Zhang, Q. and Canosa, R. L. (2014). *A comparison of histogram distance metrics for content-based image retrieval*. International Society for Optics and Photonics.
- [Zhang et al., 2012] Zhang, X., Yan, J., Feng, S., Lei, Z., Yi, D., and Li, S. Z. (2012). *Water Filling: Unsupervised People Counting via Vertical Kinect Sensor*. 2012 IEEE Ninth International Conference on Advanced Video and Signal-Based Surveillance.

