



Universidad
Zaragoza

Trabajo Fin de Grado

Prototipo de sistema electrónico de videovigilancia
para aplicaciones domóticas

Video surveillance electronic system prototype for smart homes

Autor

Fernando José Belaza Vallejo

Director

Bonifacio Martín del Brio

Grado en Ingeniería Electrónica y Automática

Departamento de Ingeniería Electrónica y Comunicaciones

Escuela de Ingeniería y Arquitectura de Zaragoza

Junio 2023

Resumen

En este trabajo se realizará el diseño, desarrollo y evaluación de un sistema electrónico de videovigilancia para detectar movimientos o sonidos estridentes en el hogar e informar al usuario. Será un sistema para mejorar la seguridad y/o la tranquilidad de un hogar al permitir a sus integrantes estar informados en tiempo real de la situación en la vivienda.

En primer lugar, presentaremos la utilidad de la domótica en la sociedad actual y la importancia de los sistemas de vigilancia. Estudiaremos distintas opciones para el uso de cámara, de reconocimiento de imagen y de sistemas de comunicación con el usuario, así como de la situación legal vigente relativa a los sistemas que realizan grabaciones, por si se deseara instalar en el exterior de una vivienda o zonas comunes.

Posteriormente, presentaremos el diseño tanto del hardware como del software necesario para la construcción de la cámara, con todas las decisiones que se han tomado a partir del análisis anterior. Para la realización del prototipo se cuenta con una tarjeta SBC Raspberry Pi y una webcam, además de diversos componentes electrónicos. Estudiaremos posibles alternativas de estos componentes a futuro en términos de coste y de producción y haremos un estudio del consumo eléctrico del sistema. En cuanto al software, se detallará todo el sistema programado, cuál es el objetivo de cada uno de sus componentes y con qué funcionalidades se ha desarrollado el sistema de videovigilancia. Al final, se validará el funcionamiento del prototipo desarrollado, incluyendo un estudio de consumo de energía.

Por último, se expondrán las conclusiones a partir del diseño final. Se presentarán tanto las diversas situaciones encontradas durante el desarrollo del trabajo como posibles opciones para mejorar el diseño en futuros trabajos, sea para optimizar el prototipo o para dotarlo de más funcionalidades.

Índice

1. Introducción	4
1.1 Motivación	4
1.2 Objetivo del trabajo	5
1.3 Alcance y métodos utilizados	6
1.4 Estructura del trabajo y cronograma	7
2. Planteamiento del problema	9
2.1 Hardware para procesado de imagen y alerta	9
2.2 Cámara	10
2.3 Software	11
2.4 Librerías de tratamiento de imagen	12
2.5 Sistemas de alerta al usuario	13
2.6 Consideraciones legales de seguridad y protección de datos	14
3. Diseño del hardware	16
3.1 Descripción general	16
3.2 Especificaciones técnicas	17
3.3 Energía y consumo	19
3.4 Diseño y diagramas del sistema	21
4. Desarrollo del software	22
4.1 Descripción general	22
4.2 Diseño y Diagramas	26
5. Análisis del prototipo	30
5.1 Análisis de problemas	30
5.2 Aplicación de visión por computador en la cámara	32
5.3 Consumo energético del prototipo.	35
5.4 Uso de sensores de presencia por infrarrojos	38
5.5 Mejoras propuestas de hardware	43
5.6 Mejoras propuestas de software	44
6. Conclusiones finales	46
7. Bibliografía	48

1. Introducción

1.1 Motivación

La domótica es el conjunto de sistemas que automatizan las diferentes instalaciones de una vivienda [1]. Ello incluye el control de la iluminación o temperatura, de los diversos sistemas eléctricos o, en el caso que atañe a este trabajo, la seguridad de la vivienda. La domótica se ha convertido en los últimos años en una tecnología cada vez más popular, debido principalmente a su capacidad para mejorar la calidad de vida de las personas que cohabitan en el hogar y aumentar la eficiencia energética de la vivienda.

La seguridad en el hogar es un aspecto importante de la domótica. Existen sistemas comerciales que incluyen sensores de movimiento, cámaras de vigilancia, alarmas y cerraduras inteligentes. Estos dispositivos pueden coexistir con la capacidad de informar al usuario de la situación de su hogar, de proteger la vivienda ante intrusos o incluso de alertar al propietario en caso de una emergencia. Además, algunos de estos sistemas domóticos permiten el monitoreo o acceso remoto mediante sistemas conectados a la red, como aplicaciones o páginas web. Debido a esto, el desarrollo de sistemas de videovigilancia domóticos se vuelve una opción interesante a estudiar.

En los sistemas de videovigilancia tradicionales (compuestos principalmente por cámaras de vigilancia fijas y/o por sensores de puertas conectadas a sistemas de alarma) el sistema completo suele estar conectado, bien a una central de alarmas (pagando una importante cuota mensual a la empresa), a un sistema de grabación en la nube perteneciente a terceras empresas, o bien a sistemas de grabación de vídeo local para poder ser revisado a posteriori. Esto implica depender de estos servidores de terceros (con el sobre coste que conlleva al usuario y posibles problemas de privacidad), o de no disponer de la información actualizada en todo momento. Además, estos sistemas suelen ser difíciles de integrar con el resto de la domótica de la vivienda y/o no ofrecen interfaces fáciles de usar para que el usuario pueda monitorizar la vivienda a distancia.

Integrar estos sistemas de videovigilancia en la red domótica del hogar tiene diversas ventajas y beneficios. En primer lugar, permite una mayor flexibilidad en cuanto a la ubicación de las cámaras y la monitorización, ya que las cámaras pueden ser colocadas en cualquier punto de la vivienda y ser controladas igualmente desde una sola interfaz; permitiendo así una mejor visibilidad y seguridad. Además, pueden ser programadas alertas personalizadas y/o configuraciones automáticas para responder a situaciones específicas, como movimientos sospechosos o puertas y ventanas abiertas.

Otra ventaja a tener en cuenta es la privacidad: al tener el control total de los datos recogidos en el hogar, se evita la dependencia de servidores externos y se garantiza que la información del usuario permanezca en un sitio personal, sea su hogar o los servidores especialmente elegidos por usuarios más avanzados, garantizando así que estos no se compartan con terceros. Esto es algo a tener en cuenta por la privacidad y seguridad que conlleva, ya que sistemas de videovigilancia tradicionales suelen enviar y/o almacenar sus datos en servidores externos, aumentando por ello el riesgo de violación de privacidad y seguridad de los datos.

Finalmente, otra ventaja será la económica, pues muchos de los sistemas domésticos de seguridad con una funcionalidad similar a la que se va a desarrollar en este trabajo implican el pago de una cuota mensual importante.

1.2 Objetivo del trabajo

El objetivo de este trabajo es diseñar un sistema de videovigilancia para una vivienda. En este caso, se contempla su integración con cámaras de vigilancia para la detección de movimiento. Asimismo, se enviará una alerta al usuario cuando se detecte movimiento en el campo de la cámara, incluyendo capturas de la actividad detectada para que el usuario verifique el movimiento frente a la cámara y tomar medidas acorde a ello, como activar alarmas o, en casos de actividad indeseada, avisar a la policía.

Además, el sistema también deberá incluir la capacidad de monitorizar de forma continua la actividad, permitiendo al usuario solicitar información bajo demanda del

estado actual; por ejemplo, solicitando desde un dispositivo móvil una captura actualizada de lo que ocurre frente a su hogar, dando una tranquilidad adicional al usuario sabiendo que el hogar está seguro.

Otro ejemplo importante de utilidad de este sistema sería su uso como cámara para las habitaciones de niños. Mediante la cámara de vigilancia, el sistema podrá detectar el estado de un niño en todo momento y alertar al usuario en caso de detectar algún problema, en cuyo caso el sistema enviaría una alerta al teléfono del usuario, permitiendo verificar el estado del niño, el problema y actuar si fuera necesario. Además, el sistema podría ser configurado para enviar alertas en casos de movimientos sospechosos, brindando así una mayor seguridad y tranquilidad para los padres.

1.3 Alcance y métodos utilizados

A efectos prácticos, en este trabajo se considerará que el sistema de videovigilancia será instalado en una vivienda para dos situaciones específicas, como una mirilla electrónica, o como una cámara de monitorización de niños y/o mascotas en el hogar. Para ello, se buscará crear un sistema de videovigilancia independiente capaz de detectar movimientos reales frente a una cámara de video y alertar al usuario al móvil con una captura de pantalla; uno de los objetivos del trabajo es estudiar cuál es la manera más adecuada para alertar al usuario. Asimismo, el sistema deberá permitir al usuario, bajo demanda, obtener información actualizada de la situación actual del hogar.

Para este trabajo se ha decidido usar como base de hardware un ordenador de placa única o SBC (*Single Board Computer*), al ofrecer la potencia de un ordenador completo para el análisis de imagen, pero de un tamaño muy reducido que permite su integración en cualquier entorno. Para ello se empleará una Raspberry Pi, al ser uno de los SBC más populares del mercado y que, además, permite la conexión directa de una cámara al dispositivo. Para la prueba de concepto planteada en este trabajo, se utilizará una webcam conectada por USB al usuario, con el fin de abaratar costes para el prototipo al poseer ya previamente el material.

Aunque el análisis de las imágenes mediante visión por computación para el reconocimiento de personas y/u objetos no es estrictamente un objetivo de este trabajo (podría ser objeto de otro TFG completo), sí que se usarán de forma básica librerías de visión por computador para conseguir un buen acabado final y preparar una posible futura ampliación en esta línea.

1.4 Estructura del trabajo y cronograma

En este documento se expone tanto el diseño del hardware como el desarrollo del software de videovigilancia completo. Para ello, se ha estructurado de la siguiente manera:

- En el Capítulo 2 se analizará el sistema de videovigilancia, presentando las distintas opciones disponibles para desarrollar el proyecto.
- En el Capítulo 3 se hará una descripción del hardware utilizado, incluyendo un estudio del consumo eléctrico para un posible montaje independiente de la red eléctrica.
- En el Capítulo 4 se explicará de manera detallada el funcionamiento global y el software del sistema de videovigilancia y alerta al usuario.
- En el Capítulo 5 se realizará el análisis del prototipo, se aplicarán diversas mejoras al mismo y se estudiarán posibles mejoras futuras.
- En el Capítulo 6 se mostrarán las conclusiones de la prueba de concepto, detallando los objetivos logrados en el proyecto y analizando el mismo en su globalidad.
- Por último, en los Anexos encontraremos información como el código fuente, los costes del prototipo o los resultados en bruto de los tiempos de procesado del proceso de análisis de imagen.

2. Planteamiento del problema

2.1 Hardware para procesado de imagen y alerta

Hay múltiples opciones para realizar el procesamiento de imágenes y sistema de alerta. Por un lado, una opción interesante sería ejecutar este procesado en un ordenador de escritorio. Esta opción permitiría una gran flexibilidad en cuanto al análisis de la imagen, además de tener una gran capacidad de procesamiento. Sin embargo, este sistema es costoso, requiere un espacio físico considerable y su consumo energético es elevado.

Otra opción sería el empleo de sistemas de procesamiento en la nube tales como Amazon AWS o Microsoft Azure entre otros, sistemas que permitirían el acceso remoto al sistema de manera muy sencilla, facilitando el monitoreo de la cámara. Esta opción, además de ser bastante costosa *per se* y depender de una conexión a Internet estable para su funcionamiento en todo momento, tiene el riesgo de perder la privacidad y el control de los datos relacionados con la vivienda, al pasar por servicios externos.

Una tercera opción sería utilizar sistemas embebidos para el procesado de imagen. Los sistemas embebidos o empotrados son sistemas compuestos por microprocesadores diseñados específicamente para realizar una función específica, en contraposición a los ordenadores de propósito general. Esta opción tiene la gran ventaja de ser dispositivos pequeños, muy eficientes y de bajo coste; en cambio, su potencia de procesamiento para el análisis de las imágenes es limitada.

Finalmente, la opción por la que se ha decantado en este proyecto es el uso de una SBC (*Single Board Computer*). Este tipo de dispositivos, aparte de tener un bajo coste y un consumo energético relativamente eficiente, disponen de suficiente potencia de procesamiento y son fáciles de usar. Como valor añadido, dispone además de puertos de entrada GPIO (*General Purpose Input/Output*) interesantes para su conexión con otros dispositivos electrónicos, e incluso pueden ser utilizados como central domótica de una vivienda a través de diversas aplicaciones como Home Assistant o Domoticz.

En este proyecto se usará una Raspberry Pi 3B+, al ser una SBC de las más populares del mercado, de fácil acceso para poder realizar el prototipo. Se podría estudiar, en caso de decidir hacer una producción a gran escala, el uso de una Raspberry Pi Zero, cuyo coste es considerablemente inferior.

2.2 Cámara

El sistema de videovigilancia deberá disponer de algún componente que permita la obtención de imágenes. Para ello, disponemos de una serie de soluciones posibles:

- La Raspberry Pi Foundation dispone de unos módulos oficiales de cámaras con 5 (discontinuada), 8 y 12 megapíxeles que se integran directamente en la SBC [2]. Estos dispositivos tienen la ventaja de ser baratos (entre 25 y 50 dólares, según las prestaciones deseadas), pequeños (ninguno llega a 1 cm de diagonal), de bajo consumo y muy versátiles.
- Sistemas compactos como la ArduCAM, preparados para funcionar con microcontroladores; también baratos, versátiles y de muy bajo consumo.
- Cámaras externas que funcionen vía IP/Internet, como las Foscam FI9805EP o la Avidsen Visia 8. Estas son cámaras ya profesionales que disponen de algunas de las funcionalidades que se buscarían en el proyecto, pero su consumo y precio son muy elevados y su integración complicada.

No obstante, se ha decidido realizar el prototipo de nuestro proyecto con una cámara con interfaz USB, la Logitech C525, no solo por su disponibilidad para su empleo en el prototipo, sino además por incluir en el propio dispositivo un micrófono para detectar sonidos, de forma que no es necesario incluir un componente adicional. Habrá que tener en cuenta que en caso de proceder a comercializar el producto, habría que utilizar un dispositivo de menor tamaño y consumo.

2.3 Software

El habernos decantado por el empleo de una SBC permite emplear la gran mayoría de lenguajes de programación disponibles, al poder instalar en estos cualquier sistema operativo y no estar atados al lenguaje dedicado que el sistema embebido exige.

Se ha realizado un estudio de los lenguajes más populares a partir de una serie de puntos que se han considerado de especial valor para nuestro trabajo:

- **Facilidad.** El lenguaje elegido debe ser lo suficientemente sencillo de aprender y de programar para que el simple hecho de aprender el lenguaje no sea en sí mismo un objetivo del presente trabajo. Se valora, por lo tanto, disponer de conocimientos previos sobre el lenguaje y aquellos que dispongan de una buena documentación.
- **Soporte / acceso a librerías.** El sistema va a realizar análisis y procesado de imagen, opciones que no suelen resultar sencillas sin el uso de librerías externas. Asimismo, la comunicación con sistemas externos para el alertado y para recibir avisos del usuario también pueden requerir el empleo de librerías externas. Por ello, se buscará un lenguaje que tenga un buen soporte comunitario (lo cual además contribuirá al punto anterior de facilidad) y que permita trabajar con librerías de forma sencilla.
- **Portabilidad.** Se ha decantado por la utilización de un SBC para el montaje del prototipo, pero resulta interesante permitir que el programa sea fácilmente extraíble e instalado en otro SBC o, incluso, en sistemas embebidos si se comprueba que la potencia requerida resulta entrar dentro del rango del que un sistema embebido pueda ofrecer.
- **Seguridad.** El lenguaje debe de ser lo suficientemente robusto para ofrecer una seguridad al usuario. El sistema de videovigilancia obtiene datos sensibles del usuario, como puede ser imágenes de su hogar o de dónde coloque el dispositivo, y como tal posee un interés especial que esté mínimamente protegido ante posibles ataques externos. No es objetivo de este trabajo como tal dar robustez al sistema de videovigilancia, pero sí que se buscará al menos que el lenguaje disponga de sistemas propios de seguridad.

Tras un estudio de diversos lenguajes de programación disponibles para realizar este trabajo, se ha decidido elegir Java como el lenguaje con el que se realizará el desarrollo. Con respecto a todo lo mencionado anteriormente, Java ofrece una gran facilidad de uso y amplia comunidad de desarrolladores y documentación, a lo que se suma el conocimiento previo del lenguaje por parte del autor de este trabajo; además de ser un lenguaje con una filosofía expresa de portabilidad "*Write Once, Run Anywhere*" (escribe una vez, ejecútalo en cualquier parte) [3]. Además, posee una gran cantidad de librerías a disposición del desarrollador, lo cual facilitará tanto el análisis y procesado de imagen como la comunicación para el alertado del usuario.

2.4 Librerías de tratamiento de imagen

El problema al cual nos enfrentamos en este trabajo es el desarrollo de un sistema domótico de cámara de videovigilancia, añadiendo la necesidad de estudiar el tratamiento de imágenes que permita analizar si hay movimientos externos e indeseados. Para esto, es necesario utilizar alguna librería que permita realizar un tratamiento eficiente de la imagen (el desarrollo de un sistema propio de procesamiento de imagen es algo que queda fuera del objetivo de este trabajo y podría ser considerado un trabajo de fin de grado en sí mismo).

Algunas de las posibles librerías estudiadas para el uso en este proyecto serían las siguientes:

- Librerías oficiales de Java, como "Java Image I/O" o "Java Advanced Imaging", desarrollada por Oracle (propietarios actuales de Java). Aunque permiten procesamiento de imágenes/ficheros y se integra con otras librerías también oficiales, se trata en general de librerías antiguas y no muy empleadas por la comunidad.
- OpenCV. Una de las librerías de visión por computador más populares. Además es de código abierto y gratuita, su ventaja es que dentro del procesamiento de imágenes incluye el reconocimiento de patrones y la detección y seguimiento de objetos. Dispone de versión para Java amén de para otros lenguajes.

- ImageJ. Es una librería de procesamiento de imagen desarrollada por los Institutos Nacionales de Salud de los Estados Unidos. Una de sus principales ventajas es la capacidad de extensión de la librería a través de plugins y herramientas adicionales. Además, aunque mantiene compatibilidad con versiones muy antiguas de Java, sigue siendo desarrollada a día de hoy.
- Marvin. Otra librería especializada en procesamiento de imágenes multiplataforma orientada a entornos de alta productividad y sencillez [4].

Ante las opciones aquí presentadas, se ha decidido decantarse por la librería de código abierto OpenCV debido a su potencia y popularidad, la cual permite encontrar mucha documentación para su uso.

De esta forma podremos tanto diseñar el servicio que estamos estudiando como dejar abierta la posibilidad de futuros desarrollos del trabajo. Un ejemplo de esto sería, dentro de la detección y reconocimiento de objetos, reconocer quién pasa por delante de la cámara (en el caso de empleo como mirilla) o si un niño está realizando algo extraño (en el caso de la cámara de vigilancia).

2.5 Sistemas de alerta al usuario

El sistema de videovigilancia debe disponer de alguna forma de alertar al usuario bajo ciertas circunstancias detectadas. Aunque podría instalarse un sistema más clásico (o “cinematográfico”) de alarma visual y sonora, cumpliría más con el objetivo de este proyecto que el usuario pueda enterarse de cualquier acción que pueda ocurrir desde cualquier lugar.

Para ello, se han estudiado varias opciones, como las siguientes:

- Diseño y creación de aplicaciones básicas para móvil, donde notificar la alerta y consultar sobre el estado actual de la vivienda. El diseño de una aplicación móvil quizás se escapa al objetivo del presente trabajo, pero sí que hay que destacar que con hacer aplicaciones para dos sistemas operativos se cubriría prácticamente la totalidad del mercado de smartphones en España y Europa.
- Alerta a través de correo electrónico. Una de las grandes ventajas de este sistema sería la popularidad del correo electrónico, aunque posee desventajas

como la necesidad de registrar la cámara ante un servicio de correo para poder enviar y recibir mensajes.

- Interfaz Web. Se puede diseñar un pequeño frontal web que permita el acceso para ver las alertas existentes y consultar por nuevas imágenes. Aunque su sencillez de diseño la vuelven una gran opción, lo cierto es que este sistema no permitiría un alertado continuo al sólo poder consultar las alertas desde esta interfaz. Además, se vuelve necesario contratar algún servicio de dominio online para poder acceder desde fuera de la red local del usuario, lo cual añadiría una capa de complejidad importante.

Finalmente, nos hemos decantado por dos sistemas distintos y compatibles entre ellos:

- El uso de una red de mensajería (Telegram) para la comunicación con el usuario. Las redes de mensajería están en auge, teniendo una cuota de mercado relevante. Entre las más populares, Telegram posee una API (*Application Programming Interface*, o Interfaz de Programación de la aplicación) que permitiría integrarla con nuestro dispositivo de manera sencilla. Además, facilitamos al usuario también lanzar órdenes al sistema de forma sencilla, simplemente escribiendo en el chat o con pulsaciones de botón.
- El uso de un sistema de mensajería de colas como MQTT (*Message Queues Telemetry Transport* transporte de telemetría de cola de mensajes), especialmente popular en los sistemas de domótica, para la integración con cualquiera de las aplicaciones de domótica previamente existentes, como Home Assistant u Homebridge.

2.6 Consideraciones legales de seguridad y protección de datos

En España, el organismo público que se responsabiliza de cumplir las medidas vigentes con respecto a la protección de datos es la Agencia Española de Protección de Datos o AEPD [5]. Esta agencia tiene publicada una guía sobre el uso de videocámaras para seguridad y otros afines de fácil acceso al público [6].

En esta guía, encontramos las siguientes indicaciones:

- En el caso de que la cámara grabe espacios públicos (y siempre que sea porque resulte imprescindible para dicho fin y que no pueda evitarse por la ubicación de la misma), la cámara deberá grabar la porción mínimamente imprescindible para cumplir con su finalidad.
- Con la intención de colocar este sistema de videovigilancia como mirillas, las grabaciones en zonas comunes de viviendas requieren acuerdo de la junta de propietarios, según la Ley de Propiedad Horizontal. Las mirillas digitales están excluidas de la normativa de protección al ser de utilidad doméstica, siempre que se use para facilitar el acceso a la vivienda y/o reconocer la identidad de la persona que accede a la misma. Si la mirilla reproduce y/o graba imágenes, el RGPD resultará de plena aplicación.
- La grabación en interiores de vivienda se consideran parte del ejercicio de actividades personales y domésticas, por lo que no se aplicaría la legislación de protección de datos.

Con esto, podemos tener en cuenta que podremos emplear la cámara sin ningún tipo de impedimento en el interior de la vivienda, como cámara de monitorización de mascotas y/o de niños, sin ningún perjuicio sobre la legislación. Sin embargo, habrá que tener especial atención en la instalación externa como mirilla.

3. Diseño del hardware

3.1 Descripción general

El sistema de videovigilancia consistirá inicialmente en un SBC (Raspberry Pi 3+), una cámara USB (con micrófono incorporado) y un pulsador conectado a uno de los puertos GPIO que incluye el SBC (Fig. 3.1).

Esta configuración de hardware permite grabar tanto audio como vídeo en todo momento y ser analizados y procesados dentro del mismo dispositivo, de forma que el sistema serviría como cámara de alerta en habitaciones. Además, el botón por puerto GPIO permite poder invocar la cámara de manera externa, de forma similar a la activación de un videoportero al llamar al timbre.

Este sistema, relativamente simple y fácil de mantener, permite desarrollar un prototipo y trabajar con comodidad.

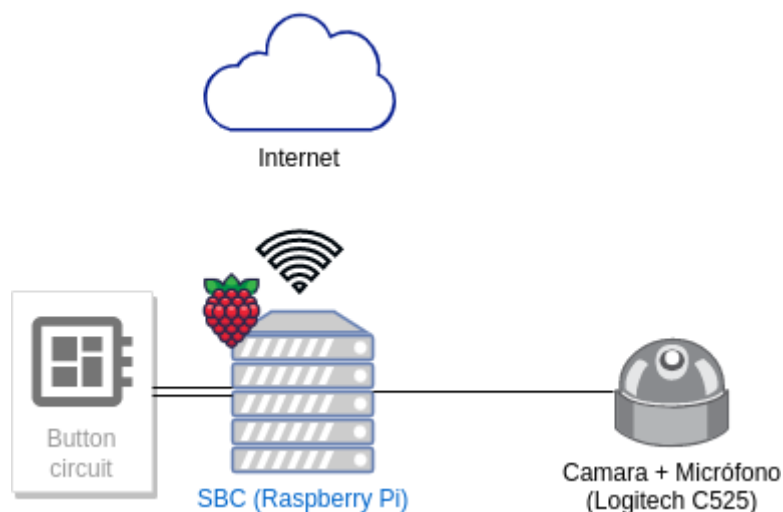


Fig. 3.1 - Diagrama general del Hardware

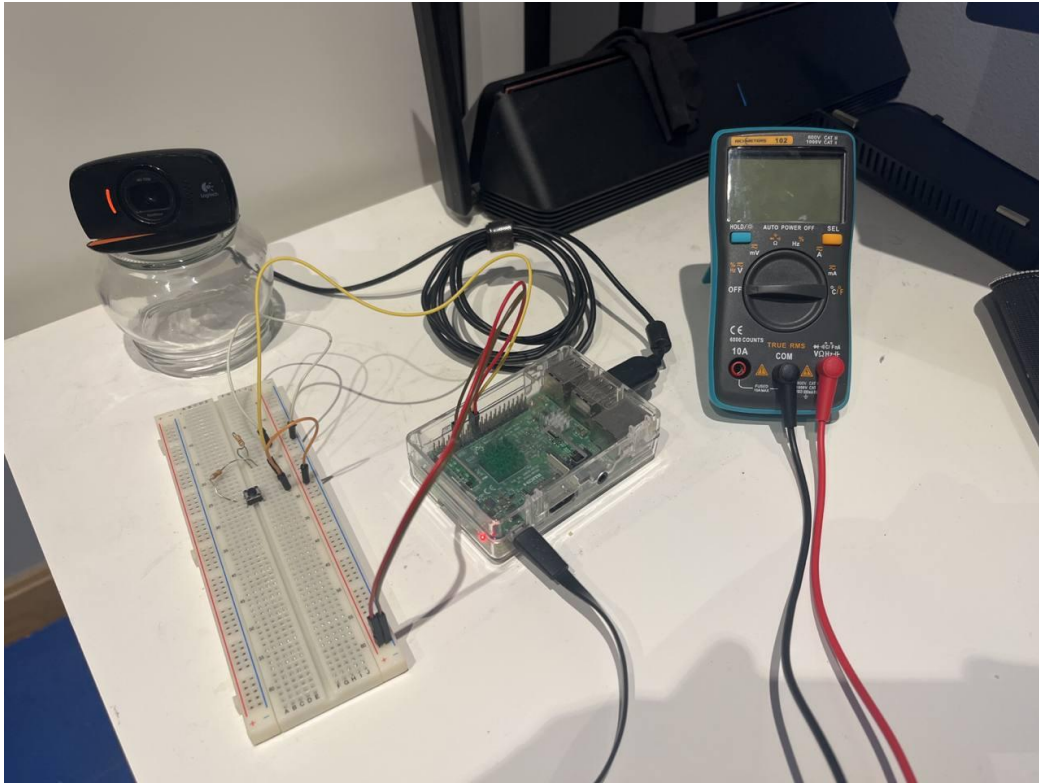


Fig. 3.2 - Foto del prototipo

3.2 Especificaciones técnicas

En el caso del SBC, se ha elegido la Raspberry Pi, modelo 3B+. (Figura 3.3). Este modelo en concreto posee un procesador Broadcom BCM2837B0, con una frecuencia de procesamiento de 1.4 GHz, y 1 GB de SDRAM incluida como memoria interna. También permite su conexión a otros equipos y redes, bien a través de cable Ethernet, bien a través de sistemas sin cables como wifi (2.4 y 5 GHz) o Bluetooth (4.2 y Bluetooth Low Energy) [7].

Además de lo comentado, posee cuatro puertos USB, un conjunto de 40 pines de entrada GPIO junto a dos entradas CSI (Camera Serial Interface) y DSI (Display Serial Interface) para conectar una cámara y una pantalla táctil; un puerto para tarjeta Micro SD como principal almacenamiento y una entrada de alimentación por micro USB (5 V).



Figura 3.3 – Raspberry Pi, modelo 3B+.

Imagen original de Raspberry Pi Foundation [8]

Hay que tener en cuenta que este producto tiene un consumo eléctrico mayor que otros SBCs, menos potentes pero adecuados para el objetivo de este trabajo, debido principalmente al hardware que posee para la gestión del sistema de Ethernet Gigabit. Se recomienda una fuente de alimentación de 2.4 A para el uso del sistema. Esto es algo que revisaremos más adelante.

La cámara usada para el prototipo será una Logitech C525 (figura 3.4). Es una cámara por interfaz USB capaz de grabar a resolución 720p y con una frecuencia de 30 imágenes por segundo. Permite la rotación de la cámara 360 grados sobre su base, lo cual facilitaría construir un sistema de seguimiento. Además, incluye un micrófono para la grabación de audio.

Por último, conectado mediante los puertos GPIO del sistema tendremos un pulsador que servirá a modo de timbre o para demostración física del conjunto. Este pulsador se ha conectado usando una resistencia *pull-up* en el que, en estado de reposo, se hará una lectura de tensión equivalente a la fuente de alimentación (en este caso, 3,3 V) y, cuando se pulse el botón, se recibirá una lectura de 0 V al conectarse la salida

directamente a tierra. Con una resistencia de $20\text{ k}\Omega$, garantizamos que la corriente utilizada en la parte física de la demo sea prácticamente despreciable ($0,2\text{ mA}$).

3.3 Energía y consumo

Para poder analizar el consumo eléctrico del sistema, se ha empleado un relé doméstico con monitorización de corriente conectado al adaptador de corriente del SBC (el Shelly 1PM, mostrado en la figura 3.5). Con el programa inicializado, el consumo se ha estabilizado en unos $3,8\text{ W}$, como se aprecia en la figura 3.6.

Este consumo se podría considerar elevado para estos tipos de dispositivo, pero era esperable al haber cogido uno de los modelos de SBC más potentes.



Fig. 3.4 - Logitech C525.

Imagen Original de Logitech [9]



Fig. 3.5 - Shelly 1PM.

Imagen Original de Shelly [10]

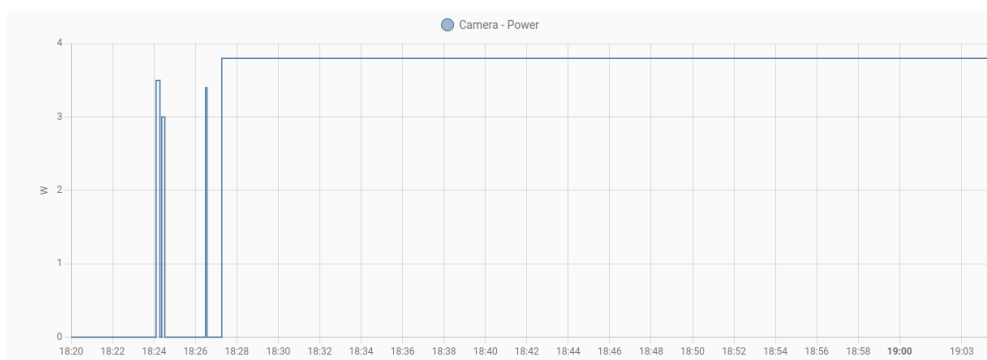


Fig. 3.6 - Consumo del sistema de videovigilancia

3.4 Diseño y diagramas del sistema

En la figura 3.7 se indica un pequeño diagrama del sistema eléctrico conectado al GPIO de la SBC para la conexión del botón que actuará como pulsador.

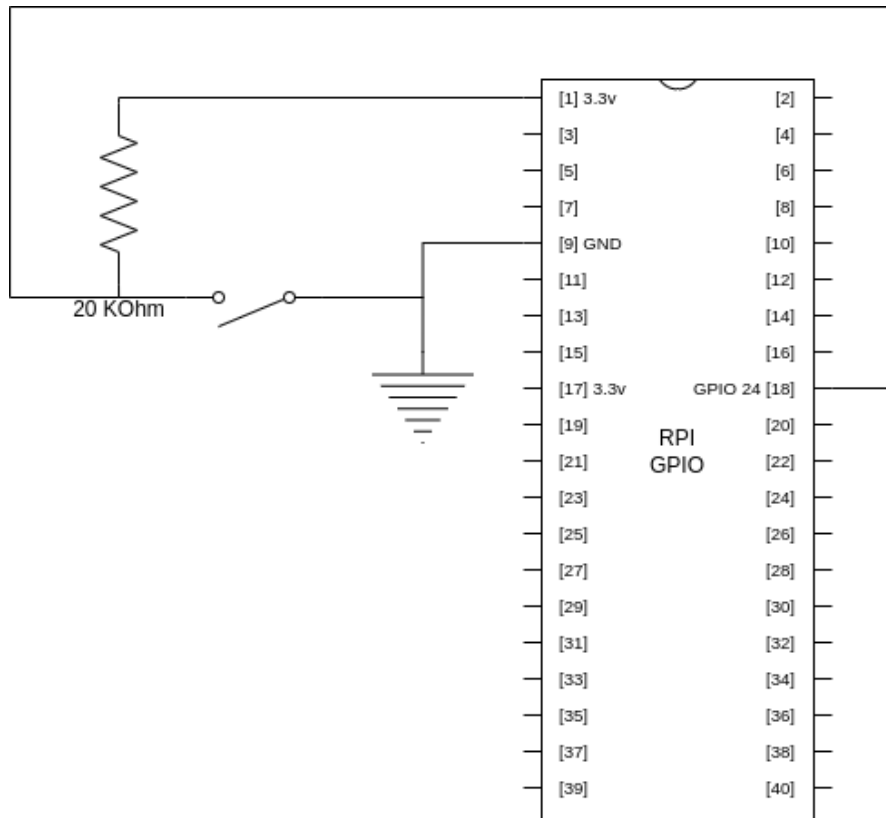


Fig. 3.7 - Diagrama del circuito GPIO (Botón de acción)

4. Desarrollo del software

4.1 Descripción general

La aplicación que forma la base de este proyecto ha sido desarrollada en Java de forma modular, utilizando para ello el entorno de desarrollo IntelliJ Idea de JetBrains.

Una clase central, «SystemCore» (o Núcleo del sistema) se encargará de la gestión global de los procesos. Esta dispondrá de una serie de servicios que le asistirán en su tarea:

- «CameraService» (Servicio de la cámara): se encargará de la gestión de la cámara en sí, como obtención de nuevas imágenes, comparación mediante visión por computación con la captura anterior...
- «HardwareService» (Servicio del Hardware): Se encarga de la gestión de aquello que tiene que ver con actuaciones físicas sobre el sistema. En este caso, monitoriza la actuación sobre un botón, que se comportaría como un timbre.
- «CommunicationService» (Servicio de Comunicaciones): se encarga de la gestión de las comunicaciones con usuario: recibir órdenes, enviar alertas y/o imágenes...
- "AudioService" (Servicio de Audio): se encarga de la gestión del micrófono. Detecta e informa si hay algún sonido fuerte cerca del dispositivo.

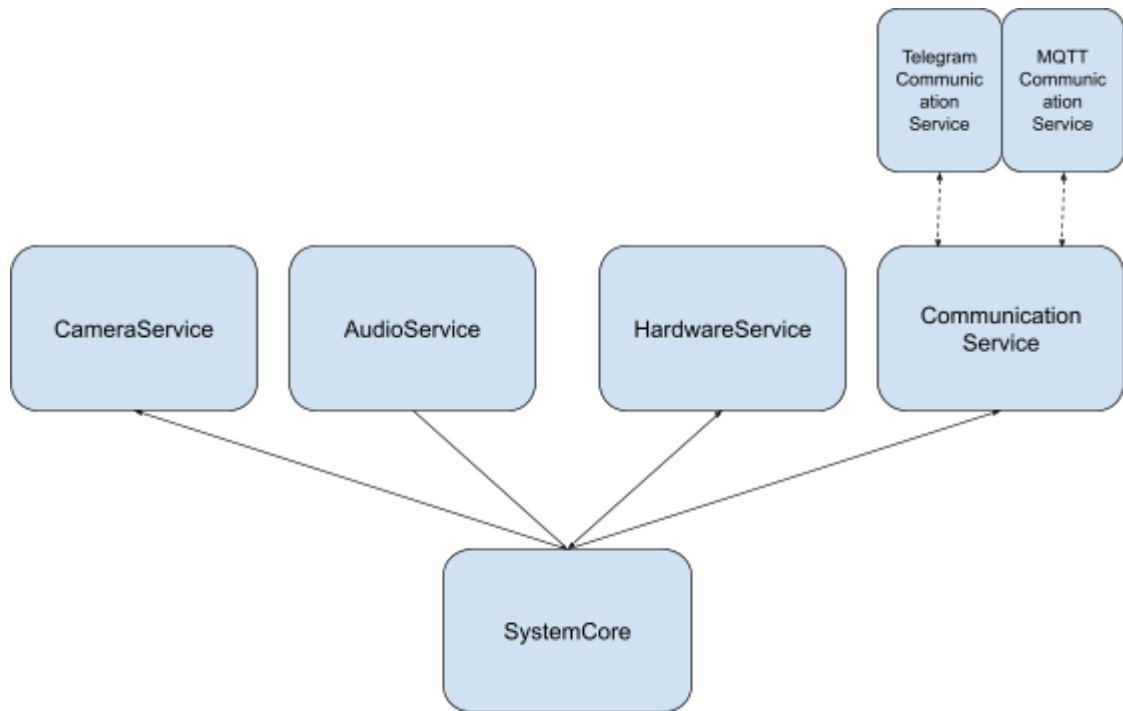


Fig. 4.1 - Diagrama de clases del sistema

La clase «SystemCore» también recoge las propiedades básicas de la aplicación, tales como el tiempo entre alertados, el tiempo de retardo entre la obtención de distintas imágenes, la sensibilidad a la hora de generar alertas, etcétera. Todas estas propiedades podrán ser posteriormente modificadas por el usuario.

En esta clase, cuatro métodos están preparados para ser invocados con una frecuencia marcada por una propiedad, definida inicialmente en 0,5 segundos:

- Un evento que se encarga de analizar la última imagen disponible y compararla con la anterior, para detectar posibles movimientos y notificar la alerta al usuario.
- Otro evento que se encarga de comprobar si ha saltado en el último periodo una alerta por sonido, para notificar en caso positivo al usuario.
- Un tercer evento que se encarga de revisar si se ha accionado el botón en este último periodo, para notificar en caso positivo al usuario.
- El último evento se encarga de comprobar si el usuario ha comunicado de alguna forma una petición, como obtener nueva imagen, desactivar alguna de

las alertas y/o modificar la sensibilidad del sistema; y actuar de forma correspondiente.

Cada uno de estos métodos se centran en comprobar si alguno de los cuatro servicios presentes en el sistema requiere de acción por el usuario, para luego enviar la orden al Servicio de Comunicaciones.

Dentro del Servicio de Comunicaciones se ha desarrollado una funcionalidad que, además de enviar mensajes, permite al usuario la petición de órdenes específicas a la cámara. Entre las órdenes que se han considerado podemos encontrar:

- Solicitar una imagen actual del sistema.
- Activar/desactivar el envío de alertas por movimiento, por sonido fuerte o ambas.
- Ajustar el tiempo de separación mínima entre los distintos tipos de alertas.
- Ajustar la sensibilidad del movimiento a partir del cual debe realizarse la foto.

Para ello, se ha construido una implementación con Telegram, una de las principales redes de mensajería en España y que permite su integración con desarrollos de software a través de las APIs diseñadas para tal uso.

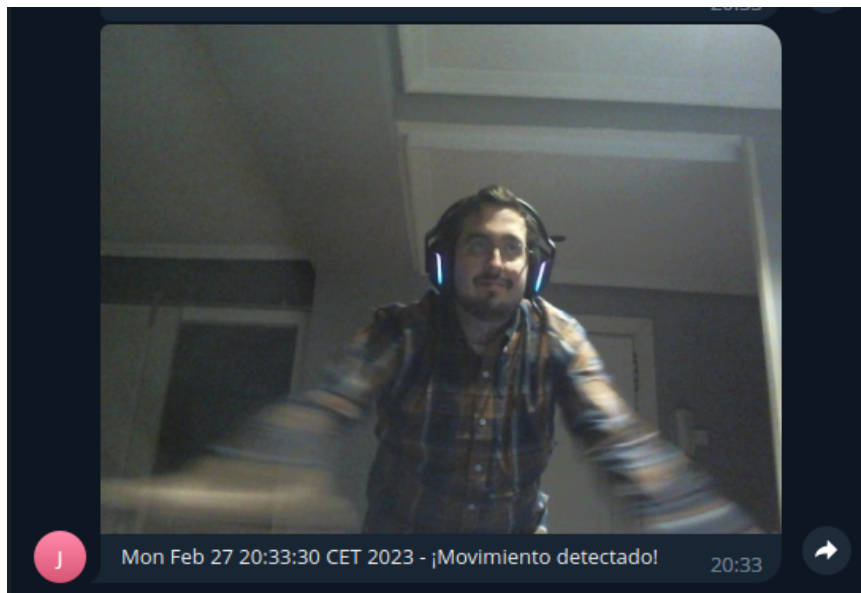


Fig. 4.2 - Ejemplo de alerta al usuario por movimiento por cámara

Además, también se ha diseñado el sistema para la implementación de otra interfaz de comunicación con MQTT, popular sistema de mensajería por colas, utilizado popularmente en el conocido "*Internet of Things*" (Internet de las Cosas). Esto permitiría la integración directa de la cámara con otras redes de domótica que empleen este sistema de mensajería.

El Servicio de Cámara se ha basado en la popular librería de visión por computador *OpenCV*. El sistema permite capturar y almacenar un histórico de dos imágenes por separado, y compararlas a través de los histogramas con el método de comparación de Chi cuadrado. Para esto, se ha dejado configurable por el usuario el nivel de sensibilidad a partir del cual esta comparación dará un resultado positivo o negativo. Esta captura una nueva imagen de la cámara cada 0,5 segundos, configurable desde la aplicación pero no por el usuario.

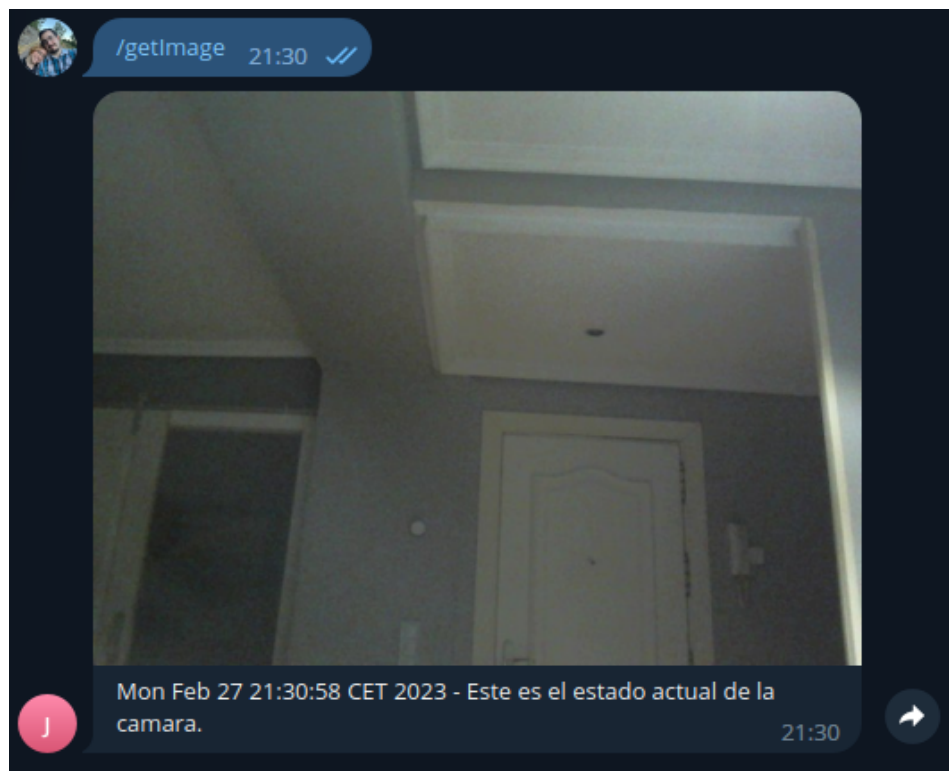


Fig. 4.3 - Ejemplo de petición por parte del usuario de imagen actual.

El Servicio de Audio, sin embargo, trabaja de manera constante para poder obtener el nivel de sonido en tiempo real. El sistema muestrea la señal del micrófono a una frecuencia de 44.100 Hz (estándar común en las grabaciones de audio), y procesa la información obtenida cada vez que se obtienen 4096 muestras (cada 0,1 segundos aproximadamente).

Si el nivel medio de las muestras en ese periodo (volumen de audio) supera una sensibilidad definida previamente, se guarda el aviso de que ha saltado la alerta para que el Núcleo de la Aplicación pueda operar a partir de este aviso. De forma análoga al Servicio de Cámara, esta sensibilidad es configurable por la aplicación, pero no es accesible por el usuario.

Por último, el Servicio de Hardware también está trabajando de manera continua analizando si se ha pulsado el botón externo. Para ello estudia el voltaje recibido en uno de los pines de la GPIO de la SBC, tal y como se ha indicado en el apartado 3.2. Cuando detecta que el voltaje aumenta en este pin, produce un aviso que el Núcleo podrá recoger de manera acorde. Se han contemplado los posibles efectos eléctricos de rebote con 3 milisegundos de tiempo de recuperación.

4.2 Diseño y Diagramas

El software utilizado se encuentra íntegramente en los anexos. El sistema se compone de los cuatro flujos en paralelo, programados para ejecutarse con una frecuencia de 0,5 segundos cada uno. El programa, una vez compilado, ocupa tan solo 2179 bytes en su totalidad.

Tras el periodo de carga inicial, el sistema se estabiliza en una ocupación de alrededor del 30% de la memoria RAM (unos 300 MB), un consumo de CPU que varía entre el 30% y un 200% (teniendo en cuenta que este hace referencia al consumo por núcleo, por lo en los momentos de mayor actividad estaría utilizando los dos núcleos de la CPU).

Con respecto a la latencia de los cuatro flujos de procesado, se han registrado un número de muestras amplias (estas están disponibles también en los anexos),

obteniendo unos tiempos medios de 15 milisegundos para la comprobación de movimiento y del orden de microsegundos para el resto de flujos (comprobación de audio, de pulsación de botón y de orden enviada por el usuario a través de los sistemas de mensajería). En caso de la verificación de imagen de un positivo, sin embargo, los tiempos superan el medio segundo, lo cual significa que puede provocar retrasos en el siguiente análisis al ser mayor que el propio periodo de frecuencia. Esto se detallará y se analizará en el siguiente capítulo de análisis.

A continuación se muestran unos diagramas con el funcionamiento general del sistema con cada uno de sus flujos y clases, de manera orientativa:

- En la figura 4.4 podemos encontrar el diagrama general de dependencias del sistema, con el funcionamiento global de la aplicación.
- En la figura 4.5 se detallan de manera más específica el flujo relacionado con el análisis de audio, separando el flujo perteneciente al núcleo de la aplicación y al servicio de audio.
- En la figura 4.6 encontraremos detallado los flujos de análisis de imagen y de alertado por hardware (es decir, pulsación de botón).

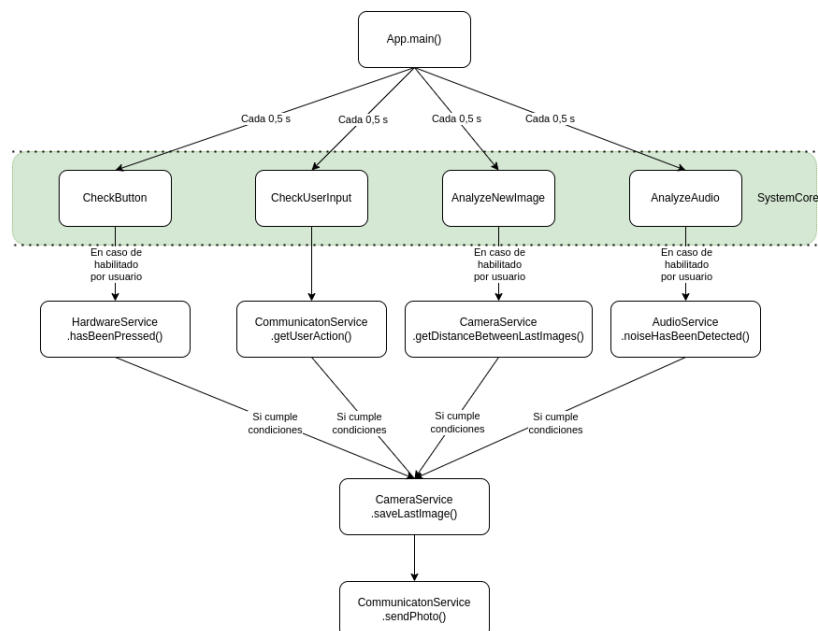


Fig. 4.4 - Diagrama general de conexiones del sistema.

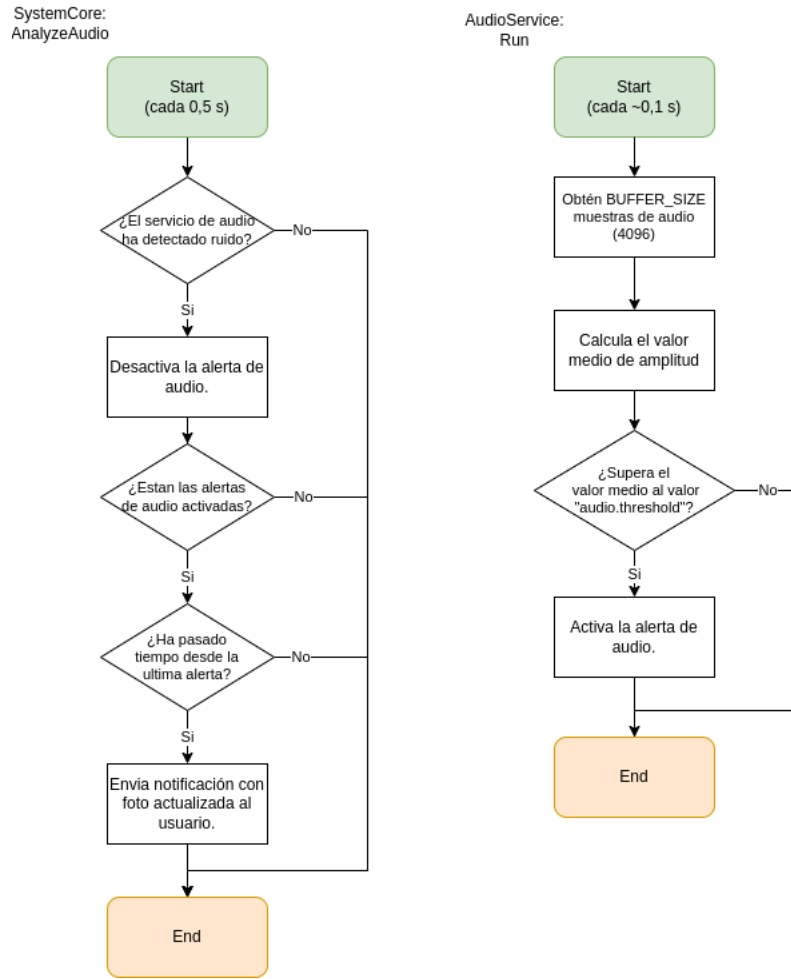
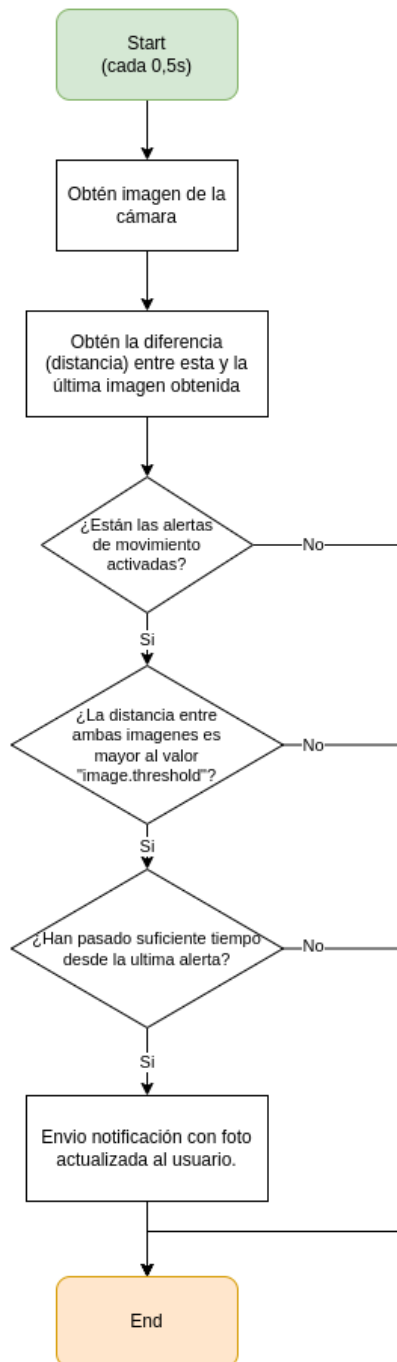


Fig. 4.5 - Diagrama de flujo del sistema de alerta por audio.

SystemCore
AnalyzeNewImage



SystemCore
CheckButton

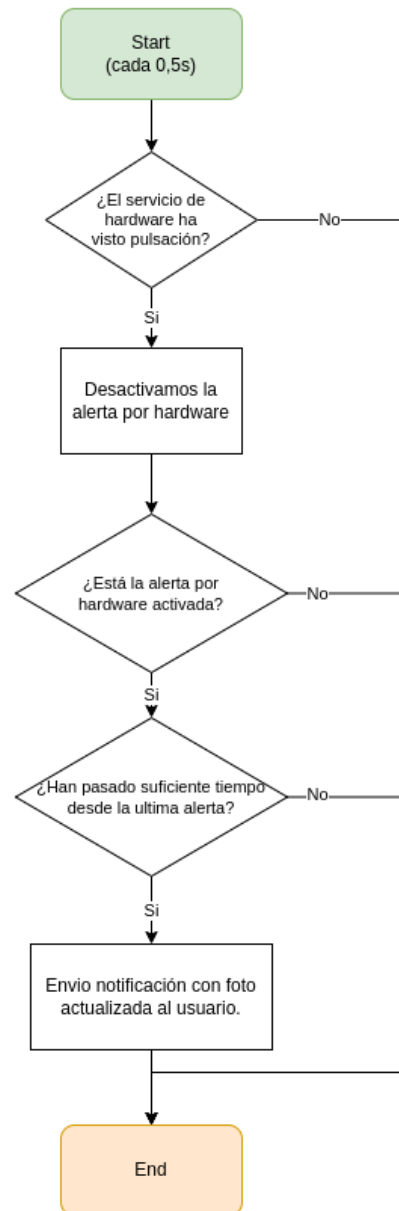


Fig. 4.6 - Diagrama de flujo de los sistemas de alerta por movimiento y hardware.

5. Análisis del prototipo

El sistema completo desarrollado, hardware y software, es plenamente funcional. En este capítulo analizaremos su funcionamiento y expondremos algunos puntos a resaltar, como problemas detectados y posibles soluciones y mejoras al prototipo inicial.

5.1 Análisis de problemas

A la hora de realizar el análisis de audio, hay que registrar la señal de audio y compararla con un nivel de referencia. Inicialmente, se había planteado medir decibelios (dB), ya que es la unidad de medida usada para expresar el nivel de potencia o intensidad de sonido. Sin embargo, se comprobó que esta unidad es relativa a un punto de referencia estándar de volumen de sonido, y definir este punto de referencia para el proyecto se volvió complicado. Esto ocasionó que obtener e interpretar lecturas de audio se volviera la parte más compleja del desarrollo.

Aunque existen formas de convertir esta señal de decibelios a otra, como a través de los dBu (con una referencia a una tensión fija, 0,77 V) o los dBm (con referencia a una potencia fija: $1 \text{ mW} = 0 \text{ dBm}$), finalmente se decidió descartar este sistema y aplicar un método más práctico.

La señal de audio se basa en una lectura de puntos de la onda de audio a partir de la vibración recibida por la membrana del micrófono. Esta señal tiene una amplitud máxima definida a partir de la señal que devuelve el micrófono al SBC.

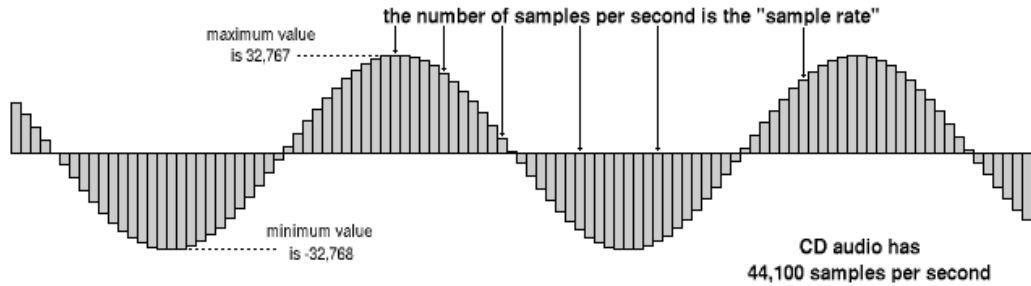


Fig. 5.1 - Representación de una señal de audio digital.

Imagen de Graham Mitchell [11]

Se decidió utilizar esta forma de la señal para obtener el nivel de audio con referencia a una amplitud media de las muestras tomadas. Un sonido fuerte genera un pico en la señal que debería detectarse, en muestras pequeñas, como un aumento de la amplitud media de la señal. Esta medida debe hacerse con muestras pequeñas para evitar que la amplitud media se vea afectada por demasiados golpes sonoros consecutivos, y/o por efectos físicos de la propia onda de sonido. Tras diversas pruebas se decidió que el análisis fuera sobre 4096 medidas, tal y como se comentó en el apartado de audio.

Otro de los problemas que se han encontrado es la generación de falsos positivos en el reconocimiento de imagen; en este caso por el uso de una cámara antigua que de vez en cuando necesita enfocar para capturar correctamente la imagen. Este proceso de enfoque y desenfoco puede generar que el histograma de la imagen sea muy distinta a la anterior por el simple hecho del cambio de brillo, lo cual genera una alerta al usuario.

Sería, por lo tanto, interesante estudiar un cambio en la forma de detectar el alertado a partir de imagen. Se estudian dos formas:

- El empleo de un sensor de presencia como complemento a la cámara. Si se establece que sea el sensor de presencia el que detecte el movimiento y, posteriormente, se use la cámara para capturar la imagen, evitamos posibles ruidos por desenfocos. Además, evitaríamos el empleo continuo de la cámara, lo cual representaría una mejora en cuanto a consumo. Esta opción sería la más correcta con respecto a la legislación estudiada en el apartado 2.6.

- El uso de sistemas de visión por computador para detectar movimientos. Esta opción permitiría tener una pequeña información de qué es lo que está detectando realmente el movimiento (sea un animal, un adulto, un niño) y así evitar dar alertas por otro tipo de movimientos, como el de un robot de limpieza, un juguete móvil, etcétera.

5.2 Aplicación de visión por computador en la cámara

Se ha creado un parámetro de configuración que permite la activación y desactivación del sistema de visión por computador, de forma que se pueda desactivar en caso de usar una SBC de menor potencia o para ahorro de batería cuando su uso es desconectado de la red eléctrica. Así, se ha implementado la siguiente funcionalidad: cuando se detecte algún movimiento en la cámara, se analizará la imagen para saber qué elemento ha podido generar el movimiento y, sólo en caso de coincidencia con una cara (detectar una persona frente a la cámara), dar la alerta al usuario.

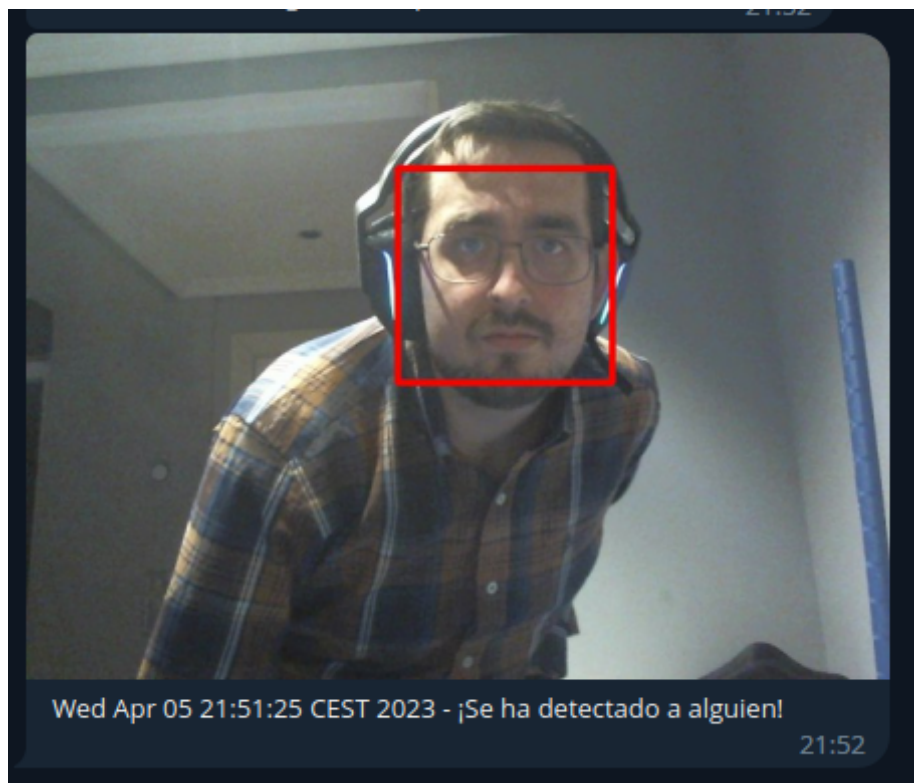


Fig. 5.2 - Sistema de videovigilancia alertando de una presencia.

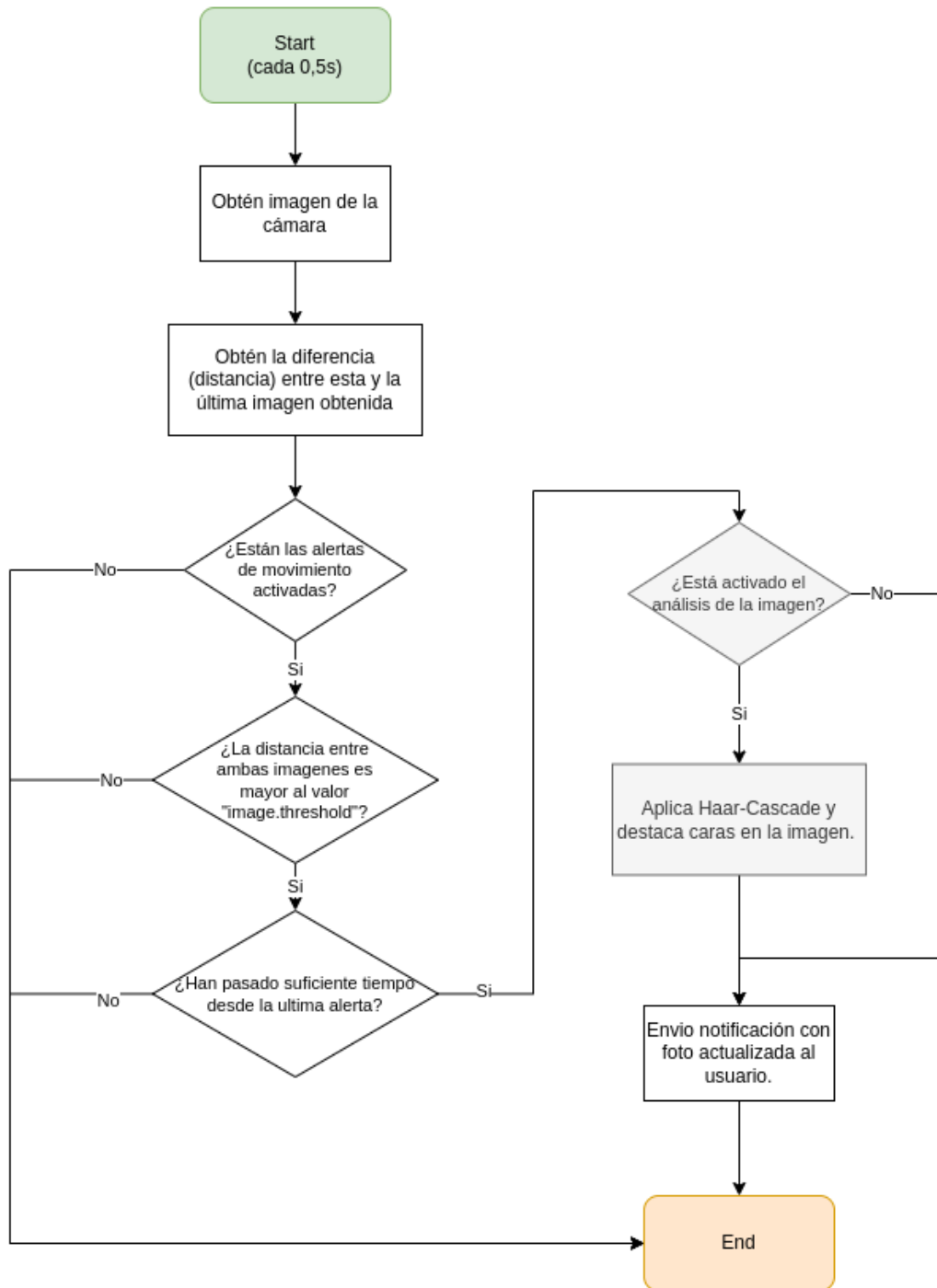


Fig. 5.3 - Diagrama de flujo de alertado por imagen.

Los cambios con respecto a la fig. 4.6 (Visión por computador) se aprecian en gris.

Para ello, se ha implementado un sistema de clasificación de imagen por Haar Cascade, basado en el uso de clasificadores previamente entrenados para encontrar patrones en la imagen. De esta forma, se analiza la imagen desplazando el mismo por toda la imagen, buscando una posible correlación con el clasificador. Este procedimiento tiene la ventaja de ser un sistema rápido y eficiente de análisis de imagen, al partir de cálculos matemáticos sencillos de realizar, algo útil en sistemas como un SBC.

Los tiempos de cálculo con y sin análisis de imagen se recogen en el Anexo 3, pero un resumen del resultado final sería el siguiente: solo con la diferencia de movimiento, el tiempo de cálculo medio es de 649 ± 134 ms, mientras que activando el sistema de análisis de imagen sube a 1250 ± 126 ms. A partir de estos datos podemos llegar a las siguientes conclusiones:

- La potencia de la Raspberry Pi 3B+ ya es justa para el procesamiento de estas imágenes, al costarle más de un segundo procesar una imagen para comprobar si debe lanzar una alerta. Se había configurado que el tiempo entre imagen fuera de medio segundo, pero al tardar tanto, debería aumentarse al menos al segundo (o más). Si se deseara usar una SBC menos potente (como las ya mencionadas Zero W o la A+) es posible que el tiempo deba aumentarse o bien usar otro sistema de alerta (sensor de presencia) en vez del basado en procesado de imagen.
- El tiempo que tarda el dispositivo en obtener la imagen y hacer el cálculo de movimiento también es elevado (0,6 s); esto es posible que se arregle al utilizar una de las posibles cámaras integradas por GPIO en vez de una cámara por USB que añade cierto procesamiento de la imagen.
- Hemos empleado uno de los sistemas de análisis más eficientes energéticamente hablando. Pero estos tiempos implican que el sistema estaría trabajando de manera continua procesando siempre la imagen, por lo que el uso de estos sistemas de visión por computador se vuelven, en apariencia, incompatibles con un objetivo de ahorro energético del sistema para su uso desenchufado de la red eléctrica. Por lo tanto, se podrían plantear dos estrategias diferentes: aplicar el análisis de la imagen si el dispositivo está

enchufado a la corriente y no hay problemas de consumo, y otro basado en un sensor de presencia en el caso de estar funcionando por batería y/o pilas.

- Por último, se han podido comprobar falsos positivos con este sistema, en situaciones en las que no se detectan caras correctamente o en posiciones erróneas. Esto puede deberse tanto a un entrenamiento erróneo del clasificador como a la baja fiabilidad del sistema (al fin y al cabo, se ha elegido uno de los sistemas de análisis más sencillos para poder aprovechar el uso de sistemas de baja potencia como los SBC).

5.3 Consumo energético del prototipo.

En este apartado vamos a estudiar el consumo del prototipo desarrollado y evaluar la posibilidad de su uso desconectado de la red eléctrica, proponiéndose mejoras de cara a un posible futuro producto comercial.

Como ya se comentó en el apartado 3.3, se ha estimado que el consumo del prototipo es, de manera continua, de 3,8 W. Para el presente trabajo, este consumo afectaría de manera considerable a la capacidad de esta cámara para su uso desconectado de la red eléctrica. Una pila estándar tipo AA tiene entre 1500 y 3000 mAh de capacidad, dependiendo de su química: alcalinas, de aleaciones níquel-cadmio (NiCd) o níquel-metal hidruro (Ni-MH), por poner algunos ejemplos. Pero esta capacidad además se ve afectada según la corriente que se demanda de la misma.

El consumo del prototipo (3,8 W) se ha medido a una tensión fija de 5 V (el voltaje estándar suministrado por el puerto USB), lo cual representaría una corriente fija de 760 mA. Si suponemos un 5% de pérdidas en la transformación de voltaje y componentes pasivos, la corriente sería de unos 800 mA.

Name	Duracell Ultra Power AA					
Cell	Duracell Ultra Power AA					
Supplier	Duracell web shop				Date:	11-2013
Size	Weight:	25.6 g	Length:	50.3 mm	Diameter:	14.1 mm
Test current (A)	0,1	0,2	0,5	1	2	3
Measured capacity (Ah)	2,819	2,378	1,723	1,242	0,869	0,67
Measured energy (Wh)	3,438	2,865	1,931	1,297	0,822	0,595

Fig. 5.4 - Valores de capacidad y corriente de la pila Duracell Ultra Power AA. [12]

Name	Eneloop AA BK-3HCC 2450mAh (Black)					
Cell	Panasonic BK-3HCC					
Supplier	Japan				Date:	2-2014
Size	Weight:	30.4 g	Length:	50.3 mm	Diameter:	14.4 mm
Info	Top:	SB	Bottom:	metal	Rated A:	
Test condition	Charge voltage:		NA	Termination:		-dv
Test current (A)	0,2	0,5	1	2	3	5
Measured capacity (Ah)	2,455	2,430	2,401	2,367	2,342	2,286
Measured energy (Wh)	3,115	3,068	2,982	2,848	2,735	2,529
PCB protection trip current (A)	NA					
Calculated internal resistance (ohm)	0,04					

Fig. 5.5 - Valores de capacidad y corriente de la pila Eneloop AA. [12]

La Duracell Ultra Power AA (Alcalina) tiene una capacidad altamente variable a partir de la corriente que se demanda (figura 5.4). Con una tensión fija de 1,5 V, la corriente que se extraería de la batería sería de ~2,7 A. Esta es una corriente muy alta para este modelo de pila: asumiendo una progresión lineal entre 2A y 3A de corriente en esta pila, nos daría una capacidad de 0,73 Ah, o lo que es lo mismo, unos 16 minutos.

El uso de dos pilas reduciría la corriente demandada a la mitad por cada pila (1,35 A), lo cual permitiría aumentar su vida útil: usando la misma progresión lineal obtenemos 1,11 Ah; unos 50 minutos. Igualmente, un tiempo medio de vida ridículo para una cámara que debería estar activa de manera continua.

El uso de otro tipo de pila como la Eneloop AA (De NiMH) nos daría una mayor durabilidad al no ser su capacidad tan dependiente de la corriente consumida, aunque el voltaje que ofrece es algo menor (de 1,2 V) (Figura 5.5). Con unos cálculos similares a los anteriores, tendríamos unos 48 minutos con una única pila, o 1 h 46 min con un sistema de doble pila.

Una alternativa sería el empleo de Power Banks (o baterías USB externas) para la alimentación. Por ejemplo, con una estándar de 10.000 mAh hablamos idealmente de unas 12,5 horas de vida, una vida mucho más adecuada, aunque forzaría al usuario a la recarga continua del dispositivo.

Dispositivo	Consumo	Duración con Pilas alcalinas (x2)	Duración con Pilas NiMH (x2)	Duración con Power Bank / Batería externa
3B+	3,7 W	50 min	1 h 45 min	12 h
Zero W (estimado)	~ 0,9 W	7 h	6 h 30 min	2 d 7 h
A+ (estimado)	~ 0,6 W	11 h 45 min	9 h 45 min	3 d 11 h

Fig. 5.6 - Consumos y duraciones de batería por SBC.

Estudios ya existentes [14] de la propia creadora de la SBC, Raspberry Pi Foundation, muestran que nuestro dispositivo tiene un consumo de entre 2,3 y 5,6 W, según si está en reposo o en momentos de actividad. Pero existen SBC con menos consumo: en la misma comparación vemos qué modelos como el A+ (con un consumo de entre 0,5 y 0,7 W), o la Zero W (de entre 0,7 y 1 W) reduciría muchísimo el consumo, pero también tendría el efecto de una menor potencia computacional, lo cual imposibilitaría aplicar lo estudiado en el apartado 5.2.

En la tabla de la figura 5.6 se muestra una comparativa con estas dos SBC, estimando un consumo elevado pero no máximo, de manera similar al consumo obtenido en la Raspberry Pi 3B+ con respecto a los resultados vistos en el estudio previamente mencionado.

Así, podemos comprobar el efecto del elevado consumo que tiene la placa usada en las pilas alcalinas, ya que en el caso de las otras dos SBC vemos que la duración estimada es mayor que para las pilas NiMH. Además, vemos que el uso de la Raspberry Pi A+ sería más conveniente, en términos puramente de consumo/duración de baterías, puesto que representa horas o incluso días más de duración en el caso de uso de Power Banks. Sin embargo, para el desarrollo de un producto comercial, debería estudiarse la conveniencia de si debería utilizarse una u otra, teniendo en cuenta la ventaja del menor tamaño de la Zero W con respecto a la A+.

Dispositivo	Consumo	Duración con Pilas alcalinas (x2)	Duración con Pilas NiMH (x2)	Duración con Power Bank / Batería externa
3B+	2,5 W	1 h 50 m	2 h	20 h
Zero W (estimado)	~ 0,7 W	9 h 30 min	8 h 15 min	2 d 23 h
A+ (estimado)	~ 0,5 W	15 h 45 min	11 h 40 min	4 d 4 h

Fig. 5.7 - Consumos y duraciones de batería posoptimización por SBC.

Un posible punto de mejora sería intentar optimizar el código en la medida de lo posible para reducir el consumo. Mediante reducción de los hilos de ejecución y optimización del bucle de comprobación de cierre de la aplicación podemos reducir el consumo de manera considerable (en pruebas efectuadas se estabilizaba sobre los 2,4 W, 2,5 W sumando un 5% de pérdidas), lo cual también nos daría una ventaja interesante en tiempos de vida, como apreciamos en la tabla de la figura 5.7.

De esta manera, podemos alcanzar tiempos más adecuados para el caso de un uso ocasional de la cámara (por ejemplo, vigilar a la mascota o a un niño durante unas horas). Para un empleo continuo de la cámara (por ejemplo, si se usa como sistema de vigilancia durante las vacaciones) seguiremos limitados, en cuyo caso el sistema tendría que estar conectado a la red eléctrica.

Finalmente, pruebas extras realizadas sobre el puerto USB muestran que el empleo de la cámara produce un consumo eléctrico que ronda el vatio, una cantidad importante con respecto al total consumido por el SBC. Este consumo se podría reducir si la cámara no estuviera activa en todo momento y/o si usáramos una cámara de mayor eficiencia; como los propios módulos integrados de Raspberry Pi Smart Camera que consumen sobre una cuarta parte (250 ms) [13].

5.4 Uso de sensores de presencia por infrarrojos

En los apartados anteriores hemos visto limitaciones importantes del sistema derivado del uso de la cámara de manera continua:

- En el 5.1 se ha comentado la existencia de falsos positivos por el empleo de un sistema de cámara antiguo que requiere reenfoque con frecuencia.
- En el 5.2 se ha visto la limitación del sistema debido al tiempo de procesado de la imagen, el cual provocaba que el sistema tardara de manera continua unos 650 ms en analizar si ha habido movimiento y alertar.
- En el 5.3 hemos comentado que el coste energético de mantener la cámara encendida todo el tiempo es importante con respecto al total consumido por el micro.

Todo esto lleva a una conclusión relevante: la conveniencia de que el alertado de movimiento no lo realice la cámara, sino algún otro sensor. Para ello se ha decidido incluir un sensor PIR (*Passive Infrared Sensor* o Sensor Pasivo de Infrarrojos). Estos sensores, aunque no permiten detectar qué es lo que produce el movimiento, sí que detecta movimiento próximo al sensor. Así, podemos replantear el sistema de alertado de la siguiente forma:

- Cada cierto tiempo (en el caso de las pruebas efectuadas, cada 0,5 segundos) se comprobará si el sensor ha detectado movimiento.
- En caso positivo, se activará la cámara para capturar y analizar la imagen.

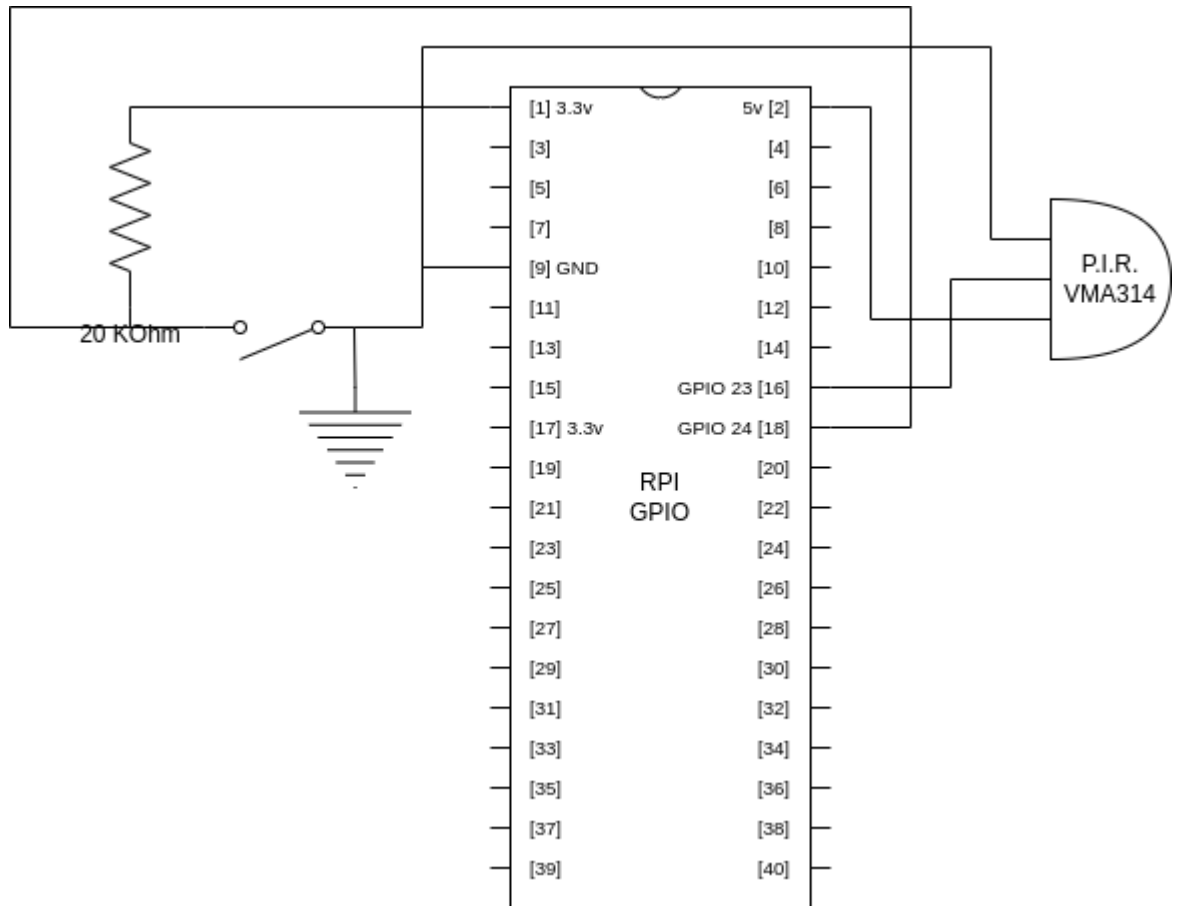


Fig. 5.8 - Diagrama eléctrico del sistema con el PIR

Para ello, se ha empleado el sensor VMA314 (figura 5.9) instalado en otro puerto del GPIO de la Raspberry Pi. Este sensor permite ajustar, mediante potenciómetros, el tiempo de medición y retardo entre alerta y alerta como la distancia máxima en la que puede detectar el movimiento. Concretamente, los rangos son de 5 a 200 s en el retardo de alertas y de 3 a 7 m en la distancia de medición [15].

Con el sensor, en los casos en los que se ha detectado movimiento y ha sido necesario utilizar la cámara, el tiempo de procesado sigue siendo alto (679 ± 79 ms). Sin embargo, esta implementación ha permitido que los tiempos sean muchísimo menores cuando el PIR no detecta movimiento alguno: 34 ± 9 μ s, tres órdenes de magnitud menos al no necesitar capturar y analizar la imagen desde la cámara. Esto debería permitir (con cámaras integradas) activar y desactivar la cámara a voluntad, bajando así el consumo considerablemente y capturando imagen solo cuando sea estrictamente necesario para el objetivo del sistema; cumpliendo así de manera holgada el RGPD.



Fig. 5.9 - Foto del sensor PIR, modelo VMA314

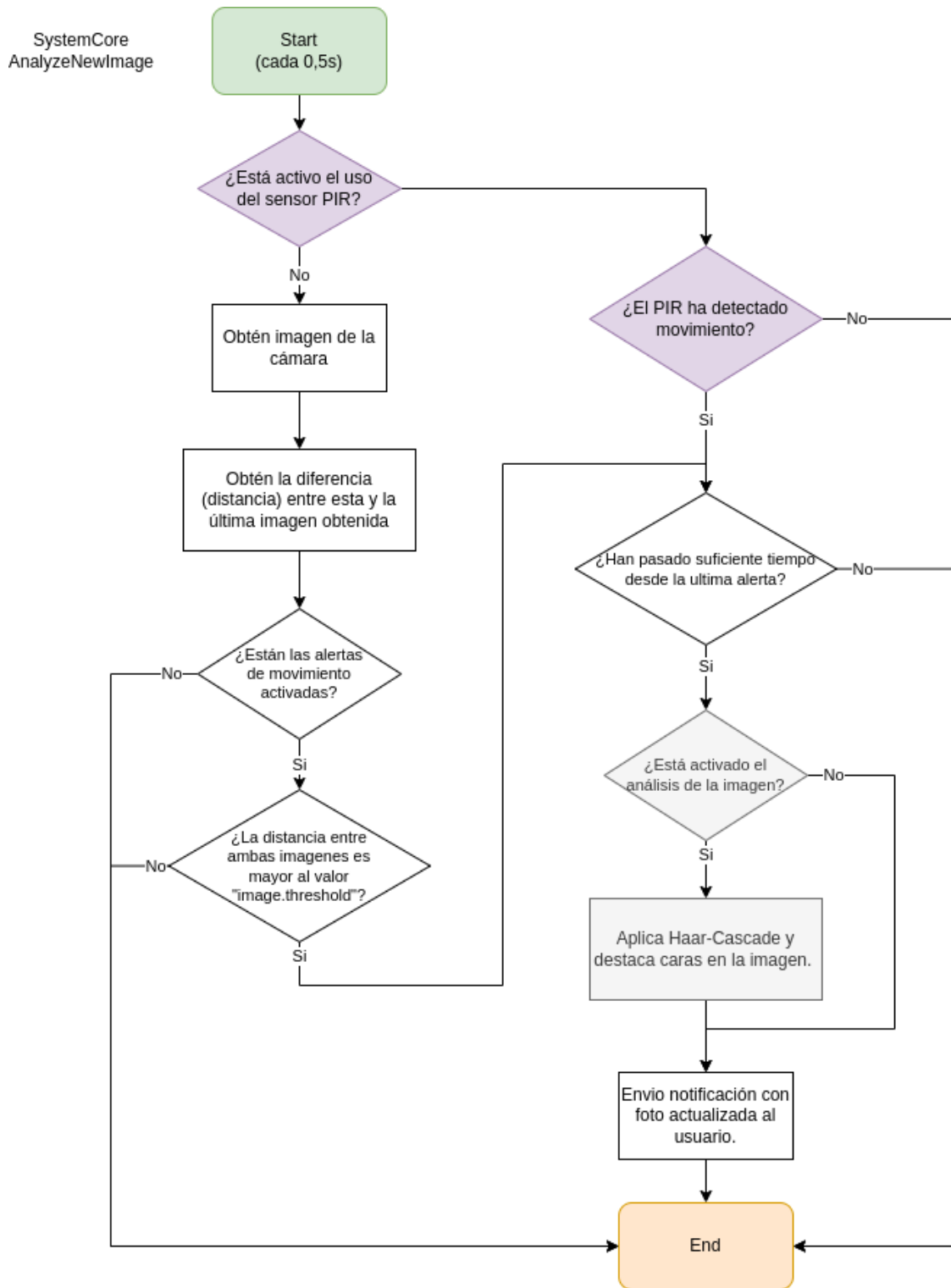


Fig. 5.10 - Diagrama de flujo de alertado por imagen.

Los cambios con respecto a la fig. 5.3 (Adición del PIR) se aprecian en morado.

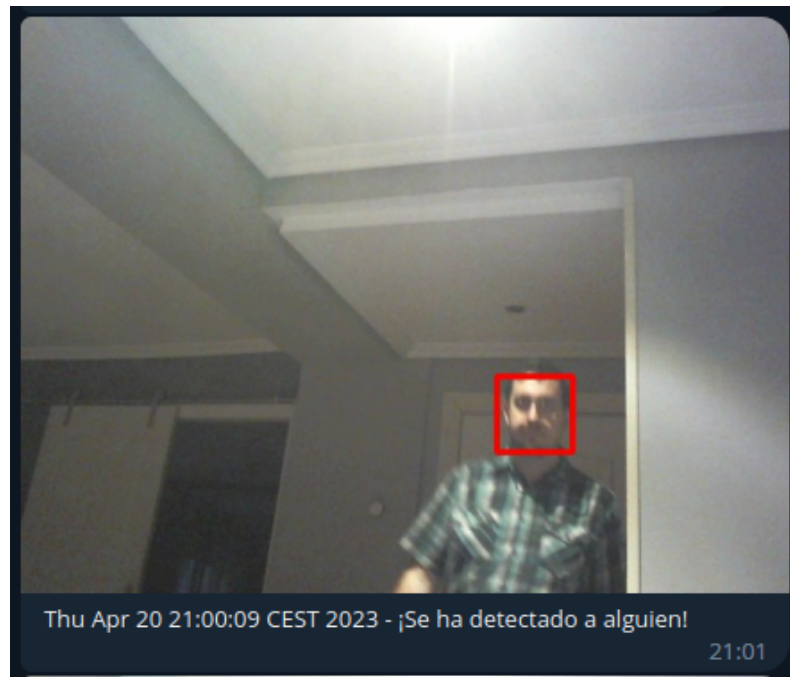


Fig. 5.11 - Captura de alertado por movimiento detectado con PIR, imagen analizada por Visión por Computador

Otra ventaja de este sistema ha sido eliminar por completo las falsas alertas. Tras un proceso de afinado para alcanzar la sensibilidad adecuada con respecto a la distancia de monitorización, el sensor PIR muestra, al no tener necesidad de enfocar y distorsionar por ello su análisis, más fiabilidad que el sistema basado en la cámara. Con esto, podemos concluir que el empleo de un sensor PIR resulta ventajoso en todos los sentidos.

5.5 Mejoras propuestas de hardware

En este apartado planteamos algunas posibles mejoras en cuanto al hardware de nuestro prototipo en el caso de querer obtener un sistema comercializable.

Por un lado, como ya hemos comentado, el consumo total es de 2,5 W de manera estable tras las mejoras expuestas. Este consumo no es muy elevado con respecto a otros sistemas (por ejemplo, la videocámara de seguridad «Advisen Homecam2 360» presenta un consumo de 4,5 W [16]), pero igualmente es un consumo alto para poder plantear la independencia del sistema de la red eléctrica mediante un sistema de

baterías/pilas. Esto, como hemos analizado en el apartado 5.3, podríamos conseguirlo utilizando alguna SBC menos potente, bajando los tiempos entre análisis de imágenes/audio, desconectando la cámara entre usos o intentando optimizar más aún el consumo del sistema. De todas formas, esto puede entrar en conflicto con algunas de las mejoras que se presentan en el apartado de software que sí que requerirían potencia de procesamiento.

Una cosa a tener en cuenta de esta videocámara de seguridad «Advisen Homecam2 360» son los extras en funcionalidades que posee, como la capacidad del análisis del entorno en 360°. Esto se podría realizar, por ejemplo y como una segunda propuesta de mejora, conectando dos motores servo al conjunto.

Una tercera mejora posible sería el empleo de alguna cámara más integrada en el sistema en vez de una cámara USB estándar como la del prototipo analizado. Esto permitiría bajar el coste e incluso plantear una producción en cadena del sistema, además de, quizás, mejorar los tiempos de obtención de imágenes para su posterior análisis. Posibles opciones serían las ya planteadas previamente en el apartado 2.2 como los módulos oficiales de Raspberry Pi Foundation; los cuales poseen además de mejor resolución (y por ende mayor fidelidad de la imagen) que la cámara usada. Habría que tener en cuenta que este tipo de módulos no incluyen micrófono, lo cual obligaría a la instalación de un micrófono independiente en el sistema, teniendo que incluir además circuitería extra, como filtro y amplificador. Si se quisiera combinar esta mejora con la anterior encontraríamos un pequeño reto, teniendo que filtrar el sonido generado por los servos en la señal del micrófono para evitar perder funcionalidad.

5.6 Mejoras propuestas de software

En cuanto al software, existen una serie de posibles mejoras detectadas durante el desarrollo, algunas de gran envergadura, que requerirían la realización de un segundo trabajo fin de grado.

En el apartado 5.2 se ha utilizado una técnica de análisis de imagen (en el caso que nos atañe, para reconocer solo caras). A partir de esto, se podría desarrollar un reconocimiento de imágenes para saber qué es lo que está produciendo el sonido, reconociendo a las diversas personas/animales presentes en la imagen para tan sólo alertar si el movimiento ha sido generado por elementos desconocidos o si lo ha generado algo en concreto (como un bebé).

También, y de manera combinada con el uso de servos para permitir un movimiento mecánico de la cámara, se podría integrar en el programa un sistema que permita reconocer hacia dónde se dirige el movimiento y movilizar la cámara en esa dirección, permitiendo así el seguimiento continuo del objeto. También con el empleo de dos micrófonos separados podría realizarse un sistema que reconociera (aproximadamente) dónde se ha generado el sonido, de forma que la cámara dirigiera su foco en esa dirección automáticamente.

Por último, también se podría plantear una mejora en el sistema de comunicación. Durante el desarrollo de este servicio sólo se ha planteado el uso de la red de mensajería Telegram y la comunicación con sistemas de domótica mediante colas, preparando el camino para otras opciones como:

- Aplicaciones propias integradas para dispositivos móviles
- Creación de interfaz web en red local, o
- Integración con otras redes de mensajería de forma que no se ata al usuario a un único servicio.

6. Conclusiones finales

En este proyecto se ha diseñado partiendo de cero un prototipo de sistema de videovigilancia capaz de integrarse en sistemas domóticos y de alertar al usuario de cualquier movimiento y/o sonidos fuertes, y de mostrar imágenes actualizadas cuando el usuario lo considere necesario. Se ha demostrado que el desarrollo del mismo es viable haciendo uso de materiales y herramientas fácilmente accesibles por cualquier usuario, como una SBC Raspberry Pi y una webcam estándar. El prototipo desarrollado es totalmente funcional, pero en el caso de querer obtener un producto comercial habría que tener en cuenta todos los aspectos y mejoras descritos en el Capítulo 5.

La domótica en general es un sistema en auge, por la capacidad de ofrecer al usuario un control completo y flexible de lo que ocurre en su hogar, por la capacidad de ahorro de electricidad, gas y agua y por la seguridad y tranquilidad que estos sistemas pueden ofrecer al usuario. Es precisamente en este último punto en el que destaca este trabajo, al ofrecer estos términos, no solo en la seguridad exterior ante posibles intrusos y/o elementos indeseados, sino también en la interior, en la monitorización de infantes, animales, etcétera.

El diseño de un sistema modular de software permitiría la incorporación de otras funcionalidades con relativa sencillez, como las presentadas en las mejoras propuestas de software (apartado 5.6) que permitirían dar una potencia mayor al proyecto. Así mismo, hay opciones para reducir el coste del mismo, lo cual permitiría el desarrollo de una versión final que fuera competitiva en términos de mercado.

El haber definido los parámetros de sensibilidad como configurables y accesibles al usuario, permite que este pueda ajustar el nivel de alerta: desde la activación de todos los sistemas, pasando por el uso de sensores de presencia para optimizar las alertas hasta incluso desactivar el sistema de imagen y activar solo el audio (en situaciones en las que el audio fuera suficiente y/o hubiera problemas de privacidad).

Sin embargo, tendríamos algunos puntos negativos: la legislación vigente limita muchísimo aplicar este sistema de videovigilancia en exteriores, al no tener una capacidad de indicar zonas de no-controlado. Esta situación se arregla parcialmente con la instalación de sensores de presencia como el sensor PIR, que permiten detectar movimiento sin necesidad de capturar imagen, pero al seguir capturando posteriormente la imagen se debe tener especial cuidado con el uso de la misma.

Además de esto, el sistema de notificaciones diseñado fuerza al usuario a poseer un sistema completo de domótica en el que integrar la cámara, o a disponer de cuenta en una red de mensajería (quizás una de las más populares a día de hoy, pero seguiría siendo una red privada perteneciente a una empresa externa). Un trabajo futuro podría ser la implementación de más sistemas de comunicación y permitir al usuario la libre elección.

Finalmente, hay que destacar que en este trabajo nos hemos encontrado con el compromiso de potencia computacional frente a consumo de energía. El prototipo desarrollado se basa en un SBC barato pero con potencia suficiente para ejecutar algoritmos de visión por computador, pero a costa de aumentar el consumo de energía, limitando el tiempo de uso como cámara aislada desconectada de la red eléctrica. Para reducir el consumo y mejorar determinados aspectos, hemos tenido que incluir un sensor de presencia (infrarrojos), de bajo coste.

Pero si se deseara disponer de una cámara de videovigilancia desconectada de la red que permita ser alimentada con baterías durante un tiempo prolongado, se requeriría rediseñar el sistema usando otros sistemas de procesamiento, como un procesador embebido (microcontrolador). Sin embargo, al tener menos potencia computacional, las posibilidades de procesamiento de imagen en este caso serán muy limitadas.

En conclusión, el prototipo de sistema de videovigilancia doméstico desarrollado no solo permitiría al usuario aumentar su seguridad y tranquilidad en la vivienda, sino que demuestra que puede realizarse con poco coste añadido a componentes de los que puede ya disponer en casa.

7. Bibliografía

- [1] Real Academia Española, *Diccionario de la Lengua Española*. (2021) [Internet] [Consulta: Ene. 3, 2023]. <https://dle.rae.es/dom%C3%B3tico>
- [2] Raspberry Pi Foundation, *About the Camera Modules* [Internet] [Consulta: Mar. 11, 2023] <https://www.raspberrypi.com/documentation/accessories/camera.html>
- [3] Sun Microsystems, «Javasoftware entrega Java 1.0» *Javasoftware Ships Java 1.0* (1996) [Internet] [Inaccesible. Consulta de archivo: Mar. 13, 2023] <http://web.archive.org/web/20070310235103/http://www.sun.com/smi/Press/sunflash/1996-01/sunflash.960123.10561.xml>
- [4] Marvin Framework, *Introduction to Marvin Framework* [Internet] [Consulta: Feb. 20, 2023] https://marvinproject.sourceforge.net/en/tutorials/01_introduction/introduction.html
- [5] Ley Orgánica 3/2018, de 5 de diciembre, de Protección de Datos y garantía de los derechos digitales, art. 30.1. Boletín Oficial del Estado, 6 de diciembre de 2018, núm. 294 [Internet] [Consulta: Mar. 11, 2023]. <https://boe.es/buscar/pdf/2018/BOE-A-2018-16673-consolidado.pdf>
- [6] Agencia Española de Protección de Datos, *Guía sobre el uso de videocámaras para seguridad y otras finalidades* [Internet] [Consulta: Feb. 10, 2023] <https://www.aepd.es/sites/default/files/2019-09/guia-videovigilancia.pdf>
- [7] Raspberry Pi Foundation, *Raspberry Pi 3 Model B+* [Internet] [Consulta: Feb. 3, 2023]. <https://www.raspberrypi.com/products/raspberry-pi-3-model-b-plus/>
- [8] Raspberry Pi Foundation, «Raspberry Pi 3B+». [Internet] [Consulta: Mar. 2, 2023] <https://www.raspberrypi.com/products/raspberry-pi-3-model-b-plus/>
- [9] Logitech, «Galería de producto: Logitech C525» [Internet] [Consulta: Mar. 2, 2023] <https://support.logi.com/hc/es/articles/360025411433--Galer%C3%ADa-de-productos-HD-Webcam-C525>
- [10] Shelly Cloud, «Shelly 1PM» [Internet] [Consulta: Abr. 20, 2023] <https://kb.shelly.cloud/knowledge-base/shelly-1pm>

- [11] G. Mitchell, «Una introducción al Audio Comprimido con OGG Vorbis» *An Introduction to Compressed Audio with Ogg Vorbis* [Internet] [Consulta: Mar. 12, 2023] http://grahamitchell.com/writings/vorbis_intro.html
- [12] H. Kjær, «Test en batería para pilas AA y AAA (Alcalinas, de litio, NiMH)» *Battery test for AA and AAA (Alkaline, lithium, NiMH) batteries* [Internet] [Consulta: Mar. 17, 2023] <https://lygte-info.dk/review/batteries2012/CommonAAIndividualTest%20UK.html>
- [13] “The Camera Module adds about 200-250mA to the power requirements of your Raspberry Pi”. Raspberry Pi Foundation, «Hardware de la Cámara» *Camera Hardware* [Internet] [Consulta: Abr. 19, 2023] https://www.raspberrypi.com/documentation/computers/camera_software.html
- [14] L. Hattersley, «Especificaciones y comparaciones de la Raspberry Pi 3B+» *Raspberry Pi 3B+ Specs and Benchmarks*, The MagPi Magazine. [Internet] [Consulta: Mar. 9, 2023] <https://magpi.raspberrypi.com/articles/raspberry-pi-3bplus-specs-benchmarks>
- [15] Velleman, “PIR MOTION SENSOR FOR ARDUINO®” [Internet] [Consulta: Abr. 17, 2023] https://www.velleman.eu/downloads/29/vma314_a4v02.pdf
- [16] Advisen, Ficha de Producto de HomeCam2 360 [Internet] [Consulta: Mar. 13, 2023] <https://www.avidsen.com/es/data/127051-ficha-producto.pdf>