



**Universidad
Europea**

UNIVERSIDAD EUROPEA DE MADRID

ESCUELA DE ARQUITECTURA, INGENIERÍA Y DISEÑO

GRADO EN INGENIERÍA INFORMÁTICA

PROYECTO FIN DE GRADO

**INTEGRACIÓN DE SISTEMAS DE APRENDIZAJE
AUTOMÁTICO EN MODELOS COGNITIVOS:
UN CASO APLICADO AL ALZHEIMER**

JENNYFER GONZÁLEZ LABRADA

Dirigido por

DR. ESTEBAN GARCÍA CUESTA

CURSO 2019-2020

TÍTULO: Integración de sistemas de aprendizaje automático en modelos cognitivos: un caso aplicado al Alzheimer

AUTOR: JENNYFER GONZÁLEZ LABRADA

TITULACIÓN: GRADO EN INGENIERÍA DE INFORMÁTICA

DIRECTOR/ES DEL PROYECTO: DR. ESTEBAN GARCÍA CUESTA

FECHA: Junio de 2020

RESUMEN

El Alzheimer es una enfermedad mental que actualmente afecta a millones de personas en todo el mundo. Esta afección actúa de manera diferente en cada persona, al ser una enfermedad degenerativa, y a medida que avanza los pacientes tienen la necesidad de un mayor cuidado llegando incluso en la mayoría de los casos a necesitar una persona que les ayude a realizar sus actividades cotidianas.

En este proyecto se plantea una solución asistiva para que estos pacientes puedan mejorar su calidad de vida y tener más autonomía en su casa. Para ello se ha creado un sistema basado en el aprendizaje de las rutinas que realizan, o deben realizar, el cual es automático y se integra con un modelo cognitivo que representa la realidad. Este sistema es capaz de aprender la rutina diaria de un paciente y le avisa de cualquier anomalía con respecto a su rutina. La información del día a día del paciente se obtendría a través de una o varias cámaras ubicadas en su casa la cual sería analizada por el sistema para detectar esos comportamientos anómalos.

La aplicación de este sistema está enfocada a este caso de uso, pero puede también ser utilizado en otros contextos. En el caso concreto probado en este proyecto se ha ejemplificado tanto el entorno real como los eventos rutinarios. Para ello se han utilizado situaciones simuladas similares a las de una rutina de un paciente con Alzheimer y posteriormente se han generado datos sintéticos a partir de esas muestras de video reales para recrear un rango posible de temporalidad y duración de dichos eventos en situaciones reales. Finalmente, el sistema puede avisar al paciente de que no ha desayunado, o que ha tardado más de lo habitual, o incluso que no lo ha realizado a su hora determinada.

Palabras clave: Alzheimer, modelo cognitivo, visión artificial, detección de anomalías, aprendizaje automático, medicina asistiva

ABSTRACT

Alzheimer's is a mental disease that currently affects millions of people worldwide. This condition acts differently in each person, as it is a degenerative disease, and as progresses, patients need more care, even in most cases needing a person to help them carry out their daily activities.

In this project, an assisted solution will be needed so that these patients can improve their quality of life and have more autonomy at home. For this, a system based on learning the routines that they performed, or must be performed, has been created, which is automatic and integrates with a cognitive model that represents reality. This system is capable of learning the daily routine of a patient and alerts him of any anomaly regarding his routine. The day-to-day information of the patient will be obtained through one or more cameras located in his home which would be analyzed by the system to detect these abnormal behaviors.

The application of this system is focused on this use case, but it can also be used in other contexts. In the specific case tested in this project, both the real environment and routine events have been exemplified. For this, simulated situations similar to those of a routine of an Alzheimer's patient have been used and subsequently generated synthetic data from those real video samples to recreate a possible range of temporality and duration of said events in real situations. Finally, the system can notify the patient that he has not had breakfast, or that it has taken him longer than usual, or even that he has not done it at his certain time.

Keywords: Alzheimer's, cognitive model, artificial vision, anomaly detection, synthetic data, machine learning, assistive medicine

AGRADECIMIENTOS

Quisiera expresar mi agradecimiento a mi tutor Esteban García Cuesta por la gran oportunidad que me brindó con el trabajo de investigación y por su compromiso e inestimable ayuda para la realización de este proyecto. Siempre ha estado dispuesto a dedicarme su tiempo, transmitiéndome sus conocimientos para estimularme y conseguir lo mejor de mí.

“La demencia se come el pensamiento del enfermo y a su vez destroza los sentimientos de los que lo quieren y lo cuidan” - Dr. Nolasac Acarín Tusell (Doctor en Medicina y especialista en Neurología, Alzheimer y Parkinson)

TABLA RESUMEN

	DATOS
Nombre y apellidos:	Jennyfer González Labrada
Título del proyecto:	Integración de sistemas de aprendizaje automático en modelos cognitivos: un caso aplicado a Alzheimer
Director del proyecto:	Esteban García Cuesta
El proyecto ha implementado un producto:	Sí
El proyecto ha consistido en el desarrollo de una investigación o innovación:	Sí
Objetivo general del proyecto:	Obtener un sistema que comunica al paciente de Alzheimer cualquier anomalía que detecte sobre su comportamiento

Índice

Capítulo 1.	RESUMEN DEL PROYECTO	11
1.1	Introducción	11
1.2	Contexto y justificación	12
1.3	Estructura de la memoria	12
1.4	Beneficios del proyecto	13
Capítulo 2.	ANTECEDENTES / ESTADO DEL ARTE	14
2.1	Estado del arte.....	14
2.2	Planteamiento del problema	15
Capítulo 3.	DESARROLLO DEL PROYECTO.....	16
3.1	Descripción global de la solución propuesta	16
3.2	Sistema de detección de objetos.....	17
3.3	Modelo cognitivo.....	20
3.4	Sistema de detección de anomalías	23
3.5	Diseño de experimentación.....	28
Capítulo 4.	RESULTADOS Y DISCUSION	34
Capítulo 5.	CONCLUSIONES Y LINEAS FUTURAS	40
Capítulo 6.	REFERENCIAS.....	42
ANEXO A.	DIAGRAMA DE GANTT	45
ANEXO B.	DESCRIPCIÓN DE TAREAS	47
ANEXO C.	PRESUPUESTO	49

Índice de Figuras

Figura 1: Esquema componentes del sistema.....	16
Figura 2: Descripción de los diferentes modelos de detección de objetos probados	17
Figura 3: Rango de aprendizaje del modelo.....	19
Figura 4: Pérdida total del aprendizaje	19
Figura 5: Imagen de la detección de desayuno (caso1)	20
Figura 6: Imagen de la detección de desayuno (caso2)	20
Figura 7: Ej. de conversión de los datos desde el sistema de detección al modelo cognitivo ...	22
Figura 8: Curva de aprendizaje en tiempo real (1).....	22
Figura 9: Curva de aprendizaje en tiempo real (2).....	22
Figura 10: Pseudocódigo de la función “insertar punto”	25
Figura 11: Pseudocódigo de la función “eliminar punto”	26
Figura 12: Desplazamiento en el árbol de un nuevo punto	26
Figura 13: Corte aleatorio RRCT (1).....	27
Figura 14: Corte aleatorio RRCT (2).....	28
Figura 15: Representación de la distribución normal	29
Figura 16: Curva de aprendizaje: Experimentación real	30
Figura 17: Curva de aprendizaje entrenamiento: Datos sintéticos.....	30
Figura 18: Curva de aprendizaje para pruebas obtenida de datos sintéticos	31
Figura 19: Línea temporal de la rutina del paciente	31
Figura 20: Experimentación real. Primera semana	32
Figura 21: Experimentación real. Segunda semana	32
Figura 22: Detección de anomalías con malos resultados	33
Figura 23: Detección de anomalías incorporando datos adicionales.....	33
Figura 24: Gráfico anomalías de los cinco días de pruebas	37
Figura 25: Gráfico anomalías de los dos primeros días de pruebas.....	37
Figura 26: Gráfico anomalías de los tres últimos días de pruebas.....	38
Figura 27: Gráfico anomalías del día 4 de pruebas	38
Figura 28: Planificación enero y febrero	46
Figura 29: Planificación marzo y abril	46

Figura 30: Planificación mayo y junio.....46

Índice de Tablas

Tabla 1: Resultados tasa de aprendizaje.....19

Tabla 2: Resultados sistema detección de objetos. Pruebas anteriores.....34

Tabla 3: Resultado sistema de detección de objetos. Acción desayunar.....34

Tabla 4: Resultados del algoritmo RRCF en el conjunto de datos de test sintético35

Tabla 5: Número de puntos anómalos reales36

Tabla 6: Tareas para el diagrama de GANTT45

Tabla 7: Tarea 1 del proyecto47

Tabla 8: Tarea 2 del proyecto47

Tabla 9: Tarea 3 del proyecto47

Tabla 10: Tarea 4 del proyecto47

Tabla 11: Tarea 5 del proyecto48

Tabla 12: Tarea 6 del proyecto48

Tabla 13: Tarea 7 del proyecto48

Tabla 14: Tarea 8 del proyecto48

Tabla 15: Tarea 9 del proyecto48

Tabla 16: Estimación del presupuesto49

Capítulo 1. RESUMEN DEL PROYECTO

1.1 Introducción

Este proyecto surge con el propósito de ayudar a los pacientes de Alzheimer a mantener su independencia en las actividades diarias en su casa. Dentro de los diferentes estadios en los que se puede presentar esta enfermedad, el trabajo realizado está especialmente enfocado a aquellos que se encuentran en fases avanzadas donde no pueden valerse por sí mismos. Para ello se propone el desarrollo de un sistema que les ayudará a llevar una vida de una manera más autónoma y segura basado en un sistema de representación del conocimiento y algoritmos de inteligencia artificial.

El Alzheimer es una de las enfermedades más comunes para aquellas personas mayores de 65 años y afecta a una de cada diez personas [1]. A medida que avanza la enfermedad, es habitual que estos pacientes tengan que depender de una tercera persona para poder realizar sus actividades cotidianas. Esto tienen un coste que en muchos casos no puede ser sostenido por los familiares el uso de la tecnología también podría ayudar a democratizar esta asistencia [25].

Con este proyecto se quiere ayudar a crear una solución a este problema para permitirles vivir de una forma más autónoma e intentar mejorar lo máximo posible su calidad de vida. El sistema propuesto está planificado para implementarse en la fase uno de detección del Alzheimer para que sea capaz de aprender las rutinas del paciente y posteriormente poder detectar anomalías sobre este comportamiento base. Se considera rutina a aquellos ritmos de actividad, prolongados en el tiempo de modo regular y estable que engloban todas aquellas actividades de la vida diaria que realizamos de modo automático llevadas a cabo a la misma hora, mismo lugar en el mismo orden. Las rutinas tienen como objetivo consolidar hábitos. Así, si el paciente deja siempre las llaves encima de la mesa y en un futuro momento no se encuentran allí sino en otro lugar, se lanzaría una alarma de aviso para notificárselo y así pueda corregir su ubicación. De este modo se pretenden mantener los hábitos diarios y su independencia a través del refuerzo de estos comportamientos rutinarios asistiéndole con el sistema inteligente desarrollado.

Para ello se tiene como objetivo la creación de un sistema de detección de anomalías personalizado que pueda identificar comportamientos no habituales a través de los datos recogidos con una cámara sobre la rutina del paciente. Estos comportamientos no habituales se detectarán como valores anómalos cuando no se adecuen al periodo de tiempo aprendido, su duración sea mayor o menor de lo esperado, o directamente no se detecte. Este objetivo general puede ser desglosado en varios sub-objetivos parciales: i) **detectar objetos** haciendo uso de técnicas de análisis de imagen en tiempo real capturadas a través de una cámara, ii) **usar un modelo cognitivo con capacidades de aprendizaje y olvido en tiempo real [19]** que representará el conocimiento que se tiene sobre el entorno de aplicación del paciente, iii) **detectar anomalías** asociadas a situaciones donde se detecta un acto no habitual en la escena, iv) realizar el **diseño experimental con pruebas reales y sintéticas** para recrear situaciones

posibles de uso, y v) **validar los resultados obtenidos** para comprobar el correcto funcionamiento del sistema completo.

Tras construir el sistema de anomalías y de acuerdo con el diseño de la experimentación realizado que contempla diversas situaciones posibles en un contexto real se ha obtenido como resultado una precisión superior al 96% para la detección de comportamientos que están fuera de la rutina del paciente específico.

1.2 Contexto y justificación

Este proyecto aporta una idea innovadora que pretende ser un trabajo incipiente y disruptivo en ámbito de casas inteligentes, en concreto en el área de asistencial, y con un gran beneficio para los pacientes de Alzheimer.

Es habitual que a medida que avanza esta enfermedad los pacientes tengan la necesidad de tener una persona pendiente de ellos constantemente para poder realizar cualquier actividad cotidiana en su entorno. Esto les provocan un sentimiento de incompetencia y frustración al sentir que no pueden valerse por sí mismos y tener que depender de alguien. El objetivo de este proyecto es conseguir aportarles cierta autonomía y evitar ese sentimiento de incapacidad, pudiendo ellos mismos mantener una rutina con la asistencia automatizada de sistemas inteligentes que les asistan.

Como ventaja asociada a este tipo de sistemas propuesto decir también que aporta mucho valor a los cuidadores/familiares sobre los que, en la gran mayoría de los casos cercano al 80% [6], recae todo el peso y asumen todo el cuidados de estos enfermos. En un estudio realizado por la fundación Vianorte-Laguna declaró que el 70% de estos cuidadores no profesionales están en riesgo de padecer trastornos orgánicos y fisiológicos debido a la sobrecarga [6].

Por todo esto, el proyecto presentado puede suponer un gran avance médico en cuanto a la parte asistida de la enfermedad Alzheimer porque no solo mejora la calidad del que lo padece, sino que al integrar este sistema en su vida cotidiana muchos de ellos no tendrán que depender de una tercera persona dando así tranquilidad y confianza tanto a ellos mismos como a familiares y cuidadores.

1.3 Estructura de la memoria

La estructura de la memoria se ha organizado de acuerdo a la siguiente descripción: i) el capítulo 1 realiza un breve resumen sobre el proyecto donde se introduce la problemática a solucionar y los aspectos más relevantes del proyecto, ii) el capítulo 2 describe y justifica la realización de este proyecto y sus antecedentes incluyendo el estado del arte, iii) el capítulo 3 expone todo el desarrollo del proyecto centrándose especialmente en la parte técnica e innovadora, iv) el capítulo 4 realiza un análisis de los resultados obtenidos en el proyecto, v) el capítulo 5 incluye las conclusiones obtenidas con los resultados previos, y vi) finalmente el capítulo 6 describe las futuras líneas de trabajo tras la finalización de este proyecto.

1.4 Beneficios del proyecto

Este proyecto puede beneficiar notablemente la vida de los pacientes que sufren Alzheimer o demencia. Les puede permitir llevar una vida más autónoma, sin tener la necesidad de depender de terceras personas que lleven el control de sus actividades cotidianas.

Este sistema será capaz de notificar al paciente cuando no efectúe una acción de un modo que ha sido aprendido previamente de acuerdo a sus rutinas diarias con capacidad de readaptar esos comportamientos aprendidos. Esto puede ser de vital importancia cuando un paciente se olvide de tomar su medicamento, o se olvide de comer.

Además, este sistema le ayudará a tener una vida más ordenada consiguiendo cumplir su rutina establecida aprendida antes de presentar graves síntomas de demencia.

Capítulo 2. ANTECEDENTES / ESTADO DEL ARTE

2.1 Estado del arte

El principal propósito que persigue este proyecto es mejorar en la medida de lo posible la calidad de vida de los pacientes que sufren Alzheimer para que puedan vivir de una forma más autónoma, sin depender constantemente de terceras personas.

Esta idea surge en el contexto de una beca de investigación del grupo DSLAB asociada al proyecto “APRENDE-R Sistema conversacional afectivo basado en aprendizaje continuo en entornos cambiantes” [20].

En este proyecto se desarrolló un sistema de representación de conocimiento [19] a partir de sensores de recuperación de información, en concreto contenidos de tipo visual, creando un modelo de respuesta automático aprendido a partir de bases de datos con un gran volumen de imágenes. En este y otros contextos existen trabajos previos que tratan de detectar anomalías a partir de sensores. En [2] por ejemplo se plantea equipar la casa con sensores de diverso tipo (p.ej. de movimiento, de apertura, o de uso eléctrico) para analizar el comportamiento de pacientes con Alzheimer y detectar posteriormente anomalías, o en [26] utilizan los datos recolectados para ver la evolución del paciente. En otro contexto, en [3] se describe un sistema que permite detectar la intrusión nocturna en casas y así promover la seguridad.

A diferencia de los trabajos nombrados anteriormente, el sistema planteado en este proyecto analiza el comportamiento que tiene una persona con respecto a los objetos cotidianos que utiliza. Esto es de gran utilidad para que el paciente pueda conocer si se ha realizado o no las distintas tareas que debe realizar a lo largo del día. Por ejemplo, la toma de pastillas médicas, si ha cenado, o si ha ubicado un objeto erróneamente en algún sitio.

Existen diferentes técnicas para la detección de anomalías. Algunos trabajos se basan en la interpretación de anomalías como *outliers* y utilizan técnicas de k-vecinos para establecer un valor de anomalía en función de la distancia a sus vecinos más cercanos [21], o la búsqueda de un subespacio donde se muestre un comportamiento anómalo [22], o utilizan *ensembles para* tratar de buscar esos subespacios donde existe heterogeneidad [23]. Es en esta última categoría es donde se encuentra el método utilizado en este trabajo.

El método se basa en la realización de cortes aleatorios sobre las variables de entrada y la generación de diferentes árboles en función de estos cortes que determinan a través de un valor el grado de anomalía en función de si existen casos similares o no [4]. Este método es muy apropiado para el problema que se desea resolver dado que permite tratar datos en “*streaming*” y provee resultados en tiempo real. El principal componente de diseño necesario para poder plantear una solución utilizando esta técnica es la determinación de cómo se representan las entradas debido a que es un método no supervisado y por tanto se debe modelar el comportamiento normal. Esto se realiza con un conjunto de datos de entrenamiento para posteriormente detectar como anómalo aquello que no se adecua a ninguno de los patrones que representan esos datos. En [5] se describe una modificación de este método para permitir

la detección de anomalías en entornos cambiantes que permite evolucionar el modelo según van los datos de entrenamiento, adaptándose así a nuevos contextos.

2.2 Planteamiento del problema

El Alzheimer es una enfermedad crónica que como síntoma principal tiene la pérdida de memoria. A medida que avanza, puede también causar problemas con el lenguaje y la comunicación, desorientación o problemas de comportamiento.

Hoy en día está creciendo el número de personas que padecen Alzheimer, se estima que aproximadamente 44 millones de personas en el mundo la sufren o una forma relacionada de demencia [7]. Este número tan alto se debe a que la población está envejeciendo a pasos acelerados, según la OMS entre 2015 y 2050, la proporción de la población mundial con más de 60 años aumentará del 12% al 22% [8], y con ello también los enfermos de Alzheimer. Resaltar que según los informes del instituto nacional sobre el envejecimiento (NIH) la prevalencia a padecer esta enfermedad aumenta cada 5 años a partir de los 65 años [7].

Se trata de una enfermedad progresiva que va a empeorando a medida que pasan los años y de una forma diferente en cada paciente. Esto hace que las personas que la padecen a medida que avanza su enfermedad tengan cada vez más dificultades para seguir su vida cotidiana porque olvidan cosas tan simples como dónde han dejados las llaves de casa.

Esta enfermedad se suele dividir en tres fases aunque puede variar dependiendo de cada persona. En la primera fase del Alzheimer se suelen olvidar de los eventos recientes que han pasado en unos 20 minutos anteriores, además de padecer una pequeña desorientación. En la segunda fase comienzan los problemas en el habla y empiezan a olvidar como realizar tareas tan sencillas como vestirse o comer. En la última fase, el paciente no reconoce a sus familiares y pueden padecer crisis epilépticas [9]. A medida que avanza la enfermedad en las diferentes fases también aumenta la dependencia de terceras personas y durante la última etapa el paciente requerirá tener a una persona 24 horas del día con de ella [10].

Esta dependencia se mide a través del baremo de valoración de la dependencia (BVD), donde se mide la capacidad de autonomía que tiene un paciente en su entorno habitual. Las tareas que se evalúan son las actividades básicas de la vida diaria (ABVD), que son como comer, beber, ir al baño, lavarse, etc. [11].

En la actualidad no existe ninguna cura para remitir esta enfermedad, pero sí existen numerosos avances como medicamentos o actividades para mejorar los síntomas de la demencia y así mejorar la calidad de vida de estos pacientes [12]. Este proyecto trata de conseguir mejorar la calidad de vida de las personas que padecen Alzheimer mitigando algunos de estos efectos a través del refuerzo de las rutinas.

Capítulo 3. DESARROLLO DEL PROYECTO

En este capítulo se describe la solución técnica propuesta junto con el diseño de la experimentación realizada. La parte de la solución técnica se centra en la explicación tanto del funcionamiento del método utilizado, y su adaptación a la problemática concreta, como a la generación de los conjuntos de datos de entrenamiento sintéticos a partir de los datos tomados en entornos reales de una casa. El objetivo de realizar una captación real de objetos a través de sensores y generar posteriormente sus variantes sintéticas es poder reflejar un conjunto de escenarios diversos a partir de la definición de la rutina buscada y de ese modo proveer un sistema con cierta flexibilidad. La ventaja principal de esta aproximación es que se generarán menos falsos positivos en la detección de anomalías que si se utilizará un método basado en una rutina basada en horarios estrictos prefijados donde cualquier mínima alteración generaría una alarma. De esta manera la flexibilidad puede ser caracterizada y aprendida automáticamente a partir de un conjunto de semillas iniciales tomadas del escenario real.

Comentar que para la realización de este proyecto se ha empleado una metodología ágil, ya que es la mejor manera para poder ir obteniendo resultados en breves periodos de tiempo y conseguir adaptar el proyecto más fácilmente a los cambios dado su intrínseco carácter innovador.

3.1 Descripción global de la solución propuesta

El proyecto se compone de cuatro fases según se muestra en la figura 1. En una primera etapa se realiza la recolección de las imágenes a través de una cámara o del robot NAO (en el caso de este proyecto se probaron ambos casos). Para poder obtener información sobre la escena que se está enfocando es necesario en una siguiente fase disponer de métodos de detección de objetos. Resaltar que sólo se detectan los objetos de interés para el caso bajo estudio. Para simplificar esta parte de la experimentación, y a modo de prueba de concepto, se limitó al objeto taza para representar la actividad de desayuno de un paciente. Esos objetos detectados son la entrada del modelo cognitivo que va a representar en tiempo real el conocimiento global que se tiene sobre el contexto donde ocurre la escena. Finalmente, esta información se le transfiere al sistema de detección de anomalías que se encarga de encontrar comportamientos atípicos en el día a día del paciente.



Figura 1: Esquema componentes del sistema

3.2 Sistema de detección de objetos

El sistema de detección de objetos es la parte sensorial del sistema y por tanto muy relevante dado que si no se detecta el objetos/s bajo análisis correctamente no podrá incorporarse al modelo de representación del conocimiento en el modelo cognitivo. El objetivo es lograr la identificación y monitorización en tiempo real de elementos del entorno que sean de interés en la rutina de un paciente. La recopilación de datos se realizará a través de la detección de objetos sobre las imágenes obtenidas por una cámara (o fija o en el robot NAO). Este sistema de detección debe estar disponible 24h al día tanto en la fase de entrenamiento como en la fase de uso.

Existen diferentes algoritmos de detección de objetos algunos los cuales se basan en la optimización de una función de coste usando el gradiente descendente o redes neuronales profundas (DNN). Con el desarrollo de las DNN y la capacidad de computación, actualmente se considera que los mejores resultados en cuanto a precisión en la detección de objetos se obtienen con la arquitectura de redes convolucionales. Para la elección del modelo más adecuado se estudiaron los valores de los parámetros precisión y velocidad de algunas redes neuronales convolucionales, conocidas como Faster R-CNN[28], SDD y YOLO[27].

- Faster R-CNN[28] reduce el tiempo de funcionamiento de las redes de detección, exponiendo el cálculo de la propuesta regional como un cuello de botella. Las RPN (Red de Propuesta de Región) se entrena de extremo a extremo para generar propuestas regionales de alta calidad, que son utilizadas por Fast R-CNN para la detección. RPN es una red convolucional que predice simultáneamente límites de objetos y puntajes de objetividad en cada posición.
- SSD MobileNet [30] es un modelo de entrenamiento cruzado de SSD a arquitectura MobileNet. MobileNets se basa en una arquitectura optimizada que utiliza convoluciones separables en profundidad para construir redes neuronales profundas y livianas. SSD se basa en una red convolucional de avance que produce una colección de tamaño fijo de cuadros delimitadores y puntajes.
- YOLO[27] se entrenan imágenes completas y optimiza directamente el rendimiento de detección. Es modelo unificado, permite un entrenamiento de extremo a extremo, divide la imagen de entrada en una cuadrícula $S \times S$. Si el centro de un objeto cae en

Figura 2: Descripción de los diferentes modelos de detección de objetos probados

Tras realizar el estudio y ver los resultados obtenidos se eligió la arquitectura SSD MobileNet, debido principalmente a las limitaciones de velocidad del marco de integración en el que se estaba trabajando, y las restricciones en cuanto a capacidades de memoria y CPU. Esta red neuronal sacrifica una cantidad razonable de precisión a favor de menor latencia y mayor velocidad, por lo tanto, es especialmente adecuada para el procesamiento en tiempo real. Aunque el rendimiento de MobileNet puede verse afectado con la detección de objetos

pequeños en comparación con otras arquitecturas, mostró buenos resultados en cuanto a la precisión y velocidad. Este modelo tiene una velocidad de 30 (ms) en cuanto a la evaluación de datos basados en el entrenamiento con el conjunto de datos de Microsoft COCO[29]. COCO es un conjunto de datos formado por fotografías, que abarca 91 categorías de objetos y 328.000 imágenes. Estas imágenes pueden contener una o varias etiquetas que forman un total de 2.5 millones. El conjunto de datos COCO de Microsoft exhibe varias características como segmentación de objetos, reconocimiento en contexto, múltiples objetos por imagen y 5 subtítulos por imagen, y se considera un referente en la comunidad de visión por computador.

El código utilizado se ha hecho en Python con la librería de Tensorflow [24] que tiene implementadas las arquitecturas CNN. Además, también proporciona modelos ya entrenados para la detección de 90 objetos aunque para este proyecto se decidió generar un modelo propio para conseguir una mayor precisión utilizando únicamente los objetos deseados ya que se trabaja en un entorno muy controlado. Como resultado del uso de la técnica SSD-mobilenet se obtiene la detección de múltiples objetos dentro de una imagen cada uno de ellos con un cuadro delimitador.

Para la fase de entrenamiento se generó una base de datos con 150 imágenes sobre los objetos desde diferentes perspectivas y diferente luminosidad. En esas imágenes se agregan las diferentes etiquetas de los objetos de interés que se encuentran. Esta etapa es de vital importancia para conseguir mayor precisión a la hora de detectar objetos ya que si no se representan todas las posibles casuísticas degradaría esta precisión. Los resultados de la detección se muestran visualmente utilizando cv2 [31] proporcionando así la posibilidad en tiempo real de ver lo que está sucediendo en la escena a través del sistema de monitorización.

Desde un punto de vista de la configuración, se modificaron y ajustaron los parámetros del modelo para mejorar los resultados obtenidos. Primero se ajustaron las medidas de la imagen, a las que tenía el visor que eran 400x300 para reducir así la cantidad de entradas a la red convolucional y por otro lado eliminar inconsistencias de escala. También se establecieron los parámetros de ratio de aprendizaje “learning_rate” para poder ayudar a la convergencia rápida del modelo. El parámetro “initial_learning_rate” (tasa de aprendizaje inicial) se aumentó a 0.009 y los parámetros “decay_steps” y “decay” se redujeron a 700 y 0.7, respectivamente. Estos cambios se realizaron de acuerdo a la siguiente fórmula:

$$step(x) = initial_learning_rate \times decay^{\frac{x}{decay_steps}}$$

la cual calcula el rango de aprendizaje del modelo en cada paso del entrenamiento. Estos ajustes se han realizado para asegurar la convergencia en un tiempo razonable. La tabla 1 muestra los valores que este parámetro ha tenido para diferentes pasos hasta la convergencia. Se puede observar cómo a medida que aumenta el número de pasos, el valor de la tasa de aprendizaje va progresivamente acercándose al cero (ver Figura 3).

Tabla 1: Resultados tasa de aprendizaje

Paso	1	10	100	700	1400	4000
Tasa de aprendizaje	0,00899	0,00895	0,00855	0,0063	0,00441	0,00117

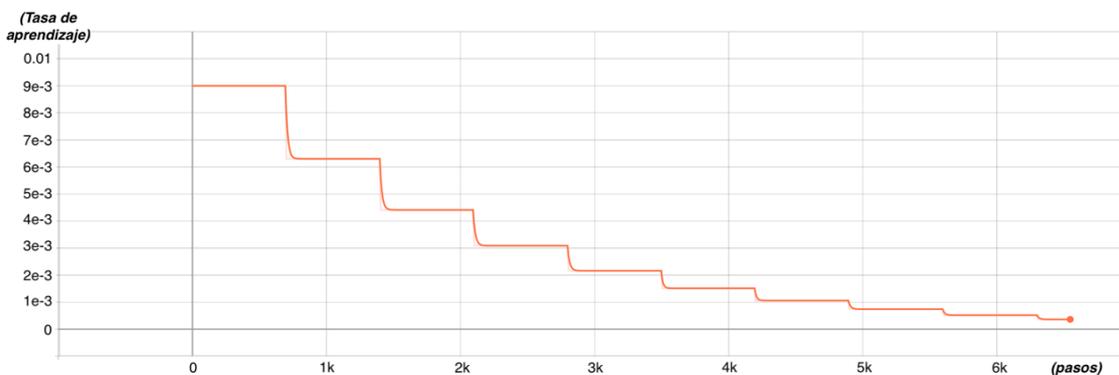


Figura 3: Rango de aprendizaje del modelo

Esto va a permitir que el modelo se ajuste mejor y de ese modo también los resultados de detección sean mejores tal y como se ve en la figura 4. Esta figura muestra el error entre los valores pronosticados y los valores reales en los datos de entrenamiento pudiéndose observar la convergencia hacia el final.

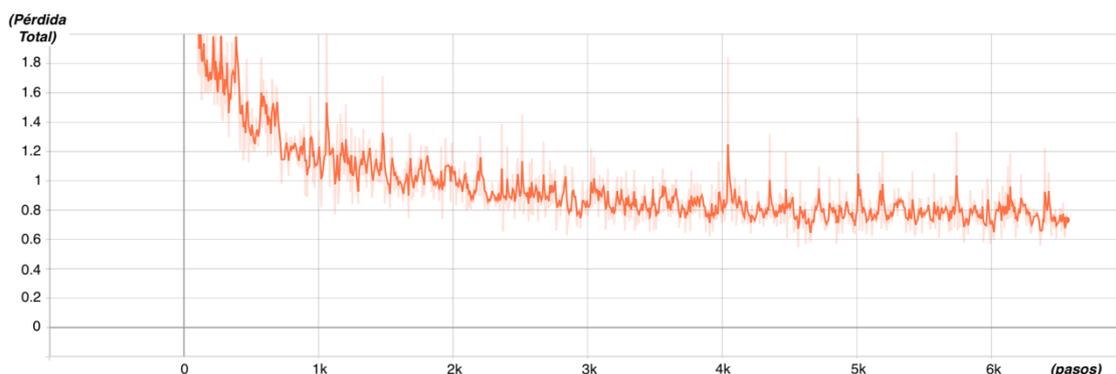


Figura 4: Pérdida total del aprendizaje

Para poder determinar que existe una relación entre objetos, además de la detección, se ha integrado el concepto de superposición la cual ocurre cuando un objeto se encuentre encima de otro. Para ello se han determinado las áreas de los objetos y definido un umbral mínimo para determinar si existe o no superposición entre ellos. Un ejemplo sería la detección de un móvil que está encima de la mesa. Esta información es útil si queremos saber si hay una persona desayunando detectando que los dos objetos están en la escena e interactúan a través de la superposición.

En la figura 5, se muestra un caso de detección de una taza y en la figura 6, se muestra un caso de detección de una persona junto a una taza. En ambos casos se categoriza como desayuno. Esto representará la acción de desayunar de un paciente para poder analizar su rutina diaria.

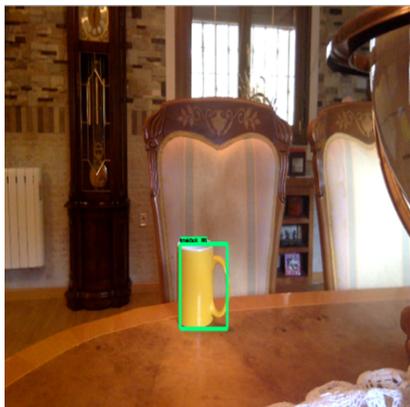


Figura 5: Imagen de la detección de desayuno (caso1)

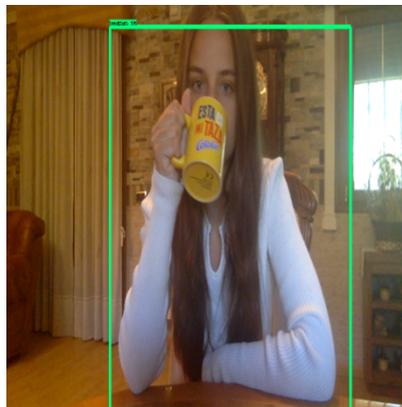


Figura 6: Imagen de la detección de desayuno (caso2)

Para integrar este modelo de visión por computadora con el modelo cognitivo se utiliza el valor de confianza en la detección proporcionado por el modelo aprendido para un objeto dado que está en la escena. Este valor de confianza p_i para un objeto dado i se le pasa al modelo cognitivo como un parámetro de entrada y que permitirá actualizar su conocimiento sobre el mundo tal y como se explica en la siguiente sección.

3.3 Modelo cognitivo

El modelo cognitivo utilizado [19] representa cada objeto o las relaciones entre ellos como una serie temporal de la verdad sobre su existencia (o su verdad) que se refuerza cuando se detecta y decae cuando no lo hace manteniendo un sentido sobre la realidad actual. Por tanto, este modelo tiene la capacidad de aprender sobre el entorno que está viendo, pero también de olvidar, creando así una curva de aprendizaje a medida que va aumentando su certeza sobre un evento. Resaltar que esta curva de aprendizaje siempre puede ser consultada como histórico por lo que también puede entenderse como un modelo de almacenamiento de información. Además, cumple con los cinco criterios introducidos por Newell [32] como deseables en las arquitecturas cognitivas inteligentes artificiales: i) comportamiento flexible, ii) tiempo real, iii) comportamiento adaptativo, iv) base de conocimiento amplia, v) comportamiento dinámico.

Este modelo utiliza la técnica simple de suavizado exponencial tal y como indican los autores en [19]. A continuación, se resume el método y también cómo interacciona con el resto de los elementos que se han diseñado.

En una época determinada (época indica un intervalo temporal), la verdad o certidumbre sobre cada objeto/relación debe actualizarse para incorporar nueva información sobre la situación actual observada. Esta información se obtiene del sistema de detección de objetos con el análisis realizado en ese momento o época el cual devuelve una certidumbre sobre que ese objeto esté

(la red neuronal no devuelve sí o no sino un valor de confianza entre 0 y 1 que es interpretado para este fin como una probabilidad). Para el primer caso, si se supone que el algoritmo de detección detecta dos objetos, llamados i y j , con confianzas p_i y p_j , respectivamente. Una primera estimación de la fuerza de la relación "objeto j está en el objeto i " viene dada por :

$$s_{ij}(t) = \begin{cases} p_i p_j ; & \text{si } i \text{ y } j \text{ estan superpuestos} \\ 0 ; & \text{de otra manera} \end{cases}$$

Por otro lado, en el caso de que se detecte un único objeto, llamado i , con una confianza de p_i . Cuya primera estimación de la fuerza de la presencia "objeto i se encuentra en escena" viene dada por:

$$s_i(t) = \begin{cases} p_i ; & \text{si } i \text{ se detecta} \\ 0 ; & \text{de otra manera} \end{cases}$$

donde un valor de cero en cualquiera de los dos casos significa que el algoritmo no lo detecta.

Hay que tener en cuenta que el conjunto de secuencias $S_{ij}(t)$ y $S_i(t)$ proporciona un modelo instantáneo del mundo, es decir que no tiene en cuenta eventos pasados. Tal representación es inherentemente inestable e introducen una gran cantidad de ruido debido al propio sistema o a eventos que suceden en la escena como puede ser que otro objeto oculte por un instante al que se está analizando. Además, también puede ocurrir que dos objetos correctamente identificados estén momentáneamente superpuestos y por tanto se detecte una relación, pero no se debe a una situación real sino casual. Por ejemplo, un usuario que lleva una taza accidentalmente pasa frente a una mesa, los algoritmos visuales pueden detectar una superposición entre los objetos "taza" y "mesa", pero el modelo no debe inferir que "la taza está sobre la mesa".

Por otro lado, cuando se encuentra constantemente una relación entre dos objetos durante épocas consecutivas, aumenta nuestra confianza en el evento. Por lo tanto, se define y actualiza la regla para los pesos de la relación "el objeto j está en el objeto i ":

$$w_{ij}(t + \Delta t) = w_{ij}(t) \left(1 - \frac{1}{\lambda}\right) + \frac{s_{ij}(t)}{\lambda}$$

También ocurriría lo mismo con la detección de un único objeto concreto, cuando se encuentra constantemente un objeto en sucesivas épocas, aumenta la confianza. De manera que, se define y actualiza la regla para los pesos de la presencia "el objeto i se encuentra en escena":

$$w_i(t + \Delta t) = w_i(t) \left(1 - \frac{1}{\lambda}\right) + \frac{s_i(t)}{\lambda}$$

Los parámetros w_{ij} (para relaciones entre i y j) y w_i (para el objeto i), son una estimación que representa si esa relación u objeto son verdad en un momento dado. Estos valores oscilan entre

0 y 1 siendo 0 cuando la certeza es nula y 1 cuando es máxima. Esta variable se calcula cada segundo que corresponderá a un punto y representa la curva de aprendizaje. En la Figura 7 se puede observar dos señales de entrada discretas, mesa (línea azul entrecortada) y hamburguesa (línea naranja entrecortada), y la señal de salida correspondiente al aprendizaje del modelo cognitivo (línea verde).

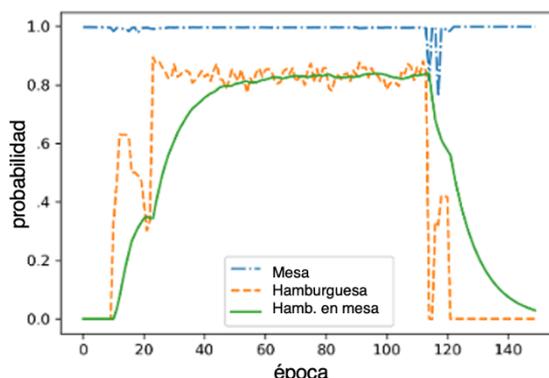


Figura 7: Ejemplo de conversión de los datos desde el sistema de detección al modelo cognitivo

El parámetro de persistencia λ establece la inercia del modelo frente a cualquier cambio tanto de refuerzo positivo (aprender) o negativo (olvidar). Si λ es grande se necesitarán varios refuerzos muy positivos para aprender una nueva relación o un nuevo objeto y, por el contrario, también se necesitan varios refuerzos negativos consecutivos para olvidarlos. Este enfoque también puede entenderse, como un filtro de paso bajo que introduce un suavizado exponencial simple de la señal con un factor de suavizado igual a $1 / \lambda$. Una persistencia alta requiere de cierto tiempo para responder a los cambios, por lo que se concluyó tomar el valor de 10 que representaba bien nuestro caso. Para visualizar las curvas de aprendizaje se utilizó la librería Matplotlib, que contiene funciones para la generación de gráficos.

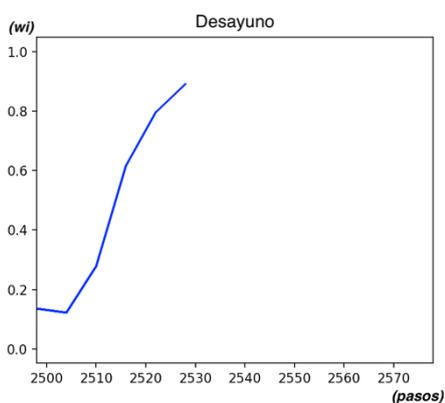


Figura 8: Curva de aprendizaje en tiempo real (1)

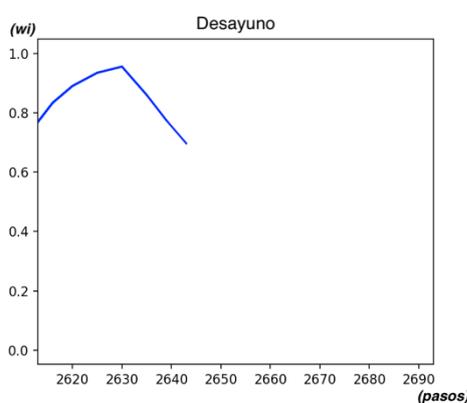


Figura 9: Curva de aprendizaje en tiempo real (2)

Realizadas con Matplotlib de Python

En la figura 8, se representa un ejemplo captado a tiempo real cuando se detecta el objeto $i = \text{“desayuno”}$. Se puede observar cómo el valor w_i aumenta a medida que el objeto permanece

más en pantalla, representado de esta manera el aprendizaje progresivo que tiene el modelo. Y por otro lado, en la figura 9, como el valor w_i disminuye cuando el objeto deja de aparecer, representando así el olvido progresivo del modelo.

3.4 Sistema de detección de anomalías

El sistema de detección de anomalías tiene como objetivo la identificación de los momentos a lo largo del día donde se rompe la rutina. Existen diferentes técnicas para la detección de anomalías y la que se ha utilizado se basa en el concepto de *ensembles para* tratar de buscar anomalías en subespacios donde existe heterogeneidad [23]. Una anomalía la definimos como un punto en la serie analizada que difiere significativamente del resto de la serie. En este caso la serie es un flujo de datos proviene del sistema de detección de objetos y del modelo cognitivo que los modela como se ha explicado en el apartado anterior (ver línea verde de la figura 5).

El método utilizado se denomina “Robust Random Cut Forest” (RRCF) el cual fue propuesto en el año 2016. Se basa en la generación de un conjunto de árboles de búsqueda binarios independientes, denominados “Random Cut Trees” (RCT) y usarlos para determinar un valor de anomalía asociado a cada punto de la serie. El algoritmo consiste en dos fases principales.

FASE DE ENTRENAMIENTO

El entrenamiento del método RRCF se realiza con los datos de la rutina que reflejan el comportamiento normal de la persona. Es decir, no debe contener ningún comportamiento que no se desee considerar como normal. Después este conjunto de datos de entrenamiento es sub-muestreado y cada una de estas sub-muestras se usa para construir un RCT que formarán las estructuras del modelo final de detección de anomalías. Antes de esto es necesario tener en cuenta tres hiper-parámetros fundamentales:

- **Número de árboles:** determina de cuantos árboles estará compuesto el modelo. Cuantos más árboles se usen mayor será la capacidad del modelo para capturar la naturaleza de los patrones que albergan los datos de entrenamiento.
- **Número de casos por árbol:** se puede utilizar una aproximación sin reemplazo donde diferentes árboles utilizan diferentes subconjuntos de entrenamiento o con reemplazo utilizando todos ellos el mismo conjunto. En general es más eficiente (aunque más costoso en tiempo) el uso de reemplazo o también denominado *bootstrapping*.
- **Tamaño del shingle:** determina cuantos puntos de la serie se seleccionan como entrada al modelo. Es muy importante elegir un valor adecuado para esta variable, ya que si es un *shingle* demasiado pequeño puede captar ruidos y detectar falsos positivos. Por otro lado, si el *shingle* tiene un tamaño demasiado grande puede dejar de detectar algunos puntos atípicos. La decisión de este tamaño tiene que ver directamente con la definición de anomalía en el contexto de aplicación.

La creación del árbol se basa en realizar cortes aleatorios en las diferentes dimensiones de los datos hasta que cada punto insertado quede aislado en una hoja individual. Los cortes se realizan teniendo en cuenta la variabilidad de cada dimensión dando más importancia a aquellas que sean mayores (y por tanto puedan desenredar mejor los diferentes casos unos de otros). Intuitivamente esto se puede explicar sabiendo que los casos anómalos son fácilmente separables del resto y por tanto quedarán en los primeros niveles del árbol. Estos árboles se conservan dinámicamente a la hora de insertar o eliminar puntos, permitiendo que el modelo se adapte a lo largo del tiempo a medida que van cambiando los datos.

En la fase de entrenamiento se realiza el proceso de inserción en los árboles con todos los puntos que han sido recogidos de la rutina del paciente generando así los árboles correspondientes y el modelo final entrenado.

La figura 10 muestra el pseudocódigo de la función “insertar punto”. En cada iteración primero se genera un corte aleatorio y se verifica si separa el conjunto de datos (S) y el nuevo punto (p). Si esto es así, se crea un nuevo nodo padre con el punto como hijo y como otro hijo el subárbol. Si no se separan, se sigue el corte existente en el árbol, si es menor que el valor de corte existente se tiene que ir al elemento secundario izquierdo, de lo contrario se tiene que ir al elemento secundario derecho. Y este proceso se repite con todos los puntos existentes en el conjunto de datos [13].

Datos: Un conjunto de puntos existentes S' divididos por el árbol T' , y un nuevo punto p

Resultado: Nuevo árbol $T'(S' \cup \{p\})$.

Sea x_i^l el valor mínimo de x con respecto a la dimensión i .

Sea x_i^h el valor máximo de x con respecto a la dimensión i .

Sea $B(S') = [x_1^l, x_1^h] \times [x_2^l, x_2^h] \times \dots \times [x_d^l, x_d^h]$ el cuadro delimitador del conjunto de puntos S' .

Sea Q el nodo raíz del árbol T' , con la dimensión de corte k y el valor de corte q .

InsertarPunto(T', p)

```

if  $S' = \emptyset$  then
    return RRCT( $p$ )
else
     $\hat{x}_i^l \leftarrow \min(p_i, x_i^l), \forall i;$ 
     $\hat{x}_i^h \leftarrow \max(p_i, x_i^h), \forall i;$ 
    Elige un número aleatorio  $r \in [0, \sum_i (\hat{x}_i^h - \hat{x}_i^l)]$ ;
     $j \leftarrow \operatorname{argmin} \{j \mid \sum_{i=1}^j (\hat{x}_i^h - \hat{x}_i^l) \geq r\};$ 
     $c \leftarrow x_j^l + \sum_{i=1}^j (\hat{x}_i^h - \hat{x}_i^l) - r$ 
    if  $c \notin [x_1^l, x_1^h]$  then
        Se crea una nueva hoja  $P$  correspondiente al punto  $p$ 
        Se crea un nuevo nodo padre  $C$  correspondiente al valor de corte  $c$  a lo largo
        de la dimensión  $j$ 
        Si el nodo raíz  $Q$  del árbol  $T'$  tiene un padre, se reemplaza  $Q$  con  $C$ 
        if  $p_j \leq c$  then
            Se establece  $P$  como el hijo izquierdo de  $C$ , y se deja que el hijo
            derecho sea la raíz  $Q$  del subárbol  $T'$ 
        else
            Se establece  $P$  como el hijo derecho de  $C$  y se deja que el hijo
            izquierdo sea la raíz  $Q$  del subárbol  $T'$ 
        end
    else
         $T'_1 \leftarrow$  Subárbol a la izquierda del nodo raíz  $Q$ 
         $T'_2 \leftarrow$  Subárbol a la derecha del nodo raíz  $Q$ 
        if  $p_k \leq q$  then
            InsertarPunto( $T'_1, p$ )
        else
            InsertarPunto( $T'_2, p$ )
        end
    end
end

```

Figura 10: Pseudocódigo de la función "insertar punto".

FASE DE VALORACIÓN DE LA ANOMALÍA (TEST)

Una vez creado ya el modelo, es decir, el conjunto de árboles generados con los datos de entrenamiento se realiza la valoración de un nuevo punto obteniendo su valor de anomalía. En esta parte se utiliza el modelo creado anteriormente, se inserta el punto a valorar, se calcula su grado de anomalía y finalmente se elimina. Este nuevo punto a valorar se debe eliminar para no generar ruido en el modelo ya que puede tratarse de un valor anómalo.

Se considera un punto anómalo cuando el árbol varía en gran medida su estructura a la hora de añadir o eliminar un punto. Los puntos que se encuentran más cerca de la raíz del árbol tienen más probabilidades de ser atípicos. También la detección de una anomalía se puede definir con respecto a la complejidad del modelo, cuando esta complejidad aumenta al añadir un punto al árbol se dice que ese punto nuevo es anómalo [4].

Para esta valoración se usan dos funciones principales, la inserción y la eliminación de un punto. La función de inserción tiene como objetivo recoger un árbol del modelo y producir uno nuevo con el punto añadido al igual que se hace en la fase de entrenamiento.

Por otro lado, está la función de eliminar punto que tiene como objetivo recoger un árbol del modelo y producir uno nuevo con el punto eliminado. Para ello quita la hoja (P) correspondiente del punto que se quiere eliminar (p), luego quita su nodo principal y finalmente reemplaza entre el hermano de (P) y el abuelo de (P), es decir, cortocircuitamos el camino desde p hasta la raíz.[13]

En las siguientes líneas se muestra el pseudocódigo de la función “eliminar punto”:

Datos: Árbol robusto de corte aleatorio T y un punto p .

Resultado: Árbol robusto de corte aleatorio T y un punto p eliminado.

EliminarPunto (T, p)

Se encuentra la hoja P en T correspondiente al punto p ;

Se deja ser R el nodo hermano de P ;

Se elimina el nodo padre de P y se reemplaza con R ;

Se devuelve el árbol modificado T' ;

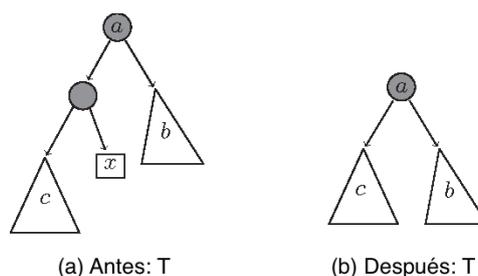


Figura 12: Desplazamiento en el árbol de un nuevo punto

En la figura 12, se muestra el desplazamiento del nuevo punto x . En (a) se puede apreciar el aumento de la complejidad del lado izquierdo del árbol al insertar el nuevo punto, en cambio cuando se elimina en (b) se puede apreciar que se reduce. Mientras que la profundidad del lado derecho del árbol no se ve afectada por esta inserción. El desplazamiento del punto x para este árbol será proporcional al número de puntos en el subárbol c .

El grado de anomalía que tiene un punto determinado se mide con el cambio de complejidad del modelo a la hora de insertar o eliminar ese punto. Esta complejidad puede representarse como la suma de las profundidades del conjunto de puntos del árbol. Por lo tanto, un punto que aumente notablemente esta complejidad cuando se inserte en el árbol, se consideraría como un valor anormal.

La estructura de estos árboles variará en función del corte aleatorio que se realice. Esto se puede ver en este pequeño ejemplo representado en las siguientes imágenes (figuras 13-14).

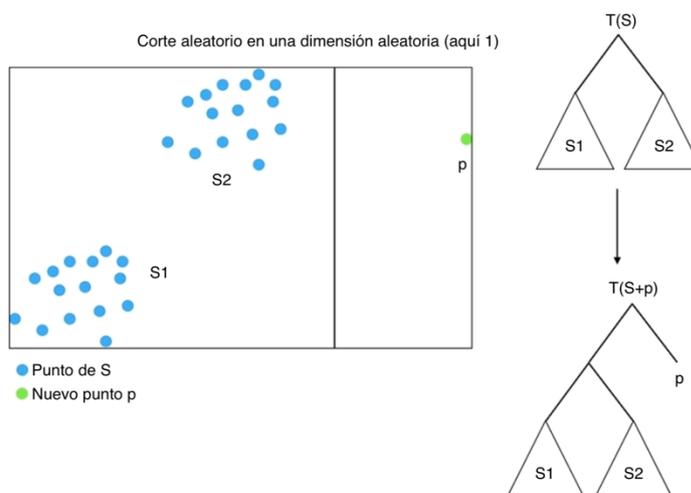


Figura 13: Corte aleatorio RRCT (1)

En la figura 13 se puede observar como este corte ha dejado por un lado el conjunto de puntos que se encuentra en el árbol y por otro lado el nuevo punto que se quiere insertar. Por lo tanto, el nuevo árbol tendrá por un lado como nodo raíz el árbol T (el árbol anterior) y por otro lado p (el nuevo punto) [14].

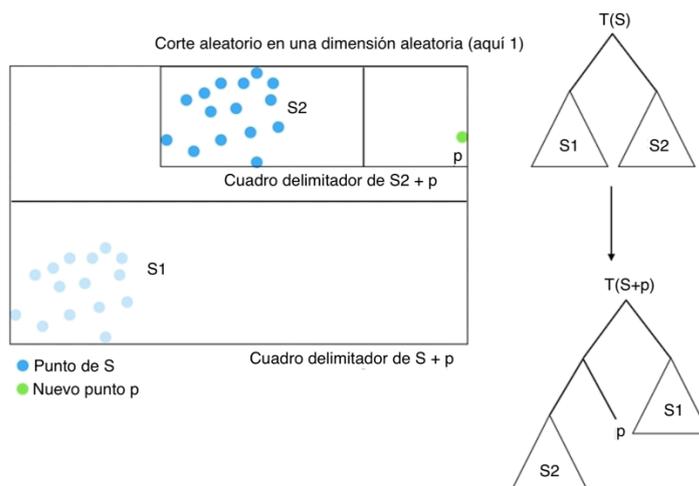


Figura 14: Corte aleatorio RRCT (2)

En cambio, en la figura 14, en el primer corte separa por un lado S2 y p y por otro lado S1. Y en el segundo corte separa por un lado S2 y por otro p. Por lo tanto, en este caso, para la creación del nuevo árbol por un lado estará como nodo principal S1 y por otro lado S2 y p [14].

La implementación se ha realizado con la librería rrcf que proporciona acceso a las funciones principales de este algoritmo.

3.5 Diseño de experimentación

El sistema consiste primero en una fase de entrenamiento, donde el modelo deberá aprender cual es la rutina del paciente y una segunda donde se verifica su funcionamiento correcto.

Para realizar la primera se han obtenido datos reales a partir de una webcam y se han registrado la detección de los objetos en un entorno simulado. Además, se han generado un conjunto de datos sintéticos a partir de estos datos reales para dar mayor flexibilidad a las rutinas diarias y así disminuir los falsos positivos debidos a pequeños cambios en la rutina.

Para realizar la segunda parte, se han generado sintéticamente y de manera aleatoria un conjunto de casos posibles dentro del contexto de hábitos diarios y se han etiquetado como válidos o no y se ha testeado que el algoritmo es capaz de reconocerlos con éxito.

Con el fin de facilitar la obtención de los datos reales se ha escalado la experimentación a un entorno más controlado simulando los eventos. Se realizó la simulación de una rutina del desayuno a escala temporal. Definimos 1 hora como el equivalente a 24 horas reales, o lo que es lo mismo un segundo de video real se corresponde 0,006 segundos simulados. Esto permite reducir un día entero de la vida del paciente a aproximadamente 60 minutos de experimentación. Una vez hecha esta equivalencia se define un rango para determinar lo que consideramos el periodo "normal" en el que tiene que desayunar. Este periodo comprende

desde las 8:00am hasta las 10:00am, que en la escala hecha para la experimentación equivale desde el minuto 20 hasta el minuto 25.

Para añadir cierta variabilidad al evento de desayuno se han generado de manera sintética un conjunto de nuevos eventos siguiendo una distribución normal alrededor de la hora media entre el comienzo y el final. Esta generación se realiza variando tanto la fecha de inicio como la duración introduciendo cierta aleatoriedad y variabilidad, o dicho de otro modo flexibilizando cuando ocurre el evento y su duración a la hora de crear el modelo. Se registraron un total de 14 nuevos eventos representando así dos semanas de experimentación completas.

Para la generación de estos datos sintéticos, primero se calcula el punto medio donde aparece el objeto en pantalla a través de la cámara y su desviación estándar, los resultados obtenidos fueron: $\mu_m = 1169$ y $\sigma_m = 104,92$. Asimismo, el punto medio donde desaparece el objeto y su desviación estándar, los resultados obtenidos fueron: $\mu_e = 1284$ y $\sigma_e = 69,89$. Estos puntos representan que el sujeto desayuna en un rango entre las 8:00am y las 10:00am.

Por otro lado, se obtuvo de la experimentación real el porcentaje de precisión del objeto detectado en cada punto, para posteriormente calcular el valor de porcentaje medio y su desviación estándar, los resultados obtenidos fueron: $\mu_p = 0,826$ y $\sigma_p = 0,263$. Esto simula la obtención del porcentaje que aparece cuando el objeto es detectado a través de la webcam. Es decir, cuando se detecta una mesa e indica un valor de 0,843, significa la probabilidad de que ese objeto fuera una mesa.

La recopilación de estos resultados es necesario para crear los datos sintéticos en la fase de entrenamiento siguiendo una distribución normal (método “*random.normal*” que proporciona la librería *numpy*) [15]. La distribución normal permite aproximar una variable aleatoria según su media y desviación típica.

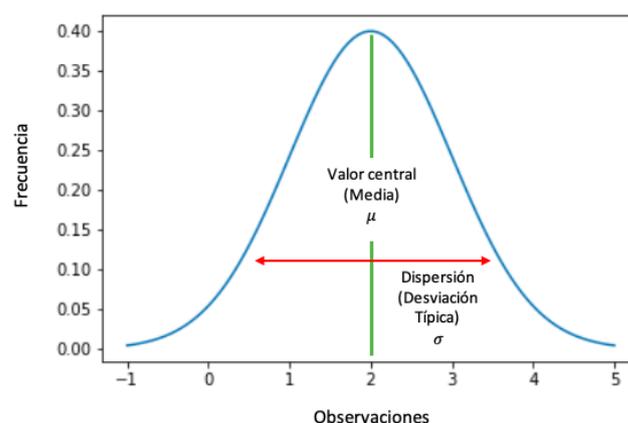


Figura 15: Representación de la distribución normal

Como se puede observar en la figura 15, el valor creado a través de la distribución normal rondará el valor medio, dando la posibilidad de que varíe según la desviación típica. Por lo tanto este valor se mantendrá dentro de la curva. De esta manera proporcionamos aleatoriedad ya que una persona no realiza estrictamente una acción siempre a la misma hora.

Su función de densidad viene dada por la siguiente fórmula:

$$F(x) = \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{1}{2}\left(\frac{x-\mu}{\sigma}\right)^2}$$

donde μ es la media y σ es la desviación estándar.

Además de estos dos parámetros se le debe indicar a la función el número de nuevos datos que se quieren generar. Para obtener un modelo de aprendizaje robusto se crean datos sintéticos simulando un mes de experimentación real. Una vez que se han calculado las diferentes probabilidades se aplica la fórmula del modelo de representación de conocimiento dinámico, que ha sido explicado anteriormente, para representar la curva de aprendizaje que tiene el modelo.

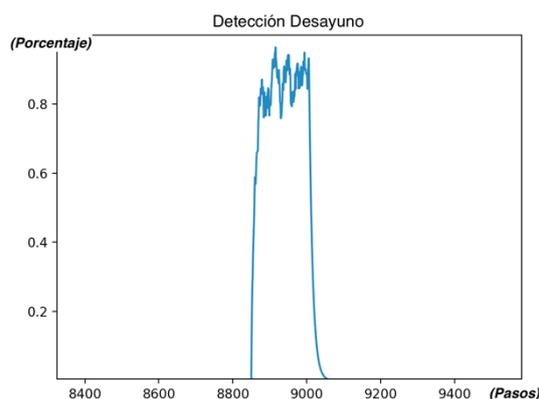


Figura 16: Curva de aprendizaje: Experimentación real

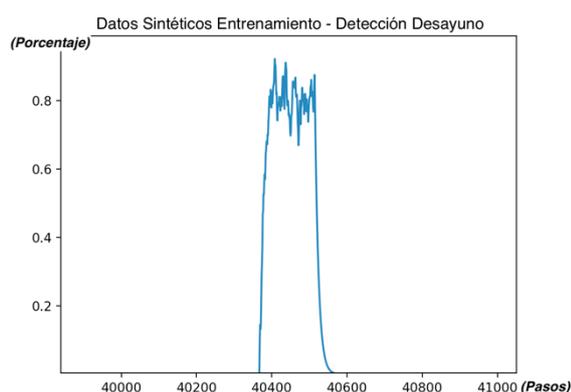


Figura 17: Curva de aprendizaje entrenamiento: Datos sintéticos

En la figura 16 se muestra el gráfico obtenido a través de la experimentación real y en la figura 17 el creado a través de datos sintéticos. Estos gráficos están formados aproximadamente por 200 puntos, donde tanto el punto de comienzo como el de final han sido calculados de la manera anteriormente explicada para representar la rutina del paciente que está comprendida entre las 8:00h y las 10:00h. Como podemos observar estos dos se asemejan a la hora de representar el modelo cognitivo sobre un objeto. Al aparecer el objeto aumenta su valor w_i y representa que el modelo cognitivo refuerza progresivamente el concepto asociado. Por otro lado, al desaparecer el objeto disminuye su valor w_i hasta llegar casi al valor cero, esto representa que el modelo cognitivo olvida progresivamente.

De modo similar se han creado los datos sintéticos para la fase de pruebas. En este caso para crear tanto el punto en el que aparece el objeto como el punto en el que desaparece, se ha ampliado el rango de horario en el que puede darse que el paciente desayune con el fin de que haya tanto casos anómalos como no anómalos. Este rango es de las 6:00am hasta las 12:00pm y, únicamente se deberá detectar como rutina a los que se encuentren dentro del rango definido anteriormente que era de 8:00h hasta las 10:00h. Todo lo demás que no se encuentre dentro se considerará como anomalía, ya que no cumple con la rutina descrita por el paciente.

Estos dos valores de inicio y fin también se generan de manera aleatoria utilizando la función “random.randint” de la librería numpy [16]. Además, se añadió un parámetro más *size*, donde se indica la cantidad de números aleatorios (o el equivalente en tiempo) que tiene que crear la función.

Para simular el porcentaje de precisión calculado por el sistema de detección de objetos, cuando detecta un objeto concreto, se utilizó la función “random.random_sample”, también provista en la librería numpy. Esta función devuelve números de tipo float que estén comprendidos entre los valores 0 y 1 [17]. Como el algoritmo de detección de objetos únicamente tiene en cuenta los objetos que son detectados con un porcentaje de precisión mayor a 0.5 se realizó el mismo filtro, para únicamente obtener valores entre 0.5 y 1.

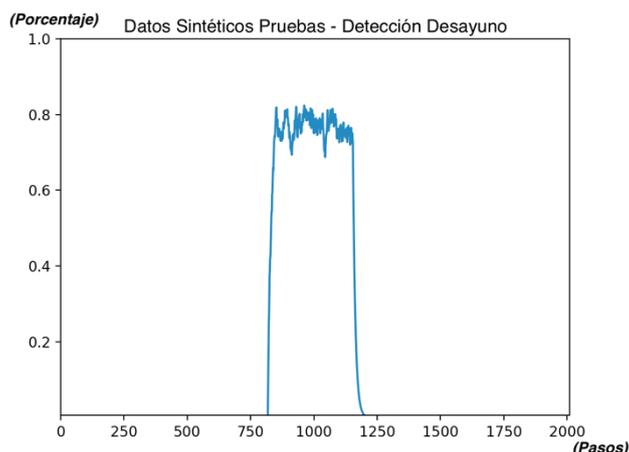


Figura 18: Curva de aprendizaje para pruebas obtenida de datos sintéticos

En la figura 18 se muestra la curva de aprendizaje del modelo, creada con datos sintéticos para la fase de evaluación. Al ser creada para probar el sistema de anomalías, puede ser similar a las que se ha visto anteriormente, por lo tanto, lo detectará como día más dentro de su rutina o ser totalmente diferentes, por lo que detectará todas las anomalías que se encuentren con respecto a su rutina.

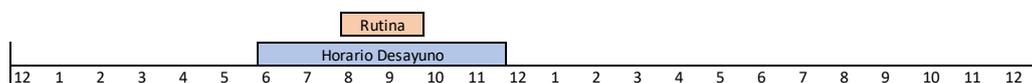


Figura 19: Línea temporal de la rutina del paciente

Para tener una imagen visual de la rutina del paciente, en la figura 19 se puede observar la línea temporal de la rutina del paciente, donde la franja de “Horario desayuno” es el periodo que en el que se podría realizar la acción. En cambio, únicamente lo que se encuentra dentro de la franja “Rutina” es detectado como comportamiento normal.

Una vez finalizada la creación de datos sintéticos tanto para la parte de entrenamiento como para la parte de pruebas, se utilizarán como conjuntos de datos de entrada al sistema de anomalías.

Cada grabación contiene el evento de desayuno simulado con la colocación del objeto taza en la pantalla. En las figuras 20 y 21 se muestra la representación de la simulación para dos semanas, donde los bloques azules representan cuando aparecería el objeto en pantalla. En el eje “Simulación” se muestran los minutos a escala y en el eje “Entorno real” los valores a los que equivale ese minuto. Como se explicó anteriormente, la rutina comprende desde las 8:00h hasta las 10:00h que en la escala propuesta equivale del minuto 20 al 25.

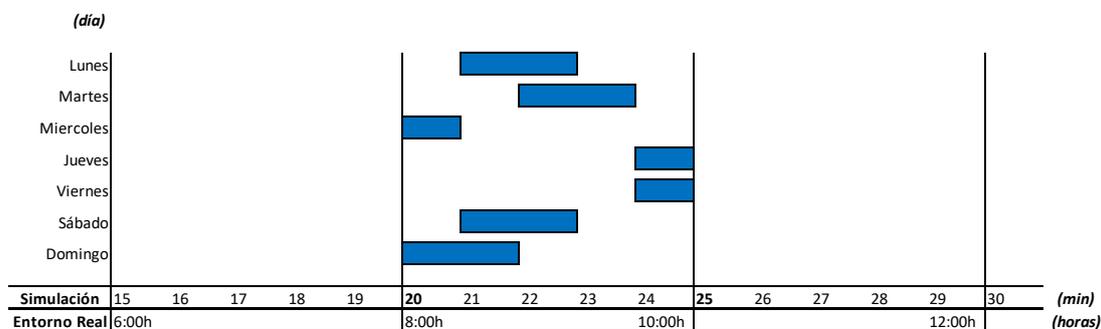


Figura 20: Experimentación real. Primera semana

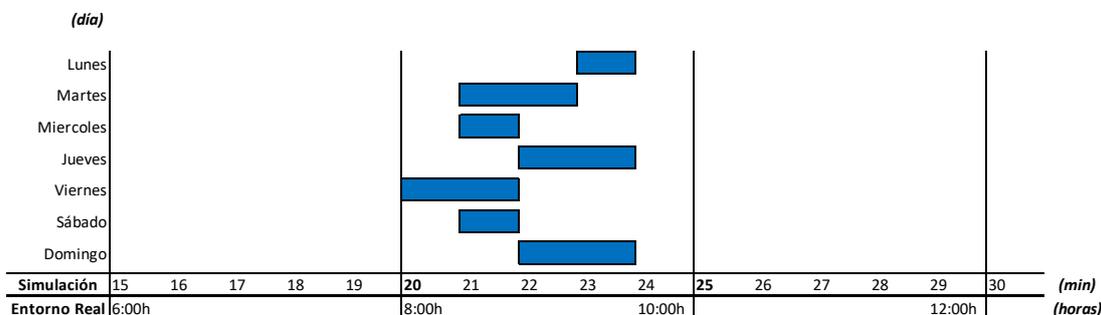


Figura 21: Experimentación real. Segunda semana

Estos nuevos eventos sintéticos generados para dos semanas completas contienen un total de 41.874 registros (1 por segundo aproximadamente o por cada 20 segundos en escala de tiempo real). Se asume que todos los datos están espaciados de manera uniforme (este tiempo depende del tiempo de ejecución del algoritmo de visión por computadora para detectar objetos puede contener muy pequeñas variaciones).

Resaltar que cuantos más datos históricos se tengan, es decir, más tiempo se haya realizado el estudio del comportamiento de los objetos del paciente, será más fácil detectar una anomalía. Ésto, la falta de facilidades para tomar más datos reales, y la flexibilidad en la detección de anomalías son los motivos principales por los que se han generado datos sintéticos.

Añadiendo únicamente estos datos no se obtuvieron los resultados esperados, siendo el porcentaje de acierto siempre muy bajo para todos los casos (Figura 22). Esto es debido a que únicamente se cuenta con un valor para entrenar el modelo.

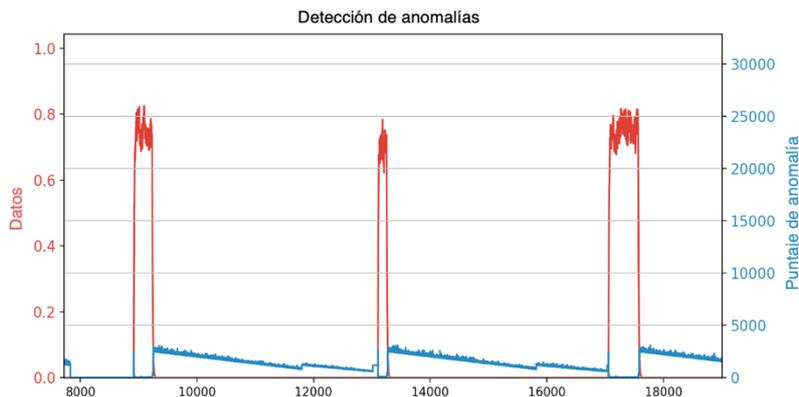


Figura 22: Detección de anomalías con malos resultados

Por esto, se han incluido características adicionales que ayuden a mejorar la detección de anomalías reales. Estas nuevas características son:

- **Duración del evento hasta ese punto w_i** : se añade como dato adicional el número de puntos que duraba el valor de w_i teniendo como umbral $> 0,5$
- **Posición diaria de w_i** : se añade la posición relativa al día de los puntos donde $w_i > 0,5$

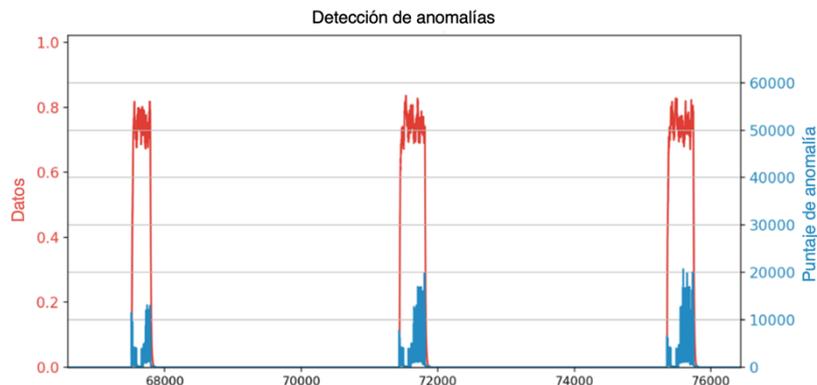


Figura 23: Detección de anomalías incorporando datos adicionales

Con la incorporación de estos datos además de indicar que los valores w_i menores de 0.00005 se transformarían en 0, se obtiene una mejora en los resultados notable. Esto puede observarse visualmente en la figura 23 donde se muestra un ejemplo obtenido con la simulación de 15 días de datos para la fase de entrenamiento y 5 días para comprobar su correcto funcionamiento. La representación se ha simplificado exponiendo únicamente 3 días de pruebas para facilitar la interpretación. El porcentaje de acierto asociado a este nuevo modelo obtenido es del 95,44%, por lo que se puede concluir que la inclusión de estas dos características mejora notablemente la detección de anomalías.

Capítulo 4. RESULTADOS Y DISCUSION

¿Qué impacto puede tener mi trabajo en un paciente que sufra Alzheimer?

Autonomía, libertad, alivio, seguridad

El programa comunicará al paciente de Alzheimer cualquier anomalía que detecte sobre sus comportamientos diarios. Es decir, si se espera que los martes y jueves a las nueve de la mañana esté el teléfono móvil sobre la mesa y llega el martes a las nueve y no está el sistema avisará al paciente de que el móvil no está en el sitio habitual. O también casos más importantes como comprobar que se ha tomado el medicamento correctamente.

Para comprobar su correcto funcionamiento se han testado en el conjunto de experimentos de test que incluyen eventos diarios de rutina. Para este proyecto en concreto se limitó únicamente en la acción desayunar para facilitar el desarrollo y su validación debido a las dificultades de realizar la experimentación con pacientes de Alzheimer en entornos reales. A continuación, se exponen los resultados parciales y globales del proyecto.

RESULTADOS DEL SISTEMA DE DETECCIÓN DE OBJETOS

Para comprobar el correcto funcionamiento del sistema de detección de objetos se calculó en que momento aparecía el objeto en pantalla y si detectaba ese objeto en concreto con cuánta precisión. Aunque para este proyecto únicamente se utilizó un objeto, en la tabla 2 se muestran los resultados con más objetos obtenidos de pruebas anteriores.

Tabla 2: Resultados sistema detección de objetos. Pruebas anteriores.

Objeto	Sofá	Fruta	Bolso	Mesa	Mando TV	Media
Precisión (%)	99,91	94,58	99,65	99,92	99,04	98,061

Se centró únicamente en la detección de una taza para representar la acción de “desayunar” del paciente. En la tabla 3 se muestra el resultado de la precisión con la que detecta el sistema el objeto en concreto.

Tabla 3: Resultado sistema de detección de objetos. Acción desayunar.

Objeto	Taza
Precisión (%)	97,14%

Se puede observar que en ambos casos se han obtenido buenos resultados, son valores que se acercan casi al 100% de precisión.

RESULTADOS GLOBALES ASOCIADOS AL SISTEMA FINAL DE DETECCIÓN DE ANOMALÍAS

Para conseguir el mejor sistema de detección de anomalías posible, se han realizado diferentes pruebas modificando los tres parámetros del modelo Random Cut Forest: i) número de shingle, ii) número de árboles, y iii) tamaño del árbol. Para su obtención se calculó primero el número de anomalías reales que se encuentran en los datos proporcionados y por otro lado, el número de anomalías que detectaba el algoritmo y así obtener el porcentaje de acierto. En la tabla 4 se muestra algunas de las pruebas realizadas.

Tabla 4: Resultados del algoritmo RRCF en el conjunto de datos de test sintético

Número de árboles	Número de shingle	Tamaño de árboles	Número total anomalías	Número de anomalías detectadas	Porcentaje de acierto
21	1	16	1602	1455	90,82%
20	5	15	1602	1431	89,32%
21	1	15	1602	1463	91,32%
20	1	14	1602	1490	93%
20	1	15	1602	1500	93,63%
20	2	15	1602	1411	88,07%

Una vez hecho este estudio, se adquirió que los valores óptimos para este algoritmo eran:

- Número de *shingle*: 2
- Número de árboles: 20
- Tamaño del árbol: 15

Los resultados descritos se obtienen a partir de datos sintéticos creados en base al diseño de la experimentación que se ha comentado en la sección “3.5 Diseño de experimentación”. Para que sea más eficaz el sistema de detección de anomalías se realiza una simulación con 15 días de datos de entrenamiento, para poder obtener un modelo robusto de detección de anomalías.

Para analizar estos resultados se realizaron las pruebas con 5 días, creados estos datos de forma sintética. Para la comprobación del correcto funcionamiento del algoritmo el primer paso es calcular los puntos anómalos de forma “manual”. Es decir, calcular el número de puntos que se encuentran fuera del rango que se definió en la fase de entrenamiento como rutina. Se examina cuando aparece y desaparece el objeto utilizando el rango de valores comprendido entre 1064 y 1274 (que son los obtenidos para el inicio por la experimentación real) y por otro lado valores

comprendidos entre 1214 y 1354 (que son los obtenidos para el final por la experimentación real). Estos valores se obtienen de la fase de experimentación.

De no cumplirse estos rangos se determina que ese día es atípico y por tanto se cuenta el número de puntos que se encuentran fuera de la rutina del paciente o que es un dato anómalo. En la tabla 5 se muestran los resultados obtenidos, con este ejemplo es con el que se realizará el análisis de los resultados del proyecto.

Tabla 5: Número de puntos anómalos reales

Día de pruebas	Número de puntos anómalos
Día 1	299
Día 2	205
Día 3	41
Día 4	No es una anomalía
Día 5	60

En estos 5 días de pruebas existen un total de 605 puntos que son anómalos. Una vez que se ha obtenido y etiquetado el número real de puntos atípicos, se calcula la cantidad de puntos detectados por el algoritmo como anómalos con respecto a estos mismos datos. El modelo detectó un total de 583 puntos, esto quiere decir que el algoritmo en este ejemplo consiguió un porcentaje de acierto del 96,36%.

En la figura 24, se muestra la gráfica donde se representan los cinco días que se han tomado para realizar pruebas. La línea de color rojo representa los valores creados sintéticamente que simulan un día cualquiera del paciente y la línea de color azul representa el grado de anomalía que tienen los puntos. Es decir, cuanto más alto sea éste último valor indicado, más anómalo es ese punto.

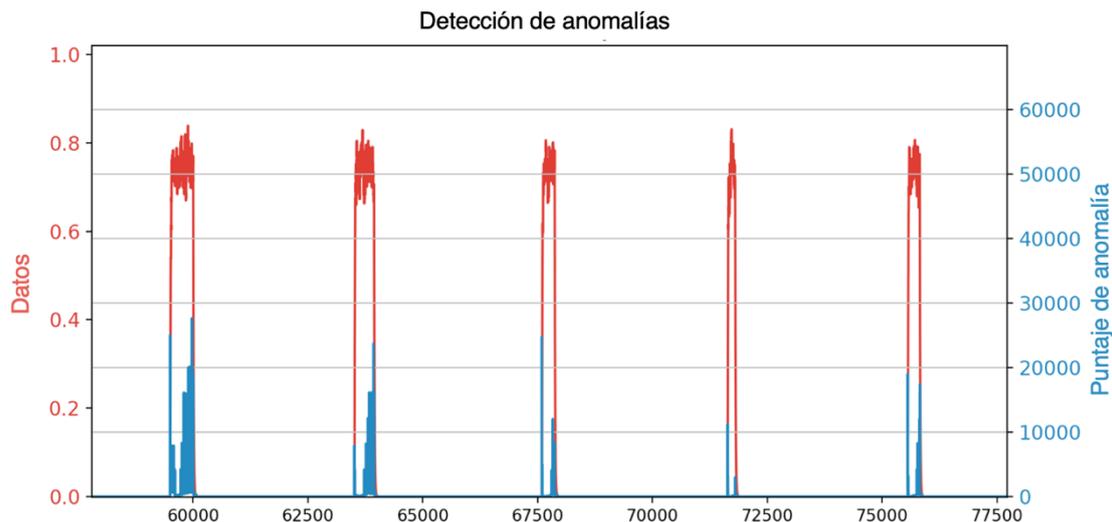


Figura 24: Gráfico anomalías de los cinco días de pruebas

Como se puede observar los dos primeros días son los que tienen más anomalías de todos y corresponde con el cálculo que se hizo anteriormente ya que el día 1 tenía 299 puntos atípicos y el día 2 tenía 205. En la figura 25 se pueden observar ampliados estos dos días. Analizamos como el grado de anomalía de la gran mayoría de puntos es muy alto con respecto a los demás.

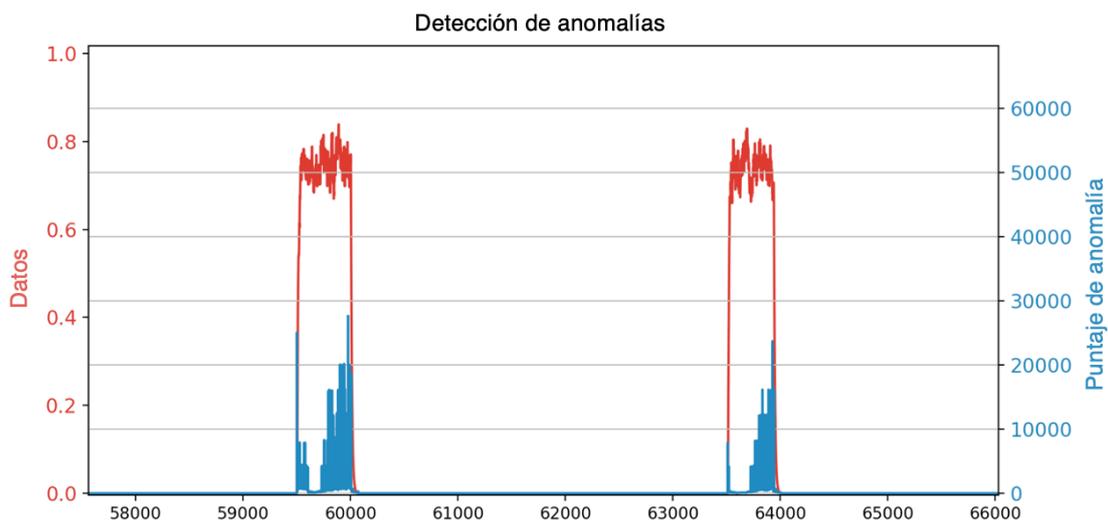


Figura 25: Gráfico anomalías de los dos primeros días de pruebas

Por otro lado, en la figura 26 se muestra el resto de los días. Se puede observar como en este caso el primer día representado en esta gráfica, que corresponde al día 3 y el último que corresponde al día 5 tienen ciertas anomalías. Esto corrobora lo calculado anteriormente donde el día 3 tenía 41 puntos anómalos y el día 5 tenía 60.

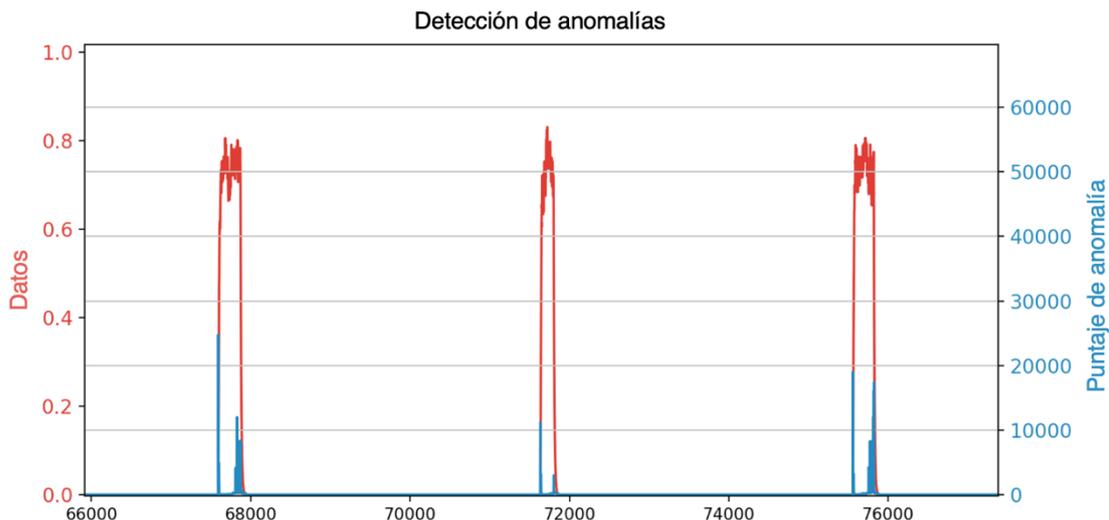


Figura 26: Gráfico anomalías de los tres últimos días de pruebas

Al contrario, el día 4 corresponde a un día dentro de la rutina. En la figura 27 se muestra la representación de este día más ampliada para comentar estos resultados. Como se puede observar los valores que tiene en este caso el grado de anomalía son nulos o muy bajos. Los picos que se distinguen son únicamente al principio y al final, es decir, cuando aparece el objeto y cuando desaparece. Para evitar que estos valores tan bajos afecten y obtener falsos positivos se ha añadido un umbral de decisión final para únicamente tener en cuenta los valores de anomalía superiores a este umbral fijándolo a través de un método de gradiente descendente. Asimismo, solo se avisa con una alarma al paciente cuando existiesen varios puntos anómalos seguidos o un mínimo tiempo anómalo (este parámetro se puede prefijar por el usuario dependiendo de las características y criticidad del evento).

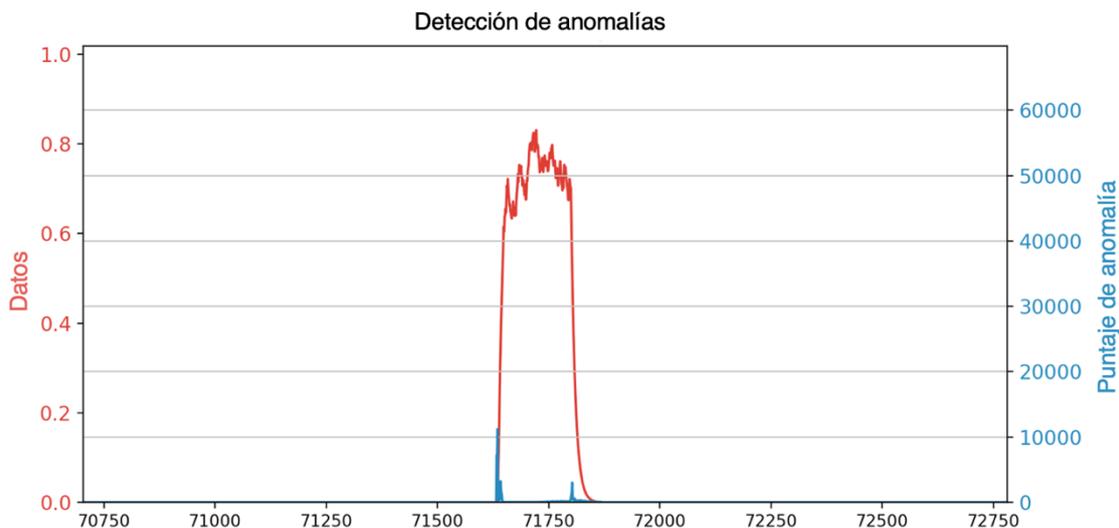


Figura 27: Gráfico anomalías del día 4 de pruebas

Ajustando el modelo propuesto el sistema final de alarma logra un 100% de éxito en la experimentación mostrada utilizando el conjunto de datos reales y sintéticos presentados. De este modo también se prueba que el sistema propuesto puede permitir mejorar a los pacientes su calidad de vida, detectando comportamientos que no se encuentren dentro de su rutina y ayudándoles a reforzarlos a través de un recordatorio o alarma.

Capítulo 5. CONCLUSIONES Y LINEAS FUTURAS

Este proyecto plantea una posible solución para conseguir mejorar la calidad de vida de las personas que padecen Alzheimer a través de un sistema de detección de comportamientos anómalos que les ayude a reforzar su rutina. Su implantación les permitirá poder estar en su casa sin tener la preocupación de olvidarse si han realizado una actividad o no, como puede ser la toma de un medicamento. Es sabido que es muy importante para estas personas seguir una rutina ya que tienen grandes dificultades para adaptarse a los cambios lo cual también les incrementa la dependencia.

Este objetivo se consigue a través del análisis de la información facilitada por un modelo cognitivo que representa la realidad observada con una cámara, y proporcionando un sistema de alertas basado en comportamientos anómalos.

Con el modelo cognitivo se ha conseguido obtener, en un entorno controlado, una representación real de lo que sucede siendo robusto a pequeñas variaciones en la escena como pueden ser reflejos u opacidades momentáneas. Para ello, este modelo utiliza características de aprendizaje y olvido asociadas a la certidumbre que provee el sistema de visión. Estas características hacen que el sistema de detección de anomalías sea también más robusto a estos cambios reduciendo las falsas alarmas.

Por otro lado, con el sistema de detección de anomalías se ha conseguido obtener un alto porcentaje de acierto con el algoritmo y detecta correctamente los comportamientos anómalos (con un acierto del 93% en detección de puntos independientes y 100% en detección de alarmas asociadas a un evento anómalo) que no se ajustan con la rutina proporcionada durante la fase de entrenamiento.

Como resultado final del proyecto se ha obtenido con éxito una prueba de concepto de un sistema de alarmas de asistencia personalizada para personas con Alzheimer. Esta prueba de concepto prueba por tanto la viabilidad de este tipo de sistemas en un entorno real aportando una mayor confianza y autonomía al paciente al permitirle realizar actividades por sí mismo sin necesidad de tener un cuidador.

Como posibles líneas futuras, se podrían implementar nuevas funcionalidades para aportar más valor a los pacientes como la incorporación de un mayor número de eventos al sistema de detección junto con un mayor número de objetos. Se podría también integrar un chatbot para mejorar la comunicación con el paciente, o se podría incluso incluir afectividad a esa comunicación. En este proyecto también se han realizado pruebas de implementación iniciales del sistema en un robot NAO como posible modo de interacción en un entorno real para dotar a la aplicación de características más humanas y mejore la interacción con el paciente. Aun así, estas pruebas se centraban en la integración del sistema más que en la interacción así que en ese ámbito sería necesario ampliar sus funcionalidades.

Por otro lado, también sería de gran utilidad la integración de un módulo de predicción de objetos para que cuando un paciente olvide donde se encuentra un objeto el sistema pueda

recordárselo. Esta nueva funcionalidad le diría donde se suele encontrar normalmente ese objeto a esa determinada hora.

Para finalizar comentar que con los resultados de este proyecto se va a contactar con la Fundación Alzheimer España (FAE) para presentarles los resultados y ver posibles vías para realizar pruebas con pacientes.

Capítulo 6. REFERENCIAS

- [1] Artaza, I. (2016). Síntomas del alzheimer, una de las enfermedades más comunes en la tercera edad. Recuperado marzo de 2020, de <https://canalsalud.imq.es/sintomas-del-alzheimer/>
- [2] Ahmad Lotfi A., Langensiepen C., Mahmoud, S. M. y Akhlaghinia M. J. (2012). Smart homes for the elderly dementia sufferers: identification and prediction of abnormal behavior. *J Ambient Intell Human Comput.* pp. 205 - 218
- [3] Radziszewski, R., Ngankam, H.K., Grégoire, V., Lorrain, D., Pigot, H. y Giroux, S. (2017). Designing calm and non-intrusive ambient assisted living system for monitoring nighttime wanderings. *International Journal of Pervasive Computing and Communications.* Vol. 13 No. 2, pp. 114-129.
- [4] Guha, S., Mishra N., Roy, G. y Schrijvers O. (2016). Robust random cut forest based anomaly detection on streams. *Proceedings of the 33rd International Conference on International Conference on Machine Learning.* Volume 48 (ICML'16). JMLR.org, 2712–2721.
- [5] Chuan Tan, S., Ming Ting, K. y Fei Liu, T. (2011). Fast Anomaly Detection for Streaming Data*. *Proceedings of the Twenty-Second International Joint Conference on Artificial Intelligence.* pp. 1511 - 1516
- [6] R. (2015). El 70% de cuidadores de personas con alzheimer está en riesgo de enfermedad. *La Vanguardia.* Recuperado febrero de 2020, de <https://www.lavanguardia.com/salud/20150921/54435391391/cuidadores-personas-alzheimer-riesgo-enfermedad.html>
- [7] Naqvi, E. (2017). Alzheimer's Disease Statistics. Recuperado febrero de 2020, de <https://alzheimersnewstoday.com/alzheimers-disease-statistics/>
- [8] World Health Organization. (2017). 10 datos sobre el envejecimiento y la salud. Recuperado febrero de 2020, de <https://www.who.int/features/factfiles/ageing/es/>
- [9] Brescané, R. (2018). ¿Cuáles son las fases del Alzheimer? Etapas y características. Recuperado marzo de 2020, de <https://knowalzheimer.com/cuantas-etapas-tiene-el-alzheimer-y-que-caracteristicas-tiene-cada-una/>
- [10] Maragall, F. P. (2019). Cuáles son los síntomas en la última etapa del Alzheimer. Recuperado marzo de 2020, de <https://blog.fpmaragall.org/ultima-etapa-del-alzheimer-sintomas>
- [11] Tomé Carruesco, G. (2016). La enfermedad de Alzheimer y la Ley de la Dependencia. [Artículo en blog]. Recuperado marzo de 2020, de <https://knowalzheimer.com/la-enfermedad-alzheimer-la-ley-la-dependencia/>
- [12] La enfermedad de Alzheimer y la demencia en España. (2018). Recuperado marzo de 2020, de <https://www.alz.org/es/demencia-alzheimer-espana.asp>

- [13]Bartos, M., Mullapudi, A. y Troutman ,S. rrcf: Implementation of the Robust Random Cut Forest algorithm for anomaly detection on streams. *Journal of Open 1 Source Software*. 4(35), 1336
- [14] MFG Labs. (2019). Anomaly detection in hostile environment. Recuperado abril de 2020, de https://medium.com/@mfg_labs/anomaly-detection-in-hostile-environment-28e4ff875621
- [15]numpy.random.normal. (2016). Recuperado mayo de 2020, de <http://lagrange.univ-lyon1.fr/docs/numpy/1.11.0/reference/generated/numpy.random.normal.html>
- [16]numpy.random.randint — NumPy v1.15 Manual. (2018). Recuperado mayo de 2020, de <https://docs.scipy.org/doc/numpy-1.15.1/reference/generated/numpy.random.randint.html>
- [17]numpy.random.random_sample — NumPy Manual. (2018). Recuperado mayo de 2020 de https://docs.scipy.org/doc/numpy-1.14.0/reference/generated/numpy.random.random_sample.html
- [18]Cerulla, N. (2017). Cómo planificar la rutina diaria. Recuperado abril de 2020, de <https://knowalzheimer.com/planificar-la-rutina-diaria/>
- [19]García-Cuesta, E., López-López, José M., Gómez-Vergel, D., Huertas-Tato, J An Adaptive Cognitive Model to Integrate Machine Learning and Visual Streaming Data. *15th International Conference on Soft Computing Models in Industrial and Environmental Applications SOCO'2020*, Burgos
- [20]APRENDE-R. Sistema conversacional afectivo basado en aprendizaje continuo en entornos cambiantes. (2018-2019), proyecto de investigación financiado por la Universidad Europea de Madrid UEM (IP Esteban García-Cuesta)
- [21]Ramaswamy, S., Rastogi, R. y Shim K. (2000). Efficient algorithms for mining outliers from large data sets. *Proceedings of the ACM SIGMOD International Conference on Management of Data*. pp. 427–438.
- [22]Aggarwal, C. C. (2013) High dimensional outlier detection: the subspace method. *Outlier Analysis*.pp. 135–167. Springer, New York, NY, USA.
- [23]Lazarevic, A. y Kumar V. (2005). Feature bagging for outlier detection. *Proceedings of theKDD-2005: 11th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. pp. 157–166. USA.
- [24]Tensorflow detection model zoo. (2018). Recuperado mayo de 2020, de https://github.com/tensorflow/models/blob/master/research/object_detection/g3doc/detecti on_model_zoo.md
- [25]Wimo, A., Winblad,B., Jonsson, L. (2007). An estimate of the total worldwide societal costs of dementia in 2005. *Alzheimer's & dementia: the journal of the Alzheimer's Association*. 3(2):81–91
- [26]Serna, A., Pigot, H., Rialle, V. (2007) Modeling the progression of Alzheimer's disease for cognitive assistance in smart homes. *User Model User Adapt Interact*. 17:415–438

-
- [27]Redmon, J., Santosh D., Girshick, R. y Farhadi, A. (2016). You Only Look Once: Unified, Real-Time Object Detection. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. pp. 779–88.
- [28]Ren, S., He, K., Girshick, R. y Sun, J. (2017). Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks. *IEEE transactions on pattern analysis and machine intelligence*. 39(6): 1137–49.
- [29]Lin, Tsung-Yi et al. (2014). “Microsoft Coco: Common Objects in Context.” In European Conference on Computer Vision, 740–55.
- [30]Howard, A.G., Zhu, M., Chen, B., Kalenichenko, D., Wang, W., Weyand, T., Andreetto y M., Adam, H. (2017). MobileNets: Efficient Convolutional Neural Networks for Mobile Vision Applications.
- [31]Open Computer Vision Library. (2018). Recuperado mayo de 2020, de <https://opencv.org>
- [32] Newell, A. (1980). Physical symbol systems. *Cognitive Science* Vol.4 pp.135–83 doi: 10.1207/s15516709cog0402_2

ANEXO A. DIAGRAMA DE GANTT

En las siguientes tablas se muestra la distribución de las tareas a lo largo de los 6 meses de realización del proyecto.

Tabla 6: Tareas para el diagrama de GANTT

Detección de objetos	
	T1 Estudio de los objetos y generación de sus imágenes
	T2 Selección y entrenamiento de modelo
	T3 Realizar código detección y superposición de objetos
Modelo cognitivo del robot	
	T4 Calcular los modelos de aprendizaje y olvido
Detección de anomalías	
	T5 Aprendizaje de situación habitual
	T6 Selección algoritmo de detección de anomalías
	T7 Diseño de la experimentación
	T8 Modelo de detección de anomalías basado en el sistema cognitivo
	T9 Diseño de pruebas
Documentación	
	T10 Documento final

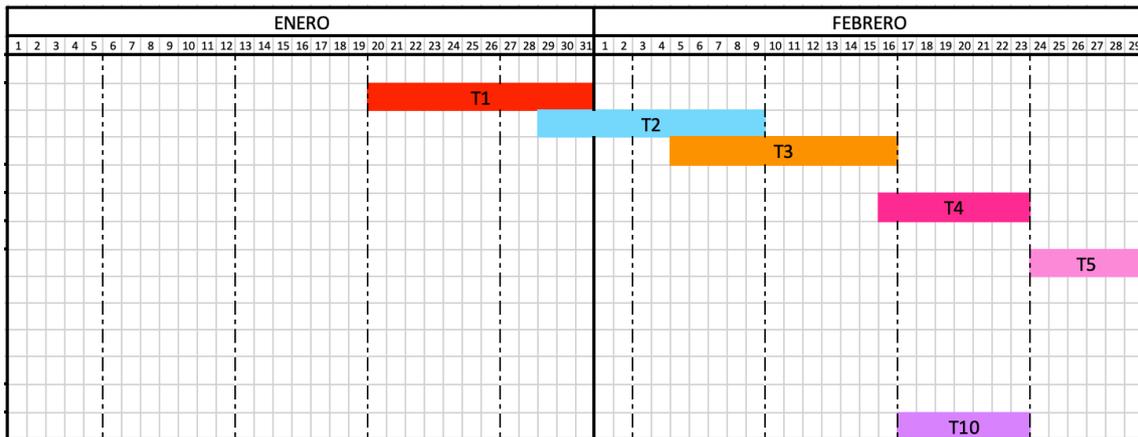


Figura 28: Planificación enero y febrero

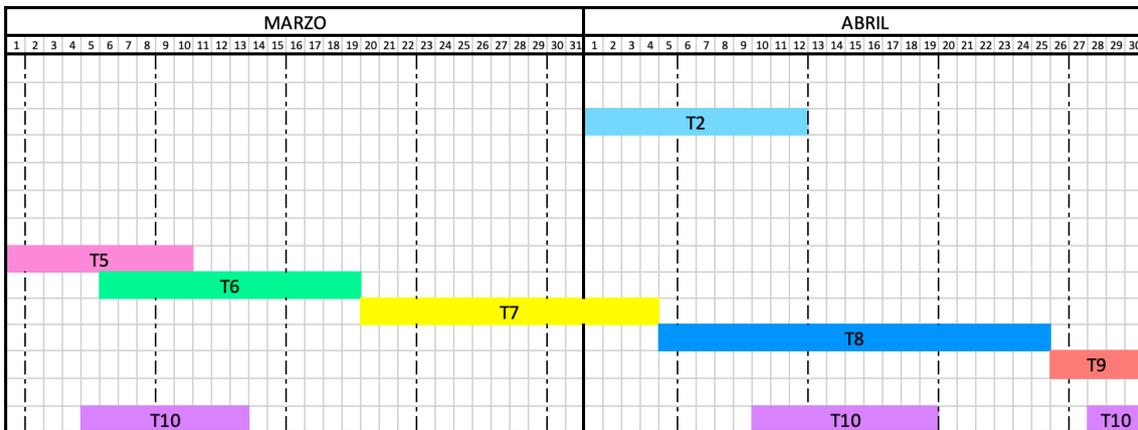


Figura 29: Planificación marzo y abril

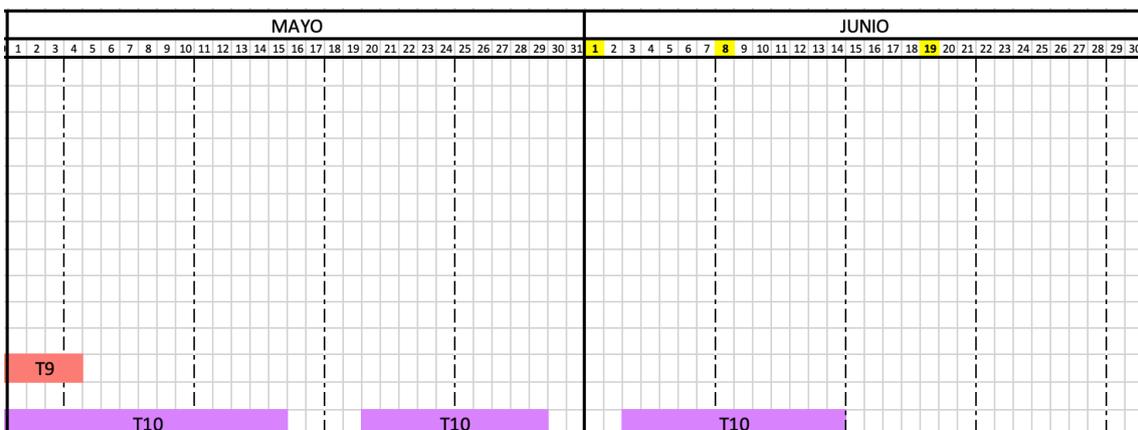


Figura 30: Planificación mayo y junio

ANEXO B. DESCRIPCIÓN DE TAREAS

En este punto se exponen las tareas específicas de los objetivos del proyecto anteriormente explicados con cada uno de sus hitos.

Tabla 7: Tarea 1 del proyecto

<u>T1. Estudio de los objetos y generación de sus imágenes</u>
Se realizará un estudio de cual son los objetos más relevantes para este tipo de pacientes. Una vez identificados, se tomará fotografías de estos desde diferentes perspectivas, luminosidad y distancia.
HITO: Obtención de una base de datos con las imágenes de los objetos a detectar

Tabla 8: Tarea 2 del proyecto

<u>T2. Selección y entrenamiento del modelo de reconocimiento de objetos</u>
Existen numerosos algoritmos para realizar la detección de objetos. Para elegir el más óptimo y el que se ajuste más a las necesidades del proyecto primero habrá que realizar un estudio de ellos. Una vez elegido el algoritmo, para optimizar la detección de los objetos se pueden modificar algunos de sus parámetros, además de adaptarlo para el proyecto. Una vez hecho esto, se entrenó el modelo con esos cambios.
HITO: Modelo entrenado

Tabla 9: Tarea 3 del proyecto

<u>T3. Realizar código de detección de relaciones por superposición</u>
Para detectar los objetos en tiempo real desde la cámara se tuvo que realizar el código. Además de implementar una función para calcular la superposición entre los objetos.
HITO: Código de detección de superposiciones entre objetos

Tabla 10: Tarea 4 del proyecto

<u>T4. Calcular modelos de aprendizaje y olvido</u>
Integrar en el código el aprendizaje si aparece el objeto y el olvido cuando ya no aparezca ese objeto.
HITO: Implementación del modelo cognitivo de aprendizaje y olvido

Tabla 11: Tarea 5 del proyecto

<u>T5. Aprendizaje de situación habitual</u>
Se analizará unas situaciones que se denominarán como habituales con el fin de que el modelo sea capaz de detectar las anomalías.
HITO: Creación de modelo situación habitual

Tabla 12: Tarea 6 del proyecto

<u>T6. Selección algoritmo de detección de anomalías</u>
Realizar un estudio sobre los diferentes algoritmos de predicción para seleccionar el que mejor encaje para este proyecto
HITO: Algoritmo de predicción

Tabla 13: Tarea 7 del proyecto

<u>T7. Diseño de la experimentación</u>
Crear datos sintéticos basados en las experimentaciones realizadas, para conseguir un modelo que detecte correctamente las anomalías.
HITO: Datos sintéticos

Tabla 14: Tarea 8 del proyecto

<u>T8. Modelo de detección de anomalías basado en el sistema cognitivo</u>
Crear modelo para detectar las anomalías que se encuentran en la situación actual respecto a las situaciones habituales en las que se ha entrenado el modelo.
HITO: Modelo detección anomalías

Tabla 15: Tarea 9 del proyecto

<u>T9. Diseño de pruebas</u>
Comprobar el correcto funcionamiento del sistema de detección de anomalías realizando una batería de pruebas. Modificar los diferentes parámetros del algoritmo seleccionado para conseguir el mejor porcentaje de acierto
HITO: Porcentaje de acierto

ANEXO C. PRESUPUESTO

Este presupuesto se realizó en base a la ejecución del proyecto en un entorno real. A continuación se describen los elementos que serían necesarios y su coste final.

Para realizar la grabación será necesario contar con una cámara 360º para poder obtener un vídeo nítido de toda la habitación donde extraer la rutina diaria del paciente. Como opción de menor presupuesto, podríamos optar con una cámara como la “Xiaomi Mi Home Security Camera 360” que vale 36,99 €. Y como opción de mayor presupuesto, la cámara “Red ABUS TVIP44510” que vale 202,06€. Estas dos se diferencian notablemente en cuanto a la calidad de grabación.

Para poder realizar el procesamiento de las imágenes se necesita un ordenador con bastante memoria RAM para que pueda soportar los grandes volúmenes de datos obtenidos diariamente a través de la cámara. Como opción de menor presupuesto, podríamos optar por un ordenador como el “MSI Modern 14 A10M-600XES” que vale 649€. Y como opción de mayor presupuesto, un ordenador como el “MSI GF65 Thin 9SEXR-236XES” que vale 1.252,51 €. Los dos tienen 16 GB de RAM pero se diferencian en cuanto al icore, tarjeta gráfica, almacenamiento...

Las herramientas de software utilizadas para el desarrollo de este sistema de anomalías han sido de software libre, por lo tanto no se ha tenido que invertir nada de dinero en este aspecto.

Por otro lado, un factor esencial para la realización de este proyecto es el personal, hay que contar con los recursos humanos necesarios para realizar este proyecto en un entorno real. En este caso serían los propios pacientes tanto con los que se haría la experimentación como los que se benefician del producto. Además de contar con un técnico para para la integración de este proyecto en un entorno real.

En la siguiente tabla se muestran los valores aproximados de la implementación de este proyecto en un entorno real.

Tabla 16: Estimación del presupuesto

Tipo de coste	Valor	Comentarios
Horas de trabajo en el proyecto	615 horas	Se estima que la duración del proyecto sea de unos 3 meses.
Equipo técnico utilizado	649 € - 1.252,51 €	Dependiendo de las prestaciones que se necesiten para hacer el proyecto se necesitará un ordenador más potente o uno con menos prestaciones.

Software utilizado	0 €	El coste es nulo ya que para la realización de este proyecto se han utilizado únicamente herramientas de software libre
Estudios e informes	0 €	Los informes y estudios de investigación a los que se ha accedido para la realización de este proyecto no han tenido ningún coste.
Materiales empleados	36,99 € - 202,06 €	Dependiendo de las prestaciones que se necesiten para hacer la detección de objetos se necesitará una cámara más potente o uno con menos prestaciones

