

PONTIFICIA UNIVERSIDAD CATÓLICA DEL PERÚ

FACULTAD DE CIENCIAS E INGENIERÍA



**DISEÑO DE UN SISTEMA DE COMPUTACIÓN DE BORDE PARA EL
MONITOREO DE DISPONIBILIDAD DE ESTACIONAMIENTOS DE UN CAMPUS
UNIVERSITARIO BASADO EN TÉCNICAS DE IA**

Tesis para obtener el título profesional de Ingeniero de las Telecomunicaciones

AUTORES:

Stuart David Cabrera Altamirano

Edgar Stiven Quispe Velasquez

ASESOR:

Luis Ángelo Velarde Criado

Lima, junio, 2025

Informe de Similitud

Yo,Luis Angelo Velarde Criado.....,
docente de la Facultad deCiencias e Ingeniería..... de la Pontificia
Universidad Católica del Perú, asesor(a) de la tesis/el trabajo de investigación titulado
.....Diseño de un Sistema de Computación de Borde para el Monitoreo de Disponibilidad de Estacionamientos
de un Campus Universitario basado en técnicas de IA
.....,

del/de la autor(a)/ de los(as) autores(as)

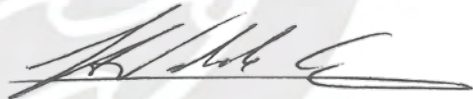
Stuart David Cabrera Altamirano,

Edgar Stiven Quispe Velasquez,

dejo constancia de lo siguiente:

- El mencionado documento tiene un índice de puntuación de similitud de 10%. Así lo consigna el reporte de similitud emitido por el software *Turnitin* el 24/06/2025.
- He revisado con detalle dicho reporte y la Tesis o Trabajo de Suficiencia Profesional, y no se advierte indicios de plagio.
- Las citas a otros autores y sus respectivas referencias cumplen con las pautas académicas.

Lugar y fecha: ...San Miguel24/06/2025.....

Apellidos y nombres del asesor / de la asesora: <u>Velarde Criado, Luis Angelo</u>	
DNI: 09646736	Firma 
ORCID: 0000-0003-4879-9918	



Edgar:

A mis padres, por su apoyo incondicional.

A mis hermanos y sobrinos, por estar siempre a mi lado.

Y, de manera especial, a mi hermana Yoshi y a mi sobrino Mathias, por ser mi luz y esperanza cuando más lo necesitaba.

Stuart:

A mis padres y a mi hermano, por su paciencia y apoyo incondicional en mi desarrollo profesional y personal.

Y, en especial, a mi abuela Dora y a los familiares que ya no están presentes físicamente; sus recuerdos me mantienen fuerte e inspiran mi voluntad para seguir adelante.

Resumen

El presente proyecto de tesis consiste en el diseño de un sistema de computación basado en técnicas de inteligencia artificial (IA), para el monitoreo de la disponibilidad de estacionamientos de campo abierto mediante cámaras dentro un campus universitario.

En el primer capítulo, se realiza un análisis sobre el desarrollo de la IA, así mismo, se presenta el contexto actual de los estacionamientos en la sociedad, las diversas soluciones implementadas, los objetivos y el alcance planteado en este proyecto. El segundo capítulo establece el modelo teórico a diseñar y se presentan los conocimientos generales de cada una de las tecnologías implicadas, se abordan conceptos de redes CNN, tecnologías de transmisión de datos y soluciones de nube. El tercer capítulo muestra la elección de los algoritmos de borde y la metodología elaborada para el diseño completo de la red, dividida en etapas de preprocesamiento, procesamiento IA, transmisión, almacenamiento y presentación. Se establecen los parámetros críticos que medirán el desempeño del sistema y se elaboran los costos por el servicio de implementación y mantenimiento tanto a nivel de este proyecto como a un costo estimado en un escenario comercial.

Por último, el cuarto capítulo define un plan de pruebas para el óptimo funcionamiento del sistema y se comparan los parámetros críticos en dos escenarios con distinta cantidad de cámaras para una misma zona de parqueo. Se brindan las conclusiones obtenidas en cada etapa y recomendaciones para una puesta en marcha a gran escala.

Índice

Resumen	i
Índice.....	ii
Lista de figuras	iv
Lista de tablas	vi
Introducción.....	1
Capítulo 1: Contexto actual del servicio de estacionamientos vehiculares dentro de la universidad ...	2
1.1 Desarrollo de las telecomunicaciones e inteligencia artificial en el país	2
1.2 Planteamiento de la problemática: estacionamientos	4
1.2.1 Contexto local.	5
1.2.2 Contexto organizacional.....	6
1.3 Estado del arte.....	9
1.3.1 Detección basada en sensores.	10
1.3.2 Detección basada en procesamiento de imágenes.	11
1.4 Objetivos.....	12
1.4.1 Objetivo general.....	12
1.4.2 Objetivos específicos.....	12
1.5 Alcance	13
Capítulo 2: Fundamento teórico sobre las tecnologías implicadas en una red inalámbrica de dispositivos con inteligencia artificial	15
2.1 Modelo teórico de la solución	15
2.2 Conceptos básicos	17
2.2.1 Inteligencia artificial.....	18
2.2.2 Ordenador de placa reducida (SBC).....	24
2.2.3 Tecnologías para transmisión de datos.....	27
2.2.4 Cloud computing.	34
2.3 Antecedentes.....	38
Capítulo 3: Metodología para el diseño de una red inalámbrica de dispositivos con inteligencia artificial aplicada en un estacionamiento del campus universitario	40
3.1 Requerimientos de diseño.....	40
3.2 Determinación de tiempo crítico.....	43
3.3 Elección del método de preprocesamiento	44
3.4 Elección del algoritmo de IA	48
3.4.1 Tareas de visión computacional.....	48

3.4.2	Dataset utilizado.	49
3.4.3	Pruebas con algoritmos escogidos.	51
3.5	Elección y configuración del dispositivo SBC.	57
3.5.1	Evaluación del dispositivo.	57
3.5.2	Configuración de la disponibilidad.	60
3.6	Elección y configuración del sistema de transporte.	61
3.7	Elección del proveedor y desarrollo del sistema en nube.	65
3.7.1	Elección del proveedor y región.	65
3.7.2	Arquitectura de la solución seleccionada.	68
3.8	Diseño de interfaz.	76
3.9	Determinación del tiempo de muestreo.	80
3.10	Elaboración de costos de servicio.	81
Capítulo 4: Prueba del sistema de disponibilidad y evaluación de los parámetros críticos.		83
4.1	Plan de pruebas.	83
4.1.1	Pruebas de funcionamiento.	84
4.1.2	Cobertura y ubicación de dispositivos.	89
4.1.3	Estadísticas.	90
Conclusiones.		92
Recomendaciones.		94
Bibliografía.		95
Anexo I.		100
Anexo II.		113
Anexo III.		124
Anexo IV.		127

Lista de figuras

Figura 1.1 La ventana de oportunidad de la inteligencia artificial (IA) [4].....	4
Figura 1.2 Subcategorías de la tecnología IA [5]	4
Figura 1.3 Mapa de estacionamiento de autos del campus PUCP [12]	7
Figura 1.4 Total de espacios libres por hora para el estacionamiento B – Mayo 2019 [Elaboración propia].....	9
Figura 1.5 Total de espacios libres por hora para el estacionamiento B – Octubre 2019 [Elaboración propia].....	9
Figura 1.6 Distribución de elementos de red [18]	11
Figura 1.7 Detección de autos a través de OCR [20].....	12
Figura 1.8 Estacionamiento B – PUCP [21]	13
Figura 2.1 Esquema del modelo teórico [Elaboración propia]	17
Figura 2.2 Modelo de detección YOLO [30]	19
Figura 2.3 Arquitectura procesamiento Fast R-CNN [32]	20
Figura 2.4 R-CNN vs SPP vs Fast R-CNN [31]	21
Figura 2.5 Arquitectura Faster R-CNN [31]	22
Figura 2.6 Raspberry pi 3B+ [38].....	25
Figura 2.7 Modelo Nvidia Jetson Nano [42].....	25
Figura 2.8 Intel 8265 para módulo Nvidia Jetson Nano [44]	26
Figura 2.9 Monitoreo de una carga pesada [52].....	28
Figura 2.10 Pila de protocolos de tecnologías de red IOT [53].....	29
Figura 2.11 Modelo de comunicación MQTT [56]	31
Figura 2.12 Arquitectura de red Sigfox [58].....	32
Figura 2.13 Autenticación entre extremos de red Sigfox [60]	34
Figura 2.14 Enlaces VPN en red Sigfox [60]	34
Figura 2.15 Plataforma de servicios AWS [67].....	37
Figura 2.16 Plataforma de servicios Microsoft Azure [69].....	38
Figura 3.1 Trayectoria recorrida hasta el estacionamiento B [72]	44
Figura 3.2 Espacios de parqueo [Elaboración propia]	45
Figura 3.3 Ejecución de etapa inicial [Elaboración propia].....	46
Figura 3.4 Ejecución de etapas posteriores [Elaboración propia]	46
Figura 3.5 Capturas del estacionamiento en cámaras 1, 3, 5 y 7 [75]	50
Figura 3.6 Imagen original (100x750) [75].....	50
Figura 3.7 Capturas del estacionamiento a las 5 p.m. y 7 p.m. [75].....	51
Figura 3.8 Capturas del estacionamiento B a las 8 p.m. [Elaboración propia]	51
Figura 3.9 de Izq. a Der. Detección con YOLO en diversas condiciones: nublado, lluvioso, soleado y noche. [Elaboración propia].....	53
Figura 3.10 Izquierda a Derecha: detección con <i>Faster</i> R-CNN en diversas condiciones: nublado, lluvioso, soleado y noche [Elaboración propia]	55
Figura 3.11 Disponibilidad del estacionamiento [Elaboración propia].....	61
Figura 3.12 Política dispositivo SBC [Elaboración propia].....	64
Figura 3.13 Diagrama de flujo computación de borde [Elaboración propia]	65
Figura 3.14 Reportes mostrados por Gartner y Forrester [79], [80]	67
Figura 3.15 Latencia navegador – región de AWS mediante Cloudping.info [Elaboración propia].....	68
Figura 3.16 Arquitectura Cloud AWS [Elaboración propia]	68
Figura 3.17 Regla configurada para el tópico “estacionamientos” [Elaboración propia].....	72

Figura 3.18 Primer patrón de acceso [Elaboración propia]	73
Figura 3.19 Primer patrón de acceso - json [Elaboración propia]	73
Figura 3.20 Segundo patrón de acceso [Elaboración propia]	74
Figura 3.21 Segundo patrón de acceso - json [Elaboración propia]	74
Figura 3.22 Tercer patrón de acceso [Elaboración propia].....	75
Figura 3.23 Registros lambda captura de ocupabilidad [Elaboración propia]	76
Figura 3.24 Vista principal - dispositivo móvil [Elaboración propia].....	77
Figura 3.25 Vista de mapa de disponibilidad - dispositivo móvil [Elaboración propia]	78
Figura 3.26 Vista de perfil de usuario - dispositivo móvil [Elaboración propia]	78
Figura 3.27 Reporte por zona de estacionamiento [Elaboración propia].....	79
Figura 3.28 Reporte por espacio de estacionamiento [Elaboración propia]	80
Figura 3.29 Panel del administrador - administración de usuarios. [Elaboración propia]	80
Figura 3.30 Tiempos obtenidos durante la ejecución [Elaboración propia]	81
Figura 4.1 Cámara 7 overcast 2015-11-25_1419 [Elaboración propia]	84
Figura 4.2 Elementos creados en dynamodb [Elaboración propia].....	85
Figura 4.3 Primer horario – dispositivo SBC [Elaboración propia].....	85
Figura 4.4 Primer horario – app móvil [Elaboración propia]	85
Figura 4.5 Segundo horario – dispositivo SBC [Elaboración propia].....	86
Figura 4.6 Segundo horario – app móvil [Elaboración propia]	86
Figura 4.7 Tiempos de procesamiento obtenidos [Elaboración propia].....	87
Figura 4.8 Distribución de los dispositivos en el estacionamiento B [Elaboración propia]	90
Figura 4.9 Gráfico de disponibilidad [Elaboración propia]	91
Figura 4.10 Gráfico de ocupabilidad por espacio [Elaboración propia].....	91



Lista de tablas

Tabla 1.1 Consecuencias por el déficit de estacionamientos [Elaboración Propia]	6
Tabla 1.2 Asistencia de los espacios reservados con letrero - Febrero 2020 [14].....	8
Tabla 2.1 Edge computing: ventajas y desventajas [23]	16
Tabla 2.2 Confusion Matrix and Related Performance Measures [36].....	24
Tabla 2.3 Comparativa Uplink - Downlink [Elaboración propia].....	33
Tabla 2.4 Comparativa on premise vs cloud [64].....	35
Tabla 3.1 Requerimientos del usuario [Elaboración propia]	41
Tabla 3.2 Ejecución de preprocesamiento [Elaboración propia].....	47
Tabla 3.3 Comparación de tareas de visión por computadora [31], [33], [73].....	48
Tabla 3.4 Precisión promedio obtenida por cada posición de parqueo - YOLO [Elaboración propia] .	53
Tabla 3.5 Parámetros promedio YOLO [Elaboración propia].....	54
Tabla 3.6 Parámetros máximos YOLO [Elaboración propia].....	55
Tabla 3.7 Precisión promedio obtenida por cada posición de parqueo – Faster R-CNN [Elaboración propia].....	56
Tabla 3.8 Parámetros promedio Faster R-CNN [Elaboración propia].....	57
Tabla 3.9 Parámetros máximos Faster R-CNN [Elaboración propia]	57
Tabla 3.10 Comparación YOLO vs Faster R-CNN [Elaboración propia].....	57
Tabla 3.11 Comparación de dispositivos SBC [Elaboración propia].....	59
Tabla 3.12 Comparación de módulos cámara [Elaboración propia].....	60
Tabla 3.13 Características de las tecnologías de transmisión [53]	62
Tabla 3.14 Comparación de proveedores FAAS [77][78].....	66
Tabla 3.15 Permisos de cada tipo de usuario [Elaboración propia]	69
Tabla 3.16 Solucionadores implementados y usuarios asociados [Elaboración propia]	71
Tabla 3.17 Costos del proyecto [Elaboración propia].....	82
Tabla 4.1 Precisión dataset overcast [Elaboración propia].....	87
Tabla 4.2 Valores de precisión por fecha [Elaboración propia]	88

Introducción

Junto al creciente desarrollo de la urbe en los últimos años, el transporte público se ha convertido en una de las causas de mayor desagrado por la población, inclusive al momento de aparcar los autos en los estacionamientos. El aumento de densidad vehicular y la mala administración de los centros de aparcamiento originan un mayor tiempo de búsqueda de una vacante libre por parte del conductor ocasionando malestares a nivel personal (estrés, ira, preocupación y ansiedad), económico (consumo excesivo de combustible) y ambiental (mayor emisión de dióxido de carbono). Se implementaron en diversas zonas del país sistemas de monitoreo de estacionamientos basados en sensores magnéticos y ultrasónicos; sin embargo, esto conlleva gastos elevados de implementación y mantenimiento, debido a la instalación al ras del suelo o arriba de cada vacante. Por ello, se requiere de un mejor enfoque: el desarrollo de un sistema de monitoreo de disponibilidad basado en el procesamiento de imágenes.

El actual proyecto de tesis consta del diseño de un sistema de monitoreo de disponibilidad de estacionamientos en tiempo real basado en el uso de algoritmos de inteligencia artificial sobre en 5 etapas: evaluación de un algoritmo CNN para la detección de autos, elección de un hardware SBC, selección de una red de transporte de información, el almacenamiento y distribución de datos en nube y la visualización mediante una app. Con ello, se pretende generar una plataforma de administración organizada para la visualización y consulta de gráficos estadísticos de la disponibilidad. El alcance del sistema será delimitado para el estacionamiento B del campus universitario, el cual dispone de 30 vacantes de parqueo. El sistema será evaluado mediante los siguientes parámetros: el tiempo crítico de actualización del sistema entre extremos y la precisión con la que los vehículos son detectados.

Capítulo 1: Contexto actual del servicio de estacionamientos vehiculares dentro de la universidad

En el contexto de los avances tecnológicos, las telecomunicaciones y la inteligencia artificial vienen innovando diversos sectores disruptivamente; sin embargo, uno de sus constantes desafíos es la gestión eficiente de los estacionamientos en entornos urbanos cada vez más saturados. En el presente capítulo, se aborda el desarrollo tecnológico en el país, la problemática relacionada con los estacionamientos, el estado del arte y establece los objetivos y el alcance de la investigación.

1.1 Desarrollo de las telecomunicaciones e inteligencia artificial en el país

El sector de las telecomunicaciones va adquiriendo mayor penetración e importancia en el Perú; las telecomunicaciones se están convirtiendo en tema de interés durante las dos últimas décadas y es que los beneficios que brinda a la sociedad son percibidos fácilmente, aunque inicialmente se requieran de inversiones considerables para el correcto aprovechamiento del sector. Con un aporte de aproximadamente el 5% al PBI nacional, se espera en el Perú una

inversión de \$1,528 millones para el 2019 teniendo un incremento de 46,5% respecto al 2018 [1]; sin embargo, parte del desarrollo de un país conlleva utilizar las herramientas que la globalización trae consigo a fin de buscar un mejor desempeño tales como las tecnologías, la industria, entre otras.

Parte de los servicios de telecomunicaciones en el Perú son los de valor añadido, donde, según el Osiptel, “utilizando como soporte los servicios portadores o finales de difusión, añaden alguna característica o facilidad al servicio que les sirve de base” [2], de esta clase de servicios deriva el uso de la tecnología IoT (*Internet of Things*) que promete una masificación futura al interconectar distintos dispositivos para fines concretos y optimizadores.

De forma horizontal a este desarrollo, la Inteligencia Artificial (IA) se encuentra entre las últimas tendencias en transformación digital que impacta en la sociedad, además de las tecnologías de nube. La irrupción de la IA en la denominada cuarta revolución industrial la encabeza como una Tecnología de Propósito General (TPG), es decir que afecta a todos los sectores, impacta a la producción de toda la economía, es masiva; un estudio realizado por la firma de consultoría Duckier Frontier y comisionado por Microsoft Latinoamérica aborda que en un escenario positivo de máxima adopción de la IA el país podría incrementar la tasa de crecimiento del PBI hasta el 7,9% de expansión anual [3], y es que el país obtiene grandes beneficios con el aprovechamiento de las oportunidades que trae consigo. La figura 1.1 muestra el panorama de cambio respecto a la IA como agente para el crecimiento, esta ventana de oportunidad se basa en dos ejes de diferenciación: el uso de *Machine Learning* dentro de la IA para el tratamiento de enormes fuentes de información y su aplicación en diversos sectores (Comercio, Entidades Bancarias, Agricultura, Manufactura, Gobierno, Tránsito, etc.). Actualmente nos encontramos en el punto de inflexión. [4]

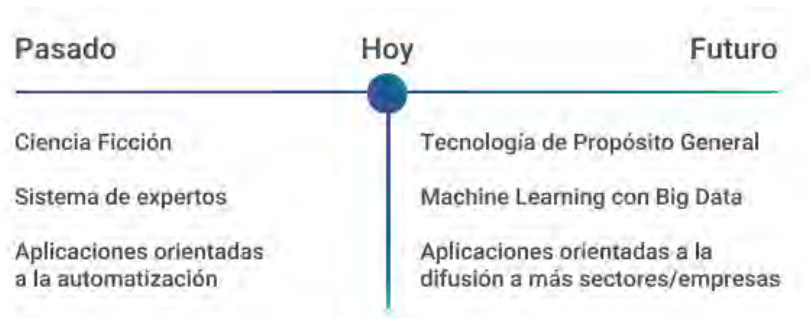


Figura 1.1 La ventana de oportunidad de la inteligencia artificial (IA) [4]

Además de lo anterior mencionado, es importante señalar los campos de aplicación de la IA, que de acuerdo con un estudio realizado por O'Reilly sobre el mercado de la IA en Norteamérica (figura 1.2) establece 5 subcategorías: Chabot, Vehículos Autónomos, Computación Cognitiva, Técnicas de Procesamiento de Imágenes y la Comprensión del Lenguaje Natural, de la cual la penúltima tiene gran importancia debido a que es una de las áreas con mayor cantidad de empresas invirtiendo en EE. UU. De esta manera, hoy en día se cuenta con sistemas que ofrecen mayor precisión y confiabilidad para la detección y clasificación de humanos, animales y diversos tipos de objetos, además se dispone de una larga lista de modelos pre entrenados por plataformas Cloud para su libre uso. [5], [6]



Figura 1.2 Subcategorías de la tecnología IA [5]

1.2 Planteamiento de la problemática: estacionamientos

Como parte del desarrollo del contexto actual, es importante explicar los lineamientos en los cuales se realiza el servicio de estacionamiento en el país, enfocado en una urbanización como Lima Metropolitana, definir el aspecto social involucrado y las consecuencias que

desencadena, para luego interiorizar en el servicio de estacionamiento ofrecido por la Pontificia Universidad Católica del Perú (PUCP).

1.2.1 Contexto local.

El servicio de estacionamientos vehicular se define según la Ley 29 461 como aquel acuerdo en virtud del cual una persona natural o jurídica, titular de un establecimiento acondicionado para el estacionamiento de vehículos, cede un espacio determinado para estacionamiento, según las condiciones ofrecidas por el titular y conforme a los alcances previstos en la ley [7]. El servicio de estacionamiento actualmente se ofrece de dos formas:

- **Servicio Principal:** Establecimiento destinado para el uso único y exclusivo del servicio de estacionamiento, brinda un espacio a un tercero.
- **Servicio Complementario:** Establecimiento destinado a una actividad distinta al servicio de estacionamiento; sin embargo, brinda de forma complementaria un espacio a un tercero.

En un escenario urbanístico donde el transporte público es considerado el segundo problema más grande por los ciudadanos de Lima y Callao (46,2% de percepción según una encuesta realizada por Lima Cómo Vamos a mediados del 2019) [8], la informalidad, el mal diseño vial, la mala gestión, falta de fiscalización y el incumplimiento de normas son puntos clave de los cuales radica el tránsito caótico. Acompañado a esto, se encuentra la abundancia de vehículos en la urbe, ya que Lima y Callao poseen el 66% del total de vehículos a nivel nacional. [9]

Con este panorama, ocurren problemas en cuanto a las áreas de aparcamiento en la ciudad y es que existen vehículos que se estacionan en zonas rígidas o en veredas, y más allá de ser un mal comportamiento por parte de los conductores, es que existe una insuficiencia de áreas automatizadas de estacionamientos y un centro control integral de estas en ciudad. Como claro ejemplo, un estudio realizado por la ONG Luz Ámbar en el 2016 reveló que Lima tiene un

déficit de al menos 45 mil espacios para estacionar, un problema que se ve evidenciado en distritos como San Isidro o Miraflores, donde los choferes no tienen reparos en dejar sus coches en la vía pública, generando congestión. [10]

Ante la falta de estacionamientos, se generan diversas consecuencias que se agravan inclusive durante las horas con mayor flujo vehicular. En la Tabla 1.1, se muestran las consecuencias a nivel personal: el estrés ocasionado al buscar un espacio disponible, la ira, preocupación y ansiedad. Todos conllevan, a largo plazo, problemas graves en la salud. Respecto al medio ambiente, se establece como factores importantes a considerar la emisión de gases contaminantes de un vehículo durante su recorrido y la contaminación sonora.

Tabla 1.1 Consecuencias por el déficit de estacionamientos [Elaboración Propia]

PERSONA	MEDIO AMBIENTE
<ul style="list-style-type: none"> • Preocupación en los conductores y transeúntes respecto al cambio en sus horarios. • Estrés debido a recorridos prolongados. • Disminución en la productividad. • Inactividad física, más propenso a enfermedades cardiovasculares y diabetes • Mayor consumo de combustible durante la búsqueda de espacios o inmovilización por ambientes saturados. 	<ul style="list-style-type: none"> • Liberación de partículas contaminantes que en el aire ocasionan enfermedades cardiovasculares y respiratorias. A largo plazo puede ser cancerígeno. • Contaminación sonora, exposición a altos niveles de ruido que puede causar sordera parcial o total

1.2.2 Contexto organizacional.

La Pontificia Universidad Católica del Perú (PUCP), ubicada en el distrito de San Miguel en la ciudad de Lima, es una universidad con autonomía normativa, académica, administrativa

y económica que permite el desarrollo de actividades de carácter educativo, cultural, humanista, científico y recreativo. Cuenta con un directorio de servicios en temas académicos, consultoría, capacitación, salud y bienestar e instalaciones, de este último deriva el servicio de estacionamiento para autos, motos y bicicletas, servicio complementario como se definió anteriormente [11]. Actualmente, consta de 12 zonas de aparcamiento para autos, como se muestra en la figura 1.3, separadas en zonas alrededor del perímetro interno del campus, en su gran mayoría ambientes amplios abiertos (*outdoor*) a excepción del estacionamiento subterráneo debajo el Pabellón A (*indoor*). [12]

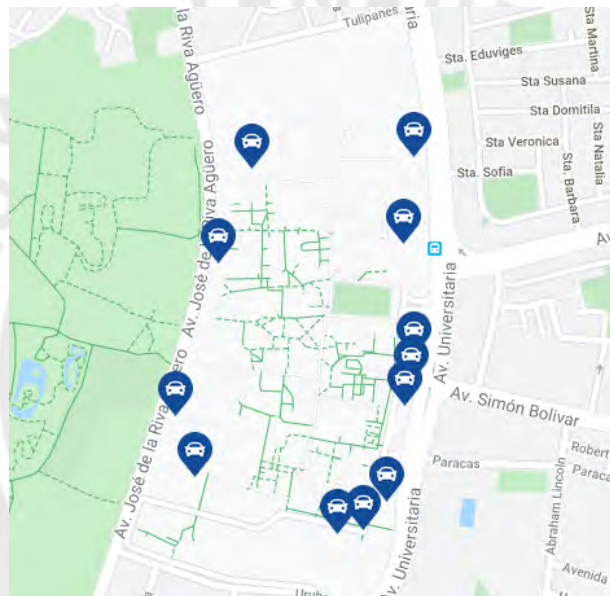


Figura 1.3 Mapa de estacionamiento de autos del campus PUCP [12]

La universidad cuenta actualmente, mediante la administración de la Oficina de Servicios Generales, con un contrato anual con el proveedor de servicios Los Portales S.A para realizar la función de administrador de las playas de estacionamiento, además de un Comité de Movilidad que evalúa con Recursos Humanos quienes son los usuarios con derecho para poder utilizar un espacio de estacionamiento dentro de la universidad; se dispone de dos puertas de ingreso para los usuarios en general, estas entradas son por la Av. Universitaria y la Av. Riva Agüero. [13]

Como parte de las funciones de administración, Los Portales se encarga del control y seguridad de los vehículos en las zonas de estacionamiento y realiza reportes de gestión mensual con información sobre la cobertura del personal, las incidencias presentadas, los auxilios mecánicos, las reservas de espacios y la asistencia diaria de los espacios reservados con letrero (en la Tabla 1.2 se muestra un resumen). Es importante aclarar que no se tiene un control sobre la ocupación de todos los espacios del estacionamiento a excepción de los reservados. [13], [14]

Tabla 1.2 Asistencia de los espacios reservados con letrero - Febrero 2020 [14]

PLAYA	CARTEL	SI	NO
O	JEFA DEPARTAMENTO ACADEMICO DE ARTES ESCENICAS	19	10
O	PRESIDENTE COM. GOB. FACULTAD DE ARTES ESCENICAS	5	24
O	RESERVADO - FACULTAD DE ARTES ESCENICAS	5	24
O	DECANO FACULTAD DE ARTE	16	13
O	JEFE DEPARTAMENTO DE ARTE	12	17
B	RESERVADO RECTORADO	0	29
B	RESERVADO RECTORADO	10	19
B	RESERVADO	15	14
B	RESERVADO	9	20
B	RESERVADO	13	16
B	RESERVADO	0	29
B	RESERVADO	0	29
B	RESERVADO con cono	6	23
B	RESERVADO con cono	15	14
B	RESERVADO con cono	20	9
B	TAXI	0	0
B	TAXI	0	0
B	TAXI	0	0
B	TAXI	0	0

Respecto al flujo de información de disponibilidad, el personal de Los Portales (anfitrión) reporta cada hora al centro de control la cantidad de espacios disponibles en cada estacionamiento de la universidad. De esta manera, en la puerta de ingreso, se conoce cuál es la disponibilidad de cada zona para finalmente guiar a los conductores que ingresan al campus; sin embargo, este reporte se realiza de manera manual con margen de una hora, dentro del cual pueden ocurrir diversos eventos, y no se muestra la disponibilidad enfocada a cada espacio de los estacionamiento [13]. Las figuras 1.4 y 1.5 muestran el total de espacios disponibles por hora registrado en los meses de Mayo y Octubre del 2019 respectivamente, para una mejor

visualización de los datos en cada gráfico se muestran 5 fechas. Se observa que la mayor ocupabilidad de los espacios ocurre en el horario de 9 a.m. a 3 p.m.

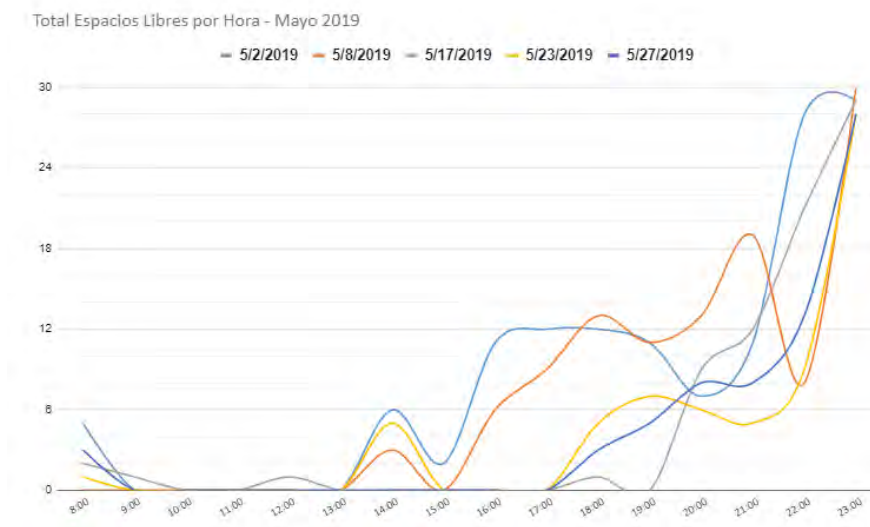


Figura 1.4 Total de espacios libres por hora para el estacionamiento B – Mayo 2019 [Elaboración propia]

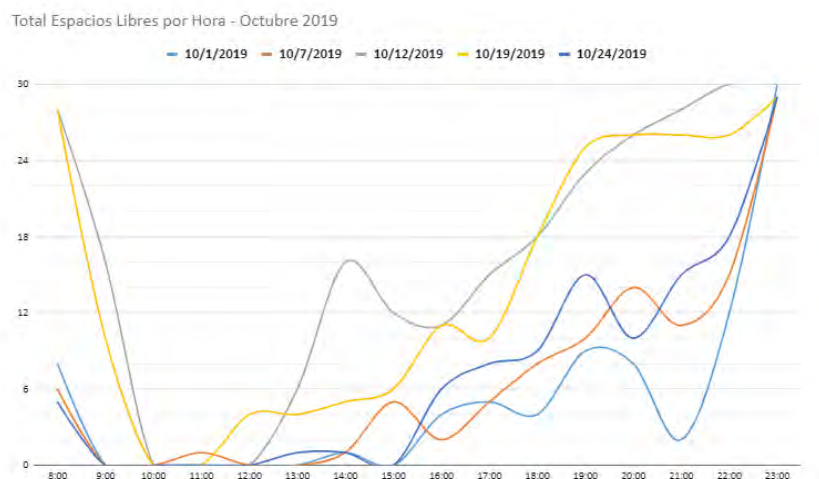


Figura 1.5 Total de espacios libres por hora para el estacionamiento B – Octubre 2019 [Elaboración propia]

1.3 Estado del arte

Las soluciones sobre sistemas de detección de disponibilidad en estacionamientos se pueden clasificar en base a 2 principales métodos: uso de sensores y procesamiento de imágenes. En las siguientes líneas, se presentarán algunos de estos casos.

1.3.1 Detección basada en sensores.

Actualmente, en el mercado se encuentran diferentes soluciones sobre sistemas para el control automatizado de estacionamientos. En el Perú, la empresa Intellisoft S.A. ofrece diversos productos para dicho fin, tales como sistemas de control y recaudación, reconocimiento de matrículas y guiado de vehículos. Por ejemplo, el Jockey Plaza [15] y La Rambla [16] son algunos de sus proyectos implementados por dicha empresa peruana. Por un lado, para la detección de espacios disponibles en estacionamientos (*INDOOR*), dicho sistema incluye sensores de ultrasonido, cuyos datos son transmitidos mediante el estándar TIA/EIA-485 hacia un concentrador. Asimismo, los datos recolectados se procesan y visualizan a través de monitores ubicados en el estacionamiento y otras plataformas. Por otro lado, en estacionamientos (*OUTDOOR*) se instalan en la superficie sensores inductivos con la finalidad de detectar variaciones, debido al acercamiento de una masa metálica como la masa de un vehículo. Después, estos dispositivos transmiten datos a través de RF usando la banda ISM hacia un concentrador. [17]

En el contexto de nuestra universidad, se encontró una tesis relacionada con el diseño de una red de sensores aplicando el protocolo ZigBee. En primer lugar, el nodo sensor está conformado por 5 elementos: sensor magnético y ultrasonido, una batería, un microcontrolador y un transceptor ZigBee. Dichos nodos son instalados sobre la superficie del pavimento de cada plaza. En segundo lugar, la central de monitoreo está conformada por un coordinador XBee y una PC, el cual dispone de un software de gestión de datos, en estos se almacena, distribuye e implementa la interfaz para la visualización de los datos. Por último, el nodo coordinador o central transfiere los datos al módulo *display* para procesarlos mediante un Arduino Micro y, finalmente, mostrarlos en una pantalla LED. [18]

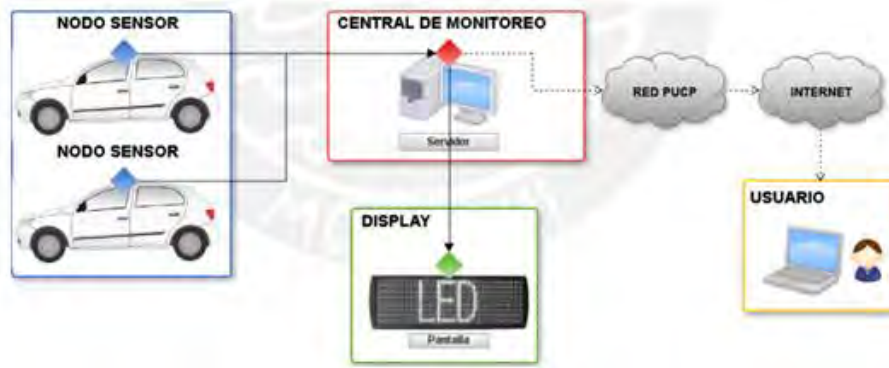


Figura 1.6 Distribución de elementos de red [18]

1.3.2 Detección basada en procesamiento de imágenes.

A diferencia del uso de sensores arriba mencionados, surge un diferente método que utiliza cámaras instaladas a cierta altura del estacionamiento y cada una de ellas captura imágenes sobre espacios asignados. Luego, las imágenes obtenidas son procesadas por algoritmos basados en inteligencia artificial para detectar la presencia de vehículos. Tal es el caso de Parkam, el cual en sus soluciones ofrece sistemas de guía, análisis de datos y disponibilidad en tiempo real. Lo resaltante de su propuesta es la posibilidad de utilizar infraestructura y cámaras de seguridad existentes [19]. En la misma línea, se muestra un caso de estudio: *Curtin University*, Australia. Se instalaron 260 cámaras, 75 postes, 84 dispositivos de comunicación punto-multipunto, pantallas LED, gabinetes de comunicaciones y demás componentes con la finalidad de cubrir alrededor de 28 estacionamientos que suman 6000 plazas aproximadamente. Así, tiene la ventaja de analizar un rango de plazas por cámara. Por último, el sistema identifica espacios disponibles e infracciones de tránsito cometidas en las mismas con una precisión de 99%.

La siguiente propuesta se basa en el procesamiento de imágenes usando Optical Character Recognition (OCR). Para empezar, es necesario pintar las plazas de estacionamiento asignándoles un número que los distingan y que contrasten con el pavimento. Luego, se instalan cámaras en la parte superior de los estacionamientos a cierto ángulo de visión y a cada una se le asigna un rango. Las imágenes obtenidas, pasan por un mejoramiento cambiando de RGB a

escala de gris y ajustando el contraste. Por último, dichas imágenes son procesadas por un algoritmo de OCR en MATLAB, el cual devuelve un arreglo de números. De esta manera, cada vez que un vehículo se parquee, el número pintado no será visible y, por tanto, en la etapa de procesamiento no será considerado. [20]



Figura 1.7 Detección de autos a través de OCR [20]

1.4 Objetivos

1.4.1 Objetivo general.

La presente tesis opta como fin principal el diseño de un sistema de computación de borde que muestre la disponibilidad de vehículos para el estacionamiento B de nuestra universidad aplicando inteligencia artificial, con la finalidad de mostrar la disponibilidad en tiempo real y disponer un historial de sus niveles de ocupación.

1.4.2 Objetivos específicos.

- Determinar los algoritmos de preprocesamiento y detección de objetos a través de imágenes.
- Determinar el dispositivo en el cual se realizará el procesamiento.
- Diseño de la arquitectura en nube para la distribución y el almacenamiento de los datos.
- Diseño de una aplicación móvil multiplataforma para la consulta y visualización de los datos.
- Integración y determinación de parámetros de funcionamiento a partir de las pruebas realizadas.

1.5 Alcance

En la presente tesis se ha optado por evaluar el comportamiento del flujo de autos en el estacionamiento B, debido a las dimensiones del área y las condiciones de iluminación y conectividad a la que se encuentra (figura 1.8).



Figura 1.8 Estacionamiento B – PUCP [21]

La presente tesis fue desarrollada en el año 2020, a inicios de la expansión de una nueva pandemia en el país causada por el virus COVID-19 (coronavirus), en el cual fue restringida toda actividad dentro de los centros públicos y privados, como los centros de educación, y la inmovilización social de personas sin pase laboral por las calles debido al estado de emergencia de salud pública declarado. Se restringe el acceso a los estacionamientos dentro del campus universitario, por ello, se propone como alternativa el uso del *dataset* de un estacionamiento con características similares al planteado inicialmente para realizar la evaluación del sistema diseñado.

Por otro lado, no se consideran las gestiones administrativas que se realizan sobre los espacios de estacionamiento como, por ejemplo, las reservas para entidades educativas y administrativas o invitados a eventos. El sistema se desarrolla sobre un estacionamiento con disponibilidad total de los espacios para todos los usuarios.

El presente proyecto busca beneficiar principalmente a toda la comunidad universitaria e invitados con correo institucional de la PUCP que tengan uso continuo de las distintas zonas

de parqueo dentro del campus universitario. Se identifican como beneficiarios directos e indirectos:

- **Beneficiarios directos:** Estudiantes, personal administrativo y académico, proveedores, conferencistas, asistentes y en general visitantes a eventos de la universidad.
- **Beneficiarios indirectos:** Autoridades encargadas de la gestión de los estacionamientos, entorno urbano al disminuir las colas de ingreso, medio ambiente al reducir las emisiones de gases contaminantes y docentes e investigadores que utilicen el proyecto como caso de estudios para posteriores innovaciones.



Capítulo 2: Fundamento teórico sobre las tecnologías implicadas en una red inalámbrica de dispositivos con inteligencia artificial

El capítulo aborda los fundamentos teóricos de las tecnologías involucradas en redes inalámbricas de dispositivos con IA (computación de borde). Se presenta el modelo teórico de la solución, seguido de una revisión de conceptos básicos de IA, el rol de las placas SBC, la transmisión de datos y los servicios de nube como elementos clave. Asimismo, se detallan los antecedentes relevantes para el diseño de la solución planteada en un campus universitario.

2.1 Modelo teórico de la solución

El presente proyecto de tesis tiene como objetivo la visualización continua de datos por los usuarios. Para ello, es necesario realizar el procesamiento computacional completo dentro de los dispositivos más cercanos a la fuente de información, en el borde de la red (*Edge Devices*). La acción de realizar el procesamiento local de datos en tiempo real es lo que se denomina computación de borde (*Edge Computing*) y marca la diferencia respecto a las soluciones de procesamiento centralizadas, ya que la información transmitida es de menor tamaño. Además, es clave para mantener bajos niveles la latencia en la red. El procesamiento

de datos se vuelve descentralizado [22]. A continuación, se muestra una tabla con las ventajas y desventajas de este modelo de procesamiento.

Tabla 2.1 Edge computing: ventajas y desventajas [23]

EDGE COMPUTING	
Ventajas	<ul style="list-style-type: none"> • Incrementa la capacidad de respuesta, la velocidad y la calidad del servicio en general. • Disminuye en gran medida la congestión de la red y los períodos de inactividad o retraso. • Mantiene el tráfico más pesado y el procesamiento más cercano a la aplicación y los dispositivos del usuario final. • Disminución del ancho de banda requerido.
Desventajas	<ul style="list-style-type: none"> • El sistema de procesamiento distribuido es mucho más complejo que una arquitectura de nube centralizada. • Requerida la implementación de fuertes medidas de seguridad en sus dispositivos finales. • Se pueden generar mayores costos de mantenimiento y administración dependiendo de la cantidad de nodos de procesamiento.

La presente sección se enfoca en establecer el diseño para cada etapa del flujo de información extremo a extremo, desde las cámaras instaladas en los postes del estacionamiento de donde se requiere observar su disponibilidad hasta su visualización final en una aplicación móvil. El modelo teórico presentado a continuación sigue el procesamiento de datos tipo Edge Computing. Consta de los siguientes 6 bloques:

- **Preprocesamiento:** método para medir la similitud entre dos imágenes. Si son distintas por un valor umbral, se inicia la etapa de procesamiento IA. Reduce el consumo computacional.

- **Inteligencia artificial (*Deep Learning*):** algoritmo utilizado para la detección de objetos, entre ellos autos, mediante el procesamiento de imágenes.
- **Ordenador de placa reducida (SBC):** hardware o dispositivo de bajo costo en el cual se realizará el procesamiento local de la información.
- **Tecnologías para la transmisión de los datos:** Sistema para el transporte unidireccional de la información en base a los protocolos de comunicación establecidos.
- **Servicios de computación en nube (*Cloud Computing*):** plataforma de almacenamiento y procesamiento de datos en la nube para su análisis y posterior visualización.
- **Aplicación Móvil:** plataforma digital diseñada para la interacción con los usuarios finales, visualización final de la información procesada.

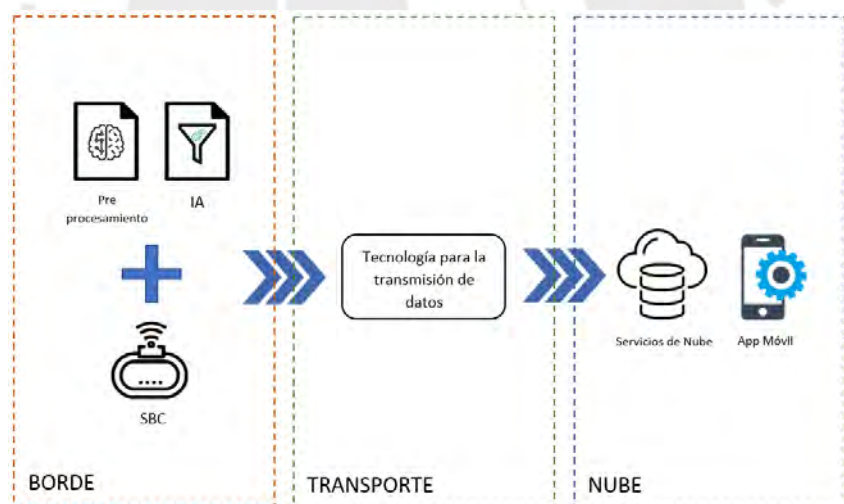


Figura 2.1 Esquema del modelo teórico [Elaboración propia]

2.2 Conceptos básicos

En esta sección se explicarán los conceptos más importantes relacionados al esquema del modelo teórico presentado.

2.2.1 Inteligencia artificial.

Se abordarán conceptos generales de los sistemas basados en redes neuronales convolucionales más utilizados actualmente, se mencionan sus características y se establecen los parámetros necesarios para compararlos.

2.2.1.1 *redes neuronales convolucionales.*

Para tener un mejor panorama sobre la base la cual se sienta la detección de objetos y más la aplicación de redes convolucionales consultar el Anexo I. A diferencia de los métodos tradicionales para la clasificación de objetos, la detección de objetos requiere de técnicas más complejas que empleen sistemas basados en el uso de redes convolucionales con aprendizaje profundo (*Deep Learning* en inglés) para una mejor predicción de múltiples objetos en una imagen. La predicción está basada en el procesamiento de miles de recortes de una imagen dentro de una red convolucional. Debido a que resulta muy ineficiente procesar una inmensa cantidad de recortes de distintos tamaños, se implementan técnicas para elaborar propuestas de regiones con objetos. Cada una de las soluciones actuales cuenta con una arquitectura extensa y muy variada para la detección de objetos, se mencionan algunas a continuación:

YOLO (*You Only Look Once*): este un sistema enfocado en la detección de objetos en tiempo real. YOLO realiza el procesamiento de una imagen completa reduciéndose a cuadros delimitadores y calculando la predicción de clase en cada una, a diferencia del sistema R-CNN que realiza procesos iterativos en los cuadros para obtener respuestas con mayor precisión, pero más lentas [30]. Entre sus beneficios destacan:

- **Detección muy rápida:** La imagen es procesada directamente en la red neuronal, no se realizan procesos repetitivos para la detección.
- **Procesamiento completo:** Durante el entrenamiento y testeo se procesan todos los píxeles de una imagen, por lo que tiene menos errores en detección de fondo que las técnicas basadas en regiones.

- **Generalización:** Aprende aspectos generales de objetos debido al entrenamiento con imágenes completas.

Como se observa en la figura 2.5, su detección consiste en dividir una imagen en rejillas (celdas) pequeñas y en cada una generar un conjunto fijo de cuadros delimitadores para realizar predicciones de manera independiente. YOLO predice los ajustes de cada cuadro delimitador hacia una posición final mediante 5 parámetros, además de los puntajes de todas las clases más una clase fondo o background. Los parámetros de ajuste son las posiciones x-y del centro del cuadro, ancho, alto, y el puntaje de confianza (confidence), este último refleja la probabilidad de que un cuadro encierre un objeto y que tan bien ajustado se encuentra a este. [30]

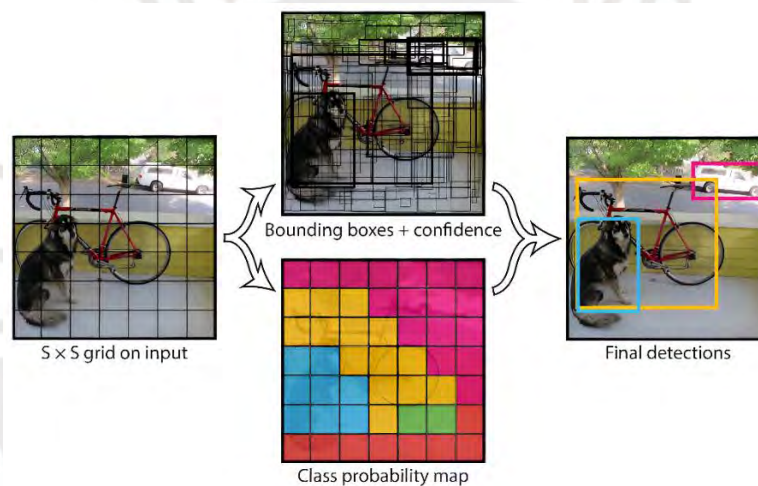


Figura 2.2 Modelo de detección YOLO [30]

Respecto al diseño de red, YOLO consta de capas convolucionales iniciales para determinar las características de una imagen y usa capas totalmente conectadas para determinar los parámetros finales y calcular la probabilidad de clase de un objeto y la proporción con su cuadro delimitador. [30]

Fast R-CNN (Region Based Convolutional Neural Networks): R-CNN es un algoritmo para la detección precisa de objetos basado en regiones, su desarrollo consiste en generar propuestas de una imagen, denominadas regiones de interés (RoI), y procesarlas, cada una, en redes convolucionales para extraer sus parámetros característicos y relacionarlas con un conjunto de clases. Entre los parámetros de salida, se obtienen 4 valores para corregir el cuadro

delimitador de un objeto detectado en la región propuesta. Como beneficio se genera una clase fondo o *background*, por lo que es el algoritmo es capaz de reconocer todo lo que no sea un objeto; sin embargo, toda la detección conlleva un gran tiempo de entrenamiento debido a sus múltiples etapas por región, además de un gran uso de recursos computacionales, por lo que se elaboraron alternativas óptimas como Fast R-CNN. [31], [32]

Fast R-CNN emplea técnicas para mejorar el tiempo de entrenamiento y prueba de una red convolucional mediante el uso de una técnica para generar regiones de interés. Entre sus beneficios destacan una mayor calidad de detección (mAP), reducción en las etapas de entrenamiento y la actualización completa de todas las capas de la red convolucional [32]. En la figura 2.6 se observa su arquitectura, una imagen completa se toma como un volumen de entrada. La imagen pasa a través de capas convolucionales y de agrupación para generar un mapa de características correspondiente, de este mapa se extraen recortes de distintas dimensiones (mediante una técnica para seleccionar regiones de interés) y son ajustados a un tamaño fijo mediante una capa de agrupación RoI. Finalmente, los recortes se procesan por capas totalmente conectadas para generar dos capas a la salida: una capa encargada de almacenar las probabilidades de clase por cada objeto más una clase de fondo y otra capa con datos agrupados en 4 valores que indican la posición adecuada del cuadro delimitador. [32]

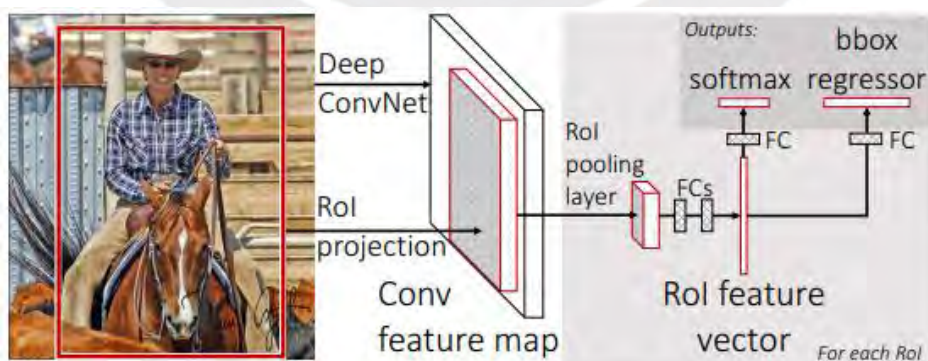


Figura 2.3 Arquitectura procesamiento Fast R-CNN [32]

En términos de velocidad, como se observa en la figura 2.7, se obtiene mejor tiempo de entrenamiento con Fast R-CNN ya que la computación compleja se reduce debido al mapa de características. Por otro lado, el tiempo de prueba disminuye a 2,3 segundos, gran parte de este

tiempo consiste en generar una por una las propuestas de regiones del mapa de características mediante una técnica como búsqueda selectiva (selective search), por lo que se genera un cuello de botella en el procesamiento. Este problema es resuelto mediante Faster R-CNN. [31]

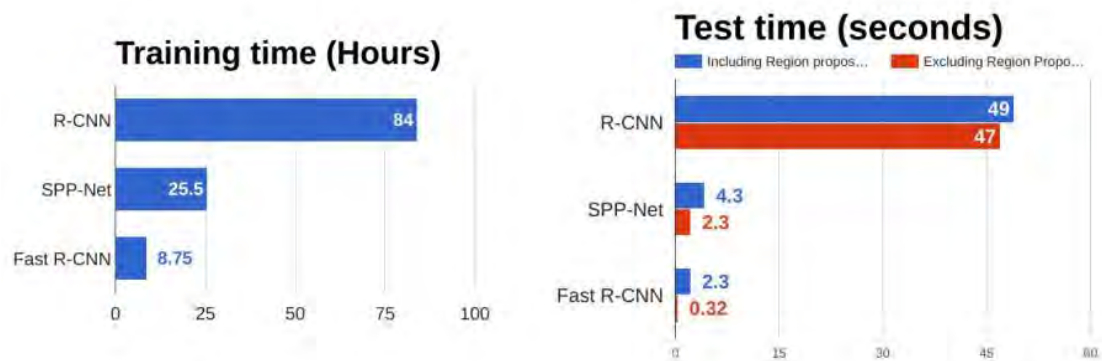


Figura 2.4 R-CNN vs SPP vs Fast R-CNN [31]

Faster R-CNN: Tiene una arquitectura similar a Fast R-CNN (ver figura 2.8), con la diferencia que ahora se usa una red de regiones propuestas (RPN) encima del mapa de características para obtener los recortes finales, la misma red se encarga de generar sus propias propuestas. Finalmente, los recortes son procesados por las demás capas al igual que en Fast R-CNN [31], [33]. En Faster R-CNN se realizan 4 tareas, la red RPN se encarga de clasificar cada región potencial y ajustar las coordenadas de su cuadro delimitador mientras que la red final realiza los ajustes finales en la clasificación y localización a la salida. Respecto a la velocidad de procesamiento, se obtuvo un valor de 0,2 segundos como tiempo de prueba frente a los 2,3 segundos que tomaba Fast R-CNN. [31], [33]

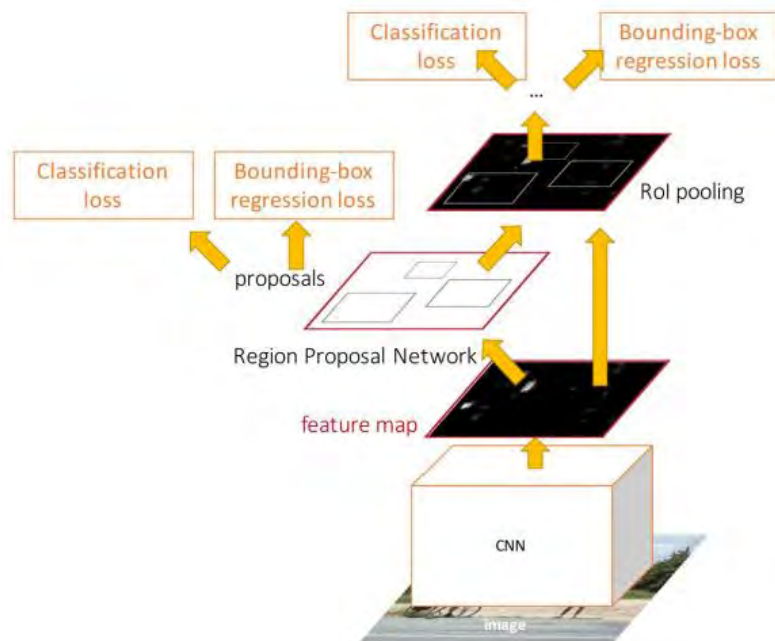


Figura 2.5 Arquitectura Faster R-CNN [31]

2.2.1.2 parámetros de medición.

Como se mencionó anteriormente, existen diferentes modelos de detección mediante el uso de redes convolucionales, cada uno con características singulares, y para determinar cuál es el más apropiado en una situación específica es importante realizar una comparación objetiva entre cada modelo entrenado mediante parámetros de validación estadísticos. [34]

La etapa de validación consiste en verificar la precisión que ha tenido el modelo en detectar las clases y cuadros delimitadores de cada objeto en una imagen respecto de los objetos y cuadros indicados manualmente en la entrada (*ground truth*). En esta etapa, se verifica que una predicción realizada (*positive*) o no (*negative*) pueda estar correcta (*true*) o incorrecta (*false*), por lo que se obtienen 4 estados [34]. En el contexto que se desee la detección de un objeto:

- **Verdad Positiva (*True Positive*, TP):** Se realizó una correcta predicción del objeto. Por ejemplo, se determinó la clase y el cuadro delimitador de un auto sin inconvenientes.

- **Verdad Negativa (*True Negative*, TN):** Se realizó una correcta no predicción del objeto. Por ejemplo, una pista o un árbol de fondo no fueron detectados por el modelo.
- **Falso Positivo (*False Positive*, FP):** Se realizó una incorrecta predicción del objeto. Por ejemplo, un auto fue clasificado como persona por el modelo.
- **Falso Negativo (*False Negative*, FN):** Se realizó una incorrecta no predicción del objeto. Por ejemplo, un auto no fue detectado por el modelo.

Según la cantidad de valores que tenga cada estado se puede determinar si la detección de objetos fue correcta o no mediante los siguientes parámetros:

- **Precisión (*Precision*):** Razón entre las predicciones correctas y el total de predicciones. Indica qué tan buenas son las predicciones realizadas por el modelo.

[35]

$$Precision [P] = TP / (TP + FP)$$

- **Recuperación (*Recall*):** Razón entre las predicciones correctas y el número total de elementos iniciales correctos en el conjunto. Indica qué tan bueno es el modelo para escoger los ítems correctos. [35]

$$Recall [R] = TP / (TP + FN)$$

La siguiente tabla muestra la matriz de confusión (*Confusion Matrix*) para determinar el rendimiento de los parámetros de medición de un modelo. [36]

Tabla 2.2 Confusion Matrix and Related Performance Measures [36]

		Ground Truth		
		+	-	
Predicción	+	Verdad Positiva (TP)	Falso Positivo (FP)	$Precision [P] = \frac{TP}{(TP + FP)}$
	-	Falso Negativo (FN)	Verdad Negativa (TN)	
		$Recall [R] = \frac{TP}{(TP + FN)}$		$Accuracy [A] = \frac{(TP + TN)}{(TP + FP + TN + FN)}$

2.2.2 Ordenador de placa reducida (SBC).

Existe una amplia gama de sistemas embebidos como candidatos en los nodos. Por un lado, se encuentran algunos basados en microcontroladores tales como Arduino. Por otro lado, se dispone de dispositivos llamados *Single Board Computer* (computadora de placa única) con mayor capacidad con la finalidad de ejecutar algoritmos más complejos. En las siguientes líneas, se mencionan algunas de ellas con sus características.

2.2.2.1 raspberry pi.

Fue creado con la finalidad de poner a disposición dispositivos económicos, con similares funcionalidades a las de una computadora, para que cualquiera pueda aprender a programar [37]. Desde entonces, se desarrollaron distintos modelos y las más recientes son el modelo 3B+ y 4. Requiere de un módulo extra para la captura de imágenes. Para ver el detalle con las especificaciones de sus componentes consultar el Anexo I sección Dispositivos SBC.



Figura 2.6 Raspberry pi 3B+ [38]

2.2.2.2 nvidia jetson.

NVIDIA es una empresa enfocada en el desarrollo de tarjetas gráficas. Entre sus patentes, se encuentra el entorno CUDA (*Compute Unified Device Architecture*), el cual optimiza la computación en paralelo aprovechando la unidad de procesamiento gráfico. Asimismo, ofrece 4 modelos con alta capacidad de procesamiento en paralelo y eficiencia energética en dispositivos compactos. [42]. Para ver las especificaciones de sus modelos consultar el Anexo I sección Dispositivos SBC.



Figura 2.7 Modelo Nvidia Jetson Nano [42]

Los diversos modelos que NVIDIA ofrece requieren de una tarjeta WiFi adicional. En la documentación del fabricante, recomienda la tarjeta Intel 8265, la cual provee conectividad para WiFi y Bluetooth y se presenta alguna de sus características.

- **Modelo:** Intel 8265AC
- **Banda de operación:** 2,4 y 5 GHz

- **Protocolo:** 802.11ac
- **Bluetooth:** 4.2
- **Dimensiones:** 22 × 30 × 2.4 mm



Figura 2.8 Intel 8265 para módulo Nvidia Jetson Nano [44]

Además, es necesario adquirir un módulo para la captura de imágenes. Dado que NVIDIA ofrece 4 modelos con diferentes capacidades, es posible instalar varias cámaras por módulo. Para ver las especificaciones del módulo consultar el Anexo I sección Dispositivos SBC.

2.2.2.3 *google coral.*

Por último, Google Coral es un dispositivo compacto y optimizado para la ejecución de redes neuronales a alta velocidad y con un bajo costo energético. El circuito Edge TPU es uno de sus principales componentes, el cual es diseñado por Google para ejecutar algoritmos IA en el borde donde las limitaciones del ancho de banda, latencia o exigencias o privacidad pueden ser críticos [46]. Al igual que los dispositivos arriba mencionados, es necesario adquirir el módulo de cámara compatible con Google Coral. Para ver las especificaciones de sus componentes consultar el Anexo I sección Dispositivos SBC.

2.2.3 Tecnologías para transmisión de datos.

2.2.3.1 sistema cableado.

Conocido como LAN (acrónimo de *Local Area Network*), es un sistema compuesto por dispositivos finales (computadoras, teléfonos IP, impresoras, servidores entre otros) conectados a *switches* o *routers* por medio de enlaces físicos, generalmente cable UTP. Este tipo de sistemas permite una mayor tasa de transmisión de datos (hasta 10 Gbps con cable de cobre y 400 Gbps con fibra óptica) [49] entre dispositivos con posición fija siguiendo una topología física de tipo estrella, bus, *hub*, anillo, malla o combinación de estas mediante el estándar IEEE 802.3. Además tiene menor interferencia que una red inalámbrica debido a que la información se envía como señales eléctricas a través de cables [50], por otro lado, se tiene mayor seguridad en cuanto al acceso a la red. Este sistema conlleva un previo diseño del cableado estructurado, acondicionamiento físico de los accesorios y rutas donde serán desplegados los enlaces, además del mantenimiento y actualización posterior. [50]

2.2.3.2 sistema inalámbrico.

Edson (2015) afirma que el término "Internet de las cosas" no es nuevo. En realidad, los profesores del MIT lo acuñaron hace casi 20 años para describir un mundo donde las "cosas", que pueden ser dispositivos o sensores, son inteligentes y están conectadas, con la capacidad de recopilar y compartir datos. Los datos que provienen de esos dispositivos y/o sensores son infinitos, y cuando se combinan y analizan con otros tipos de datos, pueden descubrir ideas que estaban fuera del alcance en el pasado. (p. 3). [51]

Actualmente, su implementación es realizable, ya que los costos de los componentes de una red IoT están disminuyendo y también la capacidad de procesamiento es asequible tanto en los servidores como en los dispositivos finales. En la figura 2.17 se muestra, como ejemplo básico, la trayectoria que sigue el flujo de información en una red IoT para el monitoreo de una

carga pesada, se observan los parámetros importantes en la transmisión y los elementos involucrados. [52]



Figura 2.9 Monitoreo de una carga pesada [52]

Desde la perspectiva del área de cobertura, se puede concluir que las tecnologías de red se pueden clasificar en grupos de corto y largo alcance. A su vez, el grupo de largo alcance se puede dividir en tecnologías licenciadas y no licenciadas. Si bien a primera vista puede parecer una gran cantidad de tecnologías, los escenarios de aplicación de IoT son tan diversos que cada tecnología de red ha encontrado su utilización [53]. La figura 2.18 muestra la pila de los protocolos que comprenden diversas tecnologías de red IoT.

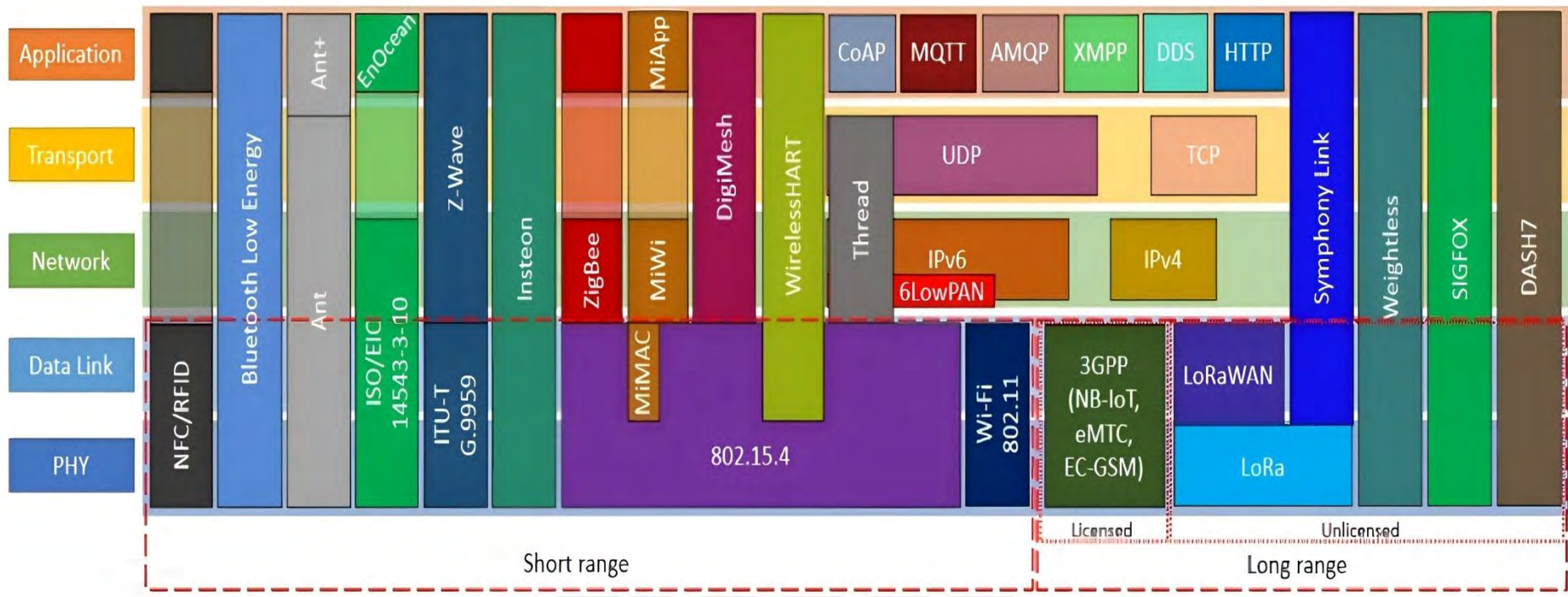


Figura 2.10 Pila de protocolos de tecnologías de red IOT [53]

2.2.3.2.1 redes de corto alcance.

WIFI: conocido como WLAN, es un sistema para la transmisión de datos sobre ondas de radiofrecuencia, permite la conexión y comunicación de dispositivos de manera inalámbrica. al igual que en la red cableada, con la diferencia que los datos se envían por el aire mediante uno de los protocolos IEEE 802.11. Este sistema permite la conexión inalámbrica de los dispositivos a internet a través de un solo router, por lo que su inversión y mantenimiento es mucho menor a una red cableada, además la capacidad de usuarios no se limita a la cantidad de interfaces en el router; sin embargo, los riesgos en seguridad son generalmente mucho mayores a una red LAN y existen problemas de conexión por interferencia [54]. A pesar de que este sistema no fue diseñado inicialmente para IoT, en conjunto de diversos protocolos de aplicación como CoAP, MQTT, entre otros, se han podido ofrecer soluciones de poca sobrecarga de datos en la red convirtiendo a este sistema en una opción viable para el desarrollo IoT en la actualidad.

Protocolo MQTT (*Message Queuing Telemetry Transport*): es un protocolo de transporte de mensajes con el modelo de comunicación de tipo publicador/suscriptor. Usa poco ancho de banda, reduce el consumo de energía, requiere de pocos recursos, soporta autenticación y posibilita un tiempo de respuesta más corto que otros protocolos web actuales. MQTT opera sobre los protocolos TCP/IP. [55]

En este protocolo un cliente envía un mensaje (Publicador) con una etiqueta específica (Tópico) hacia un nodo central (*Broker*) para que otros clientes puedan ver su contenido (Suscriptores), de esta manera, los clientes no tienen información de quien envió o recibió el mensaje, solo conocen al *Broker*. En la figura 2.19 se muestra un modelo del esquema de comunicación cuando se transmite información de humedad, el sensor publica al *Broker* con el tópico “humedad” y este lo retransmite hacia los nodos suscritos con anterioridad al mismo tópico. Es importante mencionar que la conexión hacia el *Broker* es persistente, es decir, el

cliente mantiene la conexión inclusive luego haber publicado y debe enviar un mensaje disconnect para recién liberar la comunicación, además los mensajes no pueden superar los 256 MB y puede haber varios *Brokers* en una misma red interconectados con un puente. [56]

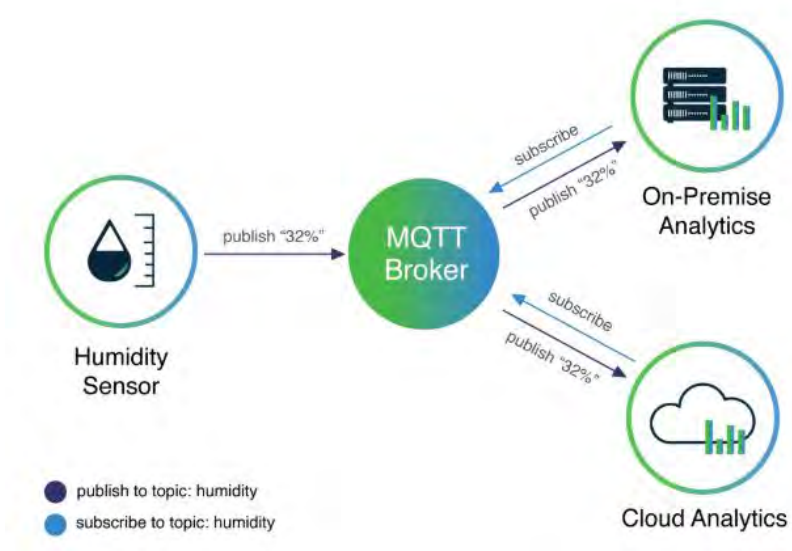


Figura 2.11 Modelo de comunicación MQTT [56]

MQTT implementa un mecanismo de calidad de servicio (QoS) dentro de sus mensajes, en el cual los clientes pueden decidir la forma en que el mensaje es entregado por el *Broker*, para esto se cuentan con 3 niveles (Conforme se aumenta el nivel se tiene mayor confiabilidad, pero habrá mayor requerimiento del ancho de banda y más latencia). [55]

- **QoS 0:** A lo mucho una vez. Existe riesgo que no se reciba el mensaje.
- **QoS 1:** Al menos una vez. Se asegura la recepción, pero es posible obtener mensajes duplicados.
- **QoS 2:** Exactamente una vez. El mensaje se recepciona sin duplicados.

2.2.3.2.2 redes de largo alcance.

A continuación, se detalla las características del protocolo Sigfox, para información del protocolo LoRa consultar el Anexo I en la sección Lora ALLIANCE.

SIGFOX: Sigfox es una compañía francesa pionera en el ecosistema de tecnologías IoT, fundada en 2010, que brinda soluciones de conectividad inalámbrica con la visión de que cada

objeto (dispositivo) pueda conectarse al mundo digital, esta conectividad se basa en ser una red simple, global, baja potencia y gran cobertura para el intercambio de mensajes de corto tamaño [57]. A continuación, se detallan características de esta tecnología.

- **Arquitectura:** en la figura 2.20 se observa una simple arquitectura de red. Consta de 4 elementos en general: Dispositivos finales, estaciones base Sigfox (BS por sus siglas en inglés), la plataforma Cloud de Sigfox y la infraestructura de TI del usuario final. Los millones de dispositivos conectados a cada BS, según su cobertura siguiendo una topología en forma de estrella, transmiten sus mensajes hacia la red de Sigfox y cada una de estas estaciones, conectadas a la única nube Sigfox mediante enlaces punto a punto, demodulan y reportan las señales para, finalmente, ser llevadas a las plataformas de TI (Servidores de los Clientes) para su visualización y posterior análisis. [58]



Figura 2.12 Arquitectura de red Sigfox [58]

- **Comunicación:** Sigfox es una tecnología de banda ultra estrecha (UNB), opera en la banda ISM. La red de Sigfox permite la comunicación bidireccional, un mensaje *Uplink* (UL) tiene un *payload* máximo de 12 bytes y se pueden transmitir hasta 140 mensajes por día, mientras que un mensaje *Downlink* (DL) tiene un *payload* de 8 bytes como máximo y se pueden transmitir hasta 4 mensajes por día; sin embargo, se tienen restricciones en la conectividad o servicio. Con el fin de obtener mayor seguridad y minimizar el consumo de energía un mensaje DL no puede ser transmitido al

dispositivo en cualquier momento, este debe solicitar a la red obtener un mensaje DL, para ello, transmite un mensaje UL con el campo de solicitud DL activado hacia las estaciones base. El mensaje es recibido por cualquier estación base en el rango y se demodula para transmitirlo a la nube Sigfox. Finalmente, el mensaje llega a los servidores de la plataforma IT, se detecta la solicitud de un mensaje DL y se transmite el payload hacia la nube Sigfox, que se encarga de programar una estación base para transmitir el mensaje al dispositivo final [59]. En la Tabla 2.3 se muestra un resumen de las características de operación entre el flujo UL y DL.

Tabla 2.3 Comparativa Uplink - Downlink [Elaboración propia]

	UPLINK	DOWNLINK
MÁXIMO NÚMERO DE MENSAJES SOBRE EL ENLACE	140 mensajes/día	4 mensajes/día
MÁXIMO TAMAÑO PAYLOAD POR MENSAJE	12 bytes	8 bytes
BITRATE	100 bps	600 bps
ANCHO DE BANDA	100 Hz	1,5 KHz
MODULACIÓN	DBPSK	GFSK

- **Seguridad:** la transmisión de mensajes se realiza hacia la red Core privada de Sigfox (nube Sigfox) mediante las estaciones base; Sigfox implementa un método de autenticación entre extremos, como se observa en la figura 2.21, mediante una única clave secreta asociada a cada dispositivo durante su fabricación y usada en los mensajes enviados hacia las estaciones base para asociar cada mensaje a una firma digital y, de esta manera, otorgarle un número de secuencia para su transmisión hacia la nube Sigfox [60]. Este método permite la autenticación del remitente y verificar la integridad del mensaje, por otra parte, la verificación del número de secuencia por la nube Sigfox descarta los intentos de retransmisión.



Figura 2.13 Autenticación entre extremos de red Sigfox [60]

La conexión entre estaciones base y la nube Sigfox es mediante enlaces punto a punto usando VPN (ver figura 2.22), la comunicación entre BS y la nube es segura, robusta y confiable. Por otro lado, la nube Sigfox es virtualizada en servidores privados dentro de data centers distribuidos en múltiples locaciones logrando una infraestructura escalable y robusta. [60]



Figura 2.14 Enlaces VPN en red Sigfox [60]

2.2.4 Cloud computing.

La computación en Nube (*Cloud Computing*) consiste en un conjunto o *pool* de recursos informáticos de fácil acceso, con seguridad y servicio de mantenimiento, que son ofrecidos a los usuarios a través de internet, ajustado a un plan de costo. Estos servicios permiten el acceso a funciones, software, plataformas de desarrollo, almacenamiento y procesamiento de información, siendo una alternativa a los métodos tradicionales (*On Premise*), la Tabla 2.4 muestra una breve comparativa de ambas soluciones. [63]

Tabla 2.4 Comparativa on premise vs cloud [64]

ON PREMISE	NUBE
Centro de datos internos	Centros de datos de terceros
Costos iniciales elevados	Bajas inversiones iniciales en infraestructura
Escalado lento	Rápidamente escalable
Proceso de compra más prolongado	Utilización eficiente de los recursos
Proceso de compra más prolongado	Reducción del proceso de compra

Entre sus características destacan:

- **Multiplataforma:** Acceso a través de cualquier dispositivo, en cualquier momento, con conexión a internet.
- **Seguridad:** Conjunto de políticas, tecnologías que protegen la información contra amenazas.
- **Confiabilidad:** Asegura mayor disponibilidad de datos frente a desastres naturales u eventos catastróficos mediante el almacenamiento de información redundante.
- **Flexibilidad:** Capacidad para agregar o eliminar recursos por el usuario según la demanda.

Respecto al uso del servicio se tienen 3 niveles. [63]

- **SaaS (Software como Servicio):** Brinda aplicaciones que se ejecutan en sistemas de la nube y se conectan al usuario a través de internet. El servicio es escalable, accesible desde cualquier dispositivo y de rápido inicio. Como ejemplo se tienen los servicios de correo (Gmail, Outlook, Yahoo), los servicios de almacenamiento (Dropbox, Mega), Adafruit.io, Ubidots o ThingSpeak.
- **PaaS (Plataforma como Servicio):** Proporciona todo el entorno - componentes y módulos preconfigurados - para el desarrollo y funcionamiento de aplicaciones, evita

la gestión de adquisición y soporte de hardware y software. Se tiene como ejemplo los servicios Google App Engine, Openshift, AWS o Microsoft Azure Web Sites.

- **IaaS (Infraestructura como Servicio):** Proporciona el control o gestión de todos los recursos informáticos a nivel de infraestructura como servidores, conmutadores, enrutadores y almacenamiento, para uso del usuario, así no se requiere de invertir en hardware adicional.

2.2.4.1 amazon web services (aws).

Es la plataforma más completa y utilizada a nivel global, consta de una amplia variedad de servicios a nivel de procesamiento y almacenamiento, bases de datos, tecnologías de aprendizaje automático, inteligencia artificial e IoT, además de una comunidad compuesta por proveedores e integradores de sistemas. AWS es la plataforma más segura ya que cumple con los protocolos necesarios de seguridad y confidencialidad de entidades bancarias, ejército y otras organizaciones. Cuenta con la infraestructura más amplia, 76 zonas de disponibilidad conectadas en 24 regiones geográficas, por lo que ofrece alta disponibilidad en los servicios implementados. Entre otras características destacan las siguientes. [66]

- Mayor funcionalidad de los recursos que ofrece, más de 175 servicios en nube acorde al usuario.
- Amplia comunidad a nivel global.
- Cumple con 90 estándares de seguridad y certificaciones de conformidad.
- Innovación continua de sus herramientas tecnológicas para uso de la comunidad en sus proyectos.
- El precio de AWS se maneja por servicios individuales bajo el esquema de pago por uso en el tiempo.

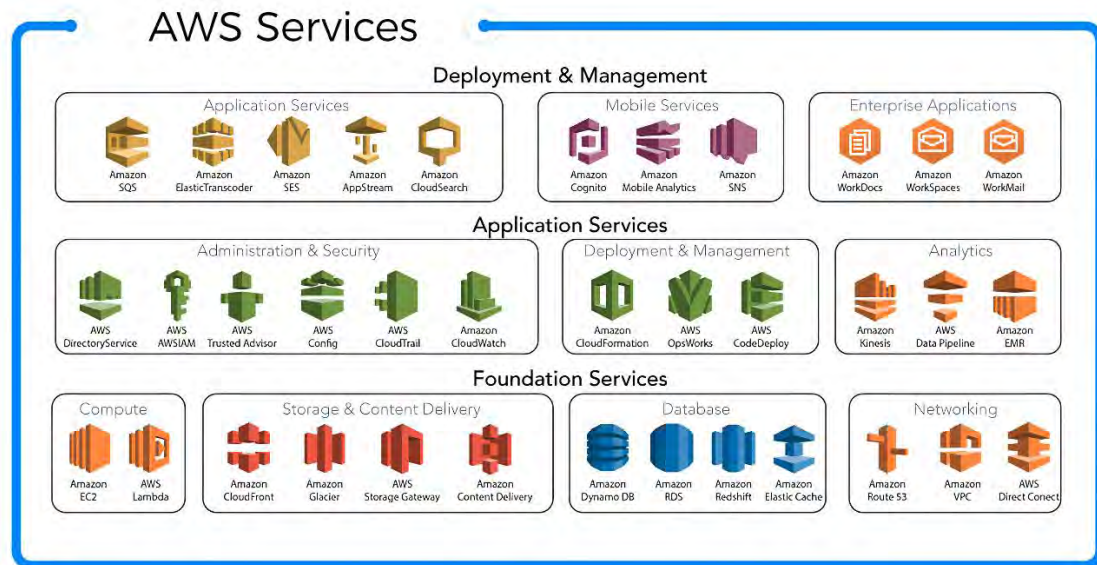


Figura 2.15 Plataforma de servicios AWS [67]

2.2.4.2 microsoft azure.

Es un conjunto de servicios en nube en continua innovación, que otorgan a los usuarios el control para desarrollar, administrar e implementar aplicaciones a través de herramientas y *frameworks*. Microsoft otorga la seguridad de sus servicios en todo nivel mediante la confianza, transparencia y cumplimiento de los estándares requeridos, un equipo de expertos a disposición e inversión continua en seguridad para la protección ante ataques, además de un foro con información detallada de los casos de uso implementados. Entre otras características destacan las siguientes: [68]

- Continua actualización de herramientas Microsoft para una mejor experiencia del usuario.
- Compatibilidad con los lenguajes y *frameworks* más utilizados.
- Integración en diversos escenarios a la nube híbrida de Microsoft.
- Más de 160 centros de datos interconectados en más de 60 regiones.
- Esquema de pago por uso, pero con menor precio respecto a AWS al usar servidores Windows, SQL PaaS y SQL IaaS.

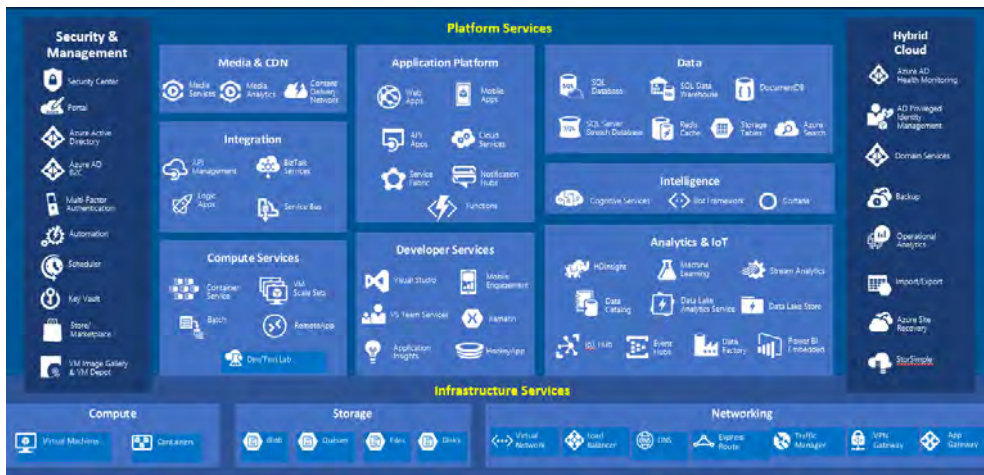


Figura 2.16 Plataforma de servicios Microsoft Azure [69]

2.3 Antecedentes

La Pontificia Universidad Católica del Perú (PUCP) cuenta con infraestructura tecnológica e informática que sirven de plataforma para los servicios destinados al quehacer académico, administrativo e investigación, se encuentra cubierta por equipos tecnológicos, diseño de procesos y sistemas de seguridad bajo la responsabilidad de la Dirección de Tecnologías de la Información (DTI), el cual repercute en la mejora de la prestación del servicio.

Según el sitio web Descubre PUCP (2013) menciona que hacia 1996 y siempre a la vanguardia, la PUCP aprovechó la tecnología Internet para sus Sistemas de Información. Lo que en ese entonces era una de las primeras Intranets nacionales, se ha convertido en el “Campus Virtual”, una completa plataforma de servicios que apoya los procesos académicos y administrativos de la institución, incluyendo el de enseñanza-aprendizaje. (p.1). [70]

La universidad dispone de la debida tecnología para realizar sus tareas diarias; sin embargo, algunas zonas no cuentan con la infraestructura apropiada para el correcto funcionamiento de los servicios de telecomunicaciones, esto se debe a que en un diseño inicial no se previeron los espacios adecuados destinados para los equipos de redes, seguridad y de cableado estructurado; asimismo, no se realizaron proyectos de reubicación que cumplan con las normas pertinentes para edificaciones que entreguen servicios de comunicaciones.

Con el transcurso de los años, se han realizado diversos proyectos para el despliegue de infraestructura en cada uno de los pabellones y edificios que eran construidos y remodelados en la universidad, además de etapas continuas para el incremento de cobertura en los servicios, entre ellos, videovigilancia y conexión inalámbrica (WIFI).

Hoy en día no se contemplan proyectos para la cobertura de wifi en los estacionamientos, por lo que son escasos los espacios que cuenten con este servicio, por otro lado, no hay cámaras de seguridad que brinden cobertura exclusiva, por grupos de espacios, de los estacionamientos. En este contexto, es importante considerar que los dispositivos que sean instalados en los estacionamientos se deben encontrar a una distancia adecuada de los espacios a evaluar y deben tener acceso al servicio de Wifi.



Capítulo 3: Metodología para el diseño de una red inalámbrica de dispositivos con inteligencia artificial aplicada en un estacionamiento del campus universitario

El diseño de una red inalámbrica de procesamiento IA aplicada a la gestión de un estacionamiento requiere una metodología clara y estructurada. Este capítulo aborda desde la definición de los requerimientos de diseño y sus parámetros esenciales hasta la elección sustentada de cada tecnología involucrada. Esta metodología a presentar es propia para el proyecto en cuestión y será detallada en base a las observaciones recabadas y experiencia en las tecnologías involucradas. Por último, se establecen los costos estimados para proyecto.

3.1 Requerimientos de diseño

Según la información teórica y descriptiva recopilada sobre cada una de las tecnologías presentes, a continuación, se mencionan los requerimientos a tomar en cuenta en el diseño del sistema:

- **Cámara:** Es indispensable el uso de un módulo para la captura de imágenes de fácil integración al sistema computacional, este módulo debe tener una resolución mayor a 5 Megapíxeles para una correcta captura de los espacios de parqueo.

- **Dispositivos SBC:** Se requiere un ordenador con disposición de una unidad de procesamiento gráfica para el tratamiento de imágenes, con recursos computacionales de memoria y procesamiento que puedan soportar la ejecución continua del algoritmo CNN, un módulo para la captura de imágenes y posibilidad de conexión inalámbrica.
- **Sistema de comunicación:** La información debe transmitirse implementando los protocolos necesarios para un menor uso del ancho de banda y asegurar la confiabilidad de transmisión.
- **Sistema de computación en nube:** Es necesario contar con los recursos en nube apropiados para el almacenamiento de los datos enviados por los dispositivos SBC y su acceso por los usuarios, realizar la gestión de los elementos involucrados y asegurar la autenticación entre extremos.
- **Interfaz de visualización:** Se requiere una interfaz gráfica para la visualización de los espacios disponibles por parte de los usuarios que se conecten a través de sus dispositivos finales, además de la elaboración de gráficos estadísticos por parte de personal autorizado.

Para el diseño del sistema en nube es necesario conocer el alcance en funcionalidad que tendrá el sistema. A continuación, en la Tabla 3.1 se presenta la lista de requerimientos funcionales que se desea implementar.

Tabla 3.1 Requerimientos del usuario [Elaboración propia]

REQUERIMIENTO	DESCRIPCIÓN
1	Se requiere la autenticación de los usuarios para el ingreso al sistema. Únicamente se podrá registrar y acceder usuarios de un dominio de correo institucional específico asociado a Google.
2	El sistema contará con 3 tipos de usuarios: administrador, analista e invitado, cada uno con distintos niveles de permisos asignados por el sistema.

3	Para todos los usuarios: Se tendrá un mapa en el que se visualizará la disponibilidad total de vacantes en tiempo real para cada sector de estacionamiento.
REQUERIMIENTO	DESCRIPCIÓN
4	El mapa debe contar con un filtro para la visualización de disponibilidad de los espacios exclusivos para personas discapacitadas.
5	Los usuarios analista y administrador contarán con una pantalla adicional que les permita observar y generar reportes estadísticos de la disponibilidad total y ocupabilidad de cada espacio de parqueo.
6	Los reportes deben incluir filtros de fecha específica y sector de estacionamiento.
7	Los reportes pueden visualizarse directamente a través de un equipo móvil o compartir la URL asociada para acceder desde otros dispositivos.
8	Todos los usuarios cuentan con una pantalla donde se muestra la información básica de su perfil creado y la opción para cerrar la sesión.
9	El usuario administrador contará con una pantalla adicional para la gestión de usuarios: la misma consta de la búsqueda de usuarios según su correo asociado, bloqueo de acceso al servicio y el cambio de rol de un usuario invitado a analista y viceversa.
10	La búsqueda de usuarios se visualizará a través de tablas con paginación.
11	La interfaz se adaptará a las dimensiones de pantalla de dispositivos móviles como de escritorio.

De acuerdo con la lista de requerimientos, se requiere un sistema con capacidad para autenticar usuarios, autorizarlos acorde a su rol, administrarlos, recibir actualizaciones de los datos en tiempo real, almacenarlos y verificar el estado de los dispositivos sensores. Adicional a lo mencionado, el sistema debe contar con redundancia para garantizar la alta disponibilidad y escalabilidad para incrementar la capacidad de los servidores en relación con la demanda de usuarios.

3.2 Determinación de tiempo crítico

Es importante mantener la información de disponibilidad del sistema actualizada hacia los usuarios finales a fin de evitar el embotellamiento a la entrada de un área de parqueo y la mala toma de decisiones causada por información errónea. Por ello, se debe establecer un tiempo de actualización óptimo del sistema (en adelante denominado Tiempo Crítico) que brinde un buen desempeño ante cualquier panorama de flujo vehicular.

Basándonos en el contexto de que la zona de estacionamiento cuente con un solo espacio disponible próximo a ser ocupado por un auto en el momento justo que otro ingresa al campus universitario por la puerta principal, el tiempo crítico debe ser tal que el conductor entrante pueda visualizar el estado ocupado del estacionamiento antes de ingresar a la zona de parqueo y se puede definir como la menor distancia recorrida por el auto entre la velocidad de conducción máxima permitida dentro del campus sumado al tiempo que demora el proceso de identificación del conductor en la puerta principal de ingreso.

$$\textit{Tiempo Crítico} = \textit{Tiempo Identificación} + \left[\frac{\textit{Distancia Recorrida}}{\textit{Velocidad Máxima Permitida}} \right]$$

En el presente proyecto, se ha considerado un tiempo de identificación aproximado de 15 segundos. Como se observa en la figura 3.1, la distancia recorrida por un auto es de 105,94 metros y la velocidad máxima permitida según el reglamento emitido por el MTC es de 30 Km/h (8,33 m/seg) [71], por lo que se obtiene un tiempo crítico de 27 segundos.

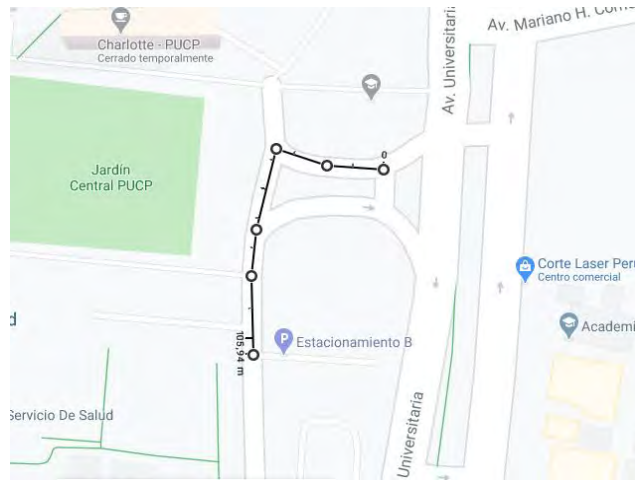


Figura 3.1 Trayectoria recorrida hasta el estacionamiento B [72]

El valor de este parámetro debe ser considerado posteriormente para establecer el periodo de muestreo de los espacios de parqueo por los dispositivos SBC.

3.3 Elección del método de preprocesamiento

Esta etapa tiene como fin reducir el consumo computacional del SBC, se evita ejecutar continuamente el algoritmo de IA para determinar la disponibilidad de espacios en imágenes continuas de la zona de parqueo. En ese sentido, solo se procede a ejecutar el algoritmo de IA siempre y cuando ambas imágenes no sean idénticas respecto a un valor umbral. Se tienen dos algoritmos muy usados para la comparación de imágenes:

- **MSE (*Mean Square Error*):** Como su nombre lo indica, este algoritmo mide la diferencia cuadrática promedio entre dos valores, en este caso los píxeles de la misma posición en cada imagen, mientras mayor sea la diferencia de intensidad mayor será el resultado y se interpretará que las imágenes son muy distintas. La igualdad entre dos imágenes se indica con un valor de cero.
- **SSIM (*Structural Similarity Method*):** Este algoritmo se enfoca en la información estructural de una imagen. Se extraen muestras de cada imagen y se comparan las intensidades promedio de píxeles, la distorsión del contraste y la pérdida de correlación. Genera un resultado entre -1 y 1 para indicar la igualdad siendo 1 el caso en que las imágenes sean idénticas. Si bien es un poco más complejo que MSE, es un

método que puede otorgar mejores resultados ante imágenes con ligeras variaciones, pero muy similares.

SSIM puede ser una gran opción para este proyecto ya que se adecúa al entorno diverso que tiene un estacionamiento abierto (cambios bruscos de luminosidad, reflejo de la luz en los autos, presencia de pequeños obstáculos, condiciones ambientales, movimientos ligeros de la cámara, entre otros); sin embargo, para seleccionar este algoritmo se debe evaluar su desempeño en un dispositivo SBC en cuanto al consumo de la RAM y tiempo de procesamiento, estas pruebas serán realizadas, a modo informativo, en un Raspberry 3B+.

Los espacios a analizar se muestran a continuación: en total son 11 espacios de parqueo distribuidos en 4 filas que forman parte de la cobertura de la cámara 6 de un *dataset* que se va a detallar más adelante en la sección 3.4. En la siguiente figura se muestra la etiqueta (número) con la cual se identifica cada espacio.

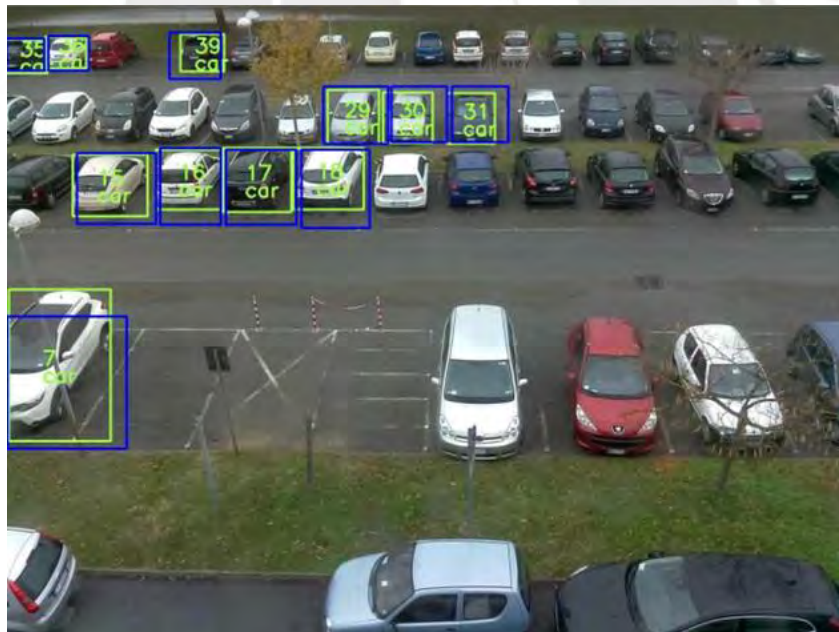


Figura 3.2 Espacios de parqueo [Elaboración propia]

Como primer análisis, se mide el desempeño al extraer las primeras capturas (*crops*) de los espacios de parqueo. En esta etapa no existe una comparación con un estado anterior, sólo se tomarán 11 recortes y se almacenarán en una carpeta para su posterior uso. Cabe resaltar que no se está almacenando la imagen entera, sino pequeños recortes con las dimensiones de cada

espacio, esto para obtener una mejor precisión al momento de hacer la comparativa. La siguiente figura muestra que esta etapa demora 47ms en ejecutarse y consume aproximadamente la tercera parte de RAM del dispositivo.

```
(YOLOv3) pi@raspberrypi:~/Downloads/Test_Preprocesamiento $ python main.py
Etapa de extraer CROPS
tiempo: 0.04772591590881348, cpu: 30.0, memoria: 327.14388155397137
```

Figura 3.3 Ejecución de etapa inicial [Elaboración propia]

En esta última fase se evalúa el preprocesamiento, en cada bucle se realiza la comparación uno por uno entre los *crops* de la imagen actual y la anterior, para obtener un resultado de “IGUAL” los 11 *crops* de cada imagen deben ser similares mayor a un valor umbral de 0,7, basta que un par sea distinto para que se rompa el bucle y se dé el resultado “DIFERENTE”. La siguiente figura muestra los valores de tiempo y recursos consumidos para las comparaciones realizadas en el *dataset*.

```
(YOLOv3) pi@raspberrypi:~/Downloads/Test_Preprocesamiento $ python main.py
La imagen /home/pi/Downloads/Final/FULL_IMAGE_1000x750/OVERCAST/2015-11-25/camera6/
2015-11-25_0718.jpg es igual a la anterior
tiempo: 0.13664007188889648 s, memoria: 326.0134137596491 MB
*****

La imagen /home/pi/Downloads/Final/FULL_IMAGE_1000x750/OVERCAST/2015-11-25/camera6/
2015-11-25_0748.jpg es diferente a la anterior
tiempo: 0.0811469554901123 s, memoria: 326.28337621799466 MB
*****

La imagen /home/pi/Downloads/Final/FULL_IMAGE_1000x750/OVERCAST/2015-11-25/camera6/
2015-11-25_0818.jpg es igual a la anterior
tiempo: 0.18426918983459473 s, memoria: 328.47682119205297 MB
*****

La imagen /home/pi/Downloads/Final/FULL_IMAGE_1000x750/OVERCAST/2015-11-25/camera6/
2015-11-25_0848.jpg es diferente a la anterior
tiempo: 0.2100536823272705 s, memoria: 328.62867507487243 MB
*****

La imagen /home/pi/Downloads/Final/FULL_IMAGE_1000x750/OVERCAST/2015-11-25/camera6/
2015-11-25_0918.jpg es diferente a la anterior
tiempo: 0.16503357887268066 s, memoria: 326.65457459021994 MB
*****

La imagen /home/pi/Downloads/Final/FULL_IMAGE_1000x750/OVERCAST/2015-11-25/camera6/
2015-11-25_0948.jpg es diferente a la anterior
tiempo: 0.187544584274292 s, memoria: 326.97515501750536 MB
*****
```

Figura 3.4 Ejecución de etapas posteriores [Elaboración propia]

Así mismo, la siguiente tabla refleja a mayor detalle todas las comparaciones hechas en el día. Se muestra el resultado de cada comparación y si este tiene concordancia con un cambio en la ocupabilidad total de los espacios marcados, además del tiempo y recursos consumidos.

Tabla 3.2 Ejecución de preprocesamiento [Elaboración propia]

FECHA	OCUPABILIDAD TOTAL	Comparación (umbral=0,7)	RAM (MB)	Tiempo (s)
07:18	0	Igual	326,01	0,136 6
07:48	1	Diferente	326,28	0,081 1
08:18	1	Igual	328,47	0,184 2
08:48	2	Diferente	328,62	0,210 0
09:18	4	Diferente	326,65	0,165 0
09:48	7	Diferente	326,97	0,187 5
10:18	9	Diferente	326,38	0,182 9
10:48	11	Diferente	326,62	0,224 7
11:18	11	Igual	327,97	0,169 4
11:48	11	Diferente	327,11	0,084 3
12:18	11	Igual	327,32	0,213 0
12:48	11	Igual	327,56	0,204 0
13:18	11	Diferente	326,70	0,135 6
13:48	11	Igual	328,03	0,186 7
14:18	11	Igual	327,16	0,198 0
14:48	11	Igual	327,39	0,210 4
15:18	11	Igual	327,32	0,230 6
15:48	11	Diferente	326,18	0,182 0
16:18	11	Igual	327,51	0,207 6
16:48	11	Diferente	326,65	0,204 4
17:18	4	Diferente	325,79	0,118 4
17:48	0	Diferente	326,03	0,113 1
Promedio			327,03	0,174 0

Con la información analizada de la tabla, se concluye que esta etapa de preprocesamiento efectivamente no consume muchos recursos del dispositivo ni requiere mucho tiempo de ejecución (observar los valores promedio y pico obtenidos); en cuanto a la comparación, se tiene una buena concordancia entre el resultado del algoritmo y la ocupabilidad real. Los casos que no cumplen esto se deben a cambios en la iluminación del día, por ejemplo a las 11:18 a.m. se muestra una imagen azul pálida mientras que a las 11:48 a.m. se torna rojiza cálida, estos pasarán a la siguiente etapa para un veredicto más preciso con la IA.

El sistema se torna más eficiente con esta etapa ya que se evita ejecutar continuamente el algoritmo pesado de IA, solo los casos filtrados como DIFERENTES serán analizados con mayor detalle.

3.4 Elección del algoritmo de IA

3.4.1 Tareas de visión computacional.

Como se mencionó anteriormente, dentro del panorama de aplicaciones de IA se encuentran las técnicas de procesamiento de imágenes. Existen variedades de tareas de visión por computadora y es importante evaluar la(s) más apropiada(s) de acuerdo con sus ventajas y desventajas. Por ello, se presenta la siguiente tabla comparativa de las tareas más importantes explicadas en el Capítulo 2.

Tabla 3.3 Comparación de tareas de visión por computadora [31], [33], [73]

TAREA	VENTAJAS	DESVENTAJAS
CLASIFICACIÓN DE IMÁGENES	<ul style="list-style-type: none"> Bajo costo computacional. Menor tiempo de detección. 	<ul style="list-style-type: none"> Reconocimiento de un único objeto en una imagen. No muestra la ubicación del objeto.
DETECCIÓN DE OBJETOS	<ul style="list-style-type: none"> Reconocimiento de múltiples objetos en una imagen. 	<ul style="list-style-type: none"> Moderado costo computacional.

	<ul style="list-style-type: none"> • Reconocimiento selectivo. • Brinda la ubicación de cada objeto mediante su cuadro delimitador. • Bajo tiempo de detección. 	
SEGMENTACIÓN DE INSTANCIAS	<ul style="list-style-type: none"> • Reconocimiento de múltiples objetos en una imagen. • Reconocimiento selectivo. • Brinda la ubicación de cada objeto mediante su cuadro delimitador. • Muestra la forma del objeto a través de una máscara. 	<ul style="list-style-type: none"> • Alto costo computacional. • Mayor tiempo de detección.

De acuerdo con el análisis realizado, se ha optado por seleccionar algoritmos dentro del campo de clasificación de imágenes y detección de objetos, ya que presentan los tiempos de detección más bajos y no ocupan muchos recursos de procesamiento a comparación de la segmentación.

3.4.2 Dataset utilizado.

En el presente año que se realiza la tesis, hay restricciones de ingreso al campus universitario. Por ello, como alternativa de diseño, se propone el uso de un *dataset* con características similares al estacionamiento B, bajo diversas condiciones climáticas y de iluminación. El *dataset*, denominado CNR-EXT, consta de un conjunto de imágenes tomadas entre noviembre 2015 hasta febrero 2016 por un grupo de 9 cámaras, cada una con distinto ángulo de perspectiva de la misma área de estacionamiento (figura 3.5). Para una mejor evaluación, se incluyeron imágenes del estacionamiento en condiciones de día soleado, día lluvioso, neblina y poca iluminación (noche). Las capturas se realizaron mediante un módulo

cámara Pi V1 (Raspberry) con un sensor OmniVision OV5647 de 5 Megapíxeles montada en una caja para exteriores e instalada frente al espacio de aparcamiento.[74]



Figura 3.5 Capturas del estacionamiento en cámaras 1, 3, 5 y 7 [75]

Se seleccionó la cámara 7 por tener un ángulo de perspectiva similar al estacionamiento inicial propuesto. Como parte de la evaluación apropiada de cada algoritmo se van a considerar las cuatro filas de parqueo dentro del estacionamiento (resolución 1000x750) como se observa en la figura 3.6.



Figura 3.6 Imagen original (100x750) [75]

Finalmente, el *dataset* original cuenta con muy poca iluminación por la noche, es casi imposible observar autos a vista simple a través de la cámara, por lo que solo se seleccionaron

las imágenes hasta las 5:30 p.m. que se tienen las mismas condiciones de iluminación que el estacionamiento B (figuras 3.7 y 3.8).

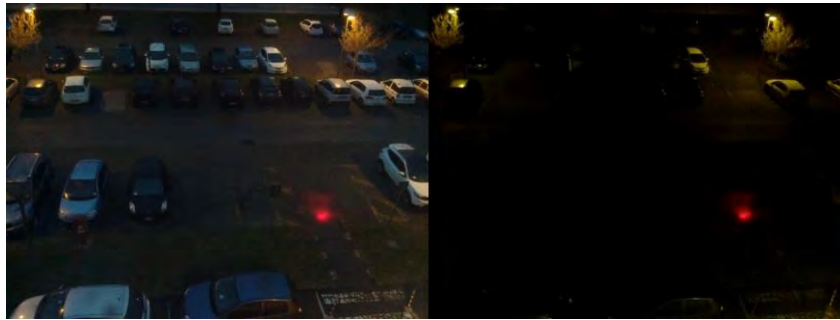


Figura 3.7 Capturas del estacionamiento a las 5 p.m. y 7 p.m. [75]



Figura 3.8 Capturas del estacionamiento B a las 8 p.m. [Elaboración propia]

3.4.3 Pruebas con algoritmos escogidos.

En esta sección, se analizará el desempeño de dos algoritmos (YOLO y Faster R-CNN) ya pre entrenados. El testeo será sobre 242 imágenes del *dataset* CNR-EXT mencionado y se evaluará el rendimiento en base a los siguientes parámetros: porcentaje de CPU promedio (% CPU USADO), memoria RAM usada promedio (MEMORIA USADA (MB)), tiempo de procesamiento promedio (TIEMPO) y las variables de la matriz de confusión (precisión [P], recuperación [R] y exactitud [A]). Todas las pruebas serán realizadas con un ordenador personal de las siguientes características:

- **Sistema Operativo:** Ubuntu 18.04 LTS
- **Tipo de S.O:** 64 bits
- **Memoria:** 3,8 GiB
- **Procesador:** Intel Core i5-4200M CPU @2.50GHz x4

- **GNOME:** 3.28.2
- **Disco:** 182,6 GB

Las siguientes figuras muestran casos concretos de todas las detecciones obtenidas por cada algoritmo en el estacionamiento en todas las condiciones climáticas y de noche. En el Anexo 1, se muestran las detecciones de todas las imágenes del *dataset*. Las pruebas se han realizado en 15 posiciones distribuidas en dos filas.

YOLO: existen diversos algoritmos de una sola visualización de imagen. De esta familia, el sobresaliente es la versión 3 de YOLO con agrupación piramidal espacial (YOLOv3-SPP) que se encarga de generar una imagen fija en la entrada de la red CNN independiente del tamaño de imagen que se vaya a procesar. Con esta ventaja se obtiene un alto valor de mAP igual a 60,6 frente a otras alternativas como YOLOv3-608 con mAP de 57,9, además de tener un menor tiempo de inferencia [76]. Este algoritmo utiliza el Framework de red neuronal Darknet y fue entrenado con el *dataset* COCO, su testeo se realizó con la biblioteca OpenCV. La figura 3.9 representa una muestra de los resultados obtenidos. La información total se muestra en el Anexo I.



Figura 3.9 de Izq. a Der. Detección con YOLO en diversas condiciones: nublado, lluvioso, soleado y noche.

[Elaboración propia]

Cada espacio de parqueo presenta distintos desafíos para su correcta detección (variación en la perspectiva, escala, obstrucción por el entorno, condiciones de iluminación, mezcla con el entorno). La Tabla 3.4 muestra la precisión promedio con la cual los vehículos fueron correctamente detectados dentro de los espacios de estacionamiento definidos. Se consideran como vehículos las clases carro, bus y camión.

Tabla 3.4 Precisión promedio obtenida por cada posición de parqueo - YOLO [Elaboración propia]

Posición	Precisión Promedio	Posición	Precisión promedio
1	0,721 5	23	0,916 1
2	0,876 4	24	0,940 4
3	0,938 7	25	0,926 4
4	0,858 1	26	0,926 8
5	0,835 5	27	0,861 0
6	0,961 3	28	0,787 3
7	0,771 5	29	0,411 6
8	0,902 5	30	0,820 6
9	0,937 9	31	0,756 9

Posición	Precisión Promedio	Posición	Precisión promedio
10	0,888 0	32	0,808 4
11	0,895 5	33	0,769 0
12	0,939 6	34	0,808 5
13	0,946 7	35	0,625 6
14	0,937 0	36	0,610 2
15	0,968 1	37	0,806 9
16	0,968 8	38	0,823 4
17	0,951 8	39	0,745 6
18	0,946 8	40	0,802 1
19	0,918 6	41	0,803 4
20	0,940 6	42	0,726 5
21	0,971 4	43	0,738 0
22	0,920 0		

De la tabla se puede analizar que los espacios 1, 29, 35 y 36 presentan los menores valores debido a que son obstruidos por las ramas de los árboles, para el caso del estacionamiento B no se cuenta con obstáculos de este tipo. Los mejores datos, superior al 90%, se encuentran en mayor proporción en la segunda y tercera fila en los espacios 8, 9 y del 12 al 26, ya que se encuentran correctamente estacionados sin obstáculos.

Acorde a la matriz de confusión presentada en el Capítulo 2, para cada una de las imágenes procesadas se obtuvieron sus valores de TP, TN, FP, FN para determinar la precisión, recuperación y exactitud con que eran detectados los objetos, estos valores además del rendimiento computacional usado para procesar cada imagen se pueden visualizar en el Anexo I. La Tabla 3.5 muestra los valores promedio obtenidos.

Tabla 3.5 Parámetros promedio YOLO [Elaboración propia]

PRECISIÓN [P]	RECUPERACIÓN [R]	EXACTITUD [A]	TIEMPO	% CPU USADO	MEMORIA USADA (MB)
1,00	0,96	0,96	2,93	89,42	682,04

La Tabla 3.6 muestra los valores máximos obtenidos, estos se toman en consideración para escoger el dispositivo SBC.

Tabla 3.6 Parámetros máximos YOLO [Elaboración propia]

PRECISIÓN [P]	RECUPERACIÓN [R]	EXACTITUD [A]	TIEMPO MÁXIMO	% CPU MÁXIMO	MEMORIA MÁXIMA (MB)
1,00	1,00	1,00	3,54	91,70	683,11

Faster R-CNN: dentro de la familia de detección con uso de algoritmos para propuestas de regiones sobresale el modelo Faster R-CNN, ya que integra la parte RPN y CNN para hacer un proceso más acelerado de las 300 regiones propuestas. Para la creación de múltiples mapas de características, utiliza el algoritmo SFP y las alimenta a la red RPN para obtener las regiones de interés (RoI). El detector trabaja sobre la red convolucional ResNet con 101 capas de profundidad y fue pre entrenado con el *Dataset* COCO. Su testeo se realizó con la librería Detectron2. Se utilizaron 2 núcleos de procesamiento en el ordenador al 100% en vez de 4. La figura 3.10 representa una muestra de los resultados obtenidos, la información total se muestra en el Anexo I.



Figura 3.10 Izquierda a Derecha: detección con *Faster R-CNN* en diversas condiciones: nublado, lluvioso, soleado y noche [Elaboración propia]

La Tabla 3.7 muestra la precisión promedio con la cual los vehículos fueron correctamente detectados dentro de los espacios de estacionamiento definidos. Se consideran como vehículos las clases carro, bus y camión.

Tabla 3.7 Precisión promedio obtenida por cada posición de parqueo – Faster R-CNN [Elaboración propia]

Posición	Precisión Promedio	Posición	Precisión promedio
1	0,841 2	23	0,858 9
2	0,845 2	24	0,773 8
3	0,844 0	25	0,817 3
4	0,774 1	26	0,821 8
5	-	27	0,825 0
6	0,982 9	28	0,800 9
7	0,890 2	29	0,771 0
8	0,950 9	30	0,863 9
9	0,942 8	31	0,806 2
10	0,923 2	32	0,792 3
11	0,884 5	33	0,692 5
12	0,899 4	34	0,742 8
13	0,920 0	35	0,561 9
14	0,901 4	36	0,483 3
15	0,951 5	37	0,569 6
16	0,966 8	38	0,576 8
17	0,953 9	39	0,624 5
18	0,966 5	40	0,638 2
19	0,876 1	41	0,577 8
20	0,853 8	42	0,615 7
21	0,861 6	43	0,653 4
22	0,871 4		

Los menores valores de precisión se encuentran en las posiciones del 35 al 43 pertenecientes a la cuarta fila (la más alejada) mientras que los mejores se encuentran en la segunda fila. Se puede intuir que, a medida que los vehículos se alejan de la cámara y disminuyen en tamaño, el algoritmo pierde eficacia en su detección. Una gran desventaja de este algoritmo se observa en la posición 5, no se tiene un valor de precisión debido a que todas las veces que un vehículo ocupaba este espacio era etiquetado con la clase “Bote”.

De la misma forma que en el caso anterior, se determinó para cada una de las imágenes procesadas sus valores de TP, TN, FP, FN, precisión, recuperación, exactitud y rendimiento computacional usado los cuales se pueden visualizar en el Anexo I. La Tabla 3.8 muestra los valores promedio obtenidos.

Tabla 3.8 Parámetros promedio Faster R-CNN [Elaboración propia]

PRECISIÓN [P]	RECUPERACIÓN [R]	EXACTITUD [A]	TIEMPO	% CPU USADO	MEMORIA USADA (MB)
0,98	0,95	0,93	6,80	50,64	1 505,20

De la misma forma, la Tabla 3.9 muestra los valores máximos obtenidos con Faster R-CNN.

Tabla 3.9 Parámetros máximos Faster R-CNN [Elaboración propia]

PRECISIÓN [P]	RECUPERACIÓN [R]	EXACTITUD [A]	TIEMPO MÁXIMO	% CPU MÁXIMO	MEMORIA MÁXIMA (MB)
1,00	1,00	1,00	8,31	66,80	1 627

Finalmente, se muestra una tabla comparativa de valores medios obtenidos por cada algoritmo. Según lo presentado, se concluye que el algoritmo YOLO muestra un mejor desempeño para el panorama presentado, realiza una mejor detección de clases de los vehículos, identifica la mayor cantidad de ellos en las condiciones presentadas, tiene menor tiempo de detección y consume poca cantidad de recursos computacionales.

Tabla 3.10 Comparación YOLO vs Faster R-CNN [Elaboración propia]

ALGORITMO	EXACTITUD [A]	TIEMPO	% CPU USADO	MEMORIA USADA (MB)
YOLO	0,96	2,93	89,42	682,04
Faster R-CNN	0,93	6,80	50,64	1 505,20

3.5 Elección y configuración del dispositivo SBC

3.5.1 Evaluación del dispositivo.

A continuación, se realizará una tabla comparativa para evaluar las principales características de los modelos de dispositivo SBC más adecuados a los requerimientos. En la evaluación, se debe considerar que el dispositivo soporte los valores máximos obtenidos en la

Tabla 3.6, además del procesamiento continuo adicional para la transmisión de los eventos hacia la plataforma Cloud a través de una tecnología de transporte, es decir, los valores de procesamiento determinados en la sección anterior irán aumentando conforme se desarrollen las demás etapas del proyecto. Por otro lado, es necesario una unidad de procesamiento gráfico para la detección de objetos del *frame* capturado.



Tabla 3.11 Comparación de dispositivos SBC [Elaboración propia]

MODELO	PROCESADOR	MEMORIA	GPU	CONECTIVIDAD	INTERFAZ CÁMARA	POTENCIA DE ENTRADA	DIMENSIONES	PRECIO DE LA PLACA
RASPBERRY PI 3B+	Broadcom BCM2837B0 Quad-Core A53 (ARMv8) 64-bit @ 1.4GHz	1GB LPDDR2 SDRAM	VideoCore IV @ 250-400MHz	LAN: Gigabit Ethernet. WLAN: 2.4GHz and 5GHz IEEE 802.11.b/g/n/ac Wireless LAN, Bluetooth 4.2, BLE	Sí	5V/2.5A DC	82 x 56 x 19,5 mm	35 USD
RASPBERRY PI 4 B	Broadcom BCM2711, Quad Core Cortex-A72 (ARM v8) 64-bit SoC @ 1.5GHz	1GB, 2GB o 4GB LPDDR4-3200 SDRAM	VideoCore VI @ 500Mhz	LAN: Gigabit Ethernet WLAN: 2.4 GHz and 5.0 GHz IEEE 802.11ac Wireless, Bluetooth 5.0, BLE	Sí	5.1V / 3.0A DC	88 x 58 x 19,5mm	Entre 35 a 55 USD
JETSON NANO	Quad-core ARM® Cortex®-A57 MPCore processor	4 GB 64-bit LPDDR4 25.6GB/s	128-core NVIDIA Maxwel	LAN: 10/100/1000 BASE-T Ethernet. WLAN: Requiere tarjeta adicional para Wifi (Intel 8265AC).	Hasta 4 cámaras	10W	69,6 x 45 mm	149 USD + 21 USD (Intel 8265AC)
JETSON TX 4GB	Dual-core Denver 1.5 64-bit CPU + Quad-core Arm® Cortex®-A57 MPCore processor	4 GB 128-bit LPDDR4 51.2GB/s	256-core NVIDIA Pascal	LAN: 10/100/1000 BASE-T Ethernet. WLAN: Requiere tarjeta adicional para Wifi (Intel 8265AC).	Hasta 6 cámaras	20W	87 x 50 mm	299 USD + 21 USD (Intel 8265AC)
DEV BOARD	NXP i.MX 8M SoC (quad Cortex-A53, Cortex-M4F)	1 GB o 2GB LPDDR4	Integrated GC7000 Lite Graphics	LAN: Gigabit Ethernet WLAN: Wi-Fi 2x2 MIMO (802.11b/g/n/ac 2.4/5GHz), Bluetooth 4.2	Sí	5V DC (USB Tipo C)	88 x 60 x 24 mm	130 USD

Adicionalmente, se muestra una comparativa de los módulos de captura de imágenes requeridos por cada modelo.

Tabla 3.12 Comparación de módulos cámara [Elaboración propia]

MODELO	RESOLUCIÓN	INTERFAZ	DIMENSIONES	COSTO
Raspberry Camera module v2	8 megapíxeles	CSI	25 x 23 x 9 mm	27 USD
Leopard Imaging LI-IMX219-MIPI-FF-NANO-H136	8 megapíxeles	CSI	25 x 24 x 15 mm	29 USD
Módulo Google Coral	5 megapíxeles	CSI	25 x 25 x 6,98 mm	20 USD

De acuerdo con las tablas analizadas, se decidió utilizar el dispositivo Raspberry PI 4 B con 2GB RAM junto al módulo Raspberry Camera V2 debido a que ofrece los recursos adecuados para el correcto funcionamiento del algoritmo CNN a un menor precio.

3.5.2 Configuración de la disponibilidad.

Luego de escoger el dispositivo, se debe realizar la configuración adecuada para establecer las variables de disponibilidad del estacionamiento conforme a los objetos detectados y finalmente construir la estructura final del mensaje que se va a transmitir. Según lo mencionado anteriormente, cada dispositivo SBC se encarga de detectar los vehículos presentes en el estacionamiento por cada imagen del *dataset* seleccionado. Como se observa en la figura 3.8, para determinar la disponibilidad total en un momento dado, se evalúan los vehículos detectados (cuadros verdes) con un conjunto de rejillas inicializadas manualmente (cuadros azules), si el centro de cada vehículo se encuentra dentro de una rejilla entonces esta será marcada como ‘ocupada’.



Figura 3.11 Disponibilidad del estacionamiento [Elaboración propia]

El sistema se encuentra monitoreando la disponibilidad del estacionamiento constantemente. El monitoreo consiste en dos etapas: el bloque de preprocesamiento, que realiza un primer filtro de cambios en los espacios de parqueo, y el bloque de procesamiento con el algoritmo IA, el cual revisa con mayor detalle sólo los cambios marcados como “diferente” por el bloque de preprocesamiento. En caso que haya un cambio de estado en los espacios de parqueo (un auto entra o sale), la información será enviada hacia el segmento *cloud* para su procesamiento.

3.6 Elección y configuración del sistema de transporte

Luego del diseño de la parte *Edge* es importante evaluar la tecnología con la que se van a transmitir los datos, para ello la Tabla 3.13 comprende de una comparativa a nivel de frecuencia de modulación, alcance, tasa máxima de transmisión, tiempo de vida de las baterías, entre otras características para poder seleccionar la tecnología más apropiada.

Tabla 3.13 Características de las tecnologías de transmisión [53]

TECNOLOGÍA	TIPO DE CONEXIÓN	BANDA DE FRECUENCIA	ALCANCE	TASA DE DATOS	VIDA DE BATERÍA	TOPOLOGÍA
Ethernet	Conexión física	100MHz, 250MHz, 500MHz (CAT5-5e, CAT6, CAT6A)	Hasta 100 m (CAT6A)	varía (Mbps a Gbps)	Días	Estrella
Wi-Fi	Conexión Inalámbrica	2.4 GHz, 5 GHz; sub-1 GHz (Wi-Fi HaLow)	100 m; 1 km (Wi-Fi HaLow)	varía (Mbps a Gbps)	Días a meses	Estrella
LoRa WAN	Conexión Inalámbrica	sub-1 GHz	10 – 15 km	50 kbps	10+ años	Estrella de estrellas
SIGFOX	Conexión Inalámbrica	sub-1 GHz	10 – 50 km	100 bps	10+ años	Estrella

La tabla analizada presenta los diversos entornos donde puede viajar la información. La tecnología Ethernet presenta desventajas, ya que una conexión alámbrica conlleva costos de implementación (adquisición de ducterías, cableado eléctrico, cableado de datos e instalación) y mantenimiento por dispositivo, por ello, se opta por soluciones inalámbricas. Sigfox resulta una alternativa atractiva por ser una tecnología de poco uso de ancho de banda y de menor costo que LoRa, además de poseer un convenio con la PUCP y tener buena cobertura de la señal; sin embargo, Sigfox maneja como límite la transmisión de 140 mensajes *uplink* por día y para realizar el diseño de un sistema en “tiempo real” es necesario determinar la cantidad de eventos que pueden ocurrir en un estacionamiento siendo un evento definido como el cambio de disponibilidad entre dos capturas contiguas tomadas desde la cámara del dispositivo SBC (un evento ocurre, por ejemplo, cuando un auto ocupa o libera un espacio de parqueo). Debido

al panorama incierto, se escoge utilizar WIFI como sistema de transporte, de esta manera se podrá observar la cantidad de eventos generados por dispositivo mediante gráficos estadísticos y se tiene una mayor libertad para estructurar la información que se desea transmitir por el dispositivo final.

Para reducir el uso de ancho de banda al transmitir los mensajes de los dispositivos SBC, se ha optado el uso del protocolo MQTT sobre TLS a través del puerto 8883 (tradicional) utilizando como *Broker* el componente Cloud IoT Core que será explicado más adelante. El QoS de transmisión será establecido en '1' para asegurar la transmisión de al menos un mensaje y los tópicos establecidos serán los siguientes:

- Estacionamientos: tópico donde se publican los cambios de disponibilidad de las zonas de parqueo.
- *Disconnect*: tópico reservado AWS IoT para indicar que un cliente MQTT se ha desconectado.

Para establecer la conexión tradicional, se necesita una clave privada, un certificado digital y un certificado de CA para la autenticación. A continuación, se presenta la configuración requerida en los dispositivos de borde:

Configuración Dispositivos SBC: se encargan de enviar las actualizaciones al *broker* (publicadores). Los dispositivos SBC utilizan las librerías AWS IoT SDK disponibles en la web para la conexión MQTT, así como el certificado para la autenticación con el *broker*.

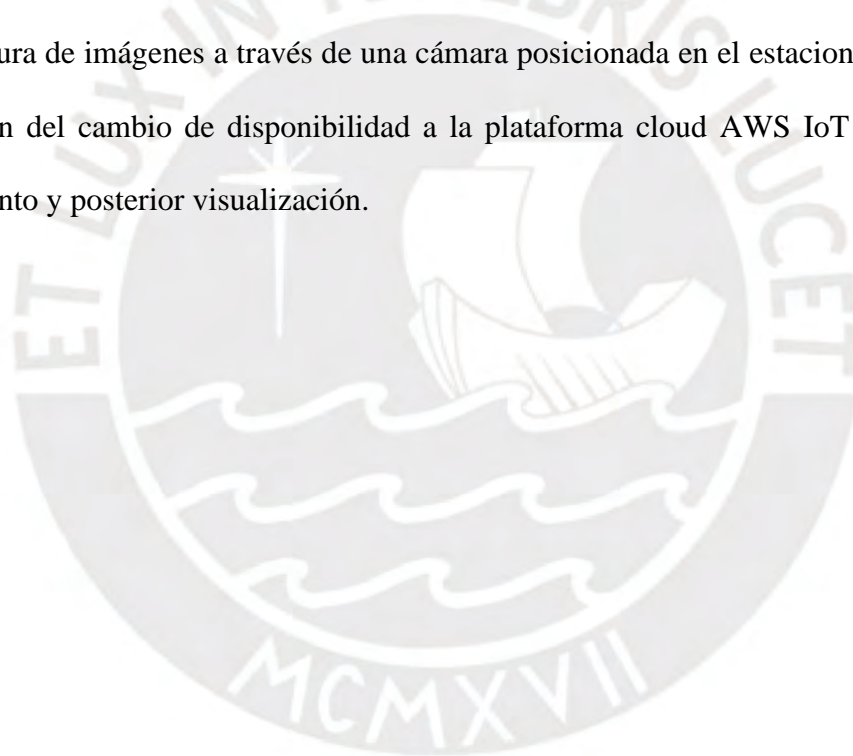
Configuración Broker: receptor en la comunicación MQTT. Por el lado de los dispositivos SBC (objetos según la plataforma) es necesario que los certificados generados se encuentren activos y asociados a la política correspondiente. En la figura 3.12 se muestra la política definida.

Documento de política

```
1 {
2   "Version": "2012-10-17",
3   "Statement": [
4     {
5       "Effect": "Allow",
6       "Action": "iot:Connect",
7       "Resource": "arn:aws:iot:us-east-1:*:client/${iot:ClientId}"
8     },
9     {
10      "Effect": "Allow",
11      "Action": "iot:Publish",
12      "Resource": "arn:aws:iot:us-east-1:*:topic/$aws/rules/estacionamiento"
13    }
14  ]
15 }
```

Figura 3.12 Política dispositivo SBC [Elaboración propia]

Finalmente, en la figura 3.13 se muestra el diagrama de flujo de información en el borde, desde la captura de imágenes a través de una cámara posicionada en el estacionamiento hasta la transmisión del cambio de disponibilidad a la plataforma cloud AWS IoT Core para su almacenamiento y posterior visualización.



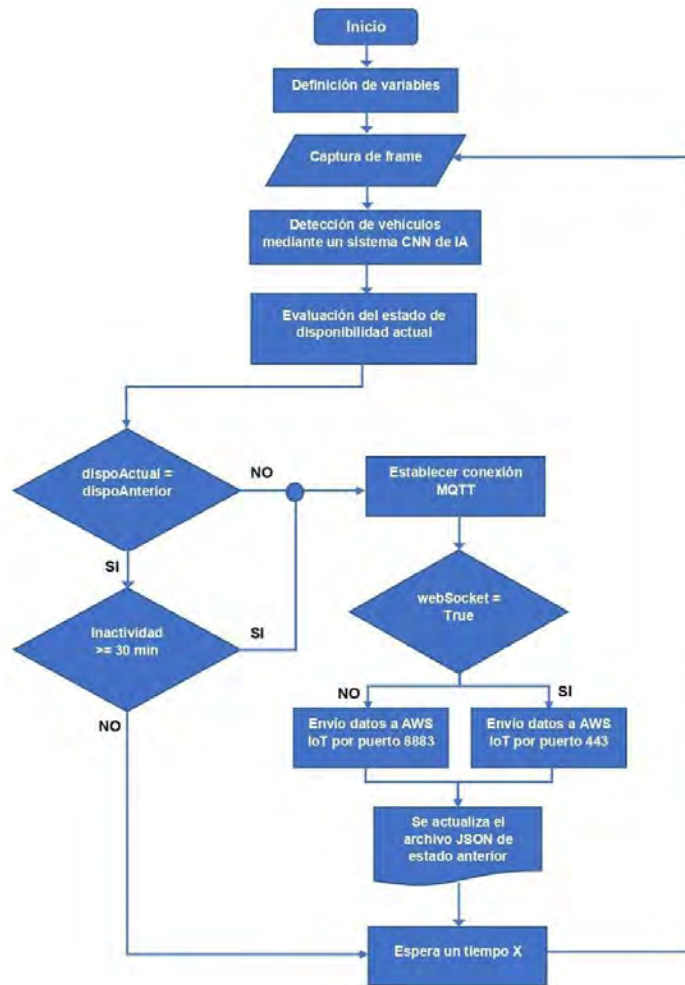


Figura 3.13 Diagrama de flujo computación de borde [Elaboración propia]

3.7 Elección del proveedor y desarrollo del sistema en nube

En esta sección, se procede a evaluar el proveedor *Cloud* para el almacenamiento de la información y a configurar los componentes requeridos para su implementación.

3.7.1 Elección del proveedor y región.

En el panorama de servicios de computación en nube sin servidores (*serverless*), existen pocos proveedores con amplia y consistente infraestructura que puedan brindar soluciones en la categoría de Funciones como Servicio (FaaS). La Tabla 3.14 muestra una comparación de las opciones FaaS más conocidas en la actualidad en base a sus características.

Tabla 3.14 Comparación de proveedores FAAS [77][78]

DESCRIPCIÓN	AWS LAMBDA	GOOGLE CLOUD FUNCTIONS	MICROSOFT AZURE FUNCTIONS
Lanzamiento	2014	2016	2016
Escalabilidad	Automática	Automática	Manual o medida (App Service Plan)
Cantidad Máxima de Funciones	Ilimitadas	1000/proyecto	Ilimitadas
Ejecuciones concurrentes	1000/cuenta/región (default)	Ilimitadas	Ilimitadas
Tiempo de ejecución máximo	15 min	9 minutos	10 min (Consumption Plan)
Granular IAM	Políticas IAM pueden adjuntarse a Lambdas	RBAC soportada en Subscription y ResourceGroup. Las funciones están contenidas	No
Almacenamiento Persistente	S3 & DynamoDB	Cloud Storage, Cloud Datastore, Cloud SQL	Blob Storage
Logeo y Monitoreo	Cloudwatch Logs & X-Ray	Stackdriver Logging, Stackdriver Error Reporting y Stackdriver Monitoring	Azure Storage & Apps Insights
Rentabilidad	1M solicitudes gratuitas, luego \$0,20/1M invocaciones, adicional \$0,00001667/GB-seg	1M solicitudes gratuitas, luego \$0,40/1M invocaciones, adicional \$0,00000231/GB-seg (tarifa separada para memoria y CPU)	1M solicitudes gratuitas, luego \$0,20/1M invocaciones, adicional \$0,000016/GB-seg

De acuerdo con la información analizada, cada proveedor presenta ventajas similares en cuanto al tratamiento y almacenamiento de la información a través de funciones; sin embargo, AWS se encuentra mejor posicionado en cuanto a escalabilidad y límites de ejecución de sus funciones a un bajo costo de servicio, por ello se opta trabajar con este proveedor. Adicionalmente, como se observa en la figura 3.14, reportes realizados por empresas dedicadas

al asesoramiento e investigación muestran el prestigio que tiene AWS al posicionarse como líder en el cuadrante mágico de Gartner ‘*Cloud Infrastructure as a Service*’ durante 9 años consecutivos y como líder de mayor presencia en el reporte de Forrester ‘*Public Cloud Enterprise Containers Platforms*’. [79], [80]



Figura 3.14 Reportes mostrados por Gartner y Forrester [79], [80]

Con el proveedor seleccionado, es importante determinar la región o el área geográfica de donde los recursos de computación y almacenamiento serán utilizados; las regiones se encuentran aisladas entre ellas y cada una tiene múltiples locaciones aisladas conocidas como *Availability Zones*, es importante seleccionar aquella que tenga menor latencia de transmisión. Para ello, se utilizará una herramienta web, cuyo dominio es CloudPing.Info, el cual indica la latencia desde el navegador hacia las distintas regiones de AWS. Dado que los usuarios están concentrados en Perú, se seleccionará la región de US-East (Virginia) que presenta menor latencia.

Region	Latency	Region	Latency
US-East (Virginia)	101 ms	Asia Pacific (Mumbai)	315 ms
US East (Ohio)	113 ms	Asia Pacific (Osaka-Local)	272 ms
US-West (California)	154 ms	Asia Pacific (Seoul)	291 ms
US-West (Oregon)	158 ms	Asia Pacific (Singapore)	340 ms
Canada (Central)	122 ms	Asia Pacific (Sydney)	282 ms
Europe (Ireland)	194 ms	Asia Pacific (Tokyo)	537 ms
Europe (London)	187 ms	South America (São Paulo)	140 ms
Europe (Frankfurt)	196 ms	China (Beijing)	297 ms
Europe (Paris)	187 ms	China (Ningxia)	313 ms
Europe (Stockholm)	212 ms	AWS GovCloud (US-East)	124 ms
Middle East (Bahrain)	317 ms	AWS GovCloud (US)	161 ms
Asia Pacific (Hong Kong)	314 ms		

Figura 3.15 Latencia navegador – región de AWS mediante Cloudping.info [Elaboración propia]

3.7.2 Arquitectura de la solución seleccionada.

Esta arquitectura se basa en un enfoque *Serverless*, el cual traslada tareas de administración del servidor al proveedor de nube, con la finalidad de concentrarse más en la lógica de negocio. En este desarrollo, se usaron servicios de AWS como Cognito, IoT Core, S3, SES, DynamoDB, CloudWatch, AppSync, Lambda y Quicksight.

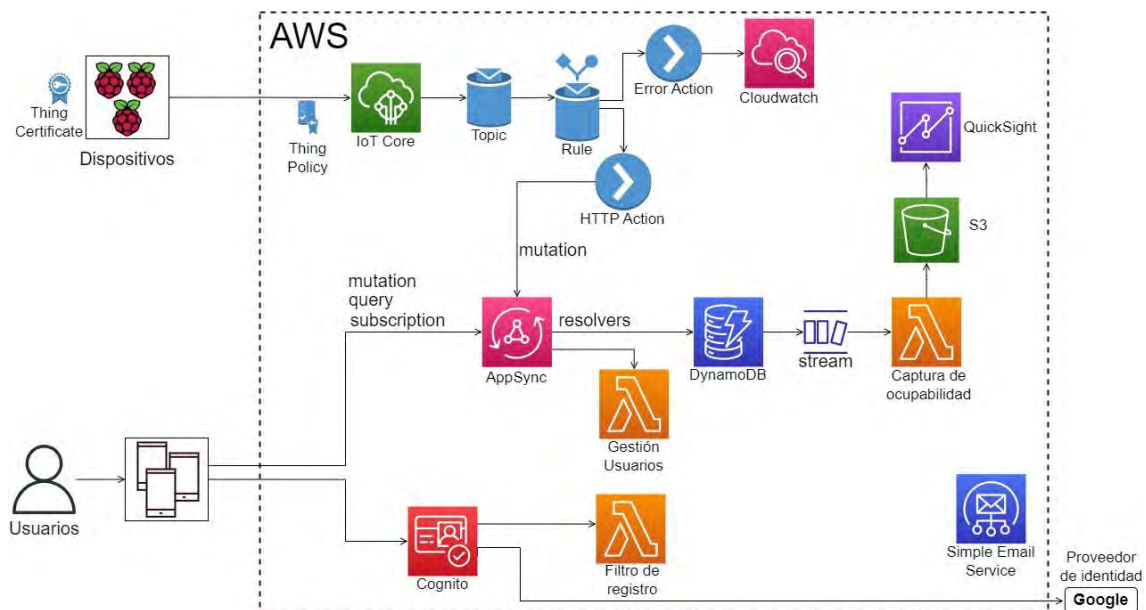


Figura 3.16 Arquitectura Cloud AWS [Elaboración propia]

A continuación, se describen los componentes involucrados y se indica la configuración realizada:

Amazon Cognito: *Cognito User Pool* es un servicio de AWS que ofrece la gestión de usuarios y administración de accesos. Asimismo, se integra con servicios como *API Gateway* o *AppSync* como autorizador y federarse a través de un proveedor de una identidad de tercero. Aquí, se definió como proveedor de identidad a Google porque los usuarios ya disponen de un

correo institucional. Además, se definió los roles Invitado, Analista y Administrador, los cuales tienen permisos asignados mostrados en la siguiente tabla.

Tabla 3.15 Permisos de cada tipo de usuario [Elaboración propia]

Permiso	Invitado	Analista	Administrador
Listar el resumen de todos los estacionamientos, así como el detalle de cada uno.	✓	✓	✓
Visualizar reportes de la ocupabilidad del estacionamiento		✓	✓
Buscar usuarios, permitir o denegar acceso a la aplicación, permitir o denegar a visualizar reportes			✓

Adicionalmente, se agregó un evento de pre registro en Cognito para el flujo de autenticación a fin de filtrar los dominios de correo. Así, solo los que estén incluidos como permitidos continuarán con el registro y acceso a la aplicación.

Lambda: Lambda es un servicio para la computación sin servidor, es decir, posibilita ejecutar código sin necesidad de administrar un servidor. Para este desarrollo, se usó como tiempo de ejecución a NodeJs en la versión 20. Asimismo, se cuenta con 3 lambdas, las cuales se detallan a continuación:

- Gestión usuarios: se consulta por correo electrónico, permite o deniega acceso a la aplicación o visualizar reporte, generar url para visualizar reporte y enviar la url al correo electrónico para visualizar en cualquier dispositivo a través del navegador.
- Filtro de registro: se deniega continuar el flujo de registro si no pertenece al dominio institucional permitido.

- Captura de ocupabilidad: se dispara por el flujo de eventos enviado por DynamoDB *Streams*. Aquí se calcula la ocupabilidad teniendo en cuenta el estado anterior y actual, y almacenarlo en S3.

AppSync: es un servicio GraphQL *serverless* en el cual se utiliza un lenguaje de consultas para leer o manipular datos. Además, provee una capacidad PUB/SUB en el cual los datos son distribuidos en tiempo real hacia los clientes suscriptores (equipos móviles). AppSync cuenta con 3 componentes principales en su implementación: *Schemas*, *Resolvers* y *DataSources*.

- *Schemas:* definen la forma de la data y cómo será recuperada o manipulada. AppSync se reserva 3 tipos: *Query* (operaciones de sólo lectura), *Mutation* (realizan acciones de inserción, modificación o eliminación en los *DataSources*) y *Subscription* (se establece una conexión *Websocket* entre el cliente y el servidor; cuando ocurre una modificación o mutación de datos, el cliente suscrito recibe las actualizaciones).
- *DataSources:* se define como el origen de datos al cual los *resolvers* acceden, procesan y retornan al cliente. AppSync soporta distintos tipos de fuentes de datos, en el proyecto se usarán: DynamoDB y Lambda.
- *Resolver:* definen cómo se manejan uno o varios campos dentro de un tipo de esquema durante una solicitud hecha por un cliente hacia los *dataSources*. Los *Resolvers* se asocian a campos específicos dentro de un type en el esquema.

A continuación, se muestra abajo una tabla con los *resolvers* (solucionadores) implementados en el presente proyecto, los usuarios con permisos de ejecución y una breve descripción de la operación realizada. Considerar que los solucionadores denominados “Actualizar” pertenecen al esquema de tipo Mutación.

Tabla 3.16 Solucionadores implementados y usuarios asociados [Elaboración propia]

Solucionador	Permiso	Descripción
Generar URL reporte	Analista y Administrador	Se envía el identificador del panel hacia la lambda “Gestión Usuarios” para generar una URL temporal que contiene de manera embebida los elementos visuales.
Enviar correo reporte	Analista y Administrador	Se envía el identificador del panel hacia la lambda “Gestión Usuarios” para enviar al correo electrónico una URL temporal y así acceder a los elementos visuales desde cualquier navegador web.
Listar grupos de usuario	Administrador	Se envía el identificador del usuario hacia la lambda “Gestión Usuarios” para retornar un listado de grupos al que pertenece.
Listar usuarios por email	Administrador	Se envía las iniciales de un correo hacia la lambda “Gestión Usuarios” para retornar un listado de usuarios que cumplen esa condición.
Actualizar usuario	Administrador	Se envía el identificador del usuario y las acciones de permitir o denegar el acceso a la aplicación, así como consultar reportes hacia la lambda “Gestión Usuarios” para actualizar dichos atributos.
Listar resumen estacionamiento	Invitado, Analista y Administrador	Se realiza una consulta hacia la tabla de DynamoDB para listar el resumen de cada zona de estacionamiento.
Detalle de una zona de estacionamiento	Invitado, Analista y Administrador	Se realiza una consulta hacia la tabla de DynamoDB para listar el detalle de una zona de estacionamiento en específico.
Actualizar estacionamiento	IoT Core Action	Cada dispositivo envía su estado en un tópico reservado hacia IoT Core. En él, se dispara la acción de realizar una petición HTTP hacia el endpoint de AppSync. De esa forma, se realiza la mutación que consiste en actualizar 2 ítems: detalle de ese dispositivo y el resumen a la zona que pertenece.
Actualizar dispositivo	IoT Core Action	Cuando un dispositivo se desconecta, se dispara una mutación en AppSync. Así, actualizamos los ítems de detalle del dispositivo y resumen de su zona a valores nulos para indicar que no tiene información.

AWS IOT Core: todo dispositivo requiere de un certificado X.590 para la autenticación con la plataforma IoT Core y, además, una política para indicar las acciones que puede ejecutar como, por ejemplo, conectarse, publicar un mensaje o suscribirse a un tópico. Para este proyecto, se cuenta con 2 tópicos reservados y definidos en la sección 3.6: Estacionamientos y *Disconnect*. El primer tópico, se usa para canalizar las actualizaciones que envían los dispositivos sobre los estados de cada espacio de parqueo. Así, se ejecuta una acción que consta de una petición HTTP hacia AppSync para realizar una mutación de “Actualizar estacionamiento”. En caso de error, se escriben tramas de log en CloudWatch.

En el segundo, se trata un tópico reservado que captura eventos cuando el dispositivo se desconecta de IoT Core. Así, se ejecuta una lectura en DynamoDB ingresando el nombre del dispositivo para obtener configuración relacionada a la zona que censa. Luego, se realiza una mutación en AppSync para registrar valores nulos a fin de indicar que no tiene información.

```

EspaciosAppSyncTopicRule:
  Type: AWS::IoT::TopicRule
  Properties:
    RuleName: !Join ["", [!Ref ProjectId, "_", !FindInMap [IoTTopicRuleEspa
    TopicRulePayload:
      ErrorAction:
        CloudwatchLogs:
          BatchMode: true
          LogGroupName: !Ref IoTRuleLogGroup
          RoleArn: !GetAtt IoTRuleInvokeRole.Arn
      Actions:
        - Http:
            Auth:
              Sigv4:
                RoleArn: !GetAtt IoTRuleInvokeRole.Arn
                ServiceName: appsync
                SigningRegion: !Sub ${AWS::Region}
            Url: !Ref AppSyncGraphQLUrl
    AwsIotSqlVersion: "2016-03-23"
    RuleDisabled: False
    Sql: !Join
      - ""
      - "SELECT "
      - "'mutation actualizarEstacionamiento($input: EspacioInput!)"
      - "'actualizarEstacionamiento' as operationName, "
      - " * as variables FROM '"
      - !Ref TopicName
      - ""
  
```

Figura 3.17 Regla configurada para el tópico “estacionamientos” [Elaboración propia]

Dynamo BD: es un servicio de base de datos NoSQL gestionado por AWS, el cual es escalable. Así mismo, los datos son distribuidos en las zonas de disponibilidad de una región, de manera que se ofrezca alta disponibilidad y son almacenados en discos SSD para mayor

durabilidad. En la implementación, se definió 1 tabla denominada “IoT-Espacios”, la cual cuenta con tres patrones de acceso con distintos usos:

- Listar el resumen de todas las zonas: permite obtener la cantidad total de espacios libres y exclusivos de discapacitados, además del nombre y código de zona. Se utilizan las llaves PK “EST” y SK que comienzan con el valor “RES_“ seguido del código de cada zona.

#	PK	SK	Ds	Re	Zo	Te
1	EST	RES_B	Dinthilac	{"rberry1":{"NULL"...	B	30
2	EST	RES_C	Ciencias	{"rbpqzk":{"M":{"L"...	C	40
3	EST	RES_D	Química	{"rbpfrm":{"M":{"L"...	D	50

Figura 3.18 Primer patrón de acceso [Elaboración propia]

```

1 {
2   "PK": "EST",
3   "SK": "RES_B",
4   "Ds": "Dinthilac",
5   "Re": {
6     "rberry1": null,
7     "rberry2": {
8       "Lnr": 6,
9       "Ocu": 2
10    },
11    "rberry3": {
12      "Ldi": 1,
13      "Lnr": 5,
14      "Ocu": 1
15    },
16    "rberry4": {
17      "Lnr": 8,
18      "Ocu": 0
19    }
20  },
21  "Te": 30,
22  "Zo": "B"
23 }

```

Figura 3.19 Primer patrón de acceso - json [Elaboración propia]

- Detalle de una zona en particular: obtiene la disponibilidad de cada espacio en específico dentro de la zona de parqueo seleccionada, además de indicar los espacios asignados para autoridades o discapacitados, el código de zona y la marca de tiempo de la última actualización. Se utilizan las llaves PK “EST” y SK que comienzan con el valor “DET_“ seguido del código de la zona escogida.

#	PK	SK	Ds	Ts	De	Zo
1	EST	DET_B_rberry4	rberry4	1425749635	["23":["S":"L"],"24"...	B
2	EST	DET_B_rberry3	rberry3	1425749635	["16":["S":"L"],"17"...	B
3	EST	DET_B_rberry2	rberry2	1425749635	["8":["S":"L"],"9":["...	B
4	EST	DET_B_rberry1	rberry1	1701624470	true	B

Figura 3.20 Segundo patrón de acceso [Elaboración propia]

```

Atributos  Ver JSON de DynamoDB
1 {
2   "PK": "EST",
3   "SK": "DET_B_rberry1",
4   "De": {
5     "1": "L",
6     "2": "L",
7     "3": "L",
8     "4": "L",
9     "5": "L",
10    "6": "L",
11    "7": "L"
12  },
13  "Ds": "rberry1",
14  "Ids": "1,2,3,4,5,6,7",
15  "Rau": "1",
16  "Rdi": "2,3",
17  "Ts": 1701833928,
18  "Zo": "B"
19 }

```

Figura 3.21 Segundo patrón de acceso - json [Elaboración propia]

- Obtener la zona asignada a un dispositivo: se obtiene el código de zona y el nombre de la PK a actualizar. En el caso de que un dispositivo SBC sea desconectado, se utiliza este patrón de acceso para actualizar la información de disponibilidad asociada al dispositivo dentro de los otros dos patrones mencionados.

```

1 {
2   "PK": "CONFIG",
3   "SK": "rberry1",
4   "IotConfig": {
5     "CodigoZona": "B",
6     "Dispositivo": "rberry1",
7     "Entidad": "EST"
8   }
9 }

```

Figura 3.22 Tercer patrón de acceso [Elaboración propia]

En la implementación, se dispone de *DynamoDB Stream*, la cual captura cada modificación realizada en los ítems de la tabla y las ordena en un flujo, con una retención de 5 minutos, para su posterior cálculo de ocupabilidad y almacenamiento en S3.

S3: es un servicio de AWS para el almacenamiento de objetos. En este desarrollo, se usa para guardar los reportes generados por el Lambda “Captura de ocupabilidad”. La clase seleccionada para el almacenamiento es la “Capa Inteligente”, el cual mueve dinámicamente la clase según el patrón de acceso a fin de ahorrar costos.

QuickSight: es un servicio de AWS para el análisis de datos y generación de *dashboards* interactivos y análisis de datos. En el presente proyecto, es utilizado para generar los reportes “Por espacio de estacionamiento” y “Por zona de estacionamiento” dentro de la aplicación. Quicksight crea su *dataset* de gráficos en base a los datos almacenados en S3, la ingesta se realiza cada 24 hrs.

CloudWatch: es un servicio de monitoreo de recursos AWS y ejecución de acciones automatizadas según umbrales definidos. En la implementación, este servicio se encarga de recopilar los registros de cada acción ejecutada por los Lambdas “Gestión Usuarios”, “Filtro de Registro” y “Captura de Ocupabilidad” en la infraestructura, además almacena los errores de cada mutación hacia el App Sync realizada por el bloque IoT Core. La figura 3.23 es un ejemplo de los registros capturados del Lambda “Captura de Ocupabilidad”.

Marca temporal	Mensaje
	No hay eventos antiguos en este momento. Volver a intentar
2023-12-05T22:43:48.160-05:00	INIT_START Runtime Version: nodejs:20.v13 Runtime Version ARN: arn:aws:lambda:us-east-1:runtime:6fb7e564fae33a7add879359ec9...
2023-12-05T22:43:48.559-05:00	START RequestId: afb1f63b-1636-4ffb-b0db-d35f1ac29759 Version: \$LATEST
2023-12-05T22:43:48.562-05:00	2023-12-06T03:43:48.5622 afb1f63b-1636-4ffb-b0db-d35f1ac29759 INFO { "Records": [{ "eventID": "76482196be3936f50e577a635e4..."
2023-12-05T22:43:48.580-05:00	2023-12-06T03:43:48.5802 afb1f63b-1636-4ffb-b0db-d35f1ac29759 INFO [{ "createdAt": 1701833929, "newImage": { "De": { "1": "...
2023-12-05T22:43:48.580-05:00	2023-12-06T03:43:48.5802 afb1f63b-1636-4ffb-b0db-d35f1ac29759 INFO categoria creada -> EST_DET
2023-12-05T22:43:48.585-05:00	2023-12-06T03:43:48.5852 afb1f63b-1636-4ffb-b0db-d35f1ac29759 INFO categoria creada -> EST_RES
2023-12-05T22:43:48.643-05:00	2023-12-06T03:43:48.6432 afb1f63b-1636-4ffb-b0db-d35f1ac29759 INFO ## COLECCION ##
2023-12-05T22:43:48.644-05:00	2023-12-06T03:43:48.6442 afb1f63b-1636-4ffb-b0db-d35f1ac29759 INFO { "EST_DET": [{ "Zona": "B", "Espacio": "1", "Estado": "...
2023-12-05T22:43:49.300-05:00	2023-12-06T03:43:49.3002 afb1f63b-1636-4ffb-b0db-d35f1ac29759 INFO S3 - Invocation time of putObject (SUCCESS): 655 ms
2023-12-05T22:43:49.340-05:00	2023-12-06T03:43:49.3402 afb1f63b-1636-4ffb-b0db-d35f1ac29759 INFO S3 - Invocation time of putObject (SUCCESS): 640 ms
2023-12-05T22:43:49.341-05:00	2023-12-06T03:43:49.3412 afb1f63b-1636-4ffb-b0db-d35f1ac29759 INFO ## Guardado a S3 ok ##
2023-12-05T22:43:49.357-05:00	END RequestId: afb1f63b-1636-4ffb-b0db-d35f1ac29759
2023-12-05T22:43:49.357-05:00	REPORT RequestId: afb1f63b-1636-4ffb-b0db-d35f1ac29759 Duration: 797.57 ms Billed Duration: 798 ms Memory Size: 256 MB Max M...
	REPORT RequestId: afb1f63b-1636-4ffb-b0db-d35f1ac29759 Duration: 797.57 ms Billed Duration: 798 ms Memory Size: 256 MB Max Memory Used: 92 MB Init Duration: 397.96 ms

Figura 3.23 Registros lambda captura de ocupabilidad [Elaboración propia]

3.8 Diseño de interfaz

Para el diseño de la aplicación móvil, se usaron algunas librerías y un *framework* que se detallan a continuación.

- **React Native:** es un *framework* de código abierto para el desarrollo multiplataforma usando JavaScript y React.
- **Maps:** es una librería que incluye componentes de mapas tanto para Android como iOS.
- **Expo:** es una plataforma para el desarrollo de aplicaciones móviles en Android & iOS.
- **Amplify SDK:** es un conjunto de librerías que facilitan la integración con AWS para la autenticación y consumo de servicios.
- **Redux Toolkit:** es una librería que simplifica el manejo de estado de la aplicación.

Vista principal de ingreso: la siguiente vista representa la ventana principal, en la que se puede encontrar un botón de Google para el acceso SSO de usuarios.



Figura 3.24 Vista principal - dispositivo móvil [Elaboración propia]

Vista de lista de disponibilidad del estacionamiento: en la siguiente vista, se tiene un mapa del campus universitario con las ubicaciones de las zonas de parqueo y la cantidad total de espacios disponibles. La pantalla cuenta con un filtro para visualizar solos los espacios disponibles de personas discapacitadas. Para indicar el estado de los espacios de parqueo se cuenta con la siguiente leyenda:

- Verde: Espacio disponible para todos.
- Rojo: Espacio disponible solo para autoridades.
- Azul: Espacio disponible para personas discapacitadas.
- Auto: Espacio ocupado.

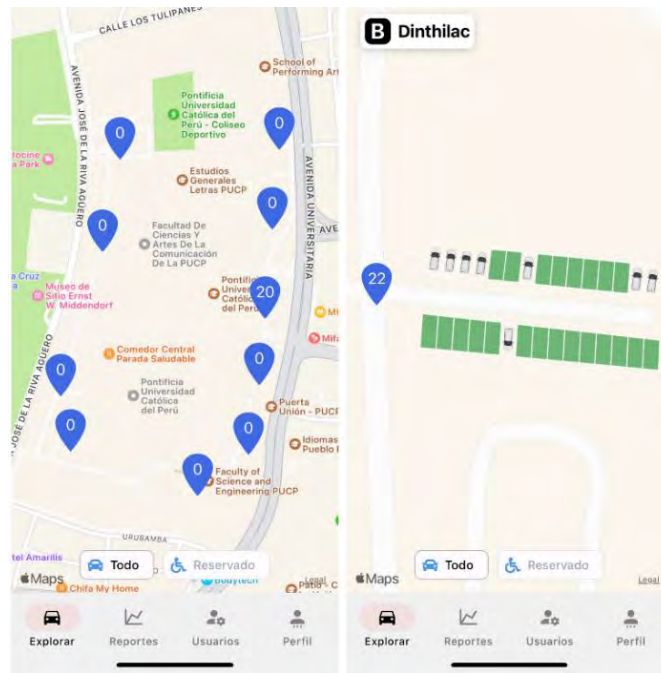


Figura 3.25 Vista de mapa de disponibilidad - dispositivo móvil [Elaboración propia]

Vista de perfil de usuario: la siguiente vista muestra el perfil básico del usuario (nombre, correo y rol asignado). La información es obtenida de la sesión de Cognito. El usuario cuenta con una opción para cerrar su sesión.

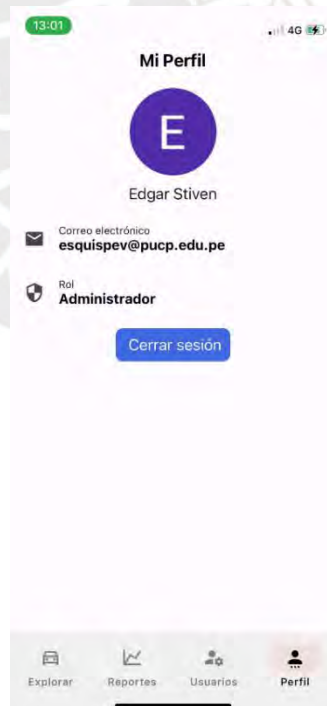


Figura 3.26 Vista de perfil de usuario - dispositivo móvil [Elaboración propia]

Vista de generar estadísticas: los usuarios analistas y administradores cuentan con esta sección para generar reportes estadísticos sobre la disponibilidad total de espacios en una zona y la ocupabilidad de cada espacio de parqueo. En esta ocasión, los gráficos se obtuvieron con datos de prueba generados manualmente por un cliente MQTT.

- Disponibilidad: se ingresan como parámetros el período o día en particular y el código del estacionamiento a la consulta. Con los filtros aplicados, se genera el gráfico mostrado en la Figura 3.27.

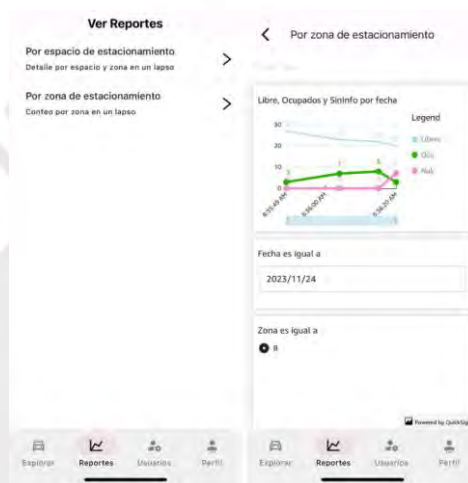


Figura 3.27 Reporte por zona de estacionamiento [Elaboración propia]

- Ocupabilidad: se calcula el tiempo que estuvo ocupado cada espacio de parqueo durante un periodo en particular. Se ingresan los filtros de periodo o día, código de estacionamiento y tiempo de duración (visualizar la data en horas, minutos o segundos).

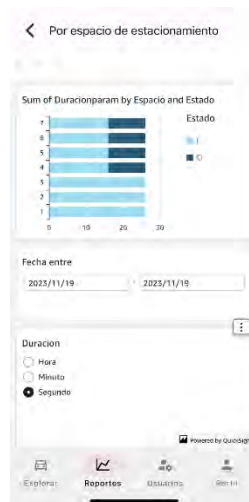


Figura 3.28 Reporte por espacio de estacionamiento [Elaboración propia]

Vista para administrar usuarios: por último, los administradores cuentan con una interfaz, la cual permite lo siguiente:

- Buscar al usuario por un correo en particular.
- Puede otorgar o denegar a un usuario el rol de analista.
- Puede habilitar o deshabilitar a un usuario.

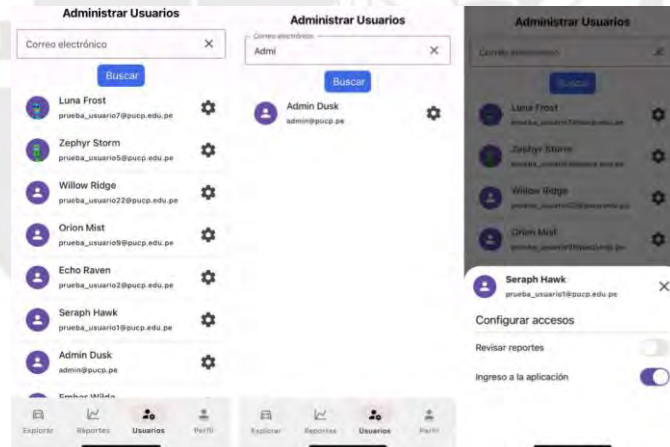


Figura 3.29 Panel del administrador - administración de usuarios. [Elaboración propia]

3.9 Determinación del tiempo de muestreo

Como se mencionó anteriormente, es importante establecer un periodo de muestreo (tiempo entre captura de *frames*) de los espacios de parqueo. Este parámetro debe ser tal que sumado a los tiempos máximos de procesamiento y transmisión a la aplicación móvil, sea menor al tiempo crítico establecido para el estacionamiento B.

$$T.Muestreo + T.Procesamiento + T.Transmisión \leq \text{Tiempo Crítico}$$

En la siguiente figura se muestran los valores de tiempo de procesamiento promedio y máximo determinados utilizando los algoritmos de IA y de configuración mencionados secciones atrás, además de los tiempos de transmisión y visualización final de los datos.

Lista de disponibilidad

último: 21/7/2020 16:19:05

```
Tiempo de procesamiento máximo obtenido: 5.268299341201782
Tiempo de procesamiento promedio obtenido: 3.0799317518870035
Último: 21/7/2020 16:19:5:741218
*****
*****
```

Figura 3.30 Tiempos obtenidos durante la ejecución [Elaboración propia]

El tiempo de procesamiento escogido es de 5,26 segundos mientras que el tiempo de transmisión es aproximadamente 1 segundo según lo visualizado, por ello el tiempo máximo de muestreo obtenido es de 20 segundos.

3.10 Elaboración de costos de servicio

Se muestran los costos de implementación y mantenimiento del sistema completo de 4 cámaras considerando 1000 usuarios activos por día, para mayor detalle de los costos en la infraestructura Cloud consultar el Anexo IV.

Tabla 3.17 Costos del proyecto [Elaboración propia]

SUBTIPO	DESCRIPCIÓN	COSTO UNITARIO	CANTIDAD	COSTO TOTAL
Sector Edge				
EQUIPO	Raspberry Pi 4 B, 4GB RAM + case con Cooler + disipadores + micro HDMI + fuente Alimentación 5V 3A + memoria 32 GB	\$ 130,88	4	\$ 523,52
MÓDULO	Raspberry Pi módulo de cámara V2, 8 megapíxeles, 1080p	\$ 26,99	4	\$ 107,96
MATERIAL	Dummy External Security Camera - HAY-772CITY	\$ 50,00	4	\$ 200,00
Sector Cloud				
INFRAESTRUCTURA	Servicio de componentes AWS IoT, Cognito, DynamoDB, Lambda, SES, AppSync, S3, QuickSight y CloudWatch con duración 1 mes	\$ 40,36	1	\$ 40,36
Sector Eléctrico				
MATERIAL	Adaptador de fuente de alimentación de 10W, 2A, CA 110V-220V a CC 5V	\$ 4,00	4	\$ 16,00
MATERIAL	Cable LH 2.5 mm Azul x 100 m	\$ 37,97	1	\$ 37,97
MATERIAL	Cable LH 2.5 mm Rojo x 100 m	\$ 37,97	1	\$ 37,97
MATERIAL	Cable LH 2.5 mm Verde/Amarillo x 100 m	\$ 37,97	1	\$ 37,97
MATERIAL	Toma Doble Universal Blanco	\$ 2,90	4	\$ 11,60
MATERIAL	Caja Universal Toma Eléctrica	\$ 1,50	4	\$ 6,00
Implementación				
SERVICIO	Mano de obra - Instalación y conexión dispositivos SBC en postes de alumbramiento por día	\$ 40,00	2	\$ 80,00
SERVICIO	Mano de obra - Configuración dispositivos SBC por día	\$ 105,00	2	\$ 210,00
MATERIAL	Uso de EPPs por día (casco, chaleco, arnés, lentes de seguridad, guantes, escalera)	\$ 30,00	2	\$ 60,00
SERVICIO	Transporte de materiales a obra	\$ 12,00	1	\$ 12,00
SERVICIO	Movilidad de personal	\$ 5,00	8	\$ 40,00

Capítulo 4: Prueba del sistema de disponibilidad y evaluación de los parámetros críticos

La implementación de este proyecto requiere de pruebas exhaustivas para garantizar su correcto funcionamiento y evaluar los parámetros esenciales. Este capítulo establece el plan de pruebas para verificar la operatividad del sistema, además de propuestas en la distribución de cámaras de acuerdo a la cobertura. Por otro lado, se visualizan las estadísticas relevantes para medir el desempeño de una zona de parqueo, a fin de realizar un monitoreo eficiente del área durante diversos periodos en el año.

4.1 Plan de pruebas

Culminada la etapa de diseño, se procede a elaborar el plan de pruebas que demuestren la eficacia del sistema, comprende de los siguientes procedimientos:

- Funcionamiento de la propuesta.
- Cobertura y ubicación de las cámaras.
- Elaboración de gráficos estadísticos.

4.1.1 Pruebas de funcionamiento.

En esta etapa, se evalúa el desempeño del sistema en base al correcto flujo de información extremo a extremo y los parámetros críticos establecidos (Tiempo Crítico y Precisión) mediante la ejecución del algoritmo en un dispositivo Raspberry Pi 3 modelo B+ (con id igual a rberry11). La cobertura de cámara se realizará en 38 espacios centrales del total de espacios comentados en el Capítulo 3, esto con el fin de tener mejor visualización de vehículos, ya que los espacios de estacionamiento ubicados en los extremos tienen dimensiones menores en comparación con los espacios centrales, y los autos en esas ubicaciones quedan solapados por sus vecinos. Se seleccionó la cámara 7 del *dataset* CNR-EXT y se evaluarán distintas condiciones climáticas (2015-11-12 *SUNNY*, 2015-11-25 *OVERCAST*, 2016-01-08 *RAINY*). La distribución de los espacios por cámara/dispositivo se realiza manualmente configurando en cada uno de ellos una grilla o cuadrícula con las áreas de interés. En la figura 4.1 se muestra el rango propuesto de 38 espacios en la cámara 7.



Figura 4.1 Cámara 7 overcast 2015-11-25_1419 [Elaboración propia]

Según la tabla de DynamoDB definida en el capítulo 3, se agrega la configuración correspondiente de ese estacionamiento y se asignan los 38 espacios seleccionados.

<input type="checkbox"/>	PK (Cadena)	SK (Cadena)	De	Ds	Ids	lotConfig
<input type="checkbox"/>	EST	DET_CNR A_rberry11	{ "1": { "S": ...	rberry11	1,10,11,12,...	
<input type="checkbox"/>	EST	RES_CNR A		CNR A		
<input type="checkbox"/>	CONFIG	rberry11				["Dispositiv...

Figura 4.2 Elementos creados en dynamodb [Elaboración propia]

Se evaluó el flujo de datos desde el dispositivo Raspberry hasta la interfaz móvil. A continuación, se muestran imágenes de las pruebas realizadas para dos horarios aleatorios del *dataset* en el mismo día. En ambos casos, se comprueba que la cantidad de espacios ocupados generada en el Raspberry es la misma que la visualizada en la aplicación.

- **HORARIO DATASET: 12/11/2015 15:17**

```

Filtrando clases vehiculos
tiempo: 39.02712297439575, cpu: 77.5, memoria: 2904.6779432235203
/home/pi/Downloads/Final/FULL_IMAGE_1000x750/SUNNY/2015-11-12/camera7/2015-11-12_1517.jpg image ...
Done!

Libres: [1, 5, 6, 10, 11, 16, 34, 38]
Ocupados: [8, 9, 20, 12, 19, 21, 15, 3, 24, 25, 27, 14, 23, 40, 4, 41, 2, 13, 28, 26, 33, 43, 39, 32, 22, 7, 37, 42, 36, 35] - TOTAL: 30
MQTT enviado ... Done!
Estado Anterior actualizado ... Done!

Tiempo de procesamiento máximo obtenido: 45.02268195152283
Tiempo de procesamiento promedio obtenido: 40.40909589038176
Último: 16/12/2023 22:2:20:320910
*****

```

Figura 4.3 Primer horario – dispositivo SBC [Elaboración propia]

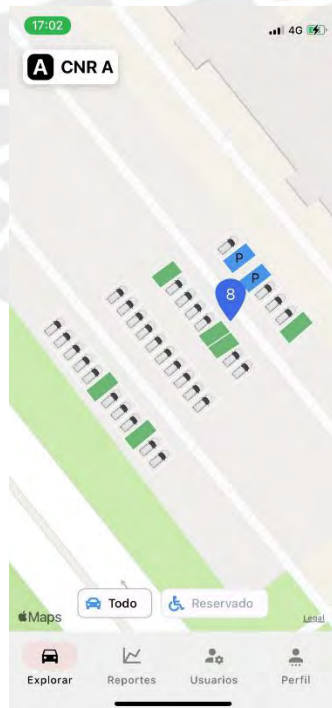


Figura 4.4 Primer horario – app móvil [Elaboración propia]

- **HORARIO DATASET: 12/11/2015 16:47**

```
Filtrando clases vehiculos
tiempo: 39.346108198165894, cpu: 76.9, memoria: 2905.791538364196
/home/pi/Downloads/Final/FULL_IMAGE_1000x750/SUNNY/2015-11-12/camera7/2015-11-12_1647.jpg image ...
Done!

Libres: [2, 4, 5, 6, 8, 16, 22, 24, 32, 34, 37, 38, 39]
Ocupados: [9, 20, 12, 15, 11, 21, 25, 14, 40, 19, 3, 43, 23, 10, 27, 13, 41, 26, 33, 36, 28, 35, 1,
7, 42] - TOTAL: 25
MQTT enviado ... Done!
Estado Anterior actualizado ... Done!

Tiempo de procesamiento máximo obtenido: 45.02268195152283
Tiempo de procesamiento promedio obtenido: 41.06014156341553
Último: 16/12/2023 22:4:50:144730
*****
*****
```

Figura 4.5 Segundo horario – dispositivo SBC [Elaboración propia]

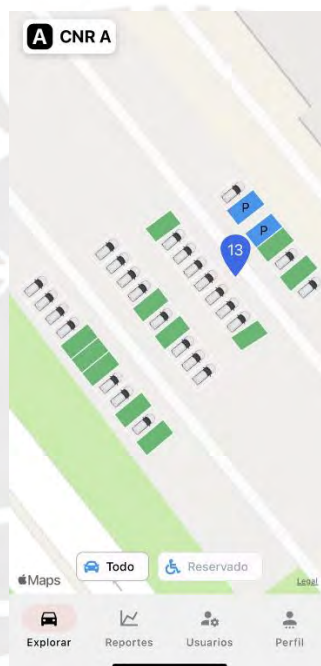


Figura 4.6 Segundo horario – app móvil [Elaboración propia]

El sistema se caracteriza según el desempeño que presenta respecto a sus parámetros críticos establecidos. El análisis de estos parámetros se explica a continuación:

TIEMPO CRÍTICO: esta sección busca dar un mayor acercamiento de cómo el tiempo crítico se relaciona con las especificaciones del dispositivo de borde. En el capítulo 3, se estableció para el estacionamiento B un tiempo crítico general de 27 segundos. Al realizar las pruebas del sistema con un ordenador personal (especificaciones dadas en el subcapítulo 3.3.2) y una sola cámara, se obtuvo un tiempo de procesamiento menor al tiempo crítico y se pudo determinar un parámetro de muestreo de 20 segundos, como se observó al final del capítulo

anterior. En este capítulo, se realizaron pruebas con un dispositivo Raspberry 3B+; el tiempo de procesamiento obtenido fue de aproximadamente 38,4 segundos con picos cercanos a los 45 segundos, valores muy superiores al tiempo crítico, por lo que no es posible establecer un tiempo de muestreo y se concluye que estos dispositivos evaluados no cuentan con los recursos apropiados para el correcto funcionamiento según los parámetros establecidos.

```
Filtrando clases vehiculos
tiempo: 38.938854455947876, cpu: 76.6, memoria: 2899.363057324841
/home/pi/Downloads/Final/FULL_IMAGE_1000x750/RAINY/2016-01-08/camera7/2016-01-08_1728.jpg image
Done!

Libres: [1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 13, 19, 20, 21, 23, 24, 25, 26, 28, 32, 33, 34, 35,
37, 38, 39, 40, 41, 42, 43]
Ocupados: [14, 15, 16, 27, 12, 22] - TOTAL: 6
MQTT enviado ... Done!
Estado Anterior actualizado ... Done!

Tiempo de procesamiento máximo obtenido: 44.6491117477417
Tiempo de procesamiento promedio obtenido: 38.41598191715422
Último: 16/12/2023 22:40:0:826745
*****
*****
```

Figura 4.7 Tiempos de procesamiento obtenidos [Elaboración propia]

PRECISIÓN: se establece como la capacidad del sistema para identificar correctamente los vehículos presentes en los espacios de disponibilidad asignados. Es determinado como la razón entre los vehículos detectados en los espacios y el total de vehículos presentes.

$$\rho = \frac{\text{vehículos detectados}}{\text{vehículos presentes}}$$

Se muestra, como ejemplo, en la Tabla 4.1 la precisión obtenida utilizando un Raspberry 3B+. La información pertenece a un día entero del *dataset*, demás fechas en el Anexo III.

Tabla 4.1 Precisión dataset overcast [Elaboración propia]

HORARIO	Ocupados Cámara 7 (38 espacios)	Vehículos presentes	Precisión (Cámara 7)
2015-11-25_0719	2	2	1,00
2015-11-25_0749	3	3	1,00
2015-11-25_0819	3	3	1,00
2015-11-25_0849	5	5	1,00
2015-11-25_0919	8	8	1,00
2015-11-25_0949	17	17	1,00
2015-11-25_1019	21	21	1,00

HORARIO	Ocupados Cámara 7 (38 espacios)	Vehículos presentes	Precisión (Cámara 7)
2015-11-25_1049	32	35	0,91
2015-11-25_1119	35	36	0,97
2015-11-25_1149	35	36	0,97
2015-11-25_1219	35	35	1,00
2015-11-25_1249	33	33	1,00
2015-11-25_1319	29	30	0,97
2015-11-25_1349	31	31	1,00
2015-11-25_1419	31	32	0,97
2015-11-25_1449	32	32	1,00
2015-11-25_1519	32	32	1,00
2015-11-25_1549	33	33	1,00
2015-11-25_1619	31	31	1,00
2015-11-25_1649	26	27	0,96
2015-11-25_1719	14	24	0,58
2015-11-25_1749	6	19	0,32
			Promedio (Cámara 7)
			0,94

La Tabla 4.2 muestra los valores de precisión promedio obtenidos del estacionamiento en cada condición climática (soleado, lluvioso, nublado). La diferencia entre estos valores se debe mayormente a variaciones de luz y obstáculos en el lente (neblina y lluvia), además en estaciones de lluvia el ocaso ocurre más pronto. Por otro lado, se observa que los espacios de parqueo más lejanos a la cámara no suelen tener una correcta detección, ya que no cuentan con un espacio amplio de grilla definida, los vehículos suelen solaparse con otros autos alrededor y su reducido tamaño en el lente de la cámara complica la detección por el algoritmo de IA.

Tabla 4.2 Valores de precisión por fecha [Elaboración propia]

Fecha	Precisión promedio (Cámara 7)
2015-11-12	0,92
2015-11-25	0,94
2016-01-08	0,83
TOTAL	0,90

4.1.2 Cobertura y ubicación de dispositivos.

A diferencia del estacionamiento B, cada cámara del *dataset* CNR-EXT se encuentra ubicado a una distancia y altura óptimas de los espacios de manera que se tiene completa visualización de las 4 filas de parqueo en cuanto a profundidad, mientras que de manera horizontal se pueden visualizar hasta 7 autos en la primera fila. Para poder determinar la correcta densidad de dispositivos a colocar en el área de interés se debe considerar la distribución de los postes de alumbramiento alrededor. Como se observa en la figura 4.11, la distancia de los postes a los espacios de parqueo es muy corta (1m), por lo que, aunque se instalen los dispositivos SBC a una elevada altura no se tendría una visualización completa de los vehículos más cercanos y por ende estos podrían no ser detectados por el algoritmo. Por ello, se propone el despliegue de 4 dispositivos/cámaras, una por cada poste, a una altura superior a los 3 metros de manera que brinden cobertura de las filas más alejadas, cada dispositivo monitorea un conjunto de 7 u 8 espacios como se muestra.

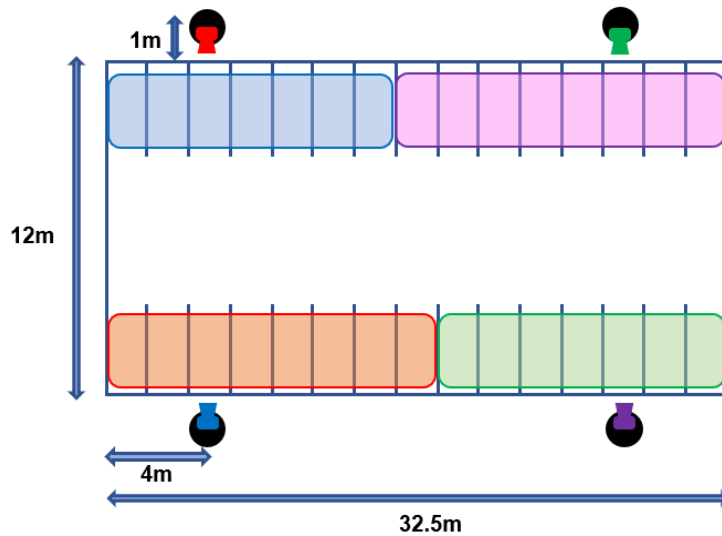


Figura 4.8 Distribución de los dispositivos en el estacionamiento B [Elaboración propia]

4.1.3 Estadísticas.

Al igual que en las anteriores pruebas, en esta etapa se busca visualizar mediante *Dashboards* en Quicksight la información enviada por el Raspberry y almacenada en la nube (bloque Amazon S3). Se aplicaron los filtros respectivos de fecha de ejecución y zona de estacionamiento CNR A.

En la figura 4.12 se visualiza el gráfico de disponibilidad obtenido, en el eje X se muestran los instantes en que los dispositivos Raspberry enviaron sus datos durante la ejecución, por lo que no guarda relación alguna con el horario de captura del *dataset*, mientras que en el eje Y se muestra la disponibilidad y ocupabilidad total del estacionamiento, además de la cantidad de espacios no reportados debido a fallas por el dispositivo SBC (Nuk). Para el presente ejemplo, se observa que a partir del mediodía se alcanza el pico de espacios ocupados, también el estacionamiento suele ocuparse casi en su totalidad. Es importante aclarar que algunos horarios del *dataset* no figuran en el reporte ya que los dispositivos no detectaron cambios respecto a un estado anterior y, por ende, no publicaron ningún mensaje.

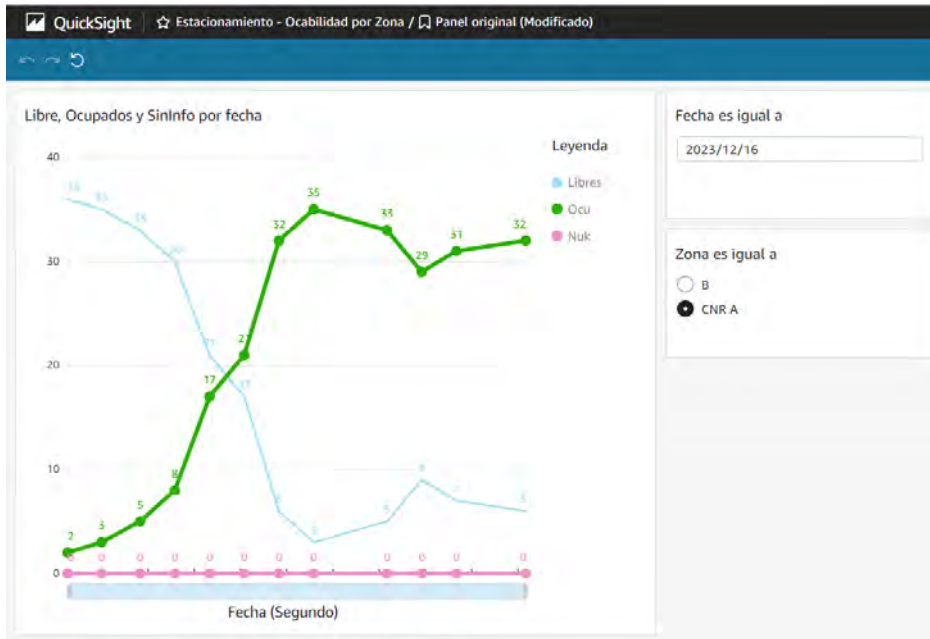


Figura 4.9 Gráfico de disponibilidad [Elaboración propia]

La figura 4.10 muestra el gráfico de ocupabilidad por cada espacio. En el ejemplo, se observa que los espacios 4 y 25 fueron los de mayor porcentaje de ocupabilidad durante el día, mientras que los espacios 5 y 6 (espacios reservados) nunca fueron ocupados.

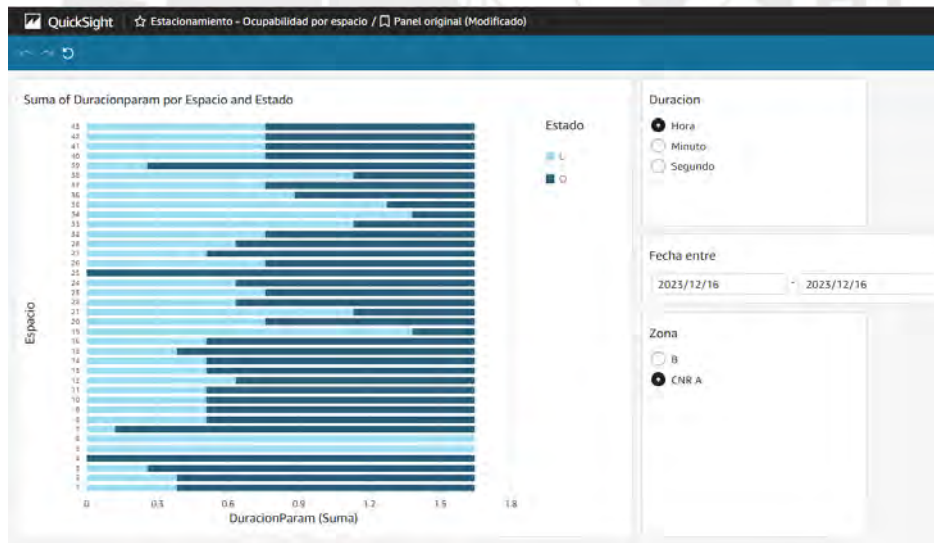


Figura 4.10 Gráfico de ocupabilidad por espacio [Elaboración propia]

Conclusiones

- Se definió el parámetro de tiempo crítico como el tiempo de actualización máximo del sistema extremo a extremo y se determinó un valor de 27 segundos para el estacionamiento B.
- Se evaluaron los algoritmos de detección YOLO y Faster R-CNN mediante el consumo de recursos computacionales, tiempo de procesamiento, precisión, recuperación y exactitud. YOLO demostró un mejor rendimiento con el *dataset* de prueba seleccionado, este último con características similares al estacionamiento propuesto.
- Se implementó una arquitectura Serverless en AWS para el almacenamiento y distribución de los datos enviados por los dispositivos SBC considerando como pilares la comunicación segura entre componentes, auto escalabilidad y optimización en rendimiento y costos.
- Se desarrolló una aplicación móvil con autenticación de usuarios por Google para que, según el tipo de rol adjunto, tengan acceso a los valores de disponibilidad, gestión de usuarios y visualización de reportes.
- Se definió la frecuencia de muestreo de los dispositivos; este parámetro, sumado a los tiempos de demora en cada etapa del sistema, debe ser siempre menor o igual al tiempo crítico.
- Se evaluó el funcionamiento del sistema en base al tiempo crítico y precisión utilizando un ordenador y un dispositivo Raspberry 3B+ y se concluyó que es necesario adquirir modelos SBC con mejores recursos gráficos y computacionales para cumplir con el tiempo crítico.
- Se determinó la cantidad de 4 dispositivos para la cobertura del estacionamiento B, una en cada poste de alumbramiento.

- Se evaluaron los gráficos estadísticos de disponibilidad y ocupabilidad generados en la aplicación móvil, la información mostrada en gráficos concuerda con el flujo de vehículos respectivo en el *dataset*.



Recomendaciones

- Para determinar los puntos de instalación de los dispositivos SBC se recomienda consultar un mapa de cobertura WIFI para garantizar la conectividad hacia la plataforma *Cloud* en el estacionamiento B.
- Para obtener un menor tiempo de detección de vehículos, se recomienda entrenar un propio modelo con un *dataset* específico de vehículos.
- Como parte de la implementación en el estacionamiento, es necesario el uso de una carcasa que brinde la protección y refrigeración adecuada a los dispositivos SBC, además de un lubricante antipolvo y lluvia para la correcta visualización de la cámara a través del lente de la carcasa.
- Con la finalidad de tener mayor seguridad y cobertura, se debe evaluar migrar el transporte de datos a Sigfox en base a la cantidad de eventos generados por día.
- En la aplicación móvil, se puede incluir un módulo que permita exportar los datos estadísticos consultados en un formato CSV o PDF.

Bibliografía

- [1] Agencia Peruana de Noticias, “Perú superará los US\$1.500M de inversión en telecomunicaciones al cierre de 2019 | AméricaEconomía | AméricaEconomía”, oct. 14, 2019. <https://www.americaeconomia.com/negocios-industrias/peru-superara-los-us1500m-de-inversion-en-telecomunicaciones-al-cierre-de-2019> (consultado jun. 20, 2020).
- [2] Osiptel, “Temas en Telecomunicaciones N° 16”, pp. 48–53, [En línea]. Disponible en: https://www.osiptel.gob.pe/Archivos/Publicaciones/clasificacion_servicios_6.pdf.
- [3] ANDINA, “Perú podría incrementar su PBI hasta 7.9% con la inteligencia artificial | Noticias | Agencia Peruana de Noticias Andina”, nov. 21, 2019. <https://andina.pe/agencia/noticia-peru-podria-incrementar-su-pbi-hasta-79-con-inteligencia-artificial-775741.aspx> (consultado jun. 20, 2020).
- [4] R. Albrieu, M. Rapetti, C. Brest López, P. Larroulet, y A. Sorrentino, “Inteligencia artificial y crecimiento económico. Oportunidades y desafíos para Perú”, oct. 2018. Consultado: jun. 20, 2020. [En línea]. Disponible en: <https://news.microsoft.com/uploads/prod/sites/41/2018/11/IA-y-Crecimiento-PERU.pdf>.
- [5] A. Naimat, “The New Artificial Intelligence Market – O’Reilly”, ago. 29, 2016. <https://www.oreilly.com/radar/the-new-artificial-intelligence-market/> (consultado jun. 20, 2020).
- [6] M. Zaforas, “Inteligencia Artificial como servicio: reconocimiento de imágenes - Paradigma”. <https://www.paradigmadigital.com/techbiz/inteligencia-artificial-servicio-reconocimiento-imagenes/> (consultado jun. 20, 2020).
- [7] Congreso de la República, “Ley29461.pdf”, 2009, [En línea]. Disponible en: <https://docs.peru.justia.com/federales/leyes/29461-nov-27-2009.pdf>.
- [8] Lima Cómo Vamos, “Tablas_EncuestaLCV2019.xlsx”, 2019. https://www.dropbox.com/s/xxuqthi7w3e0xld/Tablas_EncuestaLCV2019.xlsx?dl=0 (consultado jun. 20, 2020).
- [9] RPP, “¿Es posible vivir sin autos? Transporte alternativo para un aire más limpio”, feb. 17, 2020. <https://rpp.pe/peru/actualidad/es-posible-vivir-sin-autos-transporte-alterno-para-un-aire-mas-limpio-noticia-1246106> (consultado jun. 20, 2020).
- [10] RPP, “Lima tiene un déficit de al menos 45 mil espacios para estacionar | RPP Noticias”, mar. 21, 2016. <https://rpp.pe/lima/actualidad/lima-tiene-un-deficit-de-al-menos-45-mil-espacios-para-estacionar-noticia-947330> (consultado jun. 20, 2020).
- [11] Pontificia Universidad Católica del Perú, “Estacionamientos para autos, motos y bicicletas en el campus - PUCP | Pontificia Universidad Católica del Perú”. <https://www.pucp.edu.pe/servicio/estacionamientos-campus/> (consultado jun. 20, 2020).
- [12] Pontificia Universidad Católica del Perú, “Mapa campus PUCP”. <https://www.pucp.edu.pe/mapa-campus/> (consultado jun. 20, 2020).
- [13] L. E. Lopez Huamán, “Entrevista sobre la administración de los estacionamientos en el campus universitario PUCP”, 2019.
- [14] Los Portales S.A., “Informe Gerencial PUCP - Febrero 2020”, pp. 12–13, 2020.

- [15] Intellisoft Parking, “Intellisoft Parking - Jockey Plaza”. <http://www.intellisoftparking.com/cms/index.php/proyectos/centroscomerciales/76-jpsc> (consultado jun. 21, 2020).
- [16] Intellisoft Parking, “Intellisoft Parking - La Rambla”. <http://www.intellisoftparking.com/cms/index.php/proyectos/centroscomerciales/84-larambla> (consultado jun. 21, 2020).
- [17] CIRCONTROL, “CirPark Solutions for Efficient Parking - Product Catalogue 2019”, 2019.
- [18] P. Rivera Ramírez, “Diseño de una red inalámbrica de sensores para el monitoreo de la disponibilidad de espacios en un estacionamiento vehicular de una universidad”, Lima, 2018.
- [19] Parkam, “Smart Sign Guidance”. https://www.parkam.com/?page_id=1326 (consultado jun. 21, 2020).
- [20] A. Athira, S. Lekshmi, P. Vijayan, y B. Kurian, “Smart parking system based on optical character recognition”, en *Proceedings of the International Conference on Trends in Electronics and Informatics, ICOEI 2019*, abr. 2019, vol. 2019-April, pp. 1184–1188, doi: 10.1109/icoei.2019.8862517.
- [21] Dirección de Tecnologías de Información (DTI), “PUCP | Estacionamiento B :: Descubre PUCP”, 2013. <http://descubre.pucp.edu.pe/poi/est001> (consultado jul. 08, 2020).
- [22] M. Mangat, “Edge Computing vs Cloud Computing: Key Differences”. <https://phoenixnap.com/blog/edge-computing-vs-cloud-computing> (consultado jun. 27, 2020).
- [23] KIO Networks, “Ventajas y desventajas de Edge Computing”. <https://www.kionetworks.com/blog/data-center/ventajas-y-desventajas-de-edge-computing> (consultado jun. 27, 2020).
- [24] Stanford CS class, “Image Classification: Data-driven Approach, k-Nearest Neighbor, train/val/test splits”, 2019. <https://cs231n.github.io/classification/> (consultado jun. 21, 2020).
- [25] Stanford CS class, “Linear classification: Support Vector Machine, Softmax”, 2019. <https://cs231n.github.io/linear-classify/> (consultado jun. 21, 2020).
- [26] Stanford CS class, “Optimization: Stochastic Gradient Descent”, 2019. <https://cs231n.github.io/optimization-1/> (consultado jun. 21, 2020).
- [27] Stanford CS class, “Neural Networks Part 1: Setting up the Architecture”, 2019. <https://cs231n.github.io/neural-networks-1/> (consultado jun. 21, 2020).
- [28] Stanford CS class, “Neural Networks Part 2: Setting up the Data and the Loss”, 2019. <https://cs231n.github.io/neural-networks-2/> (consultado jun. 21, 2020).
- [29] Stanford CS class, “Convolutional Neural Networks: Architectures, Convolution / Pooling Layers”, 2019. <https://cs231n.github.io/convolutional-networks/> (consultado jun. 21, 2020).
- [30] J. Redmon, S. Divvala, R. Girshick, y A. Farhadi, “You only look once: Unified, real-time object detection”, en *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, dic. 2016, vol. 2016-Decem, pp. 779–788, doi: 10.1109/CVPR.2016.91.
- [31] F.-F. Li, J. Johnson, y S. Yeung, “Lecture 11: Detection and Segmentation”, may

- 2017.
- [32] R. Girshick, "Fast R-CNN", 2015, doi: 10.1109/ICCV.2015.169.
 - [33] Stanford University School of Engineering, *Lecture 11 | Detection and Segmentation*. 2017.
 - [34] T. Shah, "Measuring Object Detection models — mAP — What is Mean Average Precision?", ene. 26, 2018. <https://towardsdatascience.com/what-is-map-understanding-the-statistic-of-choice-for-comparing-object-detection-models-1ea4f67a9dbd> (consultado jun. 27, 2020).
 - [35] T. Shah, "Precision and Recall — An intuitive and simple explanation", dic. 03, 2017. <https://medium.com/@starang/precision-and-recall-a-brief-intro-38589a21a09> (consultado jun. 27, 2020).
 - [36] S. Rothe, S. Felix Wirtz, y D. Söfker, "Confusion matrix and related performance measures", nov. 2016. https://www.researchgate.net/figure/Confusion-matrix-and-related-performance-measures_fig2_328860691 (consultado jun. 27, 2020).
 - [37] RASPBERRY PI FOUNDATION, "What is a Raspberry Pi?" <https://www.raspberrypi.org/help/what-is-a-raspberry-pi/> (consultado jun. 21, 2020).
 - [38] RASPBERRY PI FOUNDATION, "Raspberry Pi 3 Model B+". <https://www.raspberrypi.org/products/raspberry-pi-3-model-b-plus/> (consultado jun. 21, 2020).
 - [39] RASPBERRY PI FOUNDATION, "Raspberry Pi 4 Model B specifications". <https://www.raspberrypi.org/products/raspberry-pi-4-model-b/specifications/> (consultado jun. 21, 2020).
 - [40] SEA SLUG LABS LLC, "Raspberry Pi Sigfox Board". <https://seasluglabs.io/collections/frontpage/products/wisol-raspberrypi-hat> (consultado jun. 21, 2020).
 - [41] RASPBERRY PI FOUNDATION, "Camera Module". <https://www.raspberrypi.org/documentation/hardware/camera/> (consultado jun. 21, 2020).
 - [42] NVIDIA, "JETSON NANO". <https://www.nvidia.com/es-la/autonomous-machines/embedded-systems/jetson-nano/> (consultado jun. 21, 2020).
 - [43] NVIDIA, "Embedded Systems for Next-Generation Autonomous Machines". <https://www.nvidia.com/es-la/autonomous-machines/embedded-systems/> (consultado jun. 21, 2020).
 - [44] ANRATEK, "AC8265 Wireless NIC for Jetson Nano, WiFi & Bluetooth". <https://www.antratek.com/ac8265-wireless-nic-for-jetson-nano-wifi-bluetooth> (consultado jun. 21, 2020).
 - [45] Leopard Imaging Inc., "LI-IMX219-MIPI-FF-NANO-H136". https://www.leopardimaging.com/product/nvidia-jetson-cameras/nvidia_nano_mipi_camera_kits/li-imx219-mipi-ff-nano/li-imx219-mipi-ff-nano-h136/ (consultado jun. 21, 2020).
 - [46] Google Cloud, "Edge TPU". <https://cloud.google.com/edge-tpu?hl=es> (consultado jun. 21, 2020).
 - [47] CORAL, "Dev Board". <https://coral.ai/products/dev-board/> (consultado jun. 21, 2020).
 - [48] CORAL, "Camera". <https://coral.ai/products/camera/> (consultado jun. 21, 2020).

- [49] W. Serrano, "Digital Systems in Smart City and Infrastructure: Digital as a Service", *Smart Cities*, vol. 1, núm. 1, pp. 134–153, nov. 2018, doi: 10.3390/smartcities1010008.
- [50] RF Wireless World, "WLAN vs Ethernet LAN | difference between WLAN and LAN". <https://www.rfwireless-world.com/Terminology/WLAN-vs-Ethernet-LAN.html> (consultado jun. 21, 2020).
- [51] B. Edson, "Creating the Internet of Your Things", *Microsoft Corp.*, 2015.
- [52] T. McCourt *et al.*, "Technology & Communications The Internet of Things A Study in Hype, Reality, Disruption, and Growth", 2014.
- [53] J. Mocnej, A. Pekar, W. K. G. Seah, y I. Zolotova, "Network Traffic Characteristics of the IoT Application Use Cases".
- [54] P. Christensson, "WLAN Definition", may 22, 2020. <https://techterms.com/definition/wlan> (consultado jun. 21, 2020).
- [55] M. Rouse, "What is MQTT and How Does it Work?" <https://internetofthingsagenda.techtarget.com/definition/MQTT-MQ-Telemetry-Transport> (consultado jun. 21, 2020).
- [56] BehrTech Blog, "Why You Need MQTT in Your IoT Architecture". <https://behrtech.com/blog/mqtt-in-the-iot-architecture/> (consultado jun. 21, 2020).
- [57] Sigfox, "Our story". <https://www.sigfox.com/en/sigfox-story> (consultado jun. 21, 2020).
- [58] SIGFOX, *SIGFOX TECHNOLOGY: Sigfox Network Architecture*. 2016.
- [59] SIGFOX, "Downlink Connectivity", feb. 08, 2017. https://www.youtube.com/watch?v=ZJDhWaW1b_M&feature=youtu.be (consultado jun. 21, 2020).
- [60] SIGFOX, "SIGFOX TECHNOLOGY: Security of the Sigfox Network", feb. 16, 2017. <https://www.youtube.com/watch?v=zUUUmOVlr1pQ&feature=youtu.be> (consultado jun. 21, 2020).
- [61] LoRa Alliance®, "About LoRa Alliance®". <https://lora-alliance.org/about-lora-alliance> (consultado jun. 21, 2020).
- [62] T. Rheinland, "LoRaWAN™ What is it? A technical overview of LoRa® and LoRaWAN™", San Ramon, nov. 2015. Consultado: jun. 21, 2020. [En línea]. Disponible en: https://www.tuv.com/media/corporate/products_1/electronic_components_and_lasers/TUeV_Rheinland_Overview_LoRa_and_LoRaWANtmp.pdf.
- [63] IBM, "¿Qué es cloud computing? ". <https://www.ibm.com/es-es/cloud/learn/cloud-computing> (consultado jun. 21, 2020).
- [64] G. Kuger Ríos, "Identificación de Oportunidades con la Computación en la Nube".
- [65] D. Pardo, "Tipos de cloud: nubes públicas, privadas e híbridas en el horizonte", jun. 12, 2019. <https://pandorafms.com/blog/es/tipos-de-cloud/> (consultado jun. 21, 2020).
- [66] Amazon Web Services Inc., "What is AWS", 2020. https://aws.amazon.com/what-is-aws/?nc1=h_ls (consultado jun. 21, 2020).
- [67] N. Chaturvedi, "Amazon Web Services: Important AWS Services — Part 1", mar. 03, 2020. <https://medium.com/analytics-vidhya/amazon-web-services-important-aws-services-part-1-bfad23d4e2a2> (consultado jun. 21, 2020).

- [68] Microsoft Azure, “Get to Know Azure ”. <https://azure.microsoft.com/en-us/overview/> (consultado jun. 21, 2020).
- [69] J. Arbiol, “¿Qué es Microsoft Azure y para qué sirve? Por qué elegir la plataforma cloud de Microsoft”, jul. 24, 2019. <https://www.hiberus.com/crecemos-contigo/azure-la-plataforma-cloud-de-microsoft/> (consultado jun. 21, 2020).
- [70] DescubrePUCP, “Dirección de Tecnologías de Información (DTI)”, 2013. <http://descubre.pucp.edu.pe/poi/dir013> (consultado jun. 21, 2020).
- [71] GOGO, “¿Cuáles son los Límites de Velocidad en Cada Tipo de Vía del Perú?”, mar. 11, 2018. <https://gogo.pe/noticias/cuales-son-los-limites-de-velocidad-en-cada-tipo-de-via-del-per/> (consultado jul. 21, 2020).
- [72] Google, “Google Maps”. <https://www.google.com.pe/maps/@-12.0630149,-77.0296179,13z?hl=es-419> (consultado jul. 08, 2020).
- [73] P. Sharma, “Image Classification vs. Object Detection vs. Image Segmentation”, ago. 21, 2019. <https://medium.com/analytics-vidhya/image-classification-vs-object-detection-vs-image-segmentation-f36db85fe81> (consultado jun. 27, 2020).
- [74] G. Amato, F. Carrara, F. Falchi, C. Gennaro, y C. Vairo, “Car parking occupancy detection using smart camera networks and Deep Learning”, en *Proceedings - IEEE Symposium on Computers and Communications*, ago. 2016, vol. 2016-August, pp. 1212–1217, doi: 10.1109/ISCC.2016.7543901.
- [75] G. Amato, F. Carrara, F. Falchi, C. Gennaro, C. Meghini, y C. Vairo, “CNR Parking Dataset - Dataset for visual occupancy detection of parking lots”, 2016. <http://cnrpark.it/> (consultado jul. 08, 2020).
- [76] J. Redmon y A. Farhadi, “YOLOv3: An Incremental Improvement”, *arXiv*, 2018. <https://pjreddie.com/darknet/yolo/> (consultado jul. 10, 2020).
- [77] E. Seminara y E. Auberix, “Intelligent Cloud Conference”, may 29, 2018. <https://www.slideshare.net/estelleinomniaparatus/icc-paas-vs-faas-v41> (consultado jul. 21, 2020).
- [78] R. Akiwatkar, “AWS Lambda vs Azure Functions vs Google Cloud Functions: Comparing Serverless Providers”, 2019. <https://www.simform.com/aws-lambda-vs-azure-functions-vs-google-functions/> (consultado jul. 21, 2020).
- [79] J. Barr, “AWS Named as a Leader in Gartner’s Infrastructure as a Service (IaaS) Magic Quadrant for the 9th Consecutive Year”, *Launch, News*, jul. 18, 2019. <https://aws.amazon.com/es/blogs/aws/aws-named-as-a-leader-in-gartners-infrastructure-as-a-service-iaas-magic-quadrant-for-the-9th-consecutiveyear/> (consultado jul. 21, 2020).
- [80] AWS, “Forrester New Wave: Public Cloud Enterprise Container Platforms, Q3 2019”, 2019. https://pages.awscloud.com/forrester-new-wave-public-cloud-report.html?trk=ar_card (consultado jul. 21, 2020).
- [81] AWS, “Elección entre HTTP API y REST API”, 2020. https://docs.aws.amazon.com/es_es/apigateway/latest/developerguide/http-api-vs-rest.html (consultado jul. 22, 2020).

Anexo I

TEORÍA CLASIFICACIÓN DE IMÁGENES

En el panorama de la inteligencia artificial enfocada en el reconocimiento visual de objetos, muchas tareas de la visión computacional como la detección y segmentación de objetos se basa en la clasificación de imágenes, esta se define como la tarea de predecir o corresponder a una imagen una etiqueta de un conjunto de categorías predeterminado con la mejor eficacia posible, es convertir los miles de números que representan los píxeles de una imagen a una sola etiqueta; sin embargo, presenta muchos desafíos al ser procesada por un algoritmo. [24]

- Variación en la perspectiva/orientación del objeto
- Escala
- Deformación
- Obstrucción por el entorno
- Condiciones de Iluminación
- Mezcla con el entorno
- Variedad de modelos de un objeto

El desarrollo de la clasificación de imágenes tiene un enfoque basado en datos, es decir, desarrollar un algoritmo que aprende de un conjunto de imágenes definido para la posterior predicción de nuevas entradas, el conjunto base de imágenes para el aprendizaje se denomina *training set*, mientras que su conjunto más las imágenes para la evaluación se denomina *dataset*. La etapa final del proceso consiste en evaluar el desempeño del algoritmo entrenado con un nuevo conjunto de imágenes y comparar los resultados de clasificación obtenidos [24]. Respecto al enfoque de clasificación, estos son:

- **Clasificador vecino más cercano:** Compara cada imagen de testeo con todo el conjunto de entrenamiento y se asigna la etiqueta de la imagen más cercana. La

comparación se determina como la sumatoria de cada distancia entre los píxeles de la imagen prueba y la imagen de entrenamiento. [24]

- **Clasificador K-vecinos más cercanos:** Similar al anterior, se encuentran las k imágenes más cercanas a la imagen prueba y se realiza un voto entre estos para seleccionar su etiqueta. En esta clasificación el valor de K , el método para medir la distancia y otras variables son denominados hiper-parámetros, para su ajuste se extrae de la data de entrenamiento una pequeña porción denominada conjunto de validación y con ella se evalúa el desempeño de los hiper-parámetros y, finalmente, del método de clasificación. [24]
- **Clasificador Lineal:** Consta de una función de puntaje, convierte los valores de píxeles de una imagen en puntajes asignados para cada clase. Para ello, la imagen, representada como un vector, se multiplica con una matriz de pesos y luego es sumada con un vector de sesgo. Adicionalmente se tiene una función de pérdida cuyo propósito es medir el grado de error en la predicción y corregirla, de manera que posteriormente se produzca una pérdida menor; se considera una pérdida cuando la diferencia de puntajes entre la clase correcta e incorrecta es menor a un margen dado. La función cuenta con un factor de regularización para fijar valores únicos en los pesos de cada clase y evitar ambigüedades. El objetivo es optimizar los valores del vector de pesos de manera que minimicen la función de pérdida, para ello se escoge un valor de pesos aleatorio y, siguiendo la gradiente más pronunciada de la función de pérdida, se avanza en la dirección que minimice las pérdidas. [25], [26]

Redes Neuronales: Inspirada principalmente en el modelamiento de los sistemas biológicos, las redes neuronales tienen el objetivo de lograr un desempeño eficaz en el aprendizaje automático. Los impulsos, enviados desde los axones de otras neuronas, son potenciados por la fuerza sináptica e ingresan por medio de las dendritas a una neurona en particular para ser

sumadas en el núcleo, si la suma supera un valor umbral (función de activación no lineal) entonces se genera un pico que es llevado por el axón hacia las demás neuronas. En la figura 1.1 se muestra un ejemplo de la arquitectura de una red neuronal, el modelo consta de distintas capas con conexión total entre ellas, las neuronas de distintas capas están totalmente conectadas mientras que las de una misma capa no. La arquitectura establece que las capas extremas son las de entrada y salida mientras que las intermedias se denominan capas ocultas. A diferencia de las demás, la capa de salida no cuenta con una función de activación ya que su propósito es representar los datos obtenidos como puntajes o valores. Por otro lado, la arquitectura mostrada cuenta con 41 parámetros (32 pesos asignados y 9 sesgos). [27]

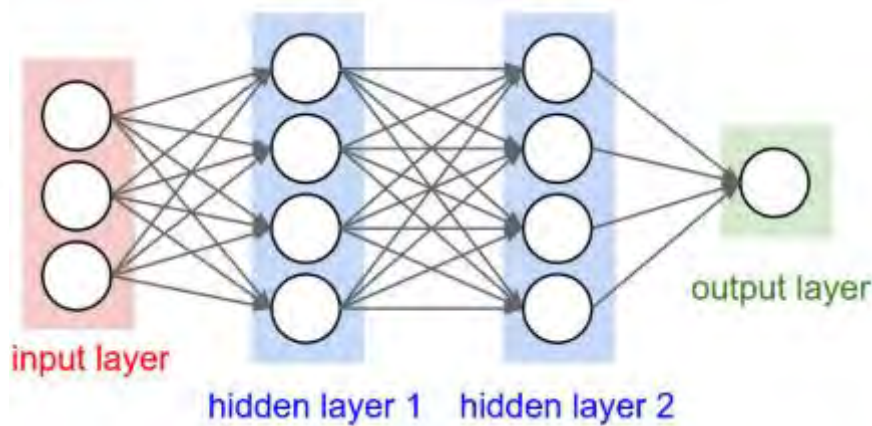


Figura 1.1 Arquitectura de una red neuronal de 3 capas con 3 entradas [27]

En el diseño de una red neuronal se debe tomar en cuenta la cantidad de capas ocultas y neuronas por cada capa. En términos generales, una red con pocas neuronas tiene en sus funciones de pérdida pocos valores mínimos muy fáciles de converger a una gran pérdida, mientras que en las redes grandes existen mayor cantidad de mínimos, pero más estables con menor variación en la pérdida final. Es recomendable el uso de una red neuronal de gran capacidad además de una técnica que permita regular el sobreajuste del modelo a sus datos de entrenamiento a fin de evitar la segmentación de pruebas y obtener un modelo más generalizado. [27]

Es importante el preprocesamiento de las entradas en la red neuronal de manera que estos se encuentren centrados en cero y normalizados, de igual manera inicializar los pesos de la red siguiendo una distribución gaussiana. [28]

Redes Convolucionales: Una red convolucional (*ConvNet*) se encarga del procesamiento de entradas con pesos y sesgos asignados a través de una serie de capas ocultas siguiendo una distribución no lineal en cada una a fin de obtener los puntajes asignados a cada clase que cumplan con los valores mínimos de la función de pérdida. La principal diferencia radica en que este tipo de redes presupone que las entradas son imágenes, por lo que hay una reducción en la cantidad de parámetros en la red y mayor eficiencia en el uso de funciones. La red convolucional surge debido a que las redes neuronales convencionales no tienen buen desempeño con las imágenes de mayor tamaño, generan una gran cantidad de entradas que aumentan considerablemente la cantidad de parámetros produciendo el sobreajuste. [29]

La figura 1.2 muestra el cambio en la arquitectura de una ConvNet respecto de una red neuronal tradicional. En este caso una imagen de entrada se convierte en un volumen de 3 dimensiones (alto, ancho y profundidad) para su procesamiento en las capas ocultas, estas generan un volumen distinto en cada salida por medio de una función diferenciable y posibles parámetros asignados. La última capa (salida) es de dimensiones $1 \times 1 \times N$ siendo N un vector con los puntajes asignados para cada clase. En esta arquitectura las neuronas de capas adyacentes no están totalmente conectadas a excepción de la última etapa. [29]

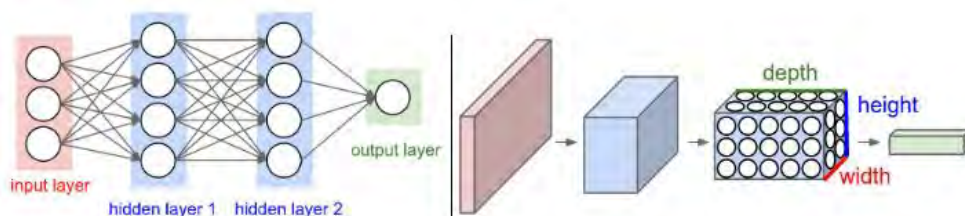


Figura 1.2 Transición a una red neuronal convolucional [29]

La arquitectura simple de una *Convnet* consta de las siguientes capas:

- **Capa convolucional:** Capa más importante de la red neuronal ya que en ella recae el mayor procesamiento computacional. Genera a la salida un volumen igual al producto punto de los pesos con una porción del volumen de entradas. Se modifica la profundidad del volumen de salida acorde a la cantidad de filtros. La capa convolucional consta de un conjunto de filtros con dimensiones pequeñas en el alto y ancho respecto al volumen de entrada, pero con su misma profundidad. Cada filtro observa solo una pequeña región de este volumen y es deslizado en este espacio para generar un mapa de activación a la salida, para esta tarea se apoya de parámetros como el tamaño del filtro, paso de deslizamiento y relleno de ceros. Finalmente, cada mapa producido se apila respecto a la profundidad y se obtiene un volumen de salida. Es importante especificar que las conexiones de cada neurona con la capa anterior son locales respecto al alto y ancho (tamaño del filtro), pero son completas respecto a la profundidad [29]. Un mejor detalle de esta capa se muestra en la figura 1.3, acá se tiene una imagen con un volumen de entrada $227 \times 227 \times 3$, a este volumen se le aplica un filtro de $11 \times 11 \times 3$ para generar un volumen de salida de $55 \times 55 \times 96$, cada sección a la salida consta de 55×55 neuronas, todas ellas compartiendo un mismo conjunto de pesos asignados ($11 \times 11 \times 3$ pesos). Por otro lado, múltiples neuronas a lo largo de la profundidad del volumen de salida tienen conexión local con solo una región espacial de la entrada, pero muestran conexión total en la profundidad. [29]

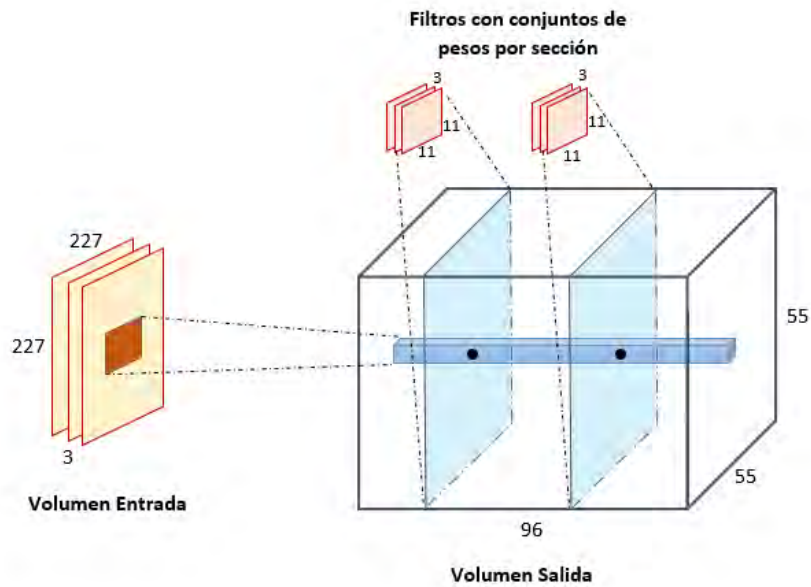


Figura 1.3: Ejemplo componentes de la capa convolucional [Elaboración propia]

- **Capa de agrupación:** Su función es reducir el tamaño espacial del volumen entrante aplicando filtros muy pequeños que se desplazan de forma descendente, la reducción es independiente por cada sección de profundidad. El proceso consiste en dividir cada región en filtros muy pequeños y tomar los máximos valores contenidos en cada uno, así se reduce el ancho y alto del volumen, pero se mantiene la profundidad. [29]
- **Capa totalmente conectada:** Última capa encargada de calcular los puntajes de cada clase, por lo que las neuronas de esta capa y la anterior se encuentran totalmente conectadas. Tiene una funcionalidad similar a la capa convolucional, con la diferencia que sus neuronas tienen visibilidad completa de toda la región del volumen de entrada y los pesos no se encuentran agrupados por secciones. [29]

Dispositivos SBC - Raspberry: a continuación, se mostrará la tabla comparativa 1.1 de las especificaciones entre los modelos 3B+ y 4B.

Tabla 1.1 Comparativa del modelo Raspberry pi 3B+ y 4

	RASPBERRY PI 3B+ [38]	RASPBERRY PI 4 B [39]
Procesador	Broadcom BCM2837B0 Quad-Core A53 (ARMv8) 64-bit @ 1.4GHz	Broadcom BCM2711, Quad Core Cortex-A72 (ARM v8) 64-bit SoC @ 1.5GHz
Memoria	1GB LPDDR2 SDRAM	1GB, 2GB o 4GB LPDDR4-3200 SDRAM
GPU	VideoCore IV @ 250-400MHz	VideoCore VI @ 500Mhz
Conectividad	<ul style="list-style-type: none"> Gigabit Ethernet 2.4GHz and 5GHz IEEE 802.11.b/g/n/ac Wireless LAN, Bluetooth 4.2, BLE 	<ul style="list-style-type: none"> Gigabit Ethernet 2.4 GHz and 5.0 GHz IEEE 802.11ac Wireless, Bluetooth 5.0, BLE
Interfaz Cámara	Sí dispone	Sí dispone
Potencia de entrada	5V/2.5A DC	5.1V / 3.0A DC
Dimensiones	82x56x19,5 mm	88 x 58 x 19,5mm

Ambos modelos ya están equipados para la transmisión de datos a través de Wifi, además es posible conectarse a la red SigFox a través de un módulo adicional (Modelo Wisol WSSFM10R2 con comunicación UART, es necesario renovar la licencia anualmente).



Figura 1.4 Módulo Sigfox para Raspberry pi [40]

A continuación, se describen algunas características del módulo de captura de imágenes.

- **Modelo:** Raspberry Camera module v2
- **Resolución:** 8 Megapixels
- **Campo de visión vertical:** 48.8 grados

- **Campo de visión horizontal:** 62.2 grados
- **Interfaz:** CSI
- **Dimensiones:** 25 x 23 x 9 mm



Figura 1.5 Raspberry camera module v2 [41]

Dispositivos SBC - Nvidia Jetson: se muestra una tabla comparativa 1.2 de la serie Nvidia.

Tabla 1.2 Comparativa de la serie Nvidia Jetson [43]

	JETSON NANO	JETSON TX 4GB	JETSON XAVIER NX	JETSON AGX XAVIER
GPU	128-core NVIDIA Maxwell™	256-core NVIDIA Pascal™	384-core NVIDIA Volta™ GPU with 48 Tensor Cores	512-core NVIDIA Volta™ GPU with 64 Tensor Cores
CPU	Quad-core ARM® Cortex®-A57 MPCore processor	Dual-core Denver 1.5 64-bit CPU + Quad-core Arm® Cortex®-A57 MPCore processor	6-core NVIDIA Carmel ARM®v8.2 64-bit CPU 6MB L2 + 4MB L3	8-core NVIDIA Carmel Arm®v8.2 64-bit CPU 8MB L2 + 4MB L3
Memoria	4 GB 64-bit LPDDR4 25,6GB/s	4 GB 128-bit LPDDR4 51,2GB/s	8 GB 128-bit LPDDR4x 51,2GB/s	32 GB 256-bit LPDDR4x 136,5GB/s
Conectividad	<ul style="list-style-type: none"> • 10/100/1000 BASE-T Ethernet. • Requiere tarjeta adicional para Wifi. 			
Potencia	10W	20W	15W	30W
Interfaz Cámara	Hasta 4 cámaras	Hasta 6 cámaras	Hasta 6 cámaras	Hasta 6 cámaras
Dimensiones	69,6 x 45 mm	87 x 50 mm	69,6 x 45 mm	100 x 87 mm

A continuación, se presente un modelo de cámara compatible para NVIDIA Jetson nano.

- **Modelo:** Leopard Imaging LI-IMX219-MIPI-FF-NANO-H136
- **Resolución:** 8 Megapíxeles
- **Campo de visión:** 90, 136 y 145 grados
- **Interfaz:** CSI
- **Dimensiones:** 25 x 24 x 15 mm



Figura 1.6 Leopard Imaging Li-imx219-mipi-ff-nano-h136 [45]

Dispositivos SBC - Google Coral: enseguida, se presenta las características de este modelo de Google.

- **CPU:** NXP i.MX 8M SoC (quad Cortex-A53, Cortex-M4F)
- **GPU:** Integrated GC7000 Lite Graphics
- **Acelerador ML:** Google Edge TPU coprocessor
- **RAM:** 1 GB o 2GB LPDDR4
- **Conectividad:** Wi-Fi 2x2 MIMO (802.11b/g/n/ac 2.4/5GHz), Bluetooth 4.2 y Gigabit Ethernet port
- **Dimensions:** 88 x 60 x 24 mm



Figura 1.7 Módulo Google Coral [47]

Se mencionan las especificaciones de su módulo de captura de imágenes.

- **Resolución:** 5 Megapíxeles
- **Campo de visión vertical:** 84 grados
- **Campo de visión horizontal:** 87 grados
- **Interfaz:** CSI
- **Dimensiones:** 25 x 25 x 6.98 mm



Figura 1.8 Módulo cámara Google Coral [48]

Redes Transmisión - LoRa ALLIANCE: LoRa Alliance es una asociación de mayor crecimiento en el sector de tecnologías IoT desde su creación en el 2015, está compuesta por la tecnología de modulación LoRa para la transmisión de datos y el protocolo LoRaWAN para el control de acceso al medio y gestión de red. Tiene la visión de que el protocolo LoRaWAN sea reconocido como estándar global para la conectividad IoT y, a través de esto, puedan

implementarse proyectos IoT de grandes proporciones [61]. A continuación, se detallan características de sus soluciones:

- **Modulación:** LoRa es la modulación inalámbrica utilizada para crear enlaces de larga distancia. Tiene un esquema de modulación basado en el espectro ensanchado de un chirp (CSS por sus siglas en inglés), desarrollada por Semtech Corporation, por lo que permite la transmisión de mensajes de baja potencia y otorga robustez frente a la interferencia logrando incrementar el rango de alcance de la comunicación. [62]
- **Protocolo:** LoRaWAN define el protocolo de comunicación y la arquitectura del sistema para la red, por lo que tiene mayor influencia en el diseño del tiempo de vida de la batería en un nodo, la capacidad de la red, calidad de servicio, seguridad y variedad de aplicaciones servidas por la red. En la figura 1.9, se muestra un resumen de la pila de protocolos LoRa, en el tope de la capa física se encuentra LoRa encargado de la parte de radio y modulación, mientras que en el tope de la capa de control de acceso al medio (MAC) se encuentra LoRaWAN para definir la configuración de la transmisión; se muestran tres clases que definen el comportamiento de los dispositivos, ya que estos no sirven para un mismo tipo de aplicación y tienen distintos requerimientos, se explican a continuación. [62]
 - **Clase A:** Es el sistema más soportado en casi todos los dispositivos, tiene mayor ahorro de energía y es usado por aplicaciones que solo requieren comunicación desde el servidor poco después de que el dispositivo haya enviado su transmisión, el dispositivo entra en modo escucha durante dos intervalos de tiempo luego de enviar un dato.
 - **Clase B:** Aparte de incorporar el comportamiento en clase A, el dispositivo de tipo B también puede usar una serie de intervalos de recepción adicionales en

horarios programados. El *Gateway* envía una sincronización al dispositivo para estos intervalos programados.

- **Clase C:** Tienen su ventana de recepción casi continuamente abierta, excepto durante la transmisión de un mensaje.

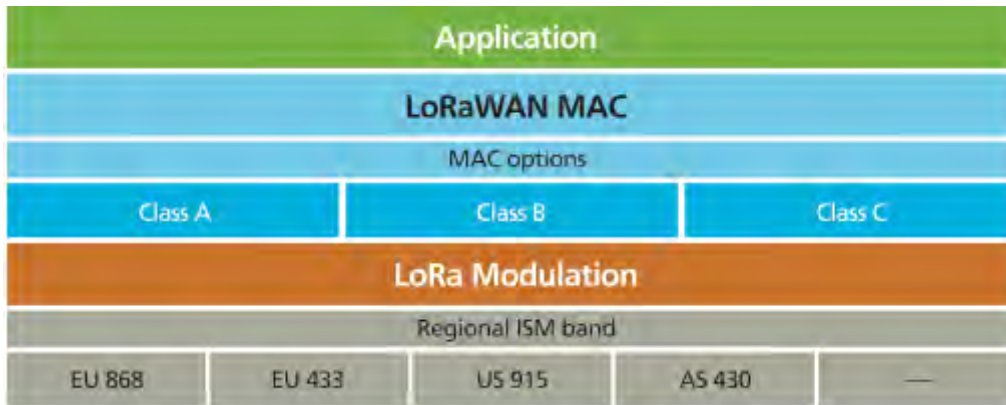


Figura 1.9 Pila de protocolos LoRa [62]

- **Arquitectura:** como se observa en la figura 1.10, su red consta de dispositivos finales (nodos), *gateways* (estaciones base o concentradores) y Servidores interconectados con la topología estrella de estrellas. La conexión entre los dispositivos y *gateways* se realiza inalámbricamente usando la banda ISM con un solo salto. El mensaje enviado por un dispositivo puede ser recibido por distintos *gateways*, para soportar la topología estos poseen gran capacidad de recepción, usan un transceptor multimodal multicanal para que se puedan recibir mensajes simultáneos en múltiples canales y asignan diversas tasas de transmisión a los nodos [62]. Finalmente, los mensajes son retransmitidos al servidor de red en la nube a través de una red *Backhaul* (Ethernet, celular, satélite o WIFI) para el filtrado de mensajes redundantes y su procesamiento.

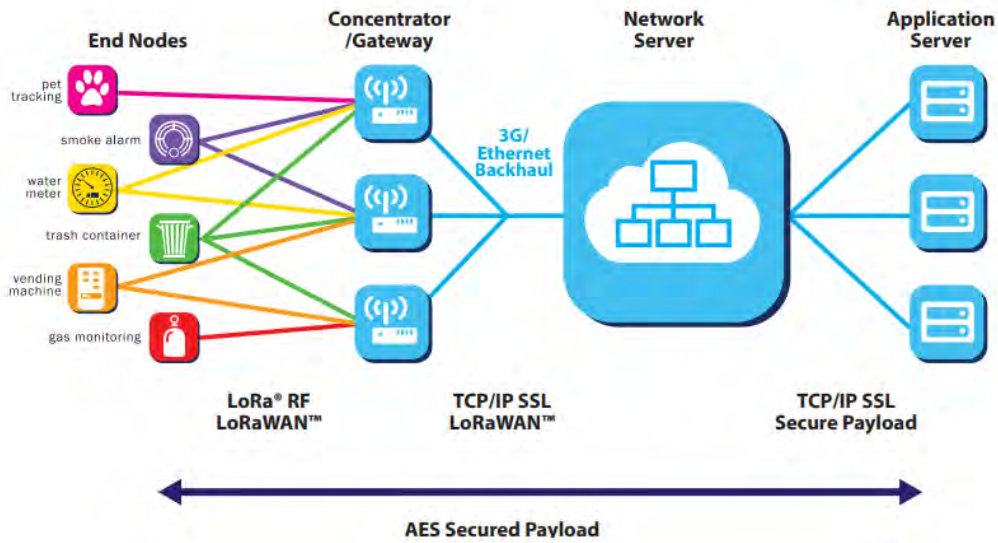


Figura 1.10 Arquitectura de red LoRa [62]

- **Seguridad:** LoRaWAN implementa dos niveles de seguridad, enfocada en red y en la aplicación. La seguridad en red asegura la autenticación del dispositivo mediante el intercambio de claves con cifrado AES de extremo a extremo utilizando un identificador IEEE EUI64 mientras que la seguridad en la aplicación evita que el operador de la red *Backhaul* tenga acceso a la información que se transmite. [62]

Anexo II

- YOLO

Tabla 2.1 Parámetros de detección obtenidos YOLO

Fecha	TP	FP	TN	FN	P	R	A	Tiempo	% CPU usado	Memoria usada (MB)
2015-11-12_0717	2	0	0	0	1,00	1,00	1,00	2,79	89,00	682,60
2015-11-12_0747	2	0	0	0	1,00	1,00	1,00	2,85	88,10	682,60
2015-11-12_0817	4	0	0	0	1,00	1,00	1,00	2,78	88,90	682,60
2015-11-12_0847	5	0	0	0	1,00	1,00	1,00	2,76	88,60	682,60
2015-11-12_0917	8	0	0	0	1,00	1,00	1,00	2,78	90,10	682,60
2015-11-12_0947	23	0	0	1	1,00	0,96	0,96	2,79	89,20	682,60
2015-11-12_1017	39	0	0	1	1,00	0,98	0,98	2,80	89,70	682,60
2015-11-12_1047	39	0	0	1	1,00	0,98	0,98	2,78	89,10	682,60
2015-11-12_1117	40	0	0	1	1,00	0,98	0,98	2,75	87,80	682,60
2015-11-12_1147	40	0	0	1	1,00	0,98	0,98	2,78	89,50	682,60
2015-11-12_1217	40	0	0	1	1,00	0,98	0,98	2,74	89,20	682,60
2015-11-12_1247	40	0	0	1	1,00	0,98	0,98	2,87	89,40	682,60
2015-11-12_1317	35	0	0	1	1,00	0,97	0,97	3,10	88,20	682,60
2015-11-12_1347	35	0	0	1	1,00	0,97	0,97	2,76	90,70	682,60
2015-11-12_1417	35	0	0	1	1,00	0,97	0,97	2,78	89,50	682,60
2015-11-12_1447	36	0	0	1	1,00	0,97	0,97	2,78	90,30	682,60
2015-11-12_1517	37	0	0	0	1,00	1,00	1,00	2,79	89,20	682,60
2015-11-12_1547	35	0	0	1	1,00	0,97	0,97	2,78	90,10	682,60
2015-11-12_1617	34	0	0	0	1,00	1,00	1,00	2,81	90,40	682,60
2015-11-12_1647	30	0	0	1	1,00	0,97	0,97	2,76	88,10	682,60
2015-11-12_1717	26	0	0	2	1,00	0,93	0,93	2,78	89,10	682,60
2015-11-20_0718	1	0	0	0	1,00	1,00	1,00	3,25	89,30	676,07
2015-11-20_0748	3	0	0	0	1,00	1,00	1,00	2,90	90,10	676,07
2015-11-20_0818	3	0	0	0	1,00	1,00	1,00	3,14	86,60	675,67
2015-11-20_0848	4	0	0	0	1,00	1,00	1,00	3,24	89,40	676,07
2015-11-20_0918	10	0	0	0	1,00	1,00	1,00	3,54	91,60	676,07
2015-11-20_0948	19	0	0	0	1,00	1,00	1,00	2,93	89,50	681,86
2015-11-20_1018	30	0	0	0	1,00	1,00	1,00	2,81	88,20	676,07
2015-11-20_1048	39	0	0	2	1,00	0,95	0,95	3,29	90,00	676,07
2015-11-20_1118	41	0	0	0	1,00	1,00	1,00	3,09	88,30	681,86
2015-11-20_1148	40	0	0	1	1,00	0,98	0,98	3,11	89,60	676,07
2015-11-20_1218	40	0	0	1	1,00	0,98	0,98	3,10	90,90	676,07
2015-11-20_1248	40	0	0	1	1,00	0,98	0,98	3,04	89,60	676,07
2015-11-20_1318	38	0	0	1	1,00	0,97	0,97	2,95	89,70	681,86
2015-11-20_1348	35	0	0	1	1,00	0,97	0,97	3,45	90,70	676,07
2015-11-20_1418	35	0	0	1	1,00	0,97	0,97	2,96	90,60	681,86
2015-11-20_1448	34	0	0	0	1,00	1,00	1,00	2,70	88,80	676,07

Fecha	TP	FP	TN	FN	P	R	A	Tiempo	% CPU usado	Memoria usada (MB)
2015-11-20_1518	32	0	0	1	1,00	0,97	0,97	2,99	89,20	681,86
2015-11-20_1548	32	0	0	0	1,00	1,00	1,00	2,95	88,80	681,86
2015-11-20_1618	30	0	0	0	1,00	1,00	1,00	2,68	90,40	675,81
2015-11-20_1648	27	0	0	0	1,00	1,00	1,00	2,95	90,80	676,07
2015-11-20_1718	15	1	0	11	0,94	0,58	0,56	3,07	87,90	676,07
2015-11-25_0719	2	0	0	0	1,00	1,00	1,00	3,03	88,90	681,86
2015-11-25_0749	3	0	0	0	1,00	1,00	1,00	2,99	88,80	681,86
2015-11-25_0819	3	0	0	0	1,00	1,00	1,00	2,95	89,50	682,08
2015-11-25_0849	5	0	0	0	1,00	1,00	1,00	3,08	90,00	681,86
2015-11-25_0919	9	0	0	0	1,00	1,00	1,00	3,02	88,40	681,86
2015-11-25_0949	20	0	0	0	1,00	1,00	1,00	3,00	89,80	681,86
2015-11-25_1019	25	0	0	1	1,00	0,96	0,96	3,03	89,00	681,86
2015-11-25_1049	40	0	0	0	1,00	1,00	1,00	2,97	90,90	681,86
2015-11-25_1119	41	0	0	0	1,00	1,00	1,00	3,04	90,20	681,86
2015-11-25_1149	41	0	0	0	1,00	1,00	1,00	3,38	88,30	681,86
2015-11-25_1219	40	0	0	0	1,00	1,00	1,00	3,02	88,90	681,86
2015-11-25_1249	38	0	0	0	1,00	1,00	1,00	2,98	88,90	681,86
2015-11-25_1319	35	0	0	0	1,00	1,00	1,00	3,29	88,10	682,08
2015-11-25_1349	36	0	0	0	1,00	1,00	1,00	3,02	90,40	681,86
2015-11-25_1419	37	0	0	0	1,00	1,00	1,00	3,04	88,70	682,08
2015-11-25_1449	37	0	0	0	1,00	1,00	1,00	2,96	87,70	681,86
2015-11-25_1519	37	0	0	0	1,00	1,00	1,00	3,06	89,50	682,08
2015-11-25_1549	38	0	0	0	1,00	1,00	1,00	3,01	88,80	681,86
2015-11-25_1619	36	0	0	0	1,00	1,00	1,00	2,97	90,50	682,08
2015-11-25_1649	32	0	0	0	1,00	1,00	1,00	2,93	89,00	681,86
2015-11-25_1719	16	0	0	13	1,00	0,55	0,55	2,98	90,10	681,86
2015-11-27_0749	1	0	0	0	1,00	1,00	1,00	2,86	87,70	682,60
2015-11-27_0819	1	0	0	0	1,00	1,00	1,00	2,78	90,80	682,60
2015-11-27_0849	2	0	0	0	1,00	1,00	1,00	2,80	88,50	682,60
2015-11-27_0919	7	0	0	1	1,00	1,00	1,00	2,77	90,10	682,60
2015-11-27_0949	21	0	0	1	1,00	1,00	1,00	2,92	89,80	682,60
2015-11-27_1019	32	0	0	2	1,00	0,94	0,94	2,89	90,00	682,60
2015-11-27_1049	38	0	0	2	1,00	0,95	0,95	2,83	90,00	682,60
2015-11-27_1119	39	0	0	1	1,00	0,98	0,98	2,78	89,70	682,60
2015-11-27_1149	40	0	0	2	1,00	0,95	0,95	2,78	88,80	682,60
2015-11-27_1219	41	0	0	1	1,00	0,98	0,98	2,79	89,40	682,60
2015-11-27_1249	40	0	0	1	1,00	0,98	0,98	2,78	88,10	682,60
2015-11-27_1319	37	0	0	1	1,00	0,97	0,97	2,79	88,90	682,60
2015-11-27_1349	35	0	0	1	1,00	0,97	0,97	2,81	88,30	682,60
2015-11-27_1419	36	0	0	1	1,00	0,97	0,97	2,77	88,90	682,60
2015-11-27_1449	35	0	0	1	1,00	0,97	0,97	2,79	91,00	682,60
2015-11-27_1519	33	0	0	3	1,00	0,92	0,92	2,89	90,10	682,60
2015-11-27_1549	31	0	0	2	1,00	0,94	0,94	3,13	87,90	682,60

Fecha	TP	FP	TN	FN	P	R	A	Tiempo	% CPU usado	Memoria usada (MB)
2015-11-27_1619	30	0	0	0	1,00	1,00	1,00	2,81	88,00	682,60
2015-11-27_1649	29	0	0	0	1,00	1,00	1,00	3,18	89,60	682,60
2015-11-27_1719	23	0	0	4	1,00	0,85	0,85	2,77	88,90	682,60
2015-12-03_0720	2	0	0	0	1,00	1,00	1,00	3,01	89,90	682,08
2015-12-03_0750	4	0	0	0	1,00	1,00	1,00	2,97	89,00	682,08
2015-12-03_0820	5	0	0	0	1,00	1,00	1,00	2,99	88,80	682,08
2015-12-03_0850	6	0	0	0	1,00	1,00	1,00	2,97	87,80	682,08
2015-12-03_0920	11	0	0	0	1,00	1,00	1,00	3,06	90,80	682,08
2015-12-03_0950	18	0	0	0	1,00	1,00	1,00	3,04	89,60	682,08
2015-12-03_1020	32	0	0	0	1,00	1,00	1,00	3,03	88,40	682,08
2015-12-03_1050	40	0	0	1	1,00	0,98	0,98	2,99	91,00	682,08
2015-12-03_1120	40	0	0	1	1,00	0,98	0,98	3,00	90,50	682,08
2015-12-03_1150	40	0	0	1	1,00	0,98	0,98	3,00	90,20	682,08
2015-12-03_1220	40	0	0	1	1,00	0,98	0,98	2,99	88,70	682,08
2015-12-03_1250	40	0	0	1	1,00	0,98	0,98	2,97	89,20	682,08
2015-12-03_1320	39	0	0	0	1,00	1,00	1,00	2,96	89,30	682,08
2015-12-03_1350	39	0	0	0	1,00	1,00	1,00	2,99	89,20	682,08
2015-12-03_1420	38	0	0	0	1,00	1,00	1,00	2,96	89,00	682,08
2015-12-03_1450	37	0	0	0	1,00	1,00	1,00	3,00	89,80	682,08
2015-12-03_1520	37	0	0	0	1,00	1,00	1,00	2,97	89,50	682,08
2015-12-03_1550	33	0	0	0	1,00	1,00	1,00	3,37	89,80	682,08
2015-12-03_1620	34	0	0	0	1,00	1,00	1,00	3,00	89,30	682,08
2015-12-03_1650	32	0	0	0	1,00	1,00	1,00	2,98	88,90	682,08
2015-12-03_1720	13	0	0	18	1,00	0,42	0,42	3,01	89,50	682,08
2015-12-10_0720	3	0	0	0	1,00	1,00	1,00	2,78	89,80	682,86
2015-12-10_0751	3	0	0	0	1,00	1,00	1,00	2,81	88,80	682,86
2015-12-10_0821	7	0	0	0	1,00	1,00	1,00	2,84	88,90	682,86
2015-12-10_0851	8	0	0	0	1,00	1,00	1,00	2,79	89,60	682,86
2015-12-10_0921	14	0	0	1	1,00	0,93	0,93	3,08	90,50	682,86
2015-12-10_0951	24	0	0	2	1,00	0,92	0,92	2,78	89,20	682,86
2015-12-10_1021	35	0	0	5	1,00	0,88	0,88	2,72	87,80	682,86
2015-12-10_1051	37	0	0	4	1,00	0,90	0,90	2,82	89,30	682,86
2015-12-10_1121	36	0	0	4	1,00	0,90	0,90	2,79	89,80	682,86
2015-12-10_1151	35	1	0	5	0,97	0,88	0,85	2,82	89,20	682,86
2015-12-10_1221	35	1	0	5	0,97	0,88	0,85	2,95	90,00	682,86
2015-12-10_1251	34	0	0	4	1,00	0,89	0,89	2,78	90,00	682,86
2015-12-10_1321	35	0	0	2	1,00	0,95	0,95	2,81	88,80	682,86
2015-12-10_1351	38	0	0	1	1,00	0,97	0,97	2,80	88,70	682,86
2015-12-10_1421	39	0	0	1	1,00	0,98	0,98	2,82	89,50	682,86
2015-12-10_1451	40	0	0	1	1,00	0,98	0,98	2,77	89,20	682,86
2015-12-10_1521	36	0	0	3	1,00	0,92	0,92	2,78	89,70	682,60
2015-12-10_1551	34	0	0	2	1,00	0,94	0,94	2,79	91,10	682,86
2015-12-10_1621	31	0	0	0	1,00	1,00	1,00	2,76	91,10	682,86

Fecha	TP	FP	TN	FN	P	R	A	Tiempo	% CPU usado	Memoria usada (MB)
2015-12-10_1651	29	0	0	0	1,00	1,00	1,00	2,83	87,30	682,86
2015-12-10_1721	8	1	0	20	0,89	0,29	0,28	2,77	89,60	682,86
2015-12-17_0724	4	0	0	0	1,00	1,00	1,00	2,81	88,80	682,86
2015-12-17_0754	4	0	0	0	1,00	1,00	1,00	2,79	90,00	682,86
2015-12-17_0824	6	0	0	0	1,00	1,00	1,00	2,81	89,40	682,86
2015-12-17_0854	7	0	0	0	1,00	1,00	1,00	2,82	89,20	682,86
2015-12-17_0924	12	0	0	0	1,00	1,00	1,00	2,81	90,50	682,86
2015-12-17_0954	17	0	0	0	1,00	1,00	1,00	2,81	88,70	682,86
2015-12-17_1024	36	0	0	3	1,00	0,92	0,92	2,79	89,50	682,86
2015-12-17_1054	35	0	0	6	1,00	0,85	0,85	3,25	89,00	682,86
2015-12-17_1124	38	0	0	3	1,00	0,93	0,93	2,82	91,00	682,86
2015-12-17_1154	40	0	0	2	1,00	0,95	0,95	2,81	90,00	682,86
2015-12-17_1224	39	0	0	2	1,00	0,95	0,95	2,83	89,00	682,86
2015-12-17_1254	39	0	0	1	1,00	0,98	0,98	2,80	89,30	682,86
2015-12-17_1324	37	0	0	1	1,00	0,97	0,97	2,81	91,00	682,86
2015-12-17_1354	39	0	0	0	1,00	1,00	1,00	2,81	91,20	682,86
2015-12-17_1424	38	0	0	1	1,00	0,97	0,97	2,80	89,30	682,86
2015-12-17_1454	41	0	0	0	1,00	1,00	1,00	2,84	89,40	682,86
2015-12-17_1524	38	0	0	1	1,00	0,97	0,97	2,86	89,80	682,86
2015-12-17_1554	36	0	0	0	1,00	1,00	1,00	2,86	90,70	682,86
2015-12-17_1624	34	0	0	0	1,00	1,00	1,00	2,85	90,50	682,86
2015-12-17_1654	33	0	0	0	1,00	1,00	1,00	2,79	88,10	682,86
2015-12-17_1724	7	0	0	23	1,00	0,23	0,23	2,82	91,20	682,86
2015-12-18_0724	2	0	0	0	1,00	1,00	1,00	3,10	90,80	682,29
2015-12-18_0754	3	0	0	0	1,00	1,00	1,00	3,04	88,50	682,08
2015-12-18_0824	3	0	0	0	1,00	1,00	1,00	3,01	90,10	682,34
2015-12-18_0854	5	0	0	0	1,00	1,00	1,00	3,06	88,50	682,34
2015-12-18_0924	7	0	0	0	1,00	1,00	1,00	3,08	90,00	682,34
2015-12-18_0954	24	0	0	0	1,00	1,00	1,00	3,34	89,40	682,34
2015-12-18_1024	41	0	0	0	1,00	1,00	1,00	3,05	88,70	682,08
2015-12-18_1054	42	0	0	0	1,00	1,00	1,00	3,02	88,40	682,08
2015-12-18_1124	42	0	0	0	1,00	1,00	1,00	2,96	88,40	682,29
2015-12-18_1154	42	0	0	0	1,00	1,00	1,00	3,37	90,00	682,29
2015-12-18_1224	42	0	0	0	1,00	1,00	1,00	3,04	90,40	682,34
2015-12-18_1254	41	0	0	0	1,00	1,00	1,00	3,04	90,30	682,08
2015-12-18_1324	36	0	0	0	1,00	1,00	1,00	3,04	89,40	682,34
2015-12-18_1354	33	0	0	0	1,00	1,00	1,00	3,02	89,60	682,08
2015-12-18_1424	37	0	0	0	1,00	1,00	1,00	3,09	90,90	682,29
2015-12-18_1454	36	0	0	0	1,00	1,00	1,00	3,06	89,40	682,29
2015-12-18_1524	35	0	0	0	1,00	1,00	1,00	3,04	89,40	682,08
2015-12-18_1554	33	0	0	0	1,00	1,00	1,00	3,11	89,40	682,08
2015-12-18_1624	31	0	0	0	1,00	1,00	1,00	3,05	89,00	682,08
2015-12-18_1654	28	0	0	1	1,00	0,97	0,97	3,04	88,50	682,29

Fecha	TP	FP	TN	FN	P	R	A	Tiempo	% CPU usado	Memoria usada (MB)
2015-12-18_1724	4	0	0	21	1,00	0,16	0,16	2,99	89,10	682,29
2015-12-22_0754	1	0	0	0	1,00	1,00	1,00	3,28	88,50	682,34
2015-12-22_0824	3	0	0	0	1,00	1,00	1,00	2,81	89,30	682,34
2015-12-22_0854	6	0	0	0	1,00	1,00	1,00	3,04	90,50	682,34
2015-12-22_0924	7	0	0	0	1,00	1,00	1,00	3,04	91,00	682,34
2015-12-22_0954	12	0	0	0	1,00	1,00	1,00	3,01	89,50	682,34
2015-12-22_1024	22	0	0	0	1,00	1,00	1,00	3,05	89,00	682,34
2015-12-22_1054	23	0	0	1	1,00	0,96	0,96	3,02	89,30	682,34
2015-12-22_1124	37	0	0	0	1,00	1,00	1,00	3,17	88,10	682,34
2015-12-22_1154	39	0	0	0	1,00	1,00	1,00	3,32	90,50	682,34
2015-12-22_1224	37	0	0	0	1,00	1,00	1,00	3,43	88,00	682,34
2015-12-22_1254	36	0	0	0	1,00	1,00	1,00	3,02	89,70	682,34
2015-12-22_1324	35	0	0	0	1,00	1,00	1,00	3,54	90,30	682,34
2015-12-22_1354	33	0	0	1	1,00	0,97	0,97	3,44	91,40	682,34
2015-12-22_1424	32	0	0	0	1,00	1,00	1,00	2,86	89,20	682,34
2015-12-22_1454	33	0	0	0	1,00	1,00	1,00	3,06	87,30	682,34
2015-12-22_1524	33	0	0	0	1,00	1,00	1,00	3,04	90,00	682,34
2015-12-22_1554	30	0	0	0	1,00	1,00	1,00	2,94	91,70	682,34
2015-12-22_1624	29	0	0	0	1,00	1,00	1,00	3,14	87,80	682,34
2015-12-22_1654	24	0	0	0	1,00	1,00	1,00	3,13	86,20	682,34
2015-12-22_1724	6	0	0	17	1,00	0,26	0,26	3,39	88,60	682,34
2016-01-08_0758	3	0	0	0	1,00	1,00	1,00	2,79	88,90	682,34
2016-01-08_0828	5	0	0	0	1,00	1,00	1,00	2,79	90,50	682,34
2016-01-08_0858	8	0	0	0	1,00	1,00	1,00	2,80	88,30	682,34
2016-01-08_0928	12	0	0	0	1,00	1,00	1,00	2,77	89,30	682,34
2016-01-08_0958	26	0	0	1	1,00	0,96	0,96	2,76	86,90	682,34
2016-01-08_1028	37	0	0	1	1,00	0,97	0,97	2,83	89,40	682,34
2016-01-08_1058	41	0	0	0	1,00	1,00	1,00	2,77	89,40	682,34
2016-01-08_1128	40	0	0	1	1,00	0,98	0,98	2,78	89,20	682,34
2016-01-08_1158	40	0	0	1	1,00	0,98	0,98	2,75	87,90	682,34
2016-01-08_1228	38	0	0	1	1,00	0,97	0,97	2,78	90,80	682,34
2016-01-08_1258	35	0	0	1	1,00	0,97	0,97	2,74	90,20	682,34
2016-01-08_1328	33	0	0	1	1,00	0,97	0,97	2,86	88,50	682,34
2016-01-08_1358	32	0	0	0	1,00	1,00	1,00	2,81	89,60	682,34
2016-01-08_1428	33	0	0	0	1,00	1,00	1,00	2,76	89,00	682,34
2016-01-08_1458	34	0	0	1	1,00	0,97	0,97	2,87	90,30	682,34
2016-01-08_1528	35	0	0	0	1,00	1,00	1,00	3,12	87,30	682,34
2016-01-08_1558	32	0	0	0	1,00	1,00	1,00	2,79	90,40	682,34
2016-01-08_1628	28	0	0	1	1,00	0,97	0,97	2,79	90,30	682,34
2016-01-08_1658	24	0	0	2	1,00	0,92	0,92	2,79	89,70	682,34
2016-01-08_1728	7	0	0	19	1,00	0,27	0,27	2,80	90,70	682,34
2016-01-18_0745	3	0	0	0	1,00	1,00	1,00	2,79	89,00	682,86
2016-01-18_0815	4	0	0	0	1,00	1,00	1,00	2,79	89,40	683,11

Fecha	TP	FP	TN	FN	P	R	A	Tiempo	% CPU usado	Memoria usada (MB)
2016-01-18_0845	10	0	0	0	1,00	1,00	1,00	2,80	89,20	683,11
2016-01-18_0915	13	0	0	1	1,00	0,93	0,93	2,86	91,00	683,11
2016-01-18_0945	21	0	0	2	1,00	0,91	0,91	2,90	90,20	682,86
2016-01-18_1015	39	0	0	2	1,00	0,95	0,95	2,82	89,30	683,11
2016-01-18_1045	39	0	0	2	1,00	0,95	0,95	2,79	89,70	683,11
2016-01-18_1115	39	0	0	2	1,00	0,95	0,95	2,80	89,40	683,11
2016-01-18_1145	40	0	0	1	1,00	0,98	0,98	2,82	88,80	683,11
2016-01-18_1215	40	0	0	0	1,00	1,00	1,00	2,80	88,90	683,11
2016-01-18_1245	40	0	0	0	1,00	1,00	1,00	2,82	89,60	683,11
2016-01-18_1315	33	0	0	1	1,00	0,97	0,97	3,02	88,70	683,11
2016-01-18_1345	31	0	0	0	1,00	1,00	1,00	2,79	89,50	683,11
2016-01-18_1415	33	0	0	0	1,00	1,00	1,00	2,81	91,00	682,86
2016-01-18_1445	35	0	0	0	1,00	1,00	1,00	2,80	89,90	683,11
2016-01-18_1515	36	0	0	0	1,00	1,00	1,00	2,80	89,70	683,11
2016-01-18_1545	35	0	0	0	1,00	1,00	1,00	2,79	91,60	683,11
2016-01-18_1615	36	0	0	0	1,00	1,00	1,00	2,82	89,10	683,11
2016-01-18_1645	32	0	0	0	1,00	1,00	1,00	2,83	89,60	683,11
2016-01-18_1715	29	0	0	0	1,00	1,00	1,00	2,79	89,30	683,11

- **Faster R-CNN**

Tabla 2.2 Parámetros de detección obtenidos Faster R-CNN

Fecha	TP	FP	TN	FN	P	R	A	Tiempo	% CPU usado	Memoria usada (MB)
2015-11-12_0717	2	0	0	0	1,00	1,00	1,00	6,67	48,90	1489,08
2015-11-12_0747	1	0	0	1	1,00	0,50	0,50	6,82	48,50	1488,90
2015-11-12_0817	3	0	0	1	1,00	0,75	0,75	6,65	49,10	1489,02
2015-11-12_0847	4	0	0	1	1,00	0,80	0,80	6,61	48,90	1488,73
2015-11-12_0917	7	0	0	1	1,00	0,88	0,88	6,73	50,20	1626,33
2015-11-12_0947	18	2	0	3	0,90	0,86	0,78	6,73	49,40	1502,64
2015-11-12_1017	36	1	0	2	0,97	0,95	0,92	6,76	50,40	1488,94
2015-11-12_1047	36	2	0	2	0,95	0,95	0,90	6,78	49,20	1488,95
2015-11-12_1117	41	0	0	0	1,00	1,00	1,00	6,79	49,00	1502,53
2015-11-12_1147	40	0	0	1	1,00	0,98	0,98	6,84	48,90	1502,53
2015-11-12_1217	41	0	0	0	1,00	1,00	1,00	6,80	49,90	1502,83
2015-11-12_1247	41	0	0	0	1,00	1,00	1,00	6,79	50,10	1488,84
2015-11-12_1317	36	0	0	0	1,00	1,00	1,00	6,70	49,20	1489,06
2015-11-12_1347	36	0	0	0	1,00	1,00	1,00	6,82	50,50	1488,69
2015-11-12_1417	36	0	0	0	1,00	1,00	1,00	6,69	48,70	1488,91
2015-11-12_1447	36	0	0	0	1,00	1,00	1,00	7,02	49,70	1626,28
2015-11-12_1517	36	0	0	1	1,00	0,97	0,97	6,61	48,80	1488,86
2015-11-12_1547	36	0	0	0	1,00	1,00	1,00	6,71	50,10	1488,93
2015-11-12_1617	34	0	0	0	1,00	1,00	1,00	6,68	50,40	1488,82

Fecha	TP	FP	TN	FN	P	R	A	Tiempo	% CPU usado	Memoria usada (MB)
2015-11-12_1647	31	0	0	0	1,00	1,00	1,00	6,79	49,30	1488,83
2015-11-12_1717	26	1	0	1	0,96	0,96	0,93	6,79	50,80	1489,00
2015-11-20_0718	1	0	0	0	1,00	1,00	1,00	7,27	61,50	1488,06
2015-11-20_0748	3	0	0	0	1,00	1,00	1,00	7,12	58,70	1488,10
2015-11-20_0818	3	0	0	0	1,00	1,00	1,00	6,23	45,30	1497,91
2015-11-20_0848	4	0	0	0	1,00	1,00	1,00	7,01	56,40	1488,06
2015-11-20_0918	11	0	0	0	1,00	1,00	1,00	7,16	58,90	1488,21
2015-11-20_0948	19	0	0	0	1,00	1,00	1,00	7,22	58,40	1488,20
2015-11-20_1018	30	0	0	0	1,00	1,00	1,00	7,06	57,80	1488,76
2015-11-20_1048	41	0	0	0	1,00	1,00	1,00	7,36	55,50	1488,09
2015-11-20_1118	41	0	0	0	1,00	1,00	1,00	7,22	57,50	1488,06
2015-11-20_1148	41	0	0	0	1,00	1,00	1,00	7,34	61,00	1488,11
2015-11-20_1218	41	0	0	0	1,00	1,00	1,00	7,85	66,80	1488,15
2015-11-20_1248	41	0	0	0	1,00	1,00	1,00	7,09	56,90	1491,51
2015-11-20_1318	39	0	0	0	1,00	1,00	1,00	7,18	57,90	1488,09
2015-11-20_1348	36	0	0	0	1,00	1,00	1,00	7,09	57,90	1488,01
2015-11-20_1418	36	0	0	0	1,00	1,00	1,00	7,18	58,80	1488,14
2015-11-20_1448	34	0	0	0	1,00	1,00	1,00	6,77	50,40	1489,10
2015-11-20_1518	33	0	0	0	1,00	1,00	1,00	7,14	57,80	1488,08
2015-11-20_1548	32	0	0	0	1,00	1,00	1,00	7,12	57,20	1488,20
2015-11-20_1618	30	0	0	0	1,00	1,00	1,00	8,31	40,80	1564,30
2015-11-20_1648	26	1	0	0	0,96	1,00	0,96	7,59	59,00	1570,59
2015-11-20_1718	15	3	0	9	0,83	0,63	0,56	7,11	57,20	1556,92
2015-11-25_0719	2	0	0	0	1,00	1,00	1,00	6,79	50,40	1488,16
2015-11-25_0749	3	0	0	0	1,00	1,00	1,00	7,19	58,20	1488,02
2015-11-25_0819	3	0	0	0	1,00	1,00	1,00	6,90	50,40	1488,22
2015-11-25_0849	4	1	0	0	0,80	1,00	0,80	6,87	52,00	1556,88
2015-11-25_0919	8	1	0	0	0,89	1,00	0,89	6,76	50,70	1625,55
2015-11-25_0949	19	1	0	0	0,95	1,00	0,95	6,81	50,50	1488,21
2015-11-25_1019	25	1	0	0	0,96	1,00	0,96	6,90	51,00	1488,20
2015-11-25_1049	39	0	0	1	1,00	0,98	0,98	7,13	56,90	1488,01
2015-11-25_1119	40	0	0	1	1,00	0,98	0,98	6,80	51,50	1501,85
2015-11-25_1149	40	1	0	0	0,98	1,00	0,98	7,30	57,30	1491,48
2015-11-25_1219	39	1	0	0	0,98	1,00	0,98	6,79	50,70	1488,10
2015-11-25_1249	37	1	0	0	0,97	1,00	0,97	6,80	50,10	1487,99
2015-11-25_1319	34	1	0	0	0,97	1,00	0,97	6,82	50,70	1501,97
2015-11-25_1349	35	1	0	0	0,97	1,00	0,97	6,73	51,80	1570,62
2015-11-25_1419	36	1	0	0	0,97	1,00	0,97	6,85	50,10	1488,09
2015-11-25_1449	36	1	0	0	0,97	1,00	0,97	6,80	50,20	1488,09
2015-11-25_1519	36	1	0	0	0,97	1,00	0,97	6,75	51,90	1488,00
2015-11-25_1549	37	1	0	0	0,97	1,00	0,97	6,72	50,60	1487,97
2015-11-25_1619	35	1	0	0	0,97	1,00	0,97	6,86	51,90	1488,20
2015-11-25_1649	31	1	0	0	0,97	1,00	0,97	7,27	58,10	1488,03

Fecha	TP	FP	TN	FN	P	R	A	Tiempo	% CPU usado	Memoria usada (MB)
2015-11-25_1719	12	1	0	16	0,92	0,43	0,41	7,17	55,10	1488,16
2015-11-27_0749	1	0	0	0	1,00	1,00	1,00	6,74	48,60	1502,71
2015-11-27_0819	1	0	0	0	1,00	1,00	1,00	6,60	50,40	1502,86
2015-11-27_0849	2	0	0	0	1,00	1,00	1,00	6,69	48,70	1489,21
2015-11-27_0919	7	0	0	0	1,00	1,00	1,00	6,61	48,10	1626,54
2015-11-27_0949	20	1	0	1	0,95	0,95	0,91	6,83	48,90	1502,90
2015-11-27_1019	31	1	0	2	0,97	0,94	0,91	6,87	50,20	1502,76
2015-11-27_1049	36	1	0	3	0,97	0,92	0,90	6,66	49,20	1489,09
2015-11-27_1119	39	0	0	1	1,00	0,98	0,98	6,69	50,30	1626,49
2015-11-27_1149	38	1	0	3	0,97	0,93	0,90	6,61	48,80	1488,96
2015-11-27_1219	41	0	0	1	1,00	0,98	0,98	6,65	49,50	1489,13
2015-11-27_1249	39	1	0	1	0,98	0,98	0,95	6,73	48,90	1488,97
2015-11-27_1319	37	0	0	1	1,00	0,97	0,97	6,74	48,90	1489,04
2015-11-27_1349	34	0	0	2	1,00	0,94	0,94	6,72	48,70	1489,01
2015-11-27_1419	36	0	0	1	1,00	0,97	0,97	6,67	49,80	1489,06
2015-11-27_1449	34	1	0	0	0,97	1,00	0,97	6,82	50,30	1489,22
2015-11-27_1519	34	0	0	2	1,00	0,94	0,94	6,74	50,40	1489,15
2015-11-27_1549	31	0	0	2	1,00	0,94	0,94	6,66	49,10	1489,19
2015-11-27_1619	30	0	0	0	1,00	1,00	1,00	6,65	48,70	1489,18
2015-11-27_1649	29	0	0	0	1,00	1,00	1,00	6,55	50,30	1488,90
2015-11-27_1719	18	1	0	8	0,95	0,69	0,67	6,80	49,20	1502,87
2015-12-03_0720	2	0	0	0	1,00	1,00	1,00	6,87	51,60	1488,27
2015-12-03_0750	4	0	0	0	1,00	1,00	1,00	6,85	50,70	1557,04
2015-12-03_0820	5	0	0	0	1,00	1,00	1,00	7,24	50,40	1488,28
2015-12-03_0850	6	0	0	0	1,00	1,00	1,00	6,94	51,30	1488,30
2015-12-03_0920	11	0	0	0	1,00	1,00	1,00	6,81	51,90	1488,27
2015-12-03_0950	18	0	0	0	1,00	1,00	1,00	6,87	51,70	1488,26
2015-12-03_1020	32	0	0	0	1,00	1,00	1,00	6,79	50,50	1488,29
2015-12-03_1050	41	0	0	0	1,00	1,00	1,00	6,83	51,80	1488,26
2015-12-03_1120	41	0	0	0	1,00	1,00	1,00	6,73	51,80	1488,23
2015-12-03_1150	41	0	0	0	1,00	1,00	1,00	6,80	51,70	1488,29
2015-12-03_1220	41	0	0	0	1,00	1,00	1,00	6,76	50,70	1556,82
2015-12-03_1250	41	0	0	0	1,00	1,00	1,00	6,74	50,60	1488,06
2015-12-03_1320	39	0	0	0	1,00	1,00	1,00	6,82	50,70	1488,19
2015-12-03_1350	39	0	0	0	1,00	1,00	1,00	6,82	49,90	1488,16
2015-12-03_1420	37	1	0	0	0,97	1,00	0,97	6,74	50,70	1488,16
2015-12-03_1450	37	0	0	0	1,00	1,00	1,00	6,91	52,00	1488,30
2015-12-03_1520	37	0	0	0	1,00	1,00	1,00	6,77	50,70	1488,22
2015-12-03_1550	33	0	0	0	1,00	1,00	1,00	6,82	51,70	1488,28
2015-12-03_1620	34	0	0	0	1,00	1,00	1,00	6,85	51,20	1488,18
2015-12-03_1650	32	0	0	0	1,00	1,00	1,00	6,78	50,60	1488,17
2015-12-03_1720	9	0	0	21	1,00	0,30	0,30	6,90	50,70	1557,01
2015-12-10_0720	3	0	0	0	1,00	1,00	1,00	6,62	48,80	1503,03

Fecha	TP	FP	TN	FN	P	R	A	Tiempo	% CPU usado	Memoria usada (MB)
2015-12-10_0751	2	1	0	0	0,67	1,00	0,67	6,70	49,00	1626,57
2015-12-10_0821	6	1	0	0	0,86	1,00	0,86	6,68	49,10	1626,54
2015-12-10_0851	7	1	0	0	0,88	1,00	0,88	6,69	48,70	1563,50
2015-12-10_0921	13	1	0	0	0,93	1,00	0,93	6,86	50,90	1489,40
2015-12-10_0951	22	1	0	3	0,96	0,88	0,85	6,83	48,50	1489,30
2015-12-10_1021	35	0	0	4	1,00	0,90	0,90	6,72	50,10	1626,72
2015-12-10_1051	36	2	0	3	0,95	0,92	0,88	6,78	48,70	1503,02
2015-12-10_1121	35	1	0	5	0,97	0,88	0,85	6,69	48,70	1626,58
2015-12-10_1151	34	1	0	6	0,97	0,85	0,83	6,74	49,10	1489,39
2015-12-10_1221	35	1	0	5	0,97	0,88	0,85	6,68	50,30	1489,11
2015-12-10_1251	37	0	0	1	1,00	0,97	0,97	6,65	49,00	1502,92
2015-12-10_1321	36	0	0	1	1,00	0,97	0,97	6,67	49,00	1626,54
2015-12-10_1351	38	1	0	0	0,97	1,00	0,97	6,63	48,20	1489,33
2015-12-10_1421	39	1	0	0	0,98	1,00	0,98	6,68	48,90	1489,34
2015-12-10_1451	39	1	0	1	0,98	0,98	0,95	6,72	48,70	1502,87
2015-12-10_1521	37	0	0	1	1,00	0,97	0,97	6,73	50,30	1489,05
2015-12-10_1551	35	0	0	1	1,00	0,97	0,97	6,71	50,30	1489,22
2015-12-10_1621	31	0	0	0	1,00	1,00	1,00	6,75	50,30	1489,21
2015-12-10_1651	29	0	0	0	1,00	1,00	1,00	6,78	49,00	1503,08
2015-12-10_1721	9	1	0	18	0,90	0,33	0,32	6,66	48,80	1626,64
2015-12-17_0724	3	1	0	0	0,75	1,00	0,75	6,66	49,10	1489,51
2015-12-17_0754	4	0	0	0	1,00	1,00	1,00	6,65	49,10	1503,18
2015-12-17_0824	3	2	0	0	0,60	1,00	0,60	6,74	48,80	1489,51
2015-12-17_0854	5	2	0	0	0,71	1,00	0,71	6,76	49,00	1489,29
2015-12-17_0924	10	2	0	0	0,83	1,00	0,83	6,72	50,10	1489,39
2015-12-17_0954	15	1	0	0	0,94	1,00	0,94	6,67	48,90	1489,31
2015-12-17_1024	32	2	0	4	0,94	0,89	0,84	6,80	48,80	1503,21
2015-12-17_1054	37	1	0	3	0,97	0,93	0,90	6,79	49,10	1489,55
2015-12-17_1124	37	1	0	2	0,97	0,95	0,93	6,74	50,30	1489,34
2015-12-17_1154	39	1	0	2	0,98	0,95	0,93	6,70	50,10	1489,29
2015-12-17_1224	40	0	0	1	1,00	0,98	0,98	6,66	49,10	1489,40
2015-12-17_1254	39	0	0	1	1,00	0,98	0,98	6,69	49,10	1489,59
2015-12-17_1324	36	0	0	2	1,00	0,95	0,95	6,73	50,40	1489,64
2015-12-17_1354	39	0	0	0	1,00	1,00	1,00	6,70	50,50	1489,45
2015-12-17_1424	39	0	0	0	1,00	1,00	1,00	6,70	49,10	1489,54
2015-12-17_1454	40	0	0	1	1,00	0,98	0,98	6,71	48,90	1489,35
2015-12-17_1524	38	0	0	1	1,00	0,97	0,97	6,72	48,80	1489,50
2015-12-17_1554	34	2	0	0	0,94	1,00	0,94	6,69	50,10	1489,56
2015-12-17_1624	33	1	0	0	0,97	1,00	0,97	6,60	50,30	1489,23
2015-12-17_1654	32	1	0	0	0,97	1,00	0,97	6,57	49,10	1489,45
2015-12-17_1724	5	2	0	23	0,71	0,18	0,17	6,80	50,40	1503,20
2015-12-18_0724	2	0	0	0	1,00	1,00	1,00	6,68	50,30	1502,22
2015-12-18_0754	3	0	0	0	1,00	1,00	1,00	6,71	50,40	1488,30

Fecha	TP	FP	TN	FN	P	R	A	Tiempo	% CPU usado	Memoria usada (MB)
2015-12-18_0824	3	0	0	0	1,00	1,00	1,00	6,78	50,10	1502,32
2015-12-18_0854	5	0	0	0	1,00	1,00	1,00	6,74	49,50	1488,54
2015-12-18_0924	7	0	0	0	1,00	1,00	1,00	6,87	50,20	1488,48
2015-12-18_0954	24	0	0	0	1,00	1,00	1,00	6,77	49,90	1488,42
2015-12-18_1024	41	0	0	0	1,00	1,00	1,00	6,63	49,50	1488,29
2015-12-18_1054	41	1	0	0	0,98	1,00	0,98	6,84	50,20	1488,30
2015-12-18_1124	41	1	0	0	0,98	1,00	0,98	6,60	49,30	1488,30
2015-12-18_1154	41	1	0	0	0,98	1,00	0,98	6,65	50,30	1488,35
2015-12-18_1224	41	1	0	0	0,98	1,00	0,98	6,72	50,50	1502,34
2015-12-18_1254	40	1	0	0	0,98	1,00	0,98	6,92	52,50	1488,43
2015-12-18_1324	36	0	0	0	1,00	1,00	1,00	6,81	50,00	1488,35
2015-12-18_1354	33	0	0	0	1,00	1,00	1,00	7,28	51,60	1491,73
2015-12-18_1424	37	0	0	0	1,00	1,00	1,00	6,75	51,30	1488,40
2015-12-18_1454	36	0	0	0	1,00	1,00	1,00	6,70	49,20	1625,88
2015-12-18_1524	35	0	0	0	1,00	1,00	1,00	6,90	52,70	1557,04
2015-12-18_1554	33	0	0	0	1,00	1,00	1,00	6,82	50,60	1488,29
2015-12-18_1624	31	0	0	0	1,00	1,00	1,00	6,79	51,10	1501,96
2015-12-18_1654	28	0	0	1	1,00	0,97	0,97	6,78	49,00	1488,46
2015-12-18_1724	5	0	0	21	1,00	0,19	0,19	6,71	48,90	1488,52
2015-12-22_0754	1	0	0	0	1,00	1,00	1,00	6,66	48,90	1488,49
2015-12-22_0824	3	0	0	0	1,00	1,00	1,00	6,67	48,20	1488,57
2015-12-22_0854	5	1	0	0	0,83	1,00	0,83	6,68	49,30	1488,66
2015-12-22_0924	6	1	0	0	0,86	1,00	0,86	6,72	50,30	1502,35
2015-12-22_0954	11	0	0	1	1,00	0,92	0,92	6,78	49,10	1502,31
2015-12-22_1024	22	0	0	0	1,00	1,00	1,00	6,67	49,50	1488,44
2015-12-22_1054	22	1	0	1	0,96	0,96	0,92	6,69	49,30	1488,36
2015-12-22_1124	35	1	0	1	0,97	0,97	0,95	6,79	49,50	1488,57
2015-12-22_1154	38	1	0	0	0,97	1,00	0,97	6,70	50,40	1502,34
2015-12-22_1224	34	1	0	2	0,97	0,94	0,92	6,71	49,40	1488,43
2015-12-22_1254	33	2	0	1	0,94	0,97	0,92	6,66	49,50	1488,58
2015-12-22_1324	33	1	0	0	0,97	1,00	0,97	6,76	50,30	1502,24
2015-12-22_1354	32	0	0	1	1,00	0,97	0,97	6,70	50,30	1488,69
2015-12-22_1424	31	0	0	0	1,00	1,00	1,00	6,83	49,20	1502,38
2015-12-22_1454	32	0	0	0	1,00	1,00	1,00	6,64	48,80	1488,52
2015-12-22_1524	32	0	0	0	1,00	1,00	1,00	6,58	49,80	1488,49
2015-12-22_1554	29	0	0	0	1,00	1,00	1,00	6,74	50,70	1626,05
2015-12-22_1624	28	0	0	0	1,00	1,00	1,00	6,79	48,90	1488,54
2015-12-22_1654	23	0	0	0	1,00	1,00	1,00	6,68	49,40	1488,64
2015-12-22_1724	5	1	0	15	0,83	0,25	0,24	6,74	49,00	1488,65
2016-01-08_0758	1	0	0	2	1,00	0,33	0,33	6,64	49,30	1488,75
2016-01-08_0828	3	0	0	1	1,00	0,75	0,75	6,61	48,70	1626,02
2016-01-08_0858	5	0	0	0	1,00	1,00	1,00	6,75	50,40	1565,04
2016-01-08_0928	6	1	0	1	0,86	0,86	0,75	6,78	48,90	1502,64

Fecha	TP	FP	TN	FN	P	R	A	Tiempo	% CPU usado	Memoria usada (MB)
2016-01-08_0958	10	1	0	1	0,91	0,91	0,83	6,90	49,30	1488,83
2016-01-08_1028	25	1	0	0	0,96	1,00	0,96	6,67	48,50	1502,38
2016-01-08_1058	34	1	0	2	0,97	0,94	0,92	6,73	49,20	1488,67
2016-01-08_1128	39	0	0	1	1,00	0,98	0,98	6,72	49,00	1488,64
2016-01-08_1158	40	1	0	0	0,98	1,00	0,98	6,71	49,30	1502,44
2016-01-08_1228	38	1	0	2	0,97	0,95	0,93	6,67	49,40	1488,73
2016-01-08_1258	38	0	0	1	1,00	0,97	0,97	6,79	50,30	1502,43
2016-01-08_1328	34	0	0	2	1,00	0,94	0,94	6,84	49,10	1502,45
2016-01-08_1358	32	0	0	2	1,00	0,94	0,94	6,75	49,70	1488,58
2016-01-08_1428	31	0	0	1	1,00	0,97	0,97	6,70	49,70	1488,77
2016-01-08_1458	32	1	0	0	0,97	1,00	0,97	6,70	49,10	1488,86
2016-01-08_1528	34	1	0	0	0,97	1,00	0,97	6,73	50,20	1502,27
2016-01-08_1558	32	1	0	2	0,97	0,94	0,91	6,75	49,40	1488,75
2016-01-08_1628	30	1	0	1	0,97	0,97	0,94	6,80	50,30	1502,50
2016-01-08_1658	27	1	0	1	0,96	0,96	0,93	6,77	50,30	1488,86
2016-01-08_1728	21	3	0	2	0,88	0,91	0,81	6,72	49,50	1502,56
2016-01-18_0745	7	0	0	19	1,00	0,27	0,27	6,77	50,20	1502,34
2016-01-18_0815	3	0	0	0	1,00	1,00	1,00	6,68	48,30	1503,22
2016-01-18_0845	4	0	0	0	1,00	1,00	1,00	6,58	49,50	1627,00
2016-01-18_0915	10	0	0	0	1,00	1,00	1,00	6,63	49,00	1479,34
2016-01-18_0945	14	0	0	0	1,00	1,00	1,00	6,83	49,90	1479,31
2016-01-18_1015	23	0	0	0	1,00	1,00	1,00	6,88	50,00	1489,53
2016-01-18_1045	40	0	0	0	1,00	1,00	1,00	6,74	48,90	1479,27
2016-01-18_1115	39	0	0	2	1,00	0,95	0,95	6,78	48,70	1492,95
2016-01-18_1145	40	0	0	1	1,00	0,98	0,98	6,78	49,50	1616,76
2016-01-18_1215	39	0	0	2	1,00	0,95	0,95	6,63	48,40	1479,41
2016-01-18_1245	40	0	0	0	1,00	1,00	1,00	6,69	48,90	1493,14
2016-01-18_1315	40	0	0	0	1,00	1,00	1,00	6,85	51,90	1479,18
2016-01-18_1345	34	0	0	0	1,00	1,00	1,00	7,18	53,60	1616,67
2016-01-18_1415	31	0	0	0	1,00	1,00	1,00	6,61	49,00	1479,35
2016-01-18_1445	32	0	0	1	1,00	0,97	0,97	6,74	50,30	1489,41
2016-01-18_1515	35	0	0	0	1,00	1,00	1,00	6,61	48,90	1479,25
2016-01-18_1545	36	0	0	0	1,00	1,00	1,00	6,70	50,00	1493,04
2016-01-18_1615	34	0	0	1	1,00	0,97	0,97	6,72	50,50	1489,57
2016-01-18_1645	36	0	0	0	1,00	1,00	1,00	6,61	48,80	1479,40
2016-01-18_1715	32	0	0	0	1,00	1,00	1,00	6,68	49,00	1616,73

Anexo III

Valores de precisión obtenidos por fecha:

Tabla 3.1 Valores de precisión obtenidos 2015-11-12

HORARIO	Ocupados Cámara 7 (38 espacios)	Vehículos presentes	Precisión (Cámara 7)
2015-11-12_0717	2	2	1,00
2015-11-12_0747	2	2	1,00
2015-11-12_0817	4	4	1,00
2015-11-12_0847	4	4	1,00
2015-11-12_0917	6	6	1,00
2015-11-12_0947	18	19	0,95
2015-11-12_1017	33	35	0,94
2015-11-12_1047	33	35	0,94
2015-11-12_1117	32	36	0,89
2015-11-12_1147	32	36	0,89
2015-11-12_1217	34	36	0,94
2015-11-12_1247	34	36	0,94
2015-11-12_1317	31	32	0,97
2015-11-12_1347	30	32	0,94
2015-11-12_1417	30	32	0,94
2015-11-12_1447	30	33	0,91
2015-11-12_1517	30	33	0,91
2015-11-12_1547	31	32	0,97
2015-11-12_1617	28	30	0,93
2015-11-12_1647	25	27	0,93
2015-11-12_1717	20	25	0,80
2015-11-12_1747	8	22	0,36
			Promedio (Cámara 7)
			0,92

Tabla 3.2 Valores de precisión obtenidos 2015-11-25

HORARIO	Ocupados Cámara 7 (38 espacios)	Vehículos presentes	Precisión (Cámara 7)
2015-11-25_0719	2	2	1,00
2015-11-25_0749	3	3	1,00
2015-11-25_0819	3	3	1,00
2015-11-25_0849	5	5	1,00
2015-11-25_0919	8	8	1,00
2015-11-25_0949	17	17	1,00
2015-11-25_1019	21	21	1,00
2015-11-25_1049	32	35	0,91
2015-11-25_1119	35	36	0,97
2015-11-25_1149	35	36	0,97
2015-11-25_1219	35	35	1,00
2015-11-25_1249	33	33	1,00
2015-11-25_1319	29	30	0,97
2015-11-25_1349	31	31	1,00
2015-11-25_1419	31	32	0,97
2015-11-25_1449	32	32	1,00
2015-11-25_1519	32	32	1,00
2015-11-25_1549	33	33	1,00
2015-11-25_1619	31	31	1,00
2015-11-25_1649	26	27	0,96
2015-11-25_1719	14	24	0,58
2015-11-25_1749	6	19	0,32
			Promedio (Cámara 7)
			0,94

Tabla 3.3 Valores de precisión obtenidos 2016-01-08

HORARIO	Ocupados Cámara 7 (38 espacios)	Vehículos presentes	Precisión (Cámara 7)
2016-01-08_0728	0	3	0,00
2016-01-08_0758	3	3	1,00
2016-01-08_0828	5	5	1,00
2016-01-08_0858	7	7	1,00
2016-01-08_0928	10	10	1,00
2016-01-08_0958	22	22	1,00
2016-01-08_1028	30	33	0,91
2016-01-08_1058	33	36	0,92
2016-01-08_1128	33	36	0,92
2016-01-08_1158	33	36	0,92
2016-01-08_1228	31	34	0,91
2016-01-08_1258	27	31	0,87
2016-01-08_1328	27	29	0,93
2016-01-08_1358	25	27	0,93
2016-01-08_1428	26	28	0,93
2016-01-08_1458	28	30	0,93
2016-01-08_1528	28	30	0,93
2016-01-08_1558	26	27	0,96
2016-01-08_1628	23	24	0,96
2016-01-08_1658	20	22	0,91
2016-01-08_1728	6	23	0,26
2016-01-08_1758	3	18	0,17
			Promedio (Cámara 7)
			0,83

Anexo IV

Las siguientes tablas 4.1 y 4.2 muestran los criterios a tomar a en consideración para establecer los costos de servicio de la parte *Cloud*.

Tabla 4.1 Variables estimadas en el diseño [Elaboración propia]

DESCRIPCIÓN	CANTIDAD
Cantidad de usuarios activos por día	1000
Cantidad de minutos que un usuario está conectado a la plataforma AWS AppSync	20
Cantidad de dispositivos SBC	4
Cantidad de días operativos de dispositivos SBC al mes	26
Cantidad de horas operativas de dispositivos SBC al día	16
Cantidad de mensajes en promedio que publica cada dispositivo SBC en 1 hora	4
Cantidad promedio que envía cada dispositivo SBC (KB)	1
Cantidad de veces que un usuario activo ingresa al día	1

Tabla 4.2 Costos estimados por componente en el norte de Virginia [Elaboración propia]

SERVICIO	DESCRIPCIÓN	COSTO
IoT Core	0,08 USD por millón de minutos de conexión	0,000 000 08
	0 USD envío a través de tópicos reservados	0
	0,15 USD por millón de cada regla evaluada y acción ejecutada	0,000 000 15
Lambda	0,0000034 GB/s, memoria de 256 MB	0,000 003 4
	0,20 USD por un millón de solicitudes	0,000 000 2
DynamoDB	0,625 USD por millón de unidades de solicitud de escritura	0,000 000 625
	0,125 USD por millón de unidades de solicitud de lectura	0,000 000 125
	0,25 USD por GB, con una capa gratuita de 25GB	0,25
	0 USD, si la captura de DynamoDB Streams es por Lambda	0
Cognito	0,0055 USD, después de la capa gratuita de 10 000 usuarios activos mensuales	0
SES	0,10 USD por cada 1000 emails que envía	0,000 01
QuickSight	24 USD por rol Autor	24

SERVICIO	DESCRIPCIÓN	COSTO
AppSync	4 USD por millón en operaciones de modificación y consulta	0,000 004
	2 USD por millón en actualizaciones en tiempo real	0,000 002
	0,08 USD por millón de minutos de conexión	0,000 000 08
	0,09 USD por GB transferido	0,09
CloudWatch	0,5 USD por GB almacenado	0,5
S3	0,023 USD por GB, primero 50TB	0,023
	0,005 USD por 1000 solicitudes PUT	0,000 005

La siguiente tabla 4.3 muestra el costo total determinado por el servicio *Cloud*.

Tabla 4.3 Costos determinados parte cloud [Elaboración propia]

Módulo	Descripción	Cantidad al mes	Precio Unitario	Cantidad disp. SBC / Usuarios	Precio USD
IoT Core	Total de minutos conectados a IoT Core por mes, por cada dispositivo SBC	24 960	0,000 000 08	4	0,007 987 2
	Cantidad promedio que emiten mensajes por dispositivo	1664	0	4	0
	Cantidad de veces que se ejecuta una regla en IoT Core	1664	0,000 000 15	4	0,000 998 4
	Cantidad de veces que se ejecuta una acción en IoT Core	1664	0,000 000 15	4	0,000 998 4
Cognito	Cantidad de usuarios activos (MAU) los 10 000 primeros son libres de cargos, para identidades federadas no hay cargos.	-	0	1000	0
	Uso del SDK para administrar usuarios (listar, agregar, remover a grupos, habilitar o deshabilitar usuarios)	-	-	-	0
AppSync	Total de minutos conectados a AppSync por mes, por cada usuario	520	0,000 000 08	1000	0,041 6
	Cantidad de mensajes que se reciben para cada usuario y dispositivo durante un mes	6656	0,000 002	1000	13,312
	Operaciones de consulta (una API del estado general y una API de detalle)	52	0,000 004	1000	0,208
	Operaciones de modificación (Lo realiza en casi su totalidad desde IoT Core a través del Action)	1664	0,000 004	4	0,026 624
	Cargos por transferencia en GB (Se considera 100KB por sesión)	52	0,000 009	1000	0,468

Módulo	Descripción	Cantidad al mes	Precio Unitario	Cantidad disp. SBC / Usuarios	Precio USD
DynamoDB	Operaciones de lectura	52	0,000 000 125	1000	0,006 5
	Operaciones de escritura	1664	0,000 000 625	4	0,004 16
	Almacenamiento (Se considera solo 100MB)	0,1	0	-	0
Lambda	Cantidad de invocaciones	1664	0,000 000 2	4	0,001 331 2
	Tiempo de ejecución de 200ms	332,8	0,000 003 4	4	0,004 526 08
SES	Considerando que se envían 1000 emails	1000	0,000 01	1	0,01
QuickSight	Licencia con el rol Autor	1	25	1	25
S3	Almacenamiento de historial mediciones de ocupabilidad (10 GB promedio)	10	0,023	-	0,23
	Operaciones PUT para guardar JSON desde Lambda	1664	0,000 005	4	0,033 28
CloudWatch	2 GB almacenados (Se considera 1 mes de retención de logs)	2	0,5	-	1
				TOTAL	40,36