

UNIVERSIDAD AUTÓNOMA DE MADRID
ESCUELA POLITÉCNICA SUPERIOR



Grado en Ingeniería Informática

TRABAJO FIN DE GRADO

Análisis de consumo energético para aplicaciones de visión artificial en Android

Pablo Sala del Real
Tutor: Juan Carlos San Miguel Avedillo
Ponente: José María Martínez Sánchez

Mayo 2017

Análisis de consumo energético para aplicaciones de visión artificial en Android

Pablo Sala del Real

Tutor: Juan Carlos San Miguel Avedillo

Ponente: José María Martínez Sánchez



Video Processing and Understanding Lab

Departamento de Tecnología Electrónica y de las Comunicaciones

Escuela Politécnica Superior

Universidad Autónoma de Madrid

Mayo 2017

Trabajo parcialmente financiado por el Ministerio de Economía y Competitividad del Gobierno de España bajo el proyecto TEC2014-53176-R (HA-Video)



Resumen

Aunque el primer teléfono inteligente o *smartphone* no apareció hasta 1994 con el llamado **IBM Simon**, no fue hasta la salida del primer iPhone cuando realmente se popularizaron los teléfonos móviles inteligentes. Desde entonces, los *smartphones* han cambiado nuestras necesidades y forma de comunicarnos. Tanto es así, que según un estudio de SurveyMonkey¹ realizado en 2015 a más 1000 personas, la batería de un teléfono móvil es la característica que más solicitan los usuarios que se mejore, con un 33 % de los encuestados. Por tanto, nace la necesidad de realizar un análisis acerca de qué componentes o módulos de un teléfono móvil tienen un mayor gasto de energía. Este es el principal objetivo de este Trabajo de Fin de Grado.

En primer lugar, se han establecido tres módulos en los que se puede dividir un teléfono móvil: captura de imágenes, procesamiento de imágenes y comunicaciones entre dispositivos. Para cada uno de los módulos se variarán distintos parámetros para observar cómo influye cada uno de ellos en el consumo energético.

Para ello, se han implementado tres aplicaciones Android para medir el consumo energético, una por cada módulo, mediante la obtención de distintas estadísticas del teléfono móvil.

Finalmente, a partir de las estadísticas extraídas del dispositivo móvil, se han obtenido distintos modelos por medio de la herramienta *MatLab*, que se utilizarán para comparar los resultados.

Palabras clave

consumo energético, Android, visión artificial, captura, procesamiento, comunicaciones, detección de personas.

¹<https://www.surveymonkey.com/>

Abstract

Although the first smartphone appeared in 1994 with the so-called IBM Simon, smartphones had not really become popular until the first iPhone was released. Since then, smartphones have changed our needs and way of communicating each other. In fact, according to a survey from SurveyMonkey¹ which was conducted in 2015 to more than 1000 people, smartphones batteries is the most requested feature to get improved, with the 33% of respondents. Therefore, it is necessary to carry out an analysis about which components or modules of a mobile phone have a higher energy expenditure. This is the Final Degree Thesis main objective.

Firstly, a smartphone has been divided into three modules: images sensing, image processing and communication between devices. For each one of the modules, different parameters will be varied in order to see how these parameters influence the power consumption.

To do that, three Android applications (one per each module) have been developed to measure the power consumption by obtaining different battery statistics.

Finally, based on the statistics from the mobile device, different models have been obtained through the MatLab tool, which will be used to compare the results.

Keywords

power consumption, Android, computer vision, sensing, processing, communications, people detection.

Agradecimientos

Quiero agradecer a toda mi familia, en especial a mis padres, hermana y cuñado, que siempre me han apoyado y dado ánimos desde el principio.

Y también agradecer a mi tutor Juan Carlos. Gracias por proponerme este trabajo y ayudarme en todo lo que he necesitado, en el que he podido aprender muchos conceptos que seguro que me servirán en un futuro.

Índice general

Resumen	v
Abstract	vii
Agradecimientos	ix
1. Introducción	1
1.1. Motivación	1
1.2. Objetivos	3
1.3. Organización de la memoria	4
2. Estado del arte	5
2.1. Introducción	5
2.2. Modelado de consumo energético	5
2.2.1. Magnitudes	5
2.2.2. Conversión entre magnitudes	6
2.3. Partes de una cámara inteligente	6
2.4. Estudio del consumo energético	7
2.4.1. En una cámara inteligente	7
2.4.2. En un dispositivo móvil basado en Android	9
2.5. Herramientas para medir el consumo energético	11
2.5.1. Polímetro	12
2.5.2. Medidor de consumo de energía	12
2.5.3. Medidor USB	12
2.5.4. Descarga de la batería mediante Software	13
2.6. Conclusiones	13
3. Diseño y desarrollo	15
3.1. Metodología	15
3.2. Medida de consumo	17
3.3. Aplicaciones Android desarrolladas	20
3.4. Herramientas utilizadas	21
3.4.1. Android	22
3.4.2. Android Studio	23
3.4.3. OpenCV	23

3.4.4. MatLab	24
4. Pruebas y resultados	25
4.1. Descripción del entorno de pruebas	25
4.2. Tiempo adecuado de muestreo	27
4.3. Resultados del experimento de captura	28
4.4. Resultados del experimento de procesamiento	31
4.5. Resultados del experimento de comunicaciones	34
5. Conclusiones y trabajo futuro	39
5.1. Conclusiones	39
5.2. Trabajo futuro	39
A. Consumo energético y potencia media por test y repetición	45
B. Requisitos para realizar los experimentos	61

Índice de figuras

1.1. Consumo energético por componentes	2
1.2. Lectura de constantes vitales mediante vídeo	2
2.1. Módulos de una cámara y relación entre ellos	7
2.2. Resultados del experimento de comunicaciones entre redes de cámaras	8
2.3. Resultados de la aplicación multi-cámara	8
2.4. Funcionamiento energético de Android	9
2.5. Resultados del experimento Wi-Fi en Android	10
2.6. Resultados del análisis del consumo por escenarios en Android	11
2.7. Polímetros	12
2.8. Medidor de consumo energético	13
2.9. Medidor USB	14
3.1. Diagrama de flujo del procedimiento general	17
3.2. Diagrama de flujo del experimento de captura	18
3.3. Diagrama de flujo del experimento de procesamiento	19
3.4. Diagrama de flujo del experimento de comunicaciones	20
3.5. Aplicaciones desarrolladas	21
3.6. Arquitectura de Android	22
4.1. Dispositivos móviles utilizados	25
4.2. Comparación del tiempo de muestreo respecto al tiempo total de re- petición	27
4.3. Comparación del tiempo de muestreo respecto al número de muestras	28
4.4. Comparativa de consumo por resoluciones en el Samsung Galaxy S4 Mini	30
4.5. Comparativa de consumo por imágenes por segundo en el Samsung Galaxy S4 Mini	31
4.6. Comparativa de consumo por resoluciones en el bq Aquaris X5 Plus .	32

4.7. Comparativa de consumo por imágenes por segundo en el bq Aquaris X5 Plus	33
4.8. Ejemplo de imagen de personas	34
4.9. Imagen procesada utilizando el algoritmo de detección de personas . .	35
4.10. Resultados del experimento de procesamiento en el Samsung Galaxy S4 Mini	36
4.11. Resultados del experimento de procesamiento en el bq Aquaris X5 Plus	37
4.12. Resultados del experimento de comunicaciones en el Samsung Galaxy S4 Mini	38
4.13. Resultados del experimento de comunicaciones en el bq Aquaris X5 Plus	38
A.1. Modo activo con la cámara sin capturar imágenes para el Samsung Galaxy S4 Mini	46
A.2. Modo activo con la cámara capturando imágenes para el Samsung Galaxy S4 Mini	47
A.3. Modo activo con la cámara sin capturar imágenes para el bq Aquaris X5 Plus (I)	48
A.4. Modo activo con la cámara sin capturar imágenes para el bq Aquaris X5 Plus (II)	49
A.5. Modo activo con la cámara capturando imágenes para el bq Aquaris X5 Plus (I)	50
A.6. Modo activo con la cámara capturando imágenes para el bq Aquaris X5 Plus (II)	51
A.7. Modo en espera con la CPU sin realizar trabajo para el Samsung Galaxy S4 Mini	52
A.8. Modo activo con la CPU procesando imágenes para el Samsung Galaxy S4 Mini	53
A.9. Modo en espera con la CPU sin realizar trabajo para el bq Aquaris X5 Plus	54
A.10. Modo activo con la CPU procesando imágenes para el bq Aquaris X5 Plus	55
A.11. Resultados del experimento con límite de 10 kbps de ancho de banda para el Samsung Galaxy S4 Mini	56
A.12. Resultados del experimento con límite de 100 kbps de ancho de banda para el Samsung Galaxy S4 Mini	57
A.13. Resultados del experimento con límite de 1000 kbps de ancho de banda para el Samsung Galaxy S4 Mini	57

A.14.Resultados del experimento sin límite de ancho de banda para el Samsung Galaxy S4 Mini	58
A.15.Resultados del experimento con límite de 10 kbps de ancho de banda para el bq Aquaris X5 Plus	58
A.16.Resultados del experimento con límite de 100 kbps de ancho de banda para el bq Aquaris X5 Plus	59
A.17.Resultados del experimento con límite de 1000 kbps de ancho de banda para el bq Aquaris X5 Plus	59
A.18.Resultados del experimento sin límite de ancho de banda para el bq Aquaris X5 Plus	60

Índice de tablas

4.1. Especificaciones técnicas de los dispositivos móviles	26
4.2. Parámetros para el experimento de captura	29
4.3. Parámetros para el experimento de procesamiento	32
4.4. Parámetros para el experimento de comunicaciones	36

Glosario

Android: Sistema Operativo para dispositivos móviles, como smartphones y tablets, dispositivos wearables, coche o televisión desarrollado por Google.

API: Application Programming Interface; interfaz de programación de aplicaciones.

ART: Android Runtime.

consumo energético: Gasto total de energía en un proceso determinado.

FPS: Frames Per Second; imágenes por segundo.

HAL: Hardware Abstraction Layer; Capa de abstracción de hardware.

IDE Integrated Development Environment; entorno de desarrollo integrado.

visión artificial: Campo de la inteligencia artificial mediante el cual los dispositivos electrónicos capturan, procesan, analizan y entienden imágenes digitales, con el fin de extraer datos del mundo real para crear información numérica o simbólica.

Capítulo 1

Introducción

1.1. Motivación

En 2007, Apple revolucionó el mercado de los teléfonos móviles con el lanzamiento del iPhone¹, uno de los primeros dispositivos móviles en utilizar una pantalla multi-táctil. Esta pantalla táctil permitía una interacción con el dispositivo directamente con el dedo, en vez de utilizar un *stylus* o teclado, al igual que poder interactuar con la pantalla con gestos de dos o más dedos.

Desde entonces, el mercado de los teléfonos inteligentes no ha parado de crecer. En 2015, aproximadamente la mitad de las personas del mundo poseían un teléfono inteligente[1]. Algunas estadísticas sobre el uso de los *smartphones*[2] son:

- El 68 % de las personas revisan su teléfono dentro de los 15 minutos después de levantarse.
- El 87 % tienen siempre su teléfono a su lado, ya sea por la mañana o por la noche.
- De media las personas comprueban su teléfono unas 150 veces al día, y gastan 177 minutos en él.
- El 82 % de las personas consultan su teléfono mientras compran en un lugar físico.
- El 91 % recurren a su teléfono móvil para encontrar ideas sobre una tarea que se tenga que hacer.

Debido a esto, el teléfono móvil se ha vuelto una herramienta indispensable en nuestra vida, por lo que es necesario un análisis del consumo energético que permita conocer

¹iPhone (Primera generacion) - https://es.wikipedia.org/wiki/iPhone_2G

qué componentes consumen más energía con el propósito de optimizar los recursos del teléfono móvil. En 2009, Google publicó una gráfica en la que se puede observar qué componentes gastan más energía de media[3].

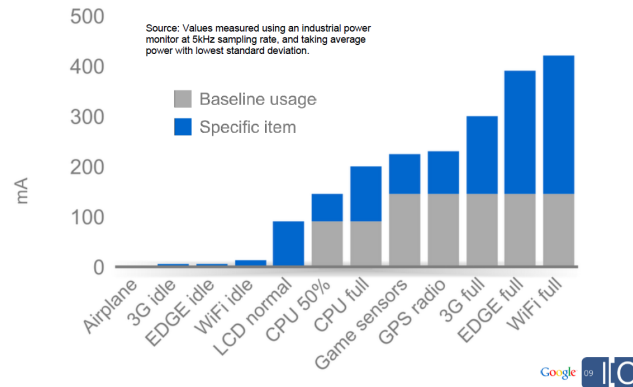


Figura 1.1: Consumo energético por componentes [3]

En la Figura 1.1 se puede comprobar que unos de los componentes que consumen más energía son las comunicaciones (Wi-Fi, 3G) cuando se encuentran activos a máxima intensidad mientras que cuando están en espera son los que menos batería gastan. A lo largo de este trabajo comprobaremos si en nuestro caso esto se cumple.

Por otro lado, la visión artificial es un campo de la inteligencia artificial mediante el cual los dispositivos electrónicos, como ordenadores o dispositivos móviles, capturan, procesan, analizan y entienden imágenes digitales, con el fin de extraer datos del mundo real para crear información numérica o simbólica. Relacionada con este área, Google creó una plataforma de realidad aumentada para dispositivos móviles llamada *Tango* en 2014.

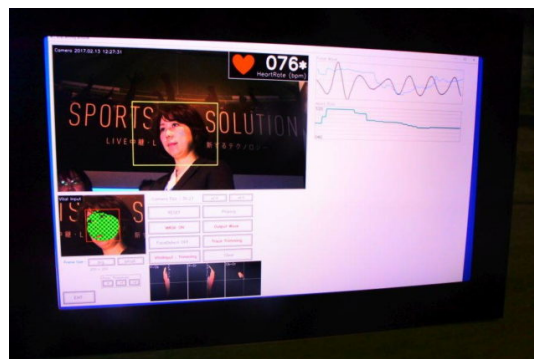


Figura 1.2: Lectura de constantes vitales mediante vídeo [4]

Algunas de las aplicaciones [5] de la visión artificial que se están investigando en *smartphones* son:

- Tomar una fotografía de la comida y mediante una aplicación controlar la dieta con su información nutricional.
- Leer constantes vitales tomando una fotografía de la cara (Figura 1.2).
- Asistir conductores usando un dispositivo móvil.

De hecho, existen aplicaciones tanto para Android como iOS que ya realizan la función de asistir en la conducción, como iOnRoad². Esta aplicación, utilizando la tecnología de la visión artificial, permite advertir sobre posibles colisiones frontales, salidas de carril, o incluso controla la distancia de seguridad, entre otras funciones. El problema de estas aplicaciones es que poseen un consumo energético elevado debido a la cantidad de recursos que utilizan del dispositivo móvil, como la captura o el procesamiento de imágenes, así como la pantalla.

No obstante, otra forma de poder emplear la visión artificial sin la necesidad de un dispositivo móvil sería una cámara inteligente. Estos dispositivos son sistemas de visión, que además de tomar imágenes, pueden extraer información de las imágenes capturadas o tomar decisiones. Las cámaras inteligentes son utilizadas como parte de un sistema inteligente y automatizado.

Debido a todos estos usos de la visión artificial mediante cámaras inteligentes (incluyendo *smartphones*, ya que pueden tener la misma finalidad), esta tecnología será muy importante en los próximos años, por lo que analizar el consumo de una aplicación que emplea la visión artificial será muy útil de cara al futuro.

1.2. Objetivos

El principal objetivo de este TFG es la creación de aplicaciones Android para analizar el consumo energético de las operaciones realizadas por un *smartphone* extrayendo diversas estadísticas de la batería del teléfono móvil representativas del patrón de consumo existente. Cada una de estas aplicaciones medirá la influencia en el consumo energético de algunos módulos del dispositivo.

Para alcanzar el principal objetivo, el TFG define los siguientes subobjetivos:

- Estudiar el estado del arte en el análisis del consumo energético en dispositivos electrónicos, así como, en aplicaciones de visión artificial.

²<http://www.ionroad.com/>

- Definir los módulos en los que se puede dividir una cámara inteligente y aplicarlo a un dispositivo móvil.
- Desarrollar aplicaciones Android para obtener estadísticas de la batería para cada uno de los módulos.
- Análisis de los resultados mediante la herramienta MatLab que permitan comparar el consumo energético.

1.3. Organización de la memoria

La memoria consta de los siguientes capítulos:

- Capítulo 1. Introducción: Motivación, objetivos y organización de la memoria.
- Capítulo 2. Estado del arte: En este capítulo se explicarán qué unidades relativas al consumo energético son importantes, así como estudios que se han realizado sobre el consumo energético, tanto en una cámara inteligente como en un dispositivo móvil.
- Capítulo 3. Diseño y desarrollo: Se explicará en este capítulo cómo se va a llevar a cabo el análisis. De la misma forma, también se mostrarán qué herramientas han sido utilizadas para llevar a cabo el análisis.
- Capítulo 4. Pruebas y resultados: En esta parte, se realizará un estudio de los resultados que se han obtenido de las pruebas.
- Capítulo 5. Conclusiones y trabajo futuro: Por último, se expondrán las conclusiones que se han extraído del análisis y el posible trabajo que se puede realizar en el futuro.
- Bibliografía.
- Apéndices:
 - ✧ Apéndice A: Consumo energético y potencia media por test y repetición. En este anexo, se mostrarán los modelos obtenidos de las pruebas de forma más detallada.
 - ✧ Apéndice B: Requisitos para realizar los experimentos. Mediante este apéndice se explicará qué es necesario para poder realizar los experimentos.

Capítulo 2

Estado del arte

2.1. Introducción

Durante este capítulo se explicará en qué estado se encuentra el análisis del consumo energético en cámaras inteligentes y, más concretamente, en dispositivos móviles basados en Android. En primer lugar, en la sección 2.2, se definirán qué magnitudes son importantes para medir el consumo energético, así como, la conversión entre unidades. Más adelante, se detallarán de qué partes o módulos puede constar una cámara inteligente (Sección 2.3). Posteriormente, se comentarán diversos estudios que se han realizado anteriormente sobre el principal tema de este TFG (Sección 2.4). Por último, en la sección 2.5, se describirán múltiples herramientas o métodos que se pueden utilizar para cuantificar el consumo energético en un dispositivo electrónico.

2.2. Modelado de consumo energético

2.2.1. Magnitudes

Las principales magnitudes relacionadas con el consumo energético son:

- Carga eléctrica (Q): propiedad intrínseca de algunas partículas que se manifiesta a través de atracciones y repulsiones determinando las interacciones electromagnéticas entre ellas. Su unidad es el culombio (C). Otras unidades habituales en las baterías eléctricas son el amperio-hora (Ah) o el miliamperio-hora (mAh).
- Voltaje (V): diferencia de potencial eléctrico entre dos puntos. Se mide en voltios (V).
- Intensidad eléctrica (I): flujo de carga eléctrica por unidad de tiempo que recorre un material. Su unidad en el Sistema Internacional es el $\frac{C}{s}$ o amperio (A).

$$I = \frac{Q}{t} \quad (2.1)$$

- Energía eléctrica (E): capacidad para producir electricidad que resulta de la existencia de una diferencial de potencial entre dos puntos. Se mide en julios (J) o en vatios-hora (Wh).

$$E = \int P dt \simeq P \cdot t \quad (2.2)$$

- Potencia eléctrica (P): relación de paso de energía de un flujo por unidad de tiempo. Su unidad es el $\frac{J}{s}$ o vatio (W).

$$P = \frac{E}{t} = \frac{E}{Q} \cdot \frac{Q}{t} = V \cdot I \quad (2.3)$$

2.2.2. Conversión entre magnitudes

Cambio de Vatios-hora (Wh) a miliAmperios-hora (mAh)

Para pasar de una unidad de energía a una de carga es necesario saber el voltaje que se ha generado.

$$Q_{(mAh)} = 10^3 \cdot \frac{E_{(Wh)}}{V_{(V)}} \quad (2.4)$$

Cambio de Voltios (V) a Julios (J)

Es posible cambiar de voltios a julios si se tiene la carga eléctrica en culombios.

$$E_{(J)} = V_{(V)} \cdot Q_{(C)} \quad (2.5)$$

2.3. Partes de una cámara inteligente

Una cámara inteligente posee varios módulos, aunque se ha decidido centrarse en tres (Figura 2.1):

- Captura de imágenes. Este módulo es el de la cámara en sí, en el que se abrirá y se procederá a la captura de imágenes en segundo plano.
- Procesamiento de imágenes. Una vez capturadas las imágenes se enviarán al módulo de procesamiento, que mediante herramientas como OpenCV, se podrían detectar personas, entre otros.

- Comunicación con otros dispositivos. Este módulo tiene la finalidad de comunicarse con otras cámaras, para enviar al resto cuando se conecta una, o con un servidor, para enviar las imágenes procesadas.

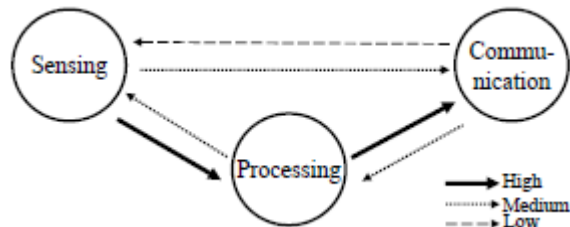


Figura 2.1: Módulos de una cámara y relación entre ellos [6]

2.4. Estudio del consumo energético

En esta sección, se mostrarán diversos estudios que se han realizado anteriormente sobre el análisis del consumo energético en dispositivos electrónicos, como una cámara inteligente o un teléfono móvil basado en Android.

2.4.1. En una cámara inteligente

Como se ha explicado anteriormente en la Sección 1.1, una cámara inteligente es un sistema que permite capturar imágenes y extraer información de las mismas.

Posteriormente, se pasará a describir un experimento que se realizó en 2009 [7] sobre el módulo de comunicaciones de una red de cámaras inteligentes. Este experimento se basaba en medir el consumo energético en tres casos:

- Cuando un objeto entraba en el campo de visión de una cámara (*New_label*).
- Cuando un objeto desaparecía del campo de visión de una cámara (*Lost_label*).
- Cuando una cámara enviaba mensajes de información al siguiente nodo (*Primitive_event*).

En el caso de los mensajes *New_label* y *Lost_label*, cuando una cámara enviaba alguno de estos mensajes a otro nodo, este le respondía con otro paquete.

Los resultados de este experimento se muestran en la Figura 2.2. Concluyeron que el intercambio de mensajes que más energía consume es la respuesta de un nodo cuando otra cámara le envía que ha desaparecido un objeto de su campo de visión. No obstante, el consumo energético depende en gran medida de la actividad de la escena.

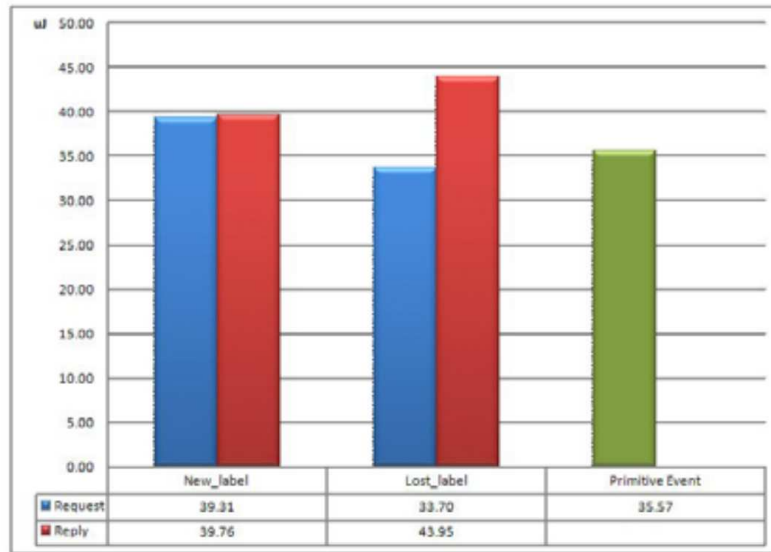


Figura 2.2: Resultados del experimento de comunicaciones entre redes de cámaras [7]

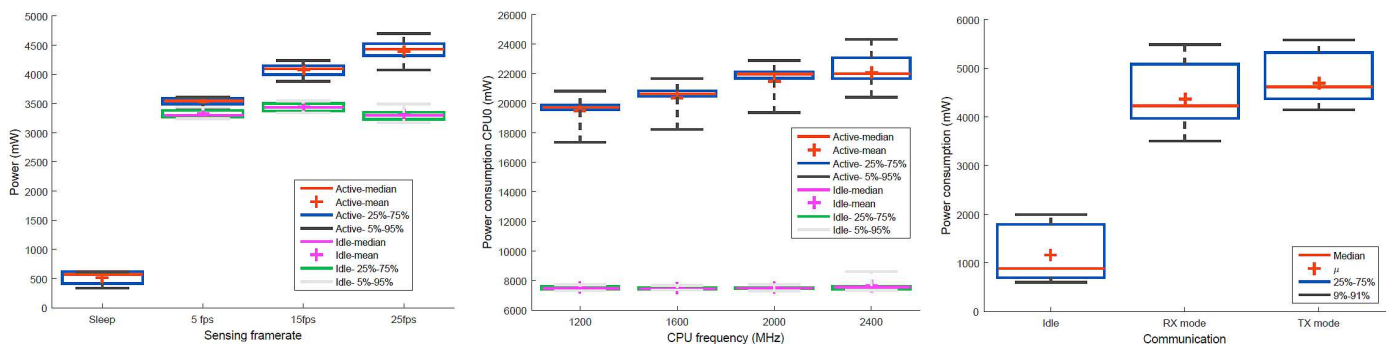


Figura 2.3: Resultados de la aplicación multi-cámara [6]

Otro experimento que se realizó [6] fue precisamente analizando el consumo energético de los tres módulos (captura, procesamiento y comunicaciones) en una aplicación de seguimiento multi-cámara, que incorporaba la tecnología de la visión artificial. Este experimento consistía en ir variando distintos parámetros de cada módulo para ver su influencia dentro del consumo de la batería. Los resultados se muestran en la Figura 2.3.

Se concluyó que el módulo que más consume es el de procesamiento, siendo este consumo entre un 45 % y un 78 % del total, seguido del módulo de captura (19%).

Por último, anteriormente también se ha analizado el consumo energético de un sensor CMOS[8] y su relación con diversos parámetros como la resolución de imagen

y la tasa de imágenes. Se concluyó que una forma de reducir el consumo energético de un sensor CMOS es disminuir la frecuencia de reloj a una óptima y un agresivo modo de espera, como, por ejemplo, una resolución de 1 MegaPixel y una tasa de 5 imágenes por segundo.

2.4.2. En un dispositivo móvil basado en Android

Android es un sistema operativo que está compuesto por el *framework* de Android y un *kernel* basado en Linux. Ambos proporcionan al sistema operativo características tales como aplicaciones y servicios. Una aplicación es un proceso que posee una interfaz gráfica, mientras que un servicio es un proceso en segundo plano. Estos procesos (aplicaciones y servicios) pueden utilizar los componentes *hardware* como la red o periféricos.

Por otra parte, las aplicaciones pueden solicitar en mayor o menor medida la CPU. En ocasiones se pueden producir *wakelocks*, que son procesos que se quedan abiertos en segundo plano y evitan que el dispositivo entre en modo de espera incluso cuando la pantalla se encuentra apagada. Esto puede producir grandes drenajes en la batería. En la Figura 2.4 se puede ver esta configuración de Android basada en el consumo energético.

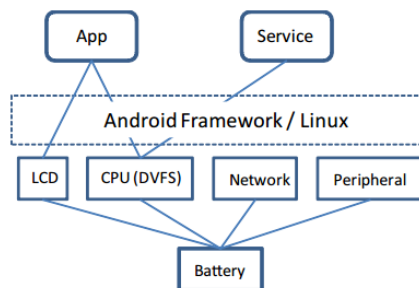


Figura 2.4: Funcionamiento energético de Android[9]

Una vez explicado el funcionamiento de Android desde el punto de vista energético, se realizó un experimento[9] sobre el módulo de Wi-Fi con el dispositivo móvil en el modo de ahorro de energía. Este experimento consistía en que el dispositivo actuaba como cliente, el cual recibía una señal del punto de acceso para indicarle que empezaba a transmitir datos.

El dispositivo se comportaba de la siguiente manera (Figura 2.5):

- Al principio, el móvil se encontraba en el estado de sueño profundo (*Deep Sleep*).
- Cuando está transmitiendo datos, pasa al estado de alta potencia (*High Power*).

- El móvil permanece en un estado inactivo de escucha (*Idle Listen*) mientras espera a que lleguen más datos.
- El consumo se reduce cuando se recibe una señal de que no van a llegar más datos y se pasa a un estado de sueño ligero (*Light Sleep*). Continúa en este estado durante un tiempo para estar preparado por si llegan más paquetes en un tiempo corto.
- El dispositivo vuelve al estado de sueño profundo cuando llega el evento del temporizador de inactividad.

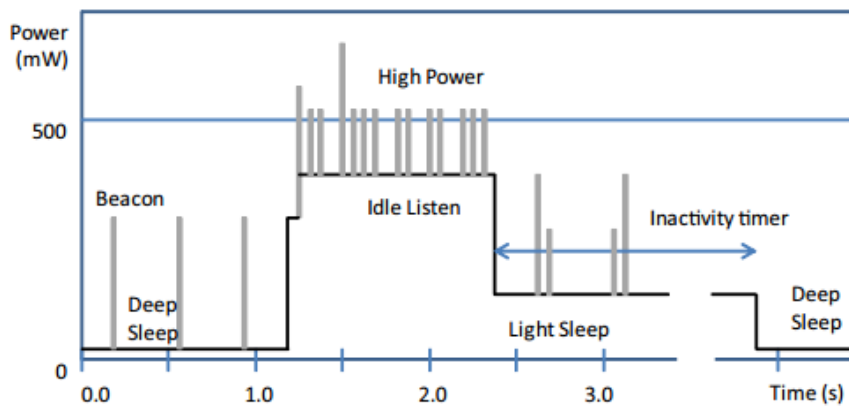


Figura 2.5: Resultados del experimento Wi-Fi en Android [9]

Otro ejemplo de experimento [10] realizado sobre el análisis de consumo energético en dispositivos móviles se basó en definir varios escenarios para comparar el rendimiento energético del dispositivo. Estos escenarios eran:

- Modo normal: es el modo en el que los usuarios utilizan sus dispositivos de la forma habitual.
- Modo de suspensión: aquel modo en el que los usuarios no utilizan el dispositivo.
- Redes de comunicaciones: las redes que se analizaron fueron la conexión Wi-Fi y los datos móviles (LTE).
- Modo avión: modo en el que se desactivan todas las comunicaciones del dispositivo.
- Modo de ahorro de energía: en este modo se aumenta la duración de la batería disminuyendo el consumo de corriente.

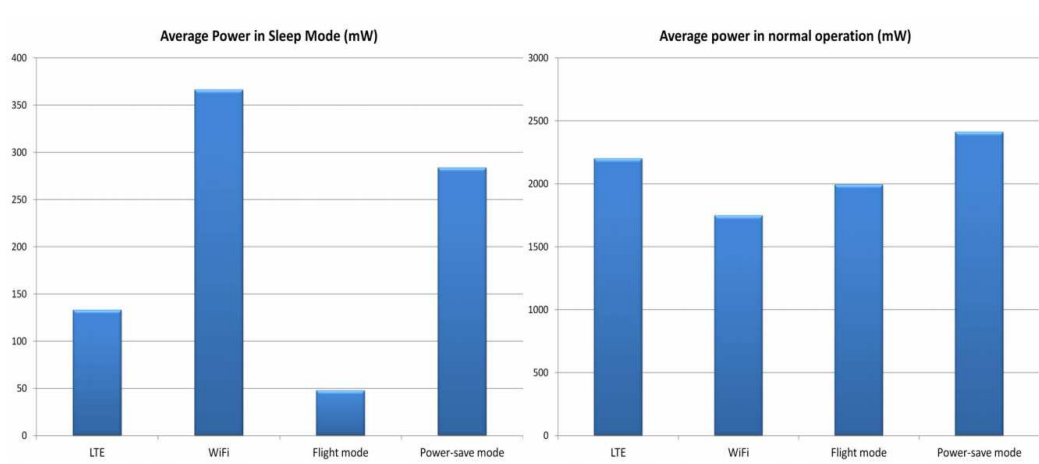


Figura 2.6: Resultados del análisis del consumo por escenarios en Android [10]

Como se puede ver en los resultados del experimento (Figura 2.6), el modo de ahorro de energía no es el modo más eficiente para reducir el consumo de energía, siendo este el modo avión. Por otro lado, la conexión LTE funciona mejor en el modo en suspensión que la conexión Wi-Fi, mientras que en el uso normal el consumo se invierte.

Otros experimentos [11] analizaban el consumo energético de dispositivos móviles detectando determinados objetos habiendo realizado previamente antes un entrenamiento con diversos algoritmos como HOG-AdaBoost o Lbp-AdaBoost. Se concluyó que el algoritmo que consumía menos batería era el HOG-AdaBoost.

De la misma forma, también se ha analizado el gasto de energía mediante un pequeño algoritmo basado en múltiples bucles anidados[12], con la conclusión de que la energía consumida está relacionada con la frecuencia de la CPU.

2.5. Herramientas para medir el consumo energético

Existen diversas herramientas que se pueden utilizar para medir el consumo energético de la batería de los dispositivos electrónicos, que van desde un polímetro para medir directamente la corriente y voltaje de la batería hasta herramientas software que proporciona el propio sistema operativo de los dispositivos electrónicos.

2.5.1. Polímetro

Un polímetro es un instrumento utilizado para medir varias magnitudes eléctricas, como el voltaje o la intensidad. Existen tanto polímetros analógicos (Figura 2.7a), que incorporan un amperímetro junto a una aguja para marcar la medida, como digitales que, en vez de utilizar una aguja como en los polímetros analógicos, poseen una pantalla que muestra la medida. En la Figura 2.7b se muestra un polímetro digital. Actualmente, los polímetros más habituales son los digitales.



(a) Polímetro analógico[13]



(b) Polímetro digital[14]

Figura 2.7: Polímetros

2.5.2. Medidor de consumo de energía

Un medidor de consumo energético (Figura 2.8) es un dispositivo que posee un enchufe de entrada y otro de salida y permite medir el consumo energético y coste de los dispositivos electrónicos conectados a él. Gracias a ellos, se pueden conocer varios parámetros, como la potencia, energía, voltaje o corriente que está fluyendo por el medidor. La mayoría de ellos tienen una pantalla que permite ver estas medidas fácilmente.

2.5.3. Medidor USB

Los medidores USB (Figura 2.9) son pequeños dispositivos que muestran en una pequeña pantalla a qué corriente y voltaje se está cargando un dispositivo electrónico. Algunos de ellos también pueden mostrar el consumo energético en forma de



Figura 2.8: Medidor de consumo energético [10]

miliAmperios-hora. Poseen un puerto USB de entrada para conectarse al dispositivo electrónico y un puerto USB de salida para conectarlo a la fuente de alimentación, como un enchufe o batería portátil.

2.5.4. Descarga de la batería mediante Software

Finalmente, los sistemas operativos basados en Linux, como Ubuntu o Android, proveen de un archivo de sistema donde se encuentra diversa información sobre la batería, como su capacidad o voltaje actuales. La ruta a este fichero en Ubuntu es `/sys/class/power_supply/BAT0/uevent` mientras que en Android es `/sys/class/power_supply/battery/uevent`. Como se puede observar, ambas rutas son parecidas, ya que como se ha explicado anteriormente, ambos sistemas están basados en Linux. Este trabajo utiliza este método, al igual que, por ejemplo, en [6].

2.6. Conclusiones

A lo largo de esta sección, se ha explicado diversos estudios que se han realizado anteriormente sobre el análisis del consumo energético tanto en cámaras inteligentes, como en dispositivos móviles Android, así como, distintas formas de medir estadísticas de la batería, tales como el voltaje o la corriente. Sin embargo, no se han realizado estudios recientes sobre qué componentes en Android consumen más energía.

Con las tres aplicaciones Android que se han desarrollado se intentará resolver esta cuestión, entre la cámara, el procesador o la conexión Wi-Fi, con la ayuda de modelos que se obtendrán por medio de la herramienta MatLab.



Figura 2.9: Medidor USB [15]

Capítulo 3

Diseño y desarrollo

Una vez se ha analizado el estado del arte, se pasará a explicar en qué consisten los experimentos y cómo se han implementado.

En primer lugar, se mostrará cual es el procedimiento general que seguirán los experimentos (Sección 3.1). Posteriormente, se explicará más en detalle cómo se realiza el experimento de cada módulo indicando en qué se basa cada experimento (Sección 3.2). En la Sección 3.3, se mostrarán el diseño de las aplicaciones Android desarrolladas, así como la estructura de clases que se seguido en el desarrollo. Por último, se definirán brevemente las herramientas que se han utilizado para la realización de este trabajo (Sección 3.4).

3.1. Metodología

Como se ha explicado anteriormente en la Sección 2.3 sobre las partes de una cámara inteligente, se va a proceder a analizar el consumo de tres módulos en Android:

- Módulo de captura.
- Módulo de procesamiento.
- Módulo de comunicaciones.

Los análisis van a consistir en tomar algunas estadísticas de la batería cada cierto tiempo (tiempo de muestreo) bajo una serie de condiciones comunes a todos los tests, para que el dispositivo móvil se encuentre en la misma situación. Las estadísticas que se recogerán van a ser:

- Voltaje instantáneo.
- Intensidad instantánea.

- Potencia instantánea.
- Carga restante.
- Energía restante.

No obstante, dependiendo del indicador de carga de la batería instalado en el dispositivo, las estadísticas disponibles pueden variar. El indicador de carga de la batería es un componente *hardware* que informa acerca de la carga y tiempo restante de la batería. En este caso, en los dos dispositivos móviles utilizados, las únicas estadísticas disponibles han sido el voltaje y la intensidad instantáneas, mientras que la potencia instantánea ha sido calculada con la Ecuación 2.3, y la energía y carga total consumidas en cada experimento mediante las Ecuaciones 3.1 y 2.4, respectivamente.

$$E_T = \sum (\bar{P} \cdot t) \quad (3.1)$$

Por otro lado, dependiendo de cada test se irán variando algunos parámetros, como la resolución de la cámara (en el análisis del módulo de captura) o la velocidad del procesador (cuando se analice el procesador), para observar cómo afectan al consumo energético.

Cada test consta de dos modos:

- Modo calibración. Consiste en analizar el consumo energético cuando el componente se encuentra desactivado o está activado sin realizar trabajo.
- Modo activo. En este caso, el componente está activado trabajando a máxima potencia.

En general, el procedimiento que se va seguir en cada test es el que se muestra en la Figura 3.1.

En primer lugar, se establecerán las condiciones generales a todos los tests. Estas condiciones generales son:

- Reducir el brillo al máximo.
- Desactivar todos los procesadores excepto uno.
- Cambiar la frecuencia del procesador activo a una determinada.
- Desactivar la conexión Wi-Fi.
- Desactivar el Bluetooth.
- Desactivar los datos móviles.

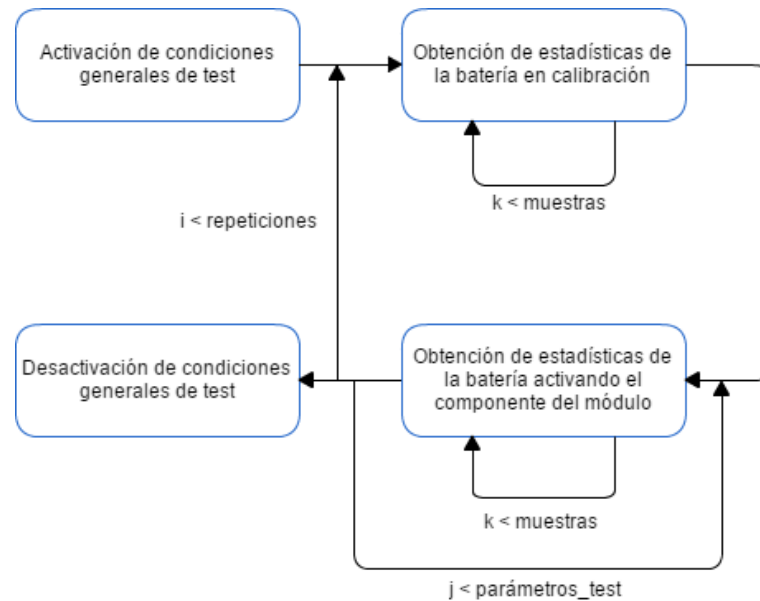


Figura 3.1: Diagrama de flujo del procedimiento general

Posteriormente, se van tomando un número determinado de muestras tanto en el modo calibración como en el activo. Esta parte del test se repetirá tantas veces como se quiera. Por último, antes de acabar el test se desactivarán las condiciones que se han establecido anteriormente para dejar el dispositivo móvil como se encontraba anteriormente. Todas las estadísticas recogidas se guardarán en archivos de texto para después generar los modelos con MatLab.

3.2. Medida de consumo

En esta sección se explicará más en profundidad en qué consiste cada test de los módulos.

Captura de imágenes

Durante este test se medirá el consumo energético que genera el módulo de la cámara. De la misma forma, se observará cómo influyen algunos parámetros de la cámara sobre el consumo de la batería. Estos parámetros son:

- Resolución: tamaño de las imágenes que se van a capturar.
- Tasa de imágenes por segundo (FPS): número de imágenes capturadas por segundo.

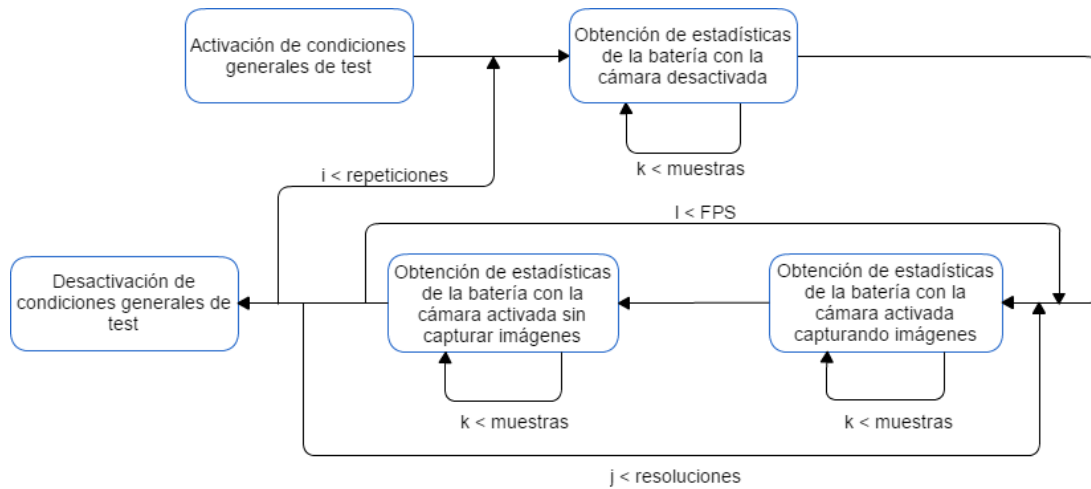


Figura 3.2: Diagrama de flujo del experimento de captura

En particular, el flujo de este test se muestra en la Figura 3.2. En este caso, el modo de calibración, que se ha explicado en la Sección 3.1, se corresponde a tomar estadísticas cuando la cámara se encuentra desactivada. Seguidamente, el modo activo se dividirá en dos:

- Cámara activada y capturando imágenes.
- Cámara activada sin capturar imágenes.

Procesamiento de imágenes

En este test se verá cómo influye el procesador en el consumo de la batería, y más en concreto, de qué manera influyen diversos parámetros, como:

- Velocidad del procesador.
- Carga de trabajo del procesador.

Para ello, en la Figura 3.3 se muestra en qué consistirá este test. De esta forma, las estadísticas de la batería se medirán en dos situaciones:

- Cuando el procesador no esté realizando ningún trabajo.
- Cuando el procesador esté trabajando. Este trabajo consistirá en procesar una imagen mediante OpenCV.

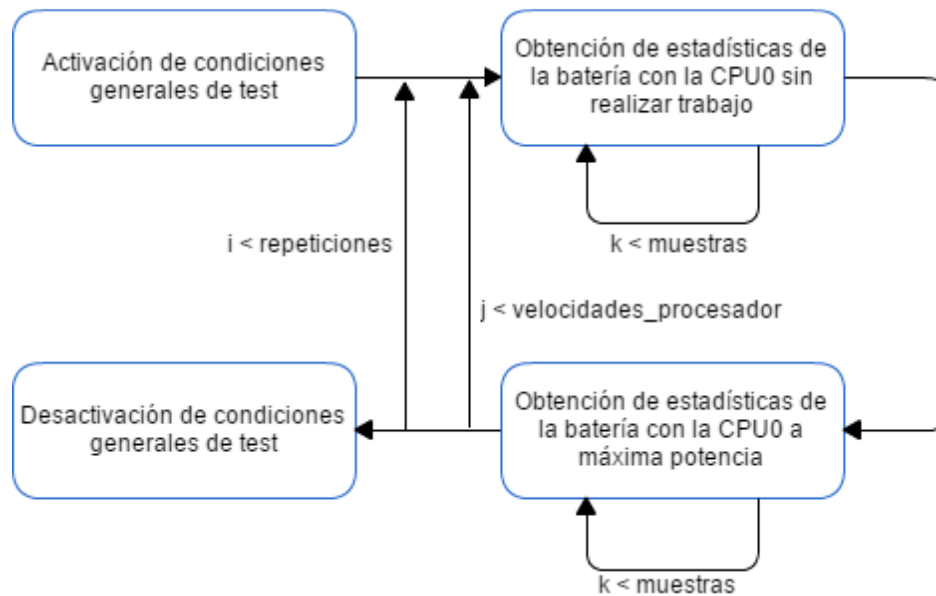


Figura 3.3: Diagrama de flujo del experimento de procesamiento

Comunicación por Wi-Fi

Por último, se medirá el consumo energético del módulo de comunicaciones, en concreto, de la conexión Wi-Fi. En este caso, se calculará la influencia de los siguientes parámetros:

- Límite de ancho de banda. Se establecerá un límite para que no se reciban / transmitan más kilobits por segundo (kbps) de los seleccionados.
- Tamaño del paquete. Definirá el tamaño del paquete que se enviará / recibirá.

Este test precisa de una conexión cliente / servidor. El cliente será el dispositivo móvil que se conectará a un servidor ubicado en un ordenador. Este servidor simplemente enviará o recibirá paquetes al dispositivo móvil.

El test (Figura 3.4) consta de tres modos en los que se obtienen medidas energéticas de la batería. En concreto son:

- Cuando el Wi-Fi se encuentra activado sin enviar ni recibir paquetes.
- Cuando el Wi-Fi está recibiendo paquetes del servidor (modo RX).
- Cuando el Wi-Fi envía paquetes al servidor (modo TX).

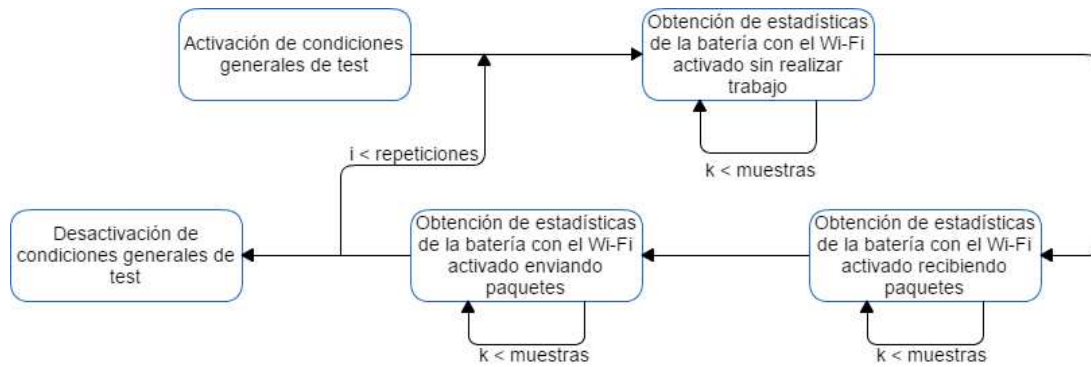


Figura 3.4: Diagrama de flujo del experimento de comunicaciones

3.3. Aplicaciones Android desarrolladas

A lo largo de esta sección, se mostrarán el diseño de las aplicaciones que se han desarrollado para Android. Estas aplicaciones implementarán los algoritmos que se han descrito anteriormente en la Sección 3.2. No obstante, en la Sección 3.1, se han introducido los parámetros comunes a todos los experimentos, siendo:

- Tiempo de espera: es el tiempo en segundos de espera inicial antes de empezar a realizar el test, ya que los ficheros del sistema tardan cierto tiempo en actualizarse.
- Tiempo de muestreo: indica cada cuantos segundos se tomará cada medida de la batería.
- Número de muestras: indica de cuantas muestras se compondrá cada repetición del test.
- Número de repeticiones: indica cuantas repeticiones se harán el test.

Todas las aplicaciones permiten modificar los parámetros de entrada seleccionando el parámetro en un ComboBox e introduciendo el valor que se desee. De la misma forma, una vez empieza el test correspondiente, el dispositivo móvil mostrará una notificación que presentará en qué punto se encuentra.

Las tres aplicaciones desarrolladas son:

- *CameraTest*: implementa el algoritmo de captura de imágenes (Figura 3.5a).
- *CPUtest*: se basa en el algoritmo descrito previamente de procesamiento de imágenes (Figura 3.5b).



Figura 3.5: Aplicaciones desarrolladas

- *WifiTest*: aplicación que realiza el algoritmo de comunicaciones. Se muestra en la Figura 3.5c.

En las tres aplicaciones se ha seguido la misma estructura de clases:

- *MainActivity*: clase que representa la pantalla principal de las aplicaciones en las que se pueden modificar los parámetros.
- *CameraTest* / *CPUTest* / *WifiTest*: pantallas que contienen toda la lógica de los tests.
- *Utilities*: clase que contienen algunos métodos de utilidad como activar / desactivar un procesador o los datos móviles.

3.4. Herramientas utilizadas

Durante esta sección, se definirán las herramientas que han sido necesarias para el principal objetivo de este TFG, las cuales han sido Android (Subsección 3.4.1), Android Studio (Subsección 3.4.2), OpenCV (Subsección 3.4.3) y MatLab (Subsección 3.4.4).

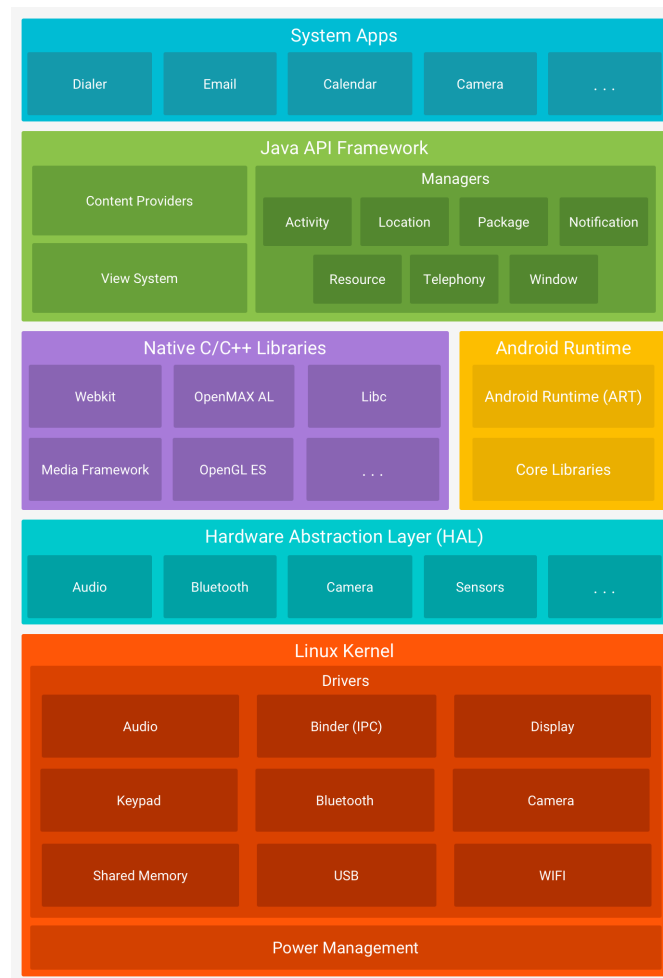


Figura 3.6: Arquitectura de Android [16]

3.4.1. Android

Android es un Sistema Operativo basado en Linux. Se usa principalmente en dispositivos táctiles, como tabletas, móviles o relojes inteligentes, aunque también existen versiones de este sistema operativo para automóviles y televisiones. Fue creado originalmente por la empresa Android, Inc. (que fue fundada en 2003), empresa que más tarde compró Google, en 2005. Android soporta diferentes tamaños de pantalla así como resoluciones.

La arquitectura de Android (Figura 3.6) está formada por los siguientes niveles o capas[16]:

- Aplicaciones. Incluyen todas las aplicaciones que el usuario instala así como las que están por defecto en el dispositivo móvil. Están codificadas con el lenguaje

Java y utilizan los servicios, API y librerías de las capas de más bajo nivel.

- *Framework* (Marco de trabajo) de aplicaciones. Está formado por todas las API con las cuales los desarrolladores pueden crear aplicaciones. Algunas de estas API son el administrador de notificaciones o de actividad.
- Librerías. Estas librerías están escritas en C/C++ a las cuales se puede acceder a través de la API de Android. Un ejemplo de estas librerías es la librería de OpenGL ES, que se utiliza para producir gráficos 2D y 3D.
- Tiempo de ejecución de Android. Está constituida por multitud de clases Java y por la máquina virtual Dalvik, antes de la versión 5.0 de Android. Después de esa versión, se sustituyó por ART.
- Capa de abstracción (). Consiste en varios módulos de librerías, y cada uno de ellos implementa un elemento *hardware* como el *Bluetooth* o la cámara.
- Núcleo Linux. El núcleo de Linux permite a los fabricantes de dispositivos desarrollar controladores de *hardware* para un kernel.

Actualmente existen 15 versiones de Android, siendo la última Android 7.1.2 Nougat.

3.4.2. Android Studio

Android Studio [17] es el entorno de desarrollo integrado (IDE) oficial para el desarrollo de aplicaciones para Android. Entre sus principales características están:

- Dispone de un emulador para probar las aplicaciones sin la necesidad de un dispositivo.
- Poder desarrollar en un mismo entorno aplicaciones para todos los dispositivos Android.
- Soporte a Google Cloud Platform, el servicio de computación en la nube de Google.

Android Studio está disponible para descargar tanto para Windows como Mac OS X y Linux. La última versión estable es la 2.3.2 lanzada en Abril de 2017.

3.4.3. OpenCV

OpenCV [18] es una librería libre de visión artificial. Su primera versión *alpha* fue desarrollada en el año 2000 por la división de investigación de Intel, Intel Research.

Más tarde, en 2006, lanzaron su primera versión oficial. Esta librería tiene multitud de funciones, como:

- Detectar y reconocer caras.
- Identificar objetos.
- Realizar el seguimiento de objetos en movimiento.
- Buscar imágenes parecidas a una base de datos de imágenes.

OpenCV está programado en C++ aunque posee *wrappers* para lenguajes como Python o Java. OpenCV funciona tanto en ordenadores (Windows, Linux, Mac OS X) como en móviles (Android y iOS), siendo la 3.2.0 la última versión estable de la librería.

3.4.4. MatLab

MatLab [19] es un entorno de desarrollo integrado para cálculo numérico y visualización. Se diferencia del resto de entornos de cálculo en su facilidad para trabajar con matrices y vectores. Algunas de sus prestaciones son:

- Manipulación de matrices.
- Representación de datos y funciones.
- Implementación de algoritmos.

MatLab está disponible para Unix, Windows, Mac OS X y Linux y, aunque es un programa de pago, posee una versión de prueba gratuita de 30 días. Su versión más reciente es la R2017a.

Capítulo 4

Pruebas y resultados

Durante este capítulo se mostrarán las pruebas que se han realizado así como los resultados que se han obtenido. En concreto, en este capítulo se hablará:

- En primer lugar (Sección 4.1), se detallarán tanto las características del teléfono móvil para las pruebas, como el algoritmo, usados.
- En la segunda sección (4.2), se explicarán algunos experimentos que se han realizado para obtener un tiempo óptimo de muestreo.
- Por último, se expondrán los resultados obtenidos de las pruebas (Secciones 4.3, 4.4 y 4.5).

4.1. Descripción del entorno de pruebas



(a) Samsung Galaxy S4 Mini
LTE [20]



(b) bq Aquaris X5 Plus[21]

Figura 4.1: Dispositivos móviles utilizados

Los *smartphones* utilizados para los experimentos es un Samsung Galaxy S4 Mini en su versión LTE (Figura 4.1a) y el bq Aquaris X5 Plus (Figura 4.1b). Las características de los móviles se muestran en la Tabla 4.1.

	Samsung Galaxy S4 Mini	bq Aquaris X5 Plus
Pantalla	<i>Super AMOLED</i> de 4.3 pulgadas	IPS de 5 pulgadas
Resolución de la pantalla	960x540px	1920x1080
Procesador	Qualcomm Snapdragon 400 Dual-core (1.7GHz)	Qualcomm Snapdragon 652 Octa-core (1.8GHz)
Frecuencias del procesador (MHz)	384; 486; 594; 702; 810; 918; 1026; 1164; 1242; 1350; 1458; 1566; 1674; 1728	400; 691.2; 806.4; 1017.6; 1190.4; 1305.6; 1382.4; 1401.6
GPU	Qualcomm Adreno 305 a 450MHz	Qualcomm Adreno 510 a 600MHz
Camera Trasera	8MP (3264x2448px) y vídeo a 1080p@30fps	16MP (4608x3456) y vídeo a 4K@30fps
Resoluciones de la cámara	3264x2448; 3264x1836; 2048x1536; 2048x1152; 1280x720; 640x480	4608x3456; 4608x2592; 4160x3120; 4000x3000; 3840x2160; 3456x3456; 3264x2448; 3264x1836; 3200x3200; 3200x2400; 2048x2048; 2048x1536; 2688x1512; 1600x1600; 1920x1080; 1600x1200; 1440x1080; 1280x960; 1280x768; 1280x720; 1200x1200; 1024x768; 800x600; 864x480; 800x480; 720x480; 640x480; 640x360; 480x640; 480x360; 480x320; 352x288; 320x240; 240x320; 176x144; 160x120; 144x176
FPS de la cámara	[4,30]	[15,15]; [10,20]; [20,20]; [10,24]; [24,24]; [10,30]; [15,30]; [30,30]
Wi-Fi	802.11 a/b/g/n	802.11 b/g/n/ac
Batería	1900 mAh	3200 mAh

Tabla 4.1: Especificaciones técnicas de los dispositivos móviles

Por otro lado, el algoritmo utilizado es la detección de personas mediante el procesamiento de imágenes con OpenCV.

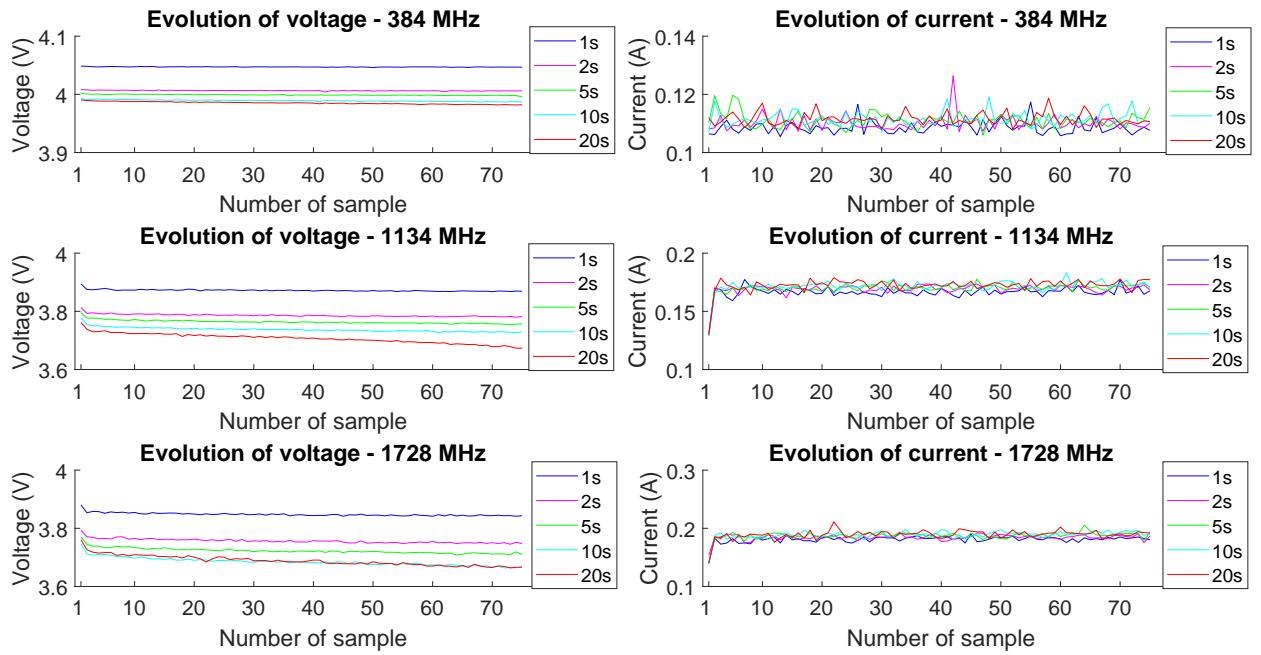


Figura 4.2: Comparación del tiempo de muestreo respecto al tiempo total de repetición

4.2. Tiempo adecuado de muestreo

Para poder encontrar un valor apropiado de cada cuanto tiempo es necesario tomar una muestra con los valores relativos a la batería, se han realizado dos experimentos en el dispositivo Samsung Galaxy S4 Mini utilizando como base el de procesamiento de imágenes:

1. Se fijará un determinado número de muestras (75 muestras) variando el tiempo de muestreo (1s / 2s / 5s / 10s / 20s).

$$T_{TR} = \#_M * T_M \quad (4.1)$$

2. Se fijará un tiempo total de repetición (200 s) variando el tiempo de muestreo (1s / 2s / 5s / 10s / 20s) teniendo, por tanto, 200, 100, 40, 20 y 10 muestras, respectivamente.

$$\#_M = T_{TR}/T_M \quad (4.2)$$

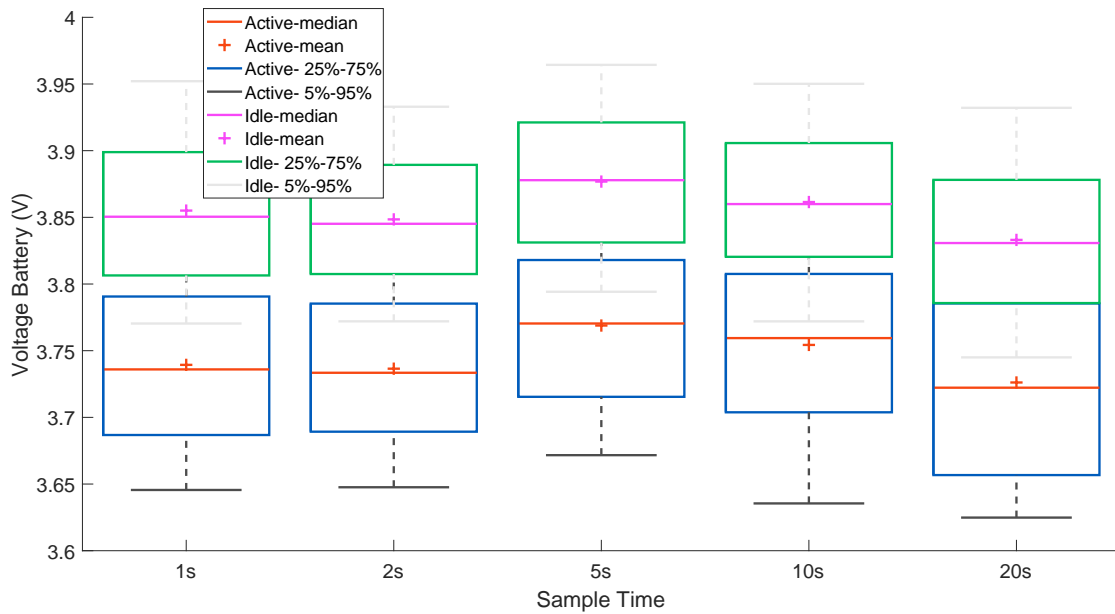


Figura 4.3: Comparación del tiempo de muestreo respecto al número de muestras

En la Figura 4.2, se puede comprobar que, cuando el número de muestras es fijo, el voltaje se va reduciendo. Esto es, porque al durar más los experimentos, los valores del voltaje de la batería son más constantes, por lo que se produce menos error en las medidas. Por esta razón, se puede ver como con un tiempo de muestreo de 1 segundo los valores menos parecidos que con el resto de tiempos.

Por el contrario, se puede ver en la Figura 4.3 como, cuando los tests duran el mismo tiempo, el voltaje de la batería se encuentra en los mismos rangos de valores, independientemente del tiempo de muestreo que se configure.

Por estas razones, se ha escogido un tiempo de muestreo para los experimentos de 5 segundos, con el cual se puede conseguir un error de medida relativamente bajo.

4.3. Resultados del experimento de captura

Configuración

Los valores de los parámetros que se han utilizado en el experimento se muestran en la Tabla 4.2.

	Samsung Galaxy S4 Mini	bq Aquaris X5 Plus
Resoluciones	640x480; 2048x1152; 2048x1536; 3264x2448	4608x3456; 1280x768; 144x176
FPS	[4, 30]	[15,15]; [30,30]
Tiempo de espera (s)	30	
Tiempo de muestreo (s)	5	
Número de muestras	70	
Número de repeticiones	5	
CPU	CPU0: 810 MHz	CPU0: 806.4 MHz
	CPU1 Desactivada	CPU1; CPU2; CPU3; CPU4; CPU5; CPU6; CPU7 desactivadas
Pantalla	Activa	
Brillo	1 %	
WiFi	Desactivado	
Bluetooth	Desactivado	
Datos móviles	Desactivado	

Tabla 4.2: Parámetros para el experimento de captura

Resultados

Samsung Galaxy S4 Mini

Las Figuras 4.4 y 4.5 muestran la comparación de consumo energético comparando entre distintas resoluciones de la cámara e imágenes por segundo (FPS) para el dispositivo Samsung Galaxy S4 Mini. En estas gráficas aparecen tanto el consumo cuando el teléfono se encuentra en estado de espera sin abrir la cámara como el consumo cuando la cámara está encendida con la diferencia de estar capturando imágenes. De esta forma, podemos concluir que:

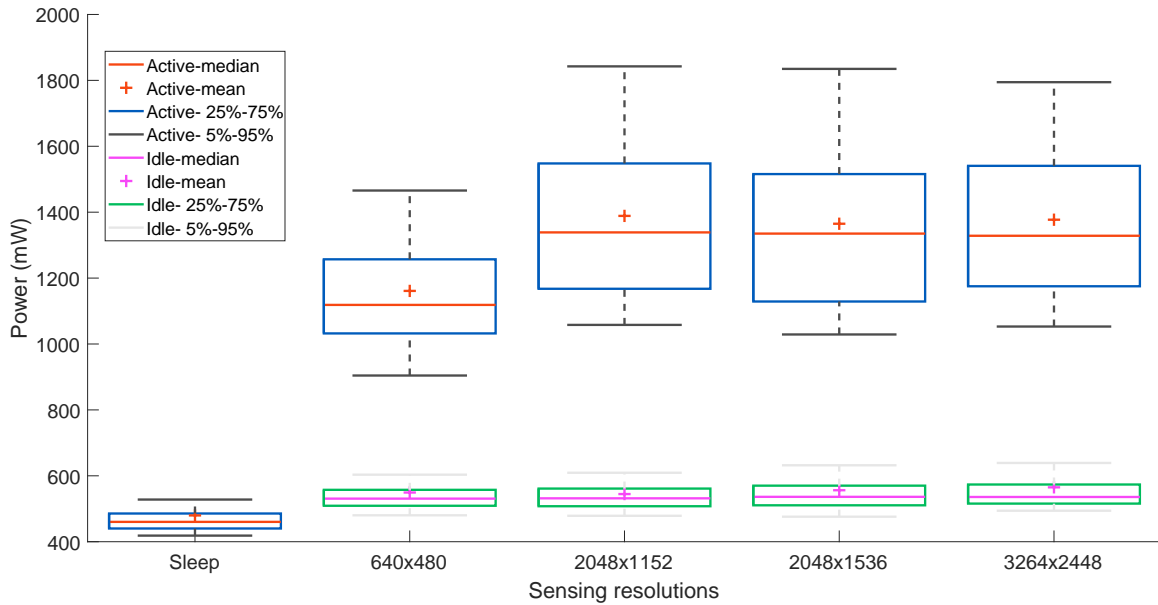


Figura 4.4: Comparativa de consumo por resoluciones en el Samsung Galaxy S4 Mini

- El aumento de energía entre la cámara capturando y sin capturar (con la cámara abierta) es significativo.
- A resoluciones altas el consumo es similar.
- Cuando la cámara está abierta, el consumo incrementa un 20 % aproximadamente respecto a cuando esta no ha abierto.
- El gasto de energía es parecido cuando no se están capturando imágenes, independientemente de la resolución configurada.

bq Aquaris X5 Plus

En las Figuras 4.6 y 4.7 se muestran los modelos obtenidos del consumo energético para el dispositivo bq Aquaris X5 Plus. De los resultados se pueden obtener los siguientes datos:

- El aumento de resolución implica un aumento del consumo.
- El número de imágenes por segundo que se capturan no influye prácticamente en el gasto de energía.

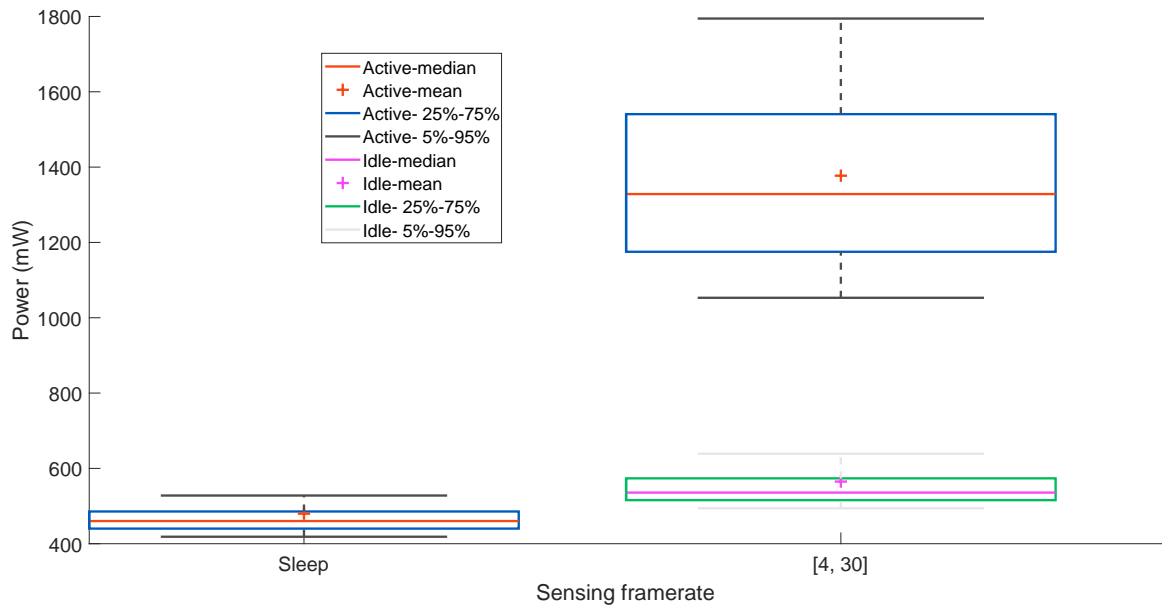


Figura 4.5: Comparativa de consumo por imágenes por segundo en el Samsung Galaxy S4 Mini

- El consumo energético cuando la cámara solo está encendida sin capturar imágenes es independiente de la resolución y el número de imágenes por segundo.
- El coste de abrir la cámara es relativamente bajo.

4.4. Resultados del experimento de procesamiento

En este experimento se mide el consumo energético asociado a la CPU, en el cual se va variando la frecuencia de reloj del primer núcleo y realizando tareas para que la CPU esté ocupada. De esta forma, cabría que esperar que a mayor frecuencia, el consumo será mayor; al igual que, cuando la CPU se encuentra activa, también será mayor.

Configuración

Los valores de los parámetros que se han utilizado en el experimento se muestran en la Tabla 4.3.

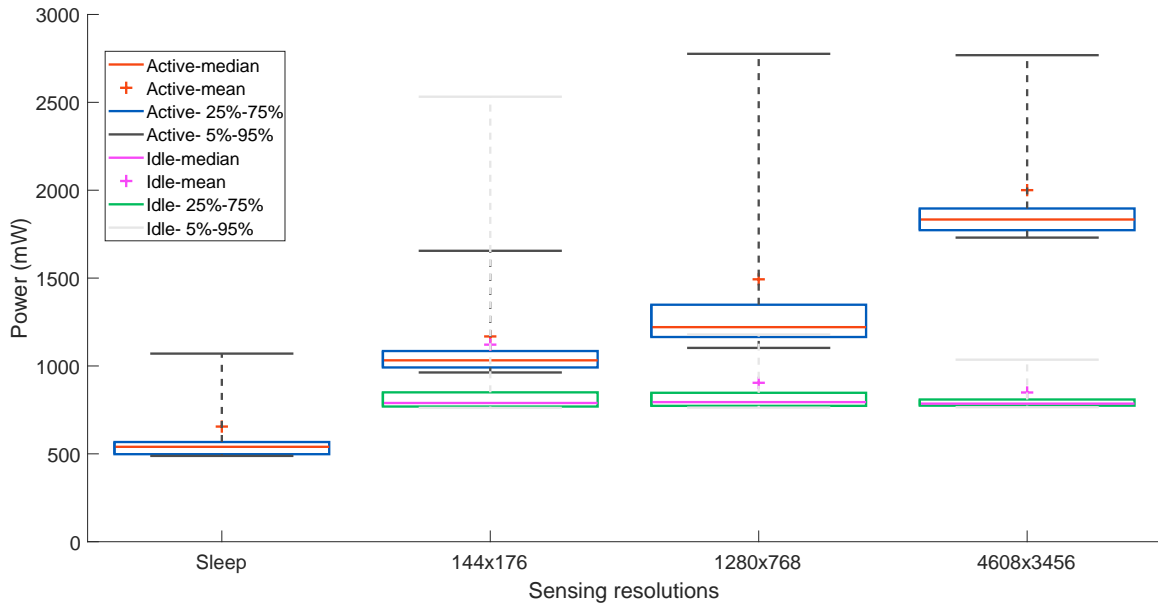


Figura 4.6: Comparativa de consumo por resoluciones en el bq Aquaris X5 Plus

	Samsung Galaxy S4 Mini	bq Aquaris X5 Plus
Frecuencias CPU0 (MHz)	384; 810; 1134; 1458; 1728	400; 806.4; 1190.4; 1305.6; 1401.6
Tiempo de espera (s)	30	
Tiempo de muestreo (s)	5	
Número de muestras	70	
Número de repeticiones	5	
Resto de CPUs	Desactivadas	
Pantalla	Activa	
Brillo	1 %	
WiFi	Desactivado	
Bluetooth	Desactivado	
Datos móviles	Desactivado	

Tabla 4.3: Parámetros para el experimento de procesamiento

Por otro lado, la imagen que se va a procesar durante el experimento es la que se muestra en la Figura 4.8:

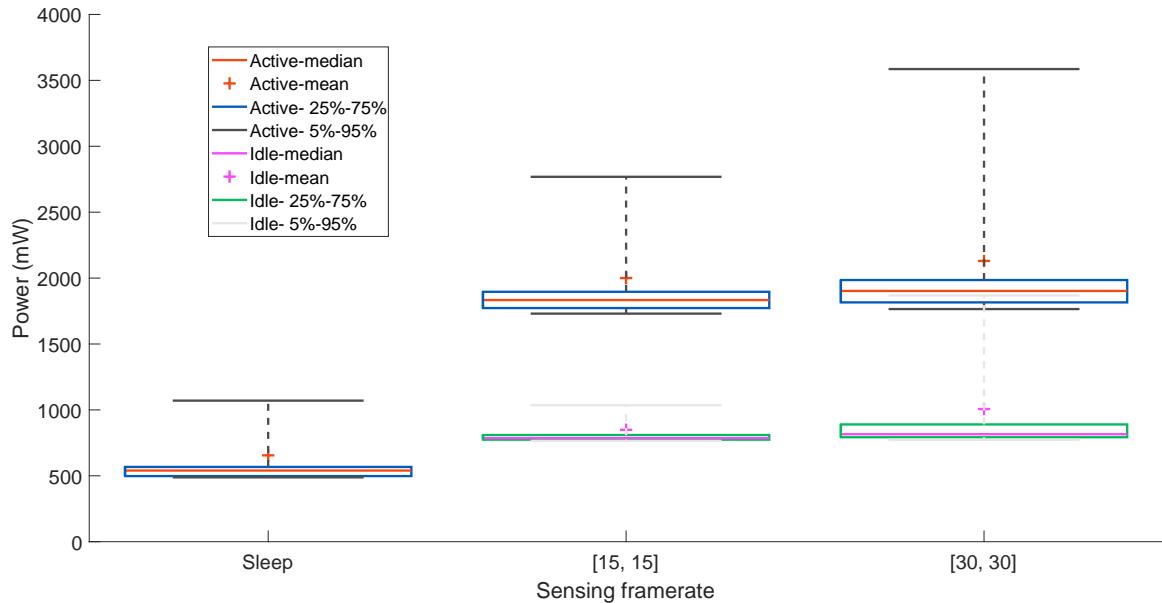


Figura 4.7: Comparativa de consumo por imágenes por segundo en el bq Aquaris X5 Plus

Resultados

El resultado de la imagen procesada con el algoritmo de detección de personas se muestra en la Figura 4.9.

Samsung Galaxy S4 Mini

Una vez realizadas las pruebas en el primer dispositivo, se puede comprobar en la Figura 4.10 que esto se cumple. Además, otros asuntos relevantes relativos a las pruebas son:

- A la menor frecuencia (384 MHz), el consumo cuando la CPU está activa o en espera es similar.
- Cuando la CPU está en espera, el consumo cuando está a la máxima frecuencia (1728 MHz) es aproximadamente un 50 % mayor que a la menor frecuencia.
- Cuando está en modo activo, el consumo aumenta un 200 % respecto a la menor frecuencia.
- Por el contrario, en el modo de espera, prácticamente el gasto es igual cuando trabaja a frecuencias bajas, mientras que a frecuencias altas el consumo aumenta de forma lineal.



Figura 4.8: Ejemplo de imagen de personas [22]

- De la misma forma el consumo en modo activo también aumenta de forma lineal.

bq Aquaris X5 Plus

Tras realizar las pruebas en el bq Aquaris X5 Plus (Figura 4.11), se puede ver que:

- Cuando el móvil se encuentra en modo activo, el consumo aumenta conforme crece la frecuencia del procesador.
- Si la CPU está en espera, el consumo es similar independientemente de la velocidad del procesador.
- Respecto al modo activo, el consumo ha aumentado cerca de un 30 % entre la menor y la mayor frecuencia.

4.5. Resultados del experimento de comunicaciones

Con este experimento se quiere cuantificar el impacto del intercambio de paquetes a través de la conexión Wi-Fi. Para ello, se medirá el coste de la conexión Wi-Fi en modo de espera y al enviar y recibir datos. Este experimento se repetirá con distintos

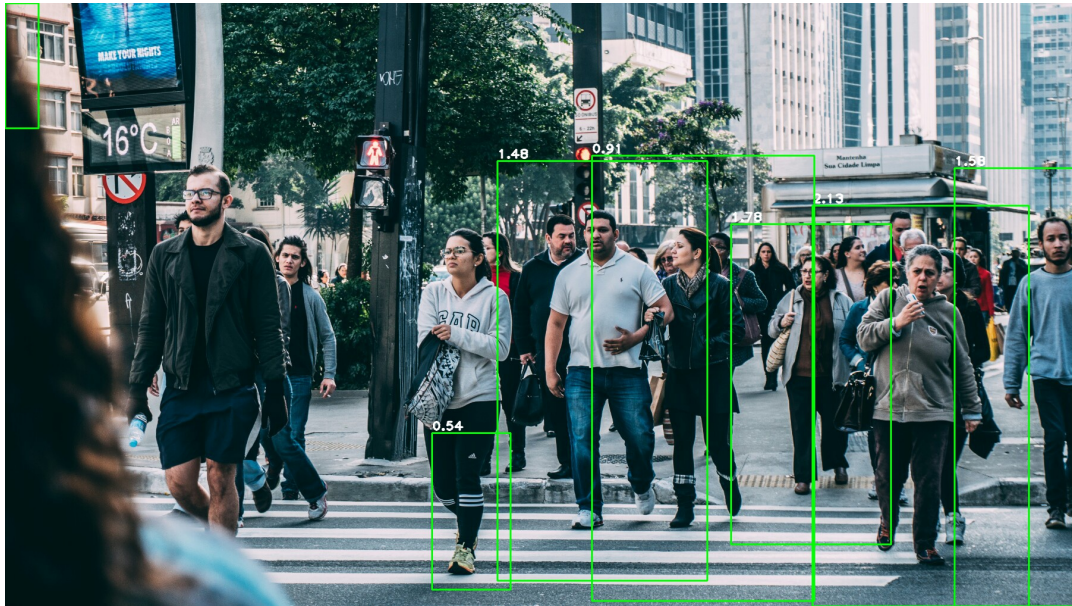


Figura 4.9: Imagen procesada utilizando el algoritmo de detección de personas

límites de ancho de banda, siendo estos de 10, 100 y 1000 kbps , así como con el límite desactivado.

Configuración

Los valores de los parámetros que se han utilizado en el experimento se muestran en la Tabla 4.4.

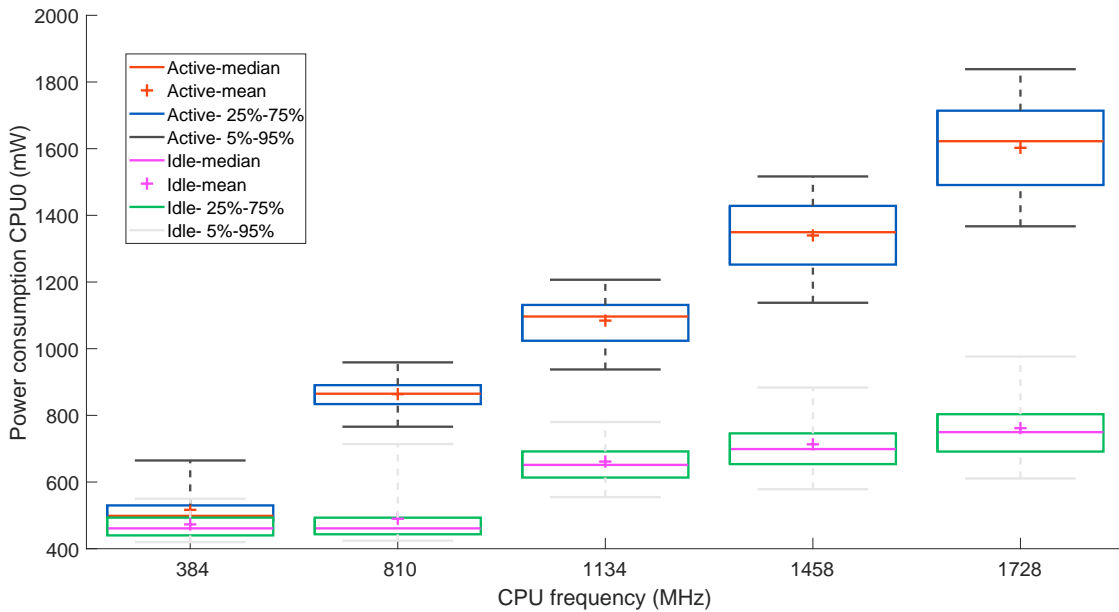


Figura 4.10: Resultados del experimento de procesamiento en el Samsung Galaxy S4 Mini

	Samsung Galaxy S4 Mini	bq Aquaris X5 Plus
WiFi	Activado	
Tamaño de paquete	50000 Bytes	
Límite de ancho de banda (kbps)	10; 100; 1000; Desactivado	
Tiempo de espera (s)	30	
Tiempo de muestreo (s)	5	
Número de muestras	70	
Número de repeticiones	5	
CPU	CPU0: 810 MHz	CPU0: 806.4 MHz
	CPU1 Desactivada	CPU1; CPU2; CPU3; CPU4; CPU5; CPU6; CPU7; desactivadas
Pantalla	Activa	
Brillo	Mínimo	
Bluetooth	Desactivado	
Datos móviles	Desactivado	

Tabla 4.4: Parámetros para el experimento de comunicaciones

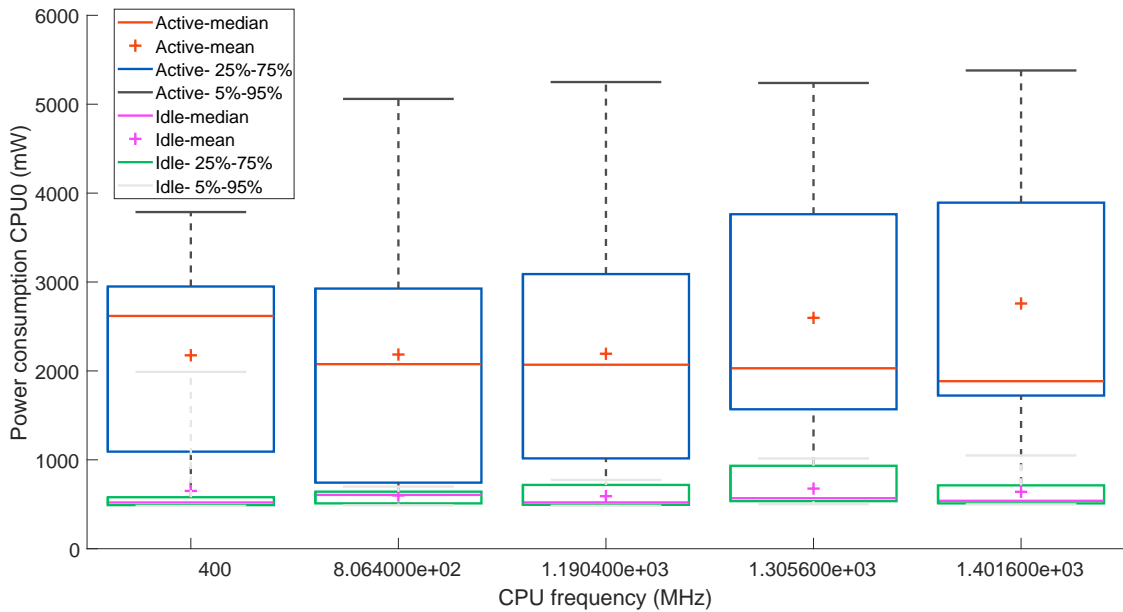


Figura 4.11: Resultados del experimento de procesamiento en el bq Aquaris X5 Plus

Resultados

Samsung Galaxy S4 Mini

El consumo obtenido en el dispositivo Samsung Galaxy S4 Mini se muestra en la Figura 4.12. Algunas apreciaciones son:

- Por lo general, la conexión Wi-Fi consume más energía al transmitir paquetes, que al recibir, excepto cuando el ancho de banda es pequeño (10 kbps).
- Conforme el ancho de banda va aumentando, el consumo también aumenta.

bq Aquaris X5 Plus

En la Figura 4.13 se muestra el modelo obtenido del experimento de comunicaciones en el dispositivo bq Aquaris X5 Plus. En este caso, se pueden ver algunas particularidades:

- Al contrario que con el dispositivo de Samsung Galaxy, con el bq Aquaris en todos los casos siempre se consume más energía cuando se envían paquetes.
- También, por lo general cuanto mayor es el ancho de banda, mayor es el consumo.

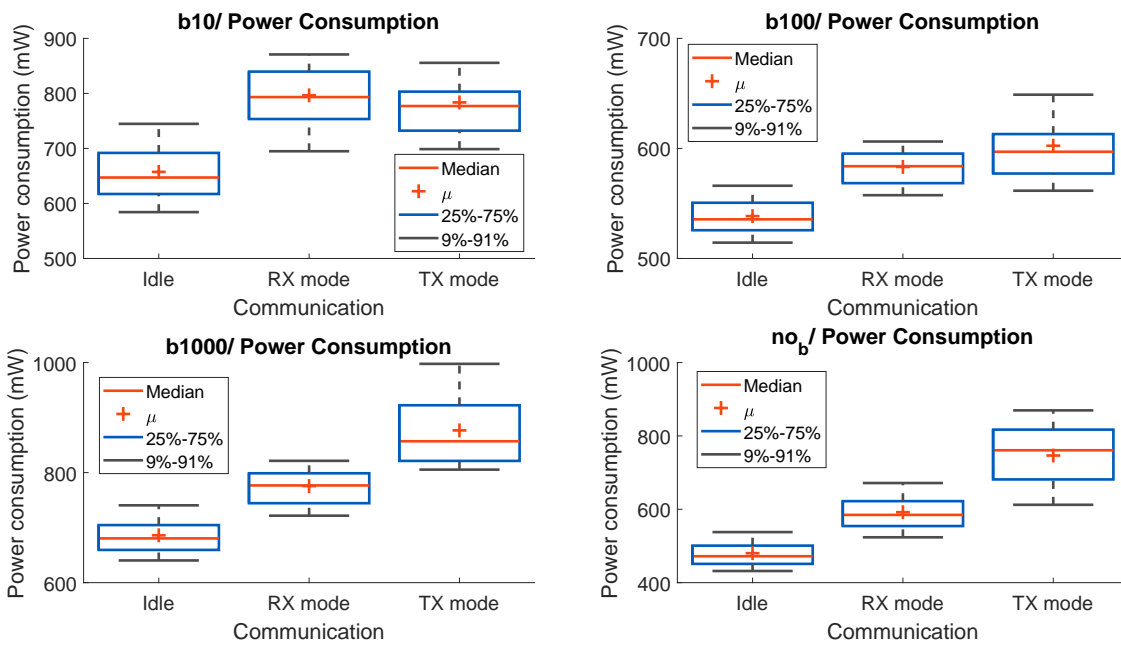


Figura 4.12: Resultados del experimento de comunicaciones en el Samsung Galaxy S4 Mini

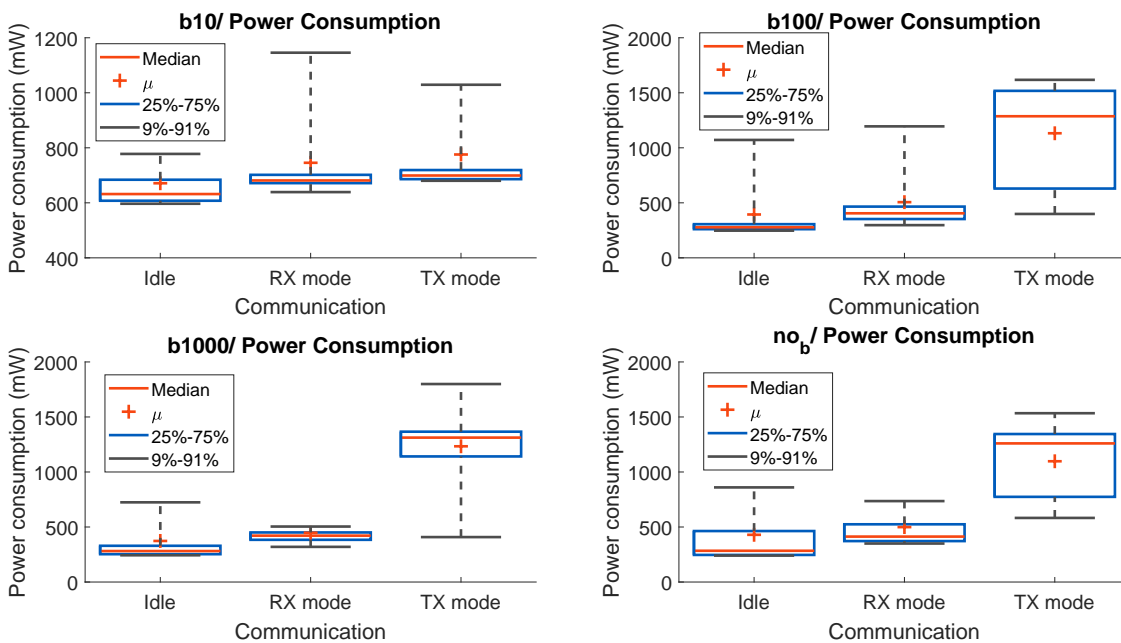


Figura 4.13: Resultados del experimento de comunicaciones en el bq Aquaris X5 Plus

Capítulo 5

Conclusiones y trabajo futuro

5.1. Conclusiones

Como se ha explicado al principio del documento, el objetivo principal de este trabajo era el análisis energético sobre distintos módulos en Android. Estos módulos en los que se ha dividido el dispositivo móvil han sido el de captura (cámara), procesamiento (procesador) y comunicaciones (Wi-Fi).

En primer lugar, se han relatado distintos estudios que se han realizado en el pasado, tanto para cámaras inteligentes como para Android. Posteriormente, se ha definido un experimento para cada módulo que permita extraer medidas energéticas de la batería para su posterior análisis. Para ello, se han variado distintos parámetros para conocer su influencia sobre la batería. Una vez definidos los experimentos se han desarrollado aplicaciones que los implementan. Por último, a partir de los resultados obtenidos de las aplicaciones, se han diseñado modelos por medio de la herramienta MatLab.

A la vista de los resultados en los dos dispositivos, se puede concluir que de los tres módulos en los que se ha dividido el dispositivo móvil el que consume un mayor gasto de energía es el módulo de procesamiento, teniendo el módulo de captura y de comunicaciones un consumo similar. No obstante, sería posible reducir el consumo energético al nivel de los otros módulos disminuyendo la velocidad del procesador o reduciendo su carga de trabajo.

5.2. Trabajo futuro

Tras la realización de este trabajo aparecen varias líneas de trabajo futuro:

- Incluir más módulos del dispositivo móvil en el análisis energético, como la

pantalla o el GPS.

- Realizar el experimento con otros dispositivos móviles tanto Android como con otros Sistemas Operativos con el fin de comparar los resultados.
- Tomando como referencia los resultados del análisis, realizar una aplicación para desarrolladores que permita predecir el consumo energético de su aplicación, con el objetivo de optimizarla.

Bibliografía

- [1] Drake Baer. This map shows the percentage of people around the world who own smartphones - <http://www.businessinsider.com/how-many-people-own-smartphones-around-the-world-2016-2>, 2016. **1**
- [2] Pawel Piejko. 16 mobile market statistics you should know in 2016 - <https://deviceatlas.com/blog/16-mobile-market-statistics-you-should-know-2016>, 2016. **1**
- [3] Jeff Sharkey. Coding for life—battery life, that is. In *Google IO 2009*, 2009. **2**
- [4] Panasonic’s new camera tech can read heart rate - <http://www.ubergizmo.com/2017/02/panasonic-camera-tech-read-heart-rate/>. **2**
- [5] Will Kelly. Computer vision and the future of mobile devices - <http://www.techrepublic.com/article/computer-vision-and-the-future-of-mobile-devices/>, 2014. **3**
- [6] Juan Carlos San Miguel; Andrea Cavallaro. Energy consumption models for smart-camera networks. *IEEE*, 2016. **7, 8, 13**
- [7] Mauricio Casares; Alvaro Pinto; Youlu Wang; Senem Velipasalar. Power consumption and performance analysis of object tracking and event detection with wireless embedded smart cameras. *ResearchGate*, 2009. **7, 8**
- [8] Robert LiKamWa, Bodhi Priyantha, Matthai Philipose, Lin Zhong, and Paramvir Bahl. Energy characterization and optimization of image sensing toward continuous mobile vision. 2013. **8**
- [9] Shin NAKAJIMA. Model-based power consumption analysis of smartphone applications. *National Institute of Informatics*, 2013. **9, 10**

- [10] Yasir Malik. Power consumption analysis of a modern smartphone. *Research-Gate*, 2014. 10, 11, 13
- [11] Depu zhou Abdelhamid Mammeri, Azzedine Boukerche. Evaluation of the power consumption of image descriptors on smartphone platforms. *School of Electrical Engineering and Computer Science, University of Ottawa*, pages 119–124, 2016. 11
- [12] Karel De Vogeleer, Gerard Memmi, Pierre Jouvelot, and Fabien Coelho. The energy/frequency convexity rule: Modeling and experimental validation on mobile devices. 2014. 11
- [13] Polímetro análogo - https://www.amazon.es/salki-8500213-Salki-Multimetro-analogico/dp/B011035WLW/ref=sr_1_3?ie=UTF8&qid=1495302619&sr=8-3&keywords=multimetro+analogico. 12
- [14] Polímetro digital - <http://electronicaroman.com/multimetros/892-multimetro-polimetro-digital-con-capacimetro-ts-34.html>. 12
- [15] Medidor usb - <https://es.aliexpress.com/item/New-LCD-USB-Charger-Capacity-Current-Voltage-Tester-Meter-For-Cell-Phone-32643296893.html?spm=2114.43010208.4.98.3q0jRx>. 14
- [16] Arquitectura de la plataforma - <https://developer.android.com/guide/platform/index.html?hl=es-419>. 22
- [17] Android studio - <https://developer.android.com/studio/index.html>. 23
- [18] OpenCV - About (<http://opencv.org/about.html>). 23
- [19] Matlab - <https://es.wikipedia.org/wiki/MATLAB>. 24
- [20] Samsung galaxy s4 mini - <https://es.webuy.com/product.php?sku=SSAMI91958GNLIBA#.WSCPx2jyjcc>. 25
- [21] Móvil - bq aq x5 plus, 5", 16gb, 4g, dual sim, negro - <https://tiendas.mediamarkt.es/p/movil-bq-aq-x5-plus-16gb-4g-dual-si-negro-1335781>. 25

- [22] Kaique Rocha. People walking on pedestrian lane during daytime - <https://www.pexels.com/photo/people-brasil-guys-avpaulista-109919/>. 34

Apéndice A

Consumo energético y potencia media por test y repetición

En este apéndice se mostrarán gráficos que se han obtenido acerca del consumo y potencia media realizando cada test.

Captura de imágenes

Samsung Galaxy S4 Mini

La Figura [A.1](#) se corresponde al modo activo sin capturar imágenes, mientras que la Figura [A.2](#) al modo activo capturando.

bq Aquaris X5 Plus

Las Figuras [A.3](#) y [A.4](#) se corresponden al modo activo sin capturar imágenes, mientras que las Figura [A.5](#) y [A.6](#) al modo activo capturando.

Procesamiento de imágenes

Samsung Galaxy S4 Mini

La Figura [A.7](#) se corresponde al modo en espera sin realizar trabajo, mientras que la Figura [A.8](#) al modo activo realizando trabajo.

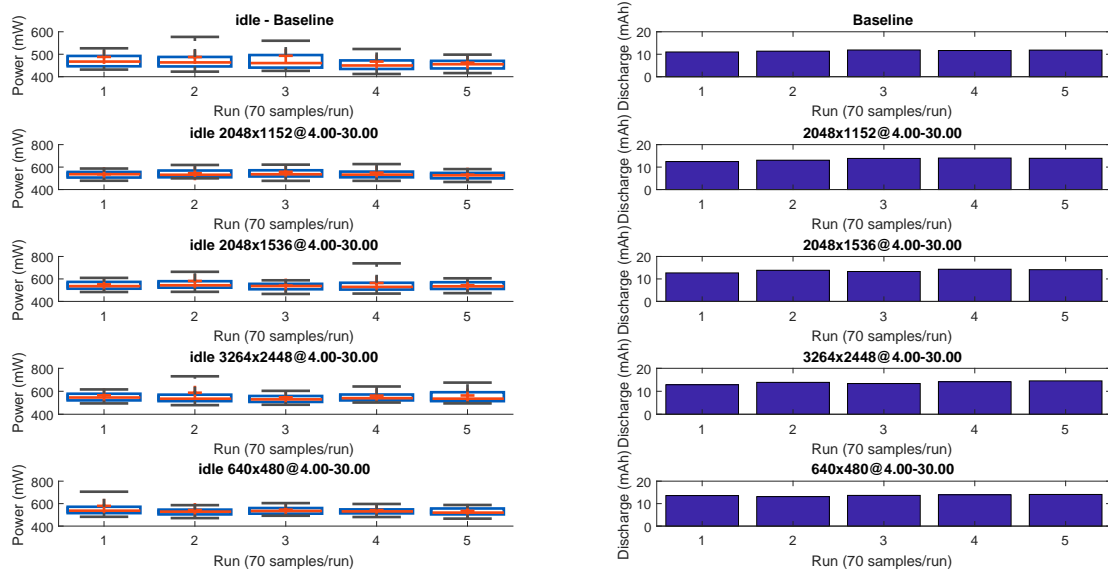


Figura A.1: Modo activo con la cámara sin capturar imágenes para el Samsung Galaxy S4 Mini

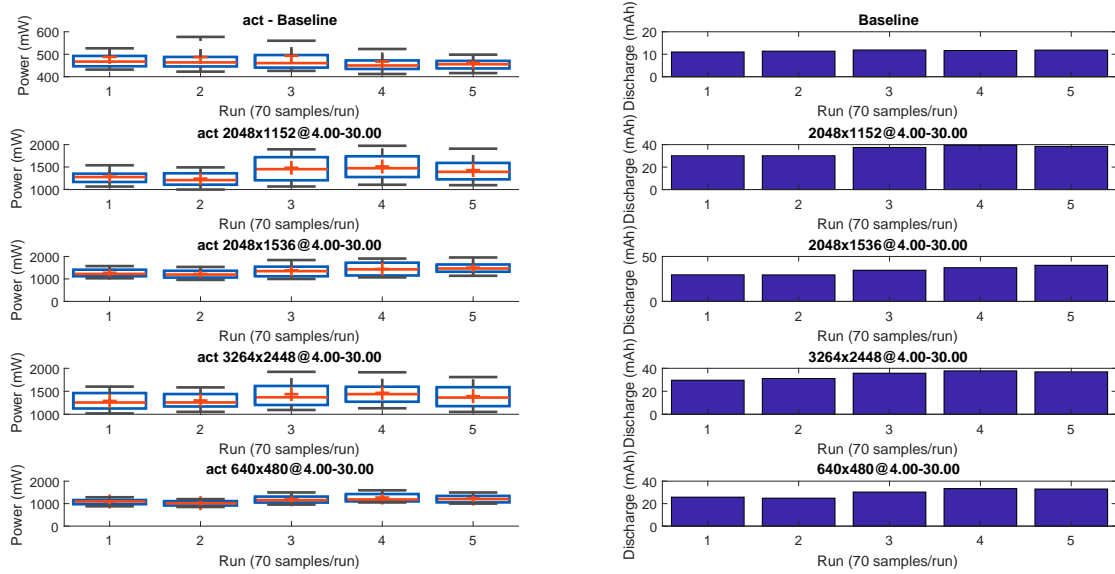


Figura A.2: Modo activo con la cámara capturando imágenes para el Samsung Galaxy S4 Mini

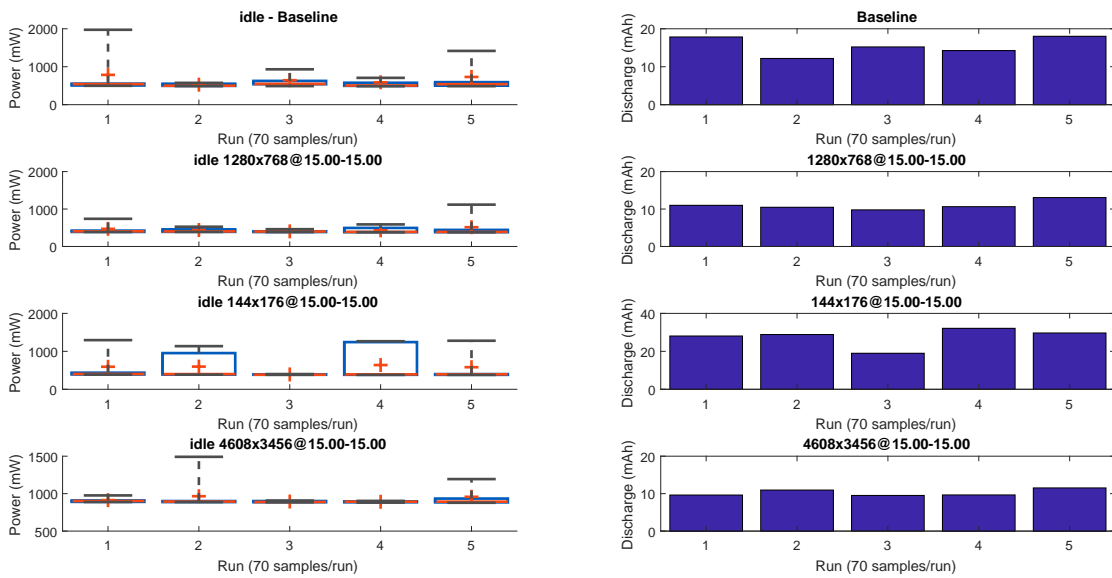


Figura A.3: Modo activo con la cámara sin capturar imágenes para el bq Aquaris X5 Plus (I)

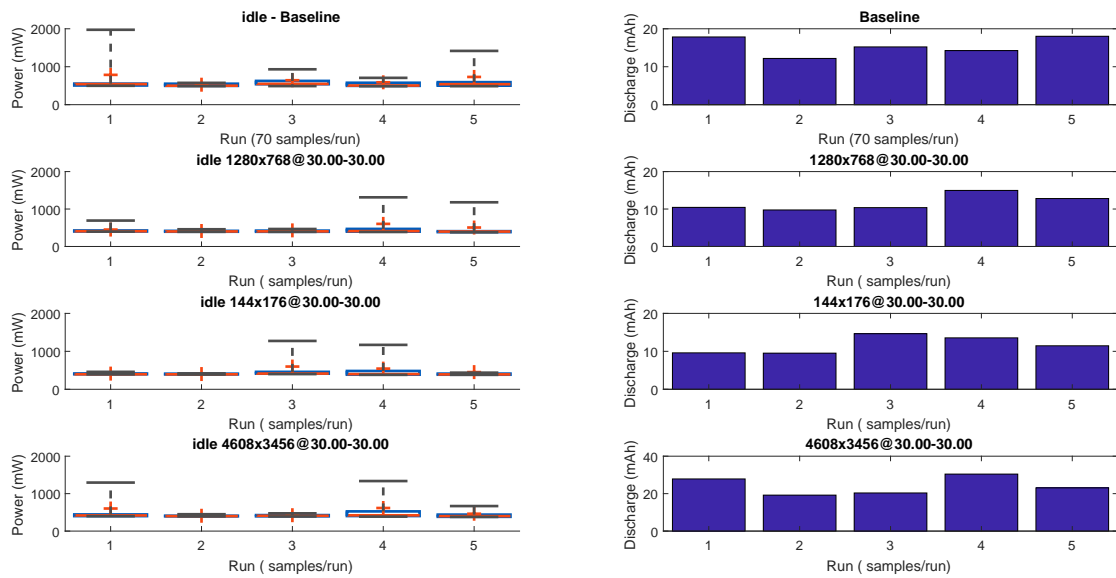


Figura A.4: Modo activo con la cámara sin capturar imágenes para el bq Aquaris X5 Plus (II)

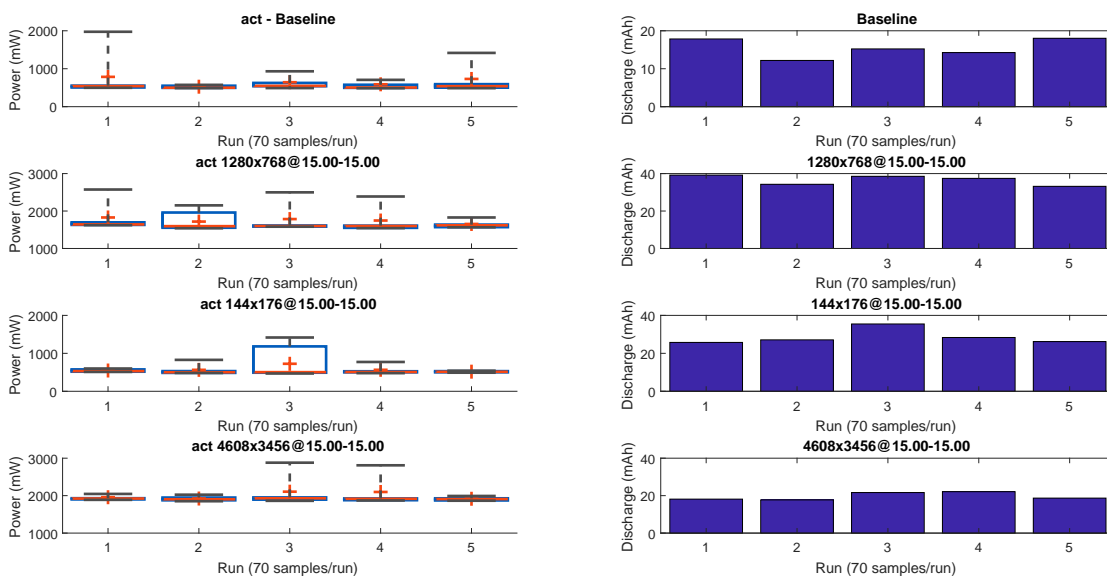


Figura A.5: Modo activo con la cámara capturando imágenes para el bq Aquaris X5 Plus (I)

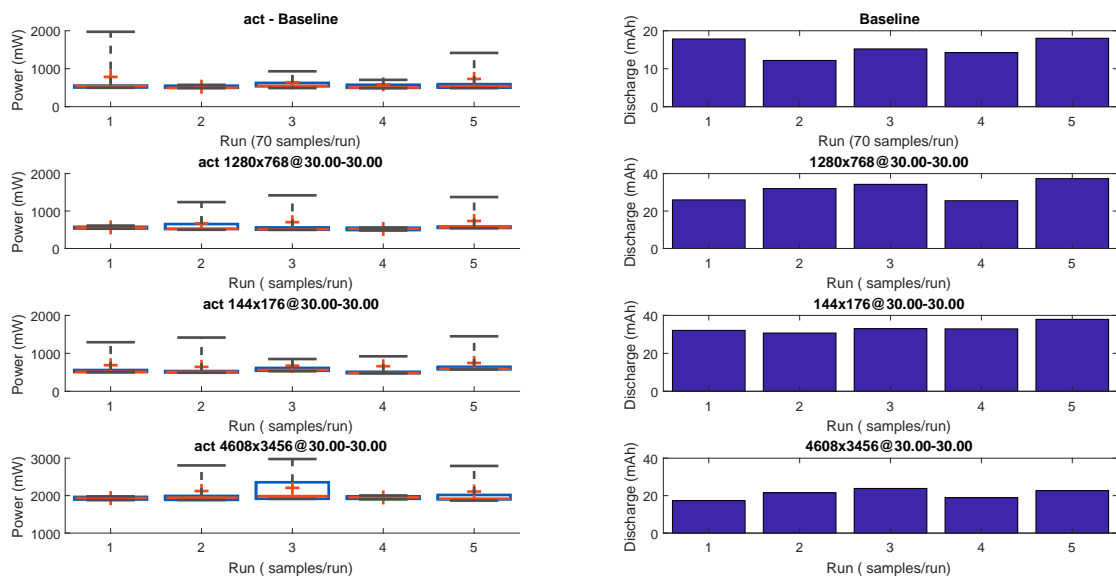


Figura A.6: Modo activo con la cámara capturando imágenes para el bq Aquaris X5 Plus (II)

52 APÉNDICE A. CONSUMO ENERGÉTICO Y POTENCIA MEDIA POR TEST Y REPETICIÓN

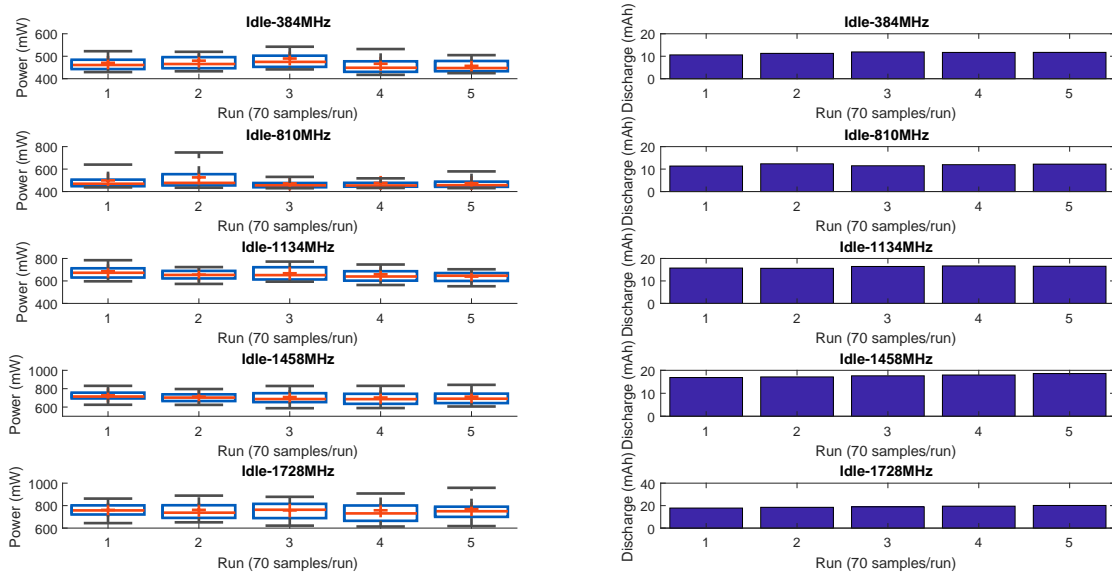


Figura A.7: Modo en espera con la CPU sin realizar trabajo para el Samsung Galaxy S4 Mini

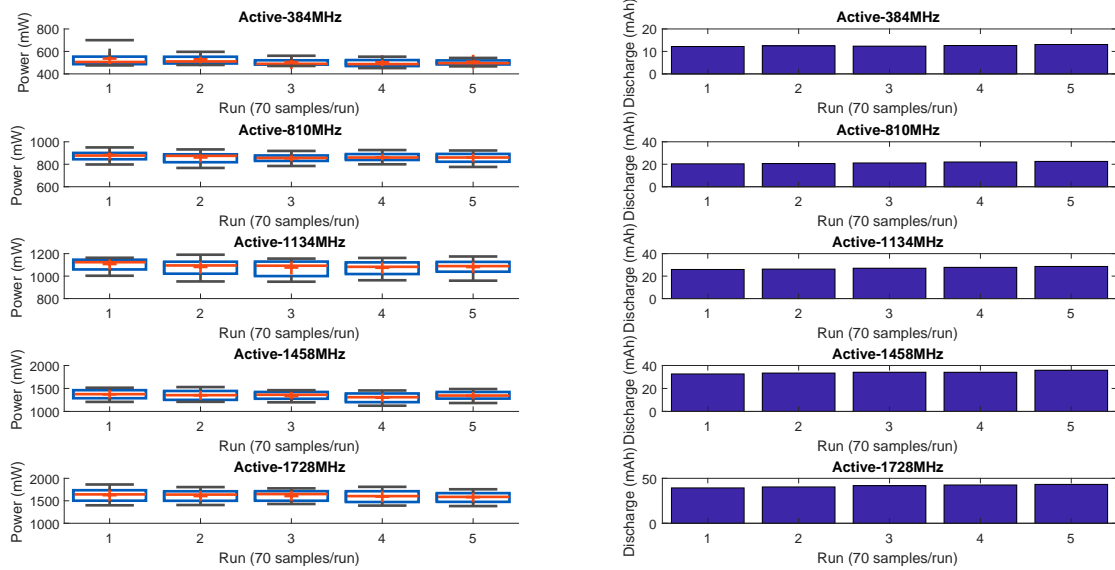


Figura A.8: Modo activo con la CPU procesando imágenes para el Samsung Galaxy S4 Mini

bq Aquaris X5 Plus

La Figura A.9 se corresponde al modo en espera sin realizar trabajo, mientras que la Figura A.10 al modo activo realizando trabajo.

Comunicaciones

Samsung Galaxy S4 Mini

Las Figuras A.11, A.12, A.13 y A.14 se corresponden con los resultados al experimento con un límite de ancho de banda de 10 kbps, 100 kbps, 1000 kbps y sin límite,

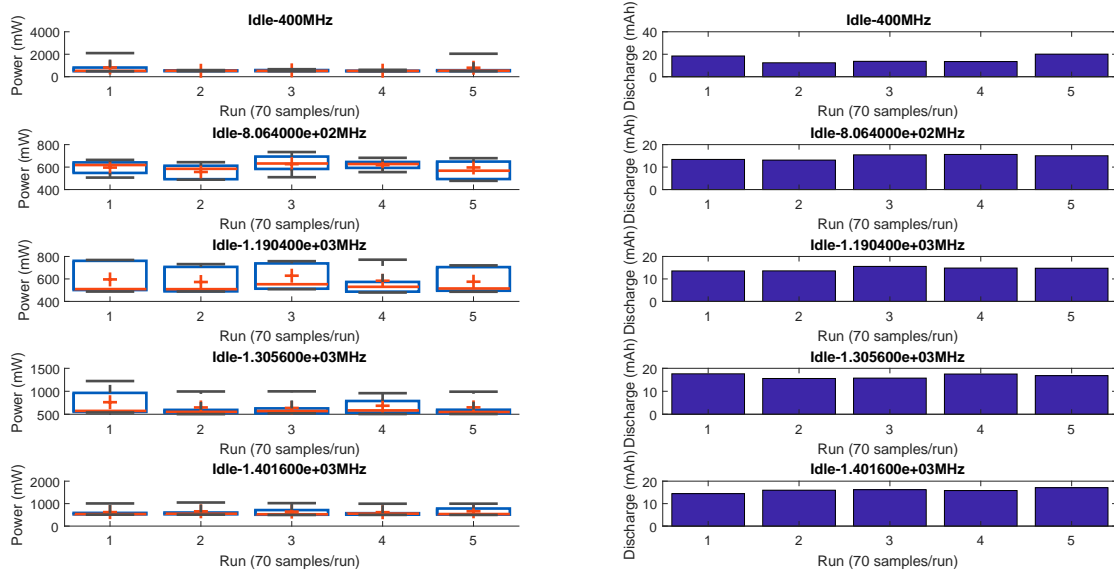


Figura A.9: Modo en espera con la CPU sin realizar trabajo para el bq Aquaris X5 Plus

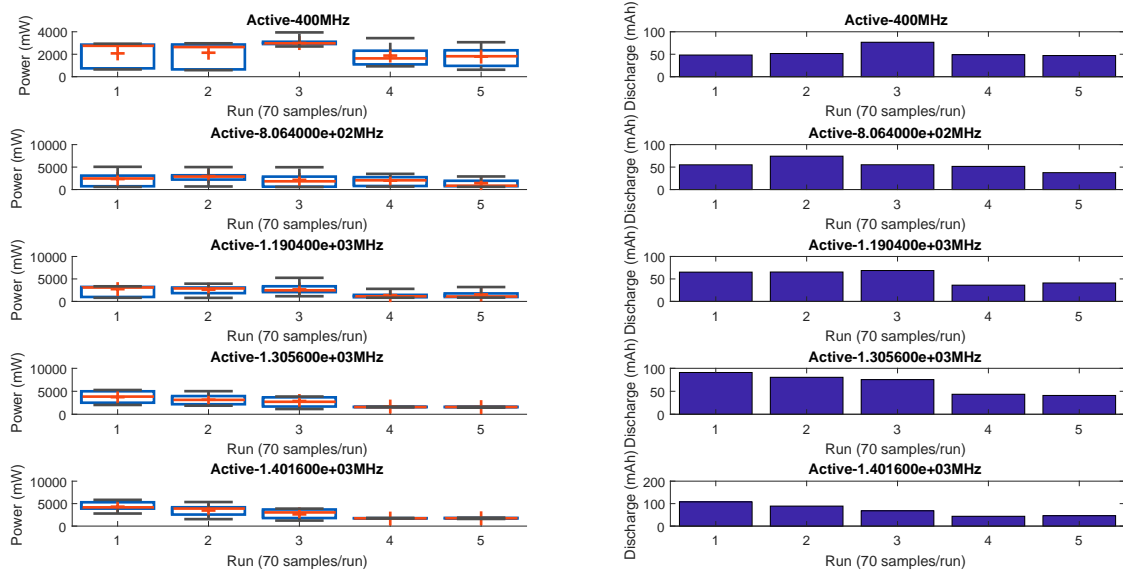


Figura A.10: Modo activo con la CPU procesando imágenes para el bq Aquaris X5 Plus

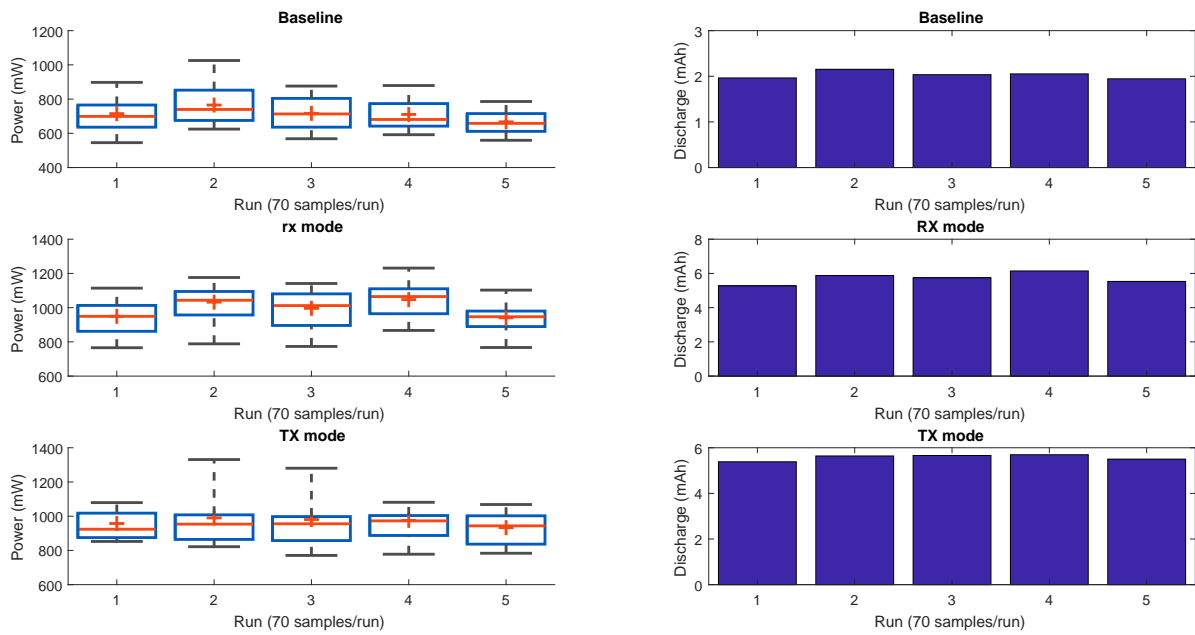


Figura A.11: Resultados del experimento con límite de 10 kbps de ancho de banda para el Samsung Galaxy S4 Mini

respectivamente.

bq Aquaris X5 Plus

Las Figuras A.15, A.16, A.17 y A.18 se corresponden con los resultados al experimento con un límite de ancho de banda de 10 kbps, 100 kbps, 1000 kbps y sin límite, respectivamente.

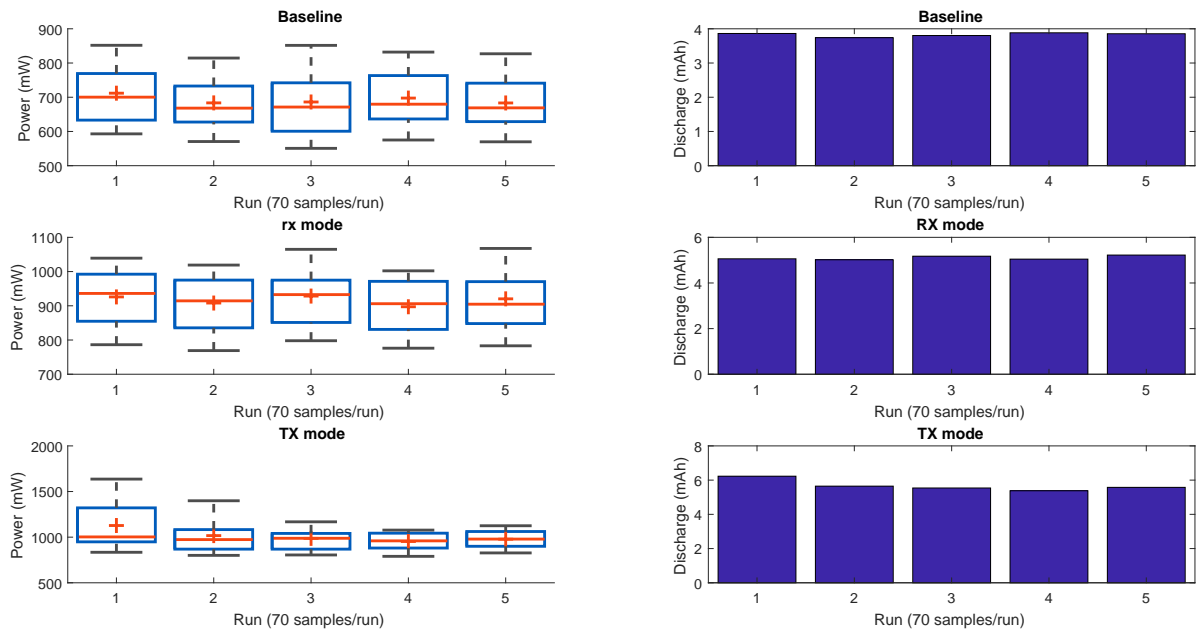


Figura A.12: Resultados del experimento con límite de 100 kbps de ancho de banda para el Samsung Galaxy S4 Mini

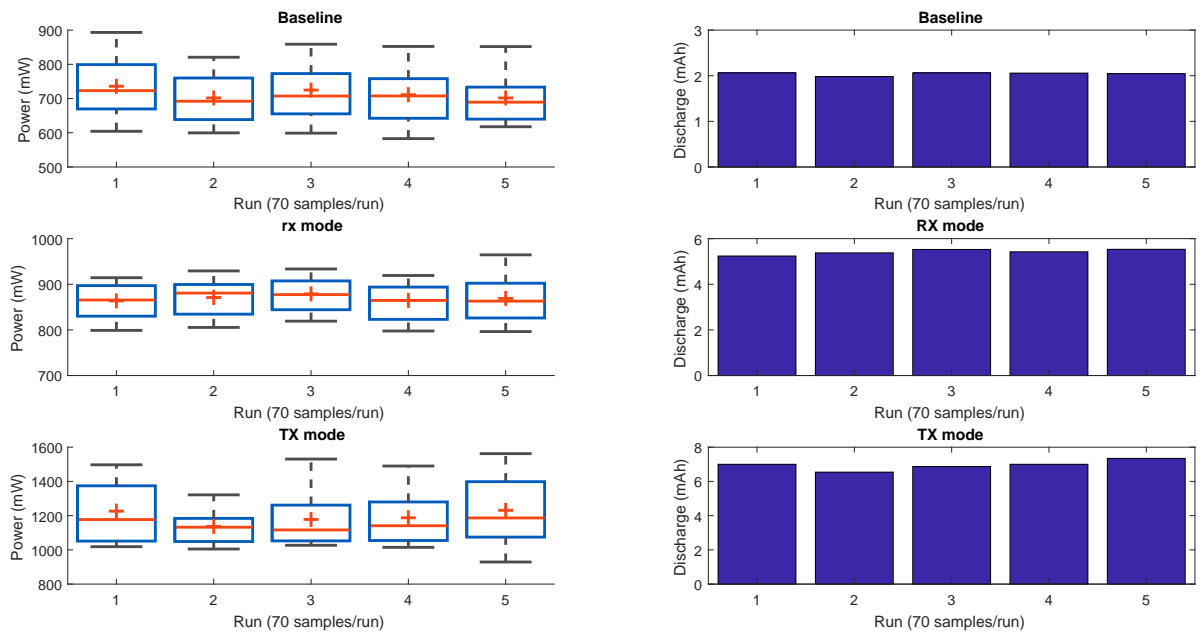


Figura A.13: Resultados del experimento con límite de 1000 kbps de ancho de banda para el Samsung Galaxy S4 Mini

58 APÉNDICE A. CONSUMO ENERGÉTICO Y POTENCIA MEDIA POR TEST Y REPETICIÓN

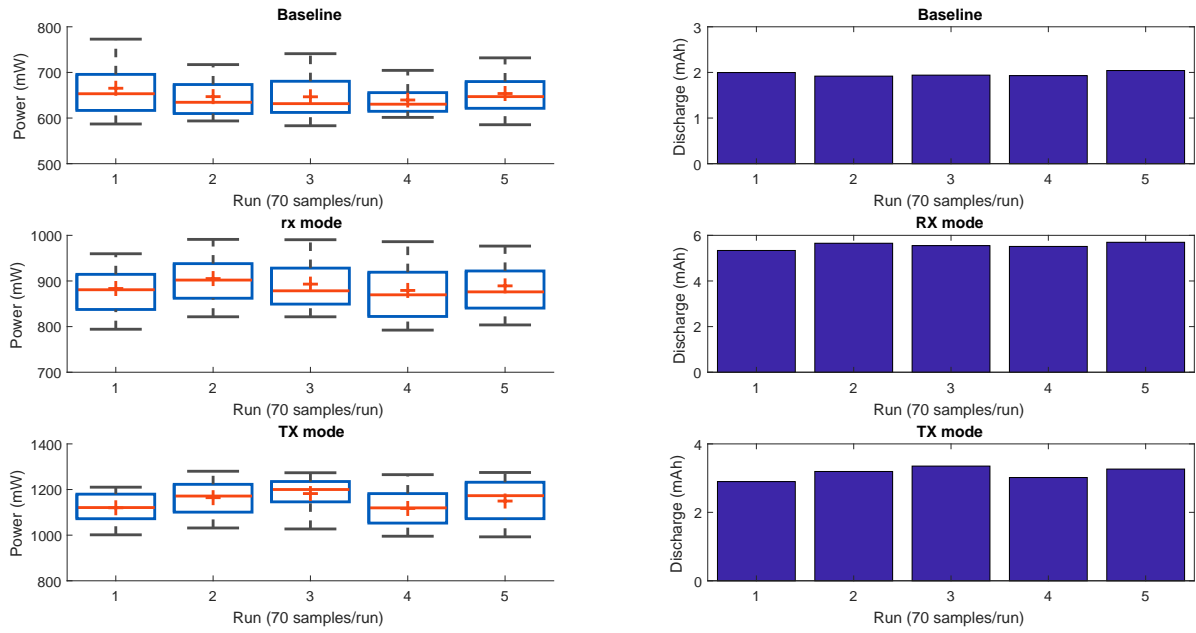


Figura A.14: Resultados del experimento sin límite de ancho de banda para el Samsung Galaxy S4 Mini

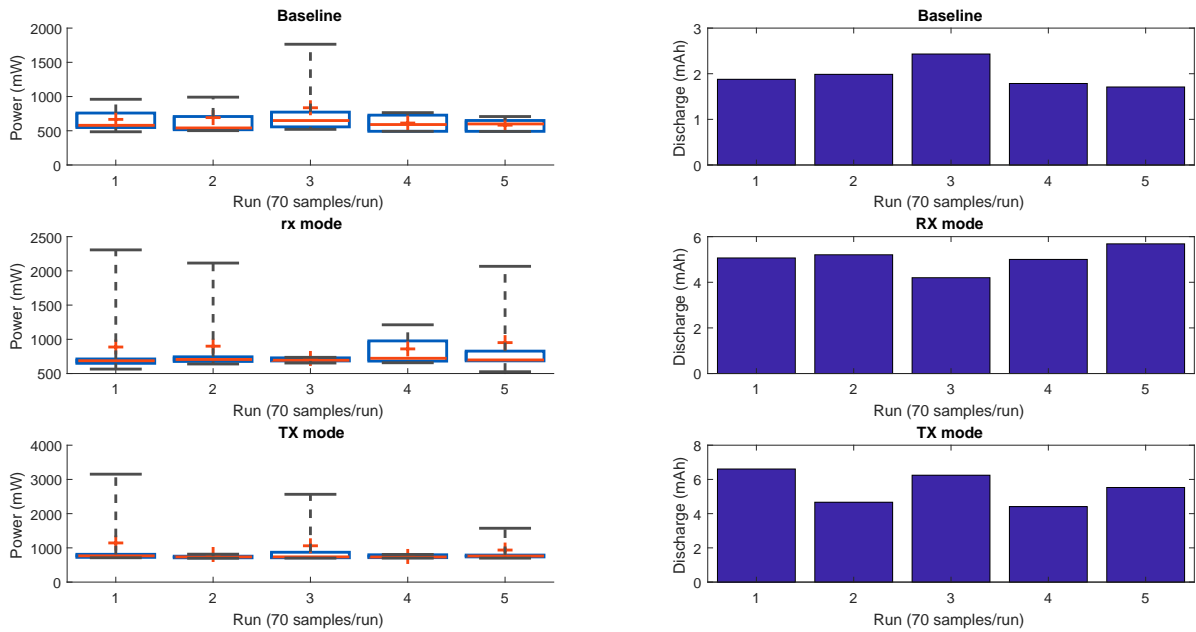


Figura A.15: Resultados del experimento con límite de 10 kbps de ancho de banda para el bq Aquaris X5 Plus

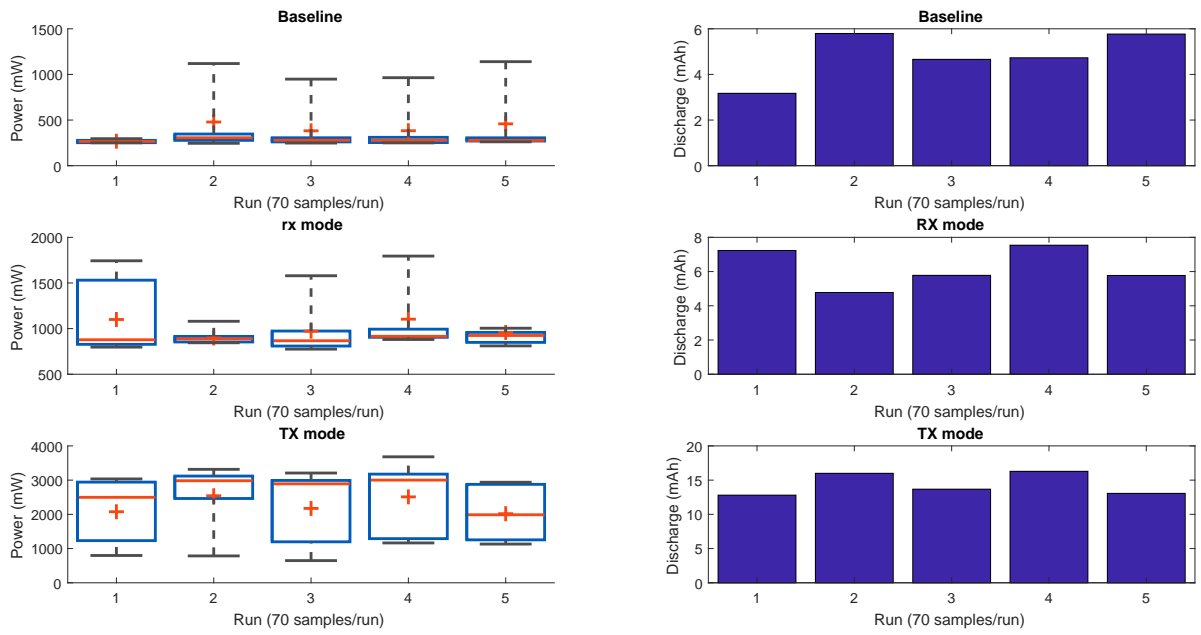


Figura A.16: Resultados del experimento con límite de 100 kbps de ancho de banda para el bq Aquaris X5 Plus

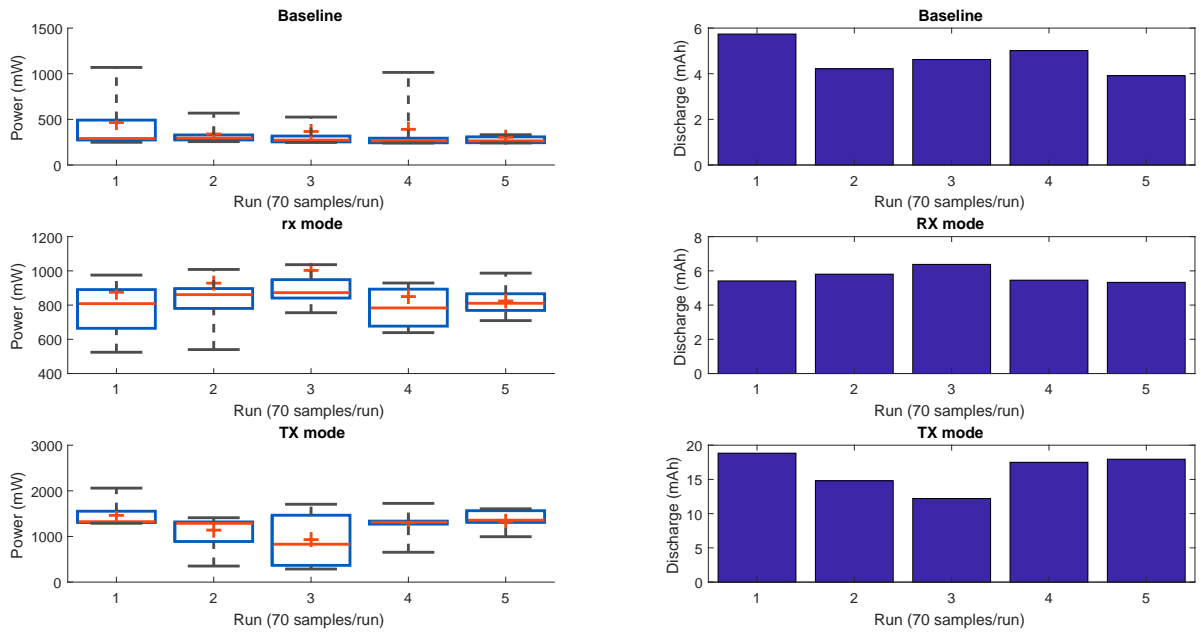


Figura A.17: Resultados del experimento con límite de 1000 kbps de ancho de banda para el bq Aquaris X5 Plus

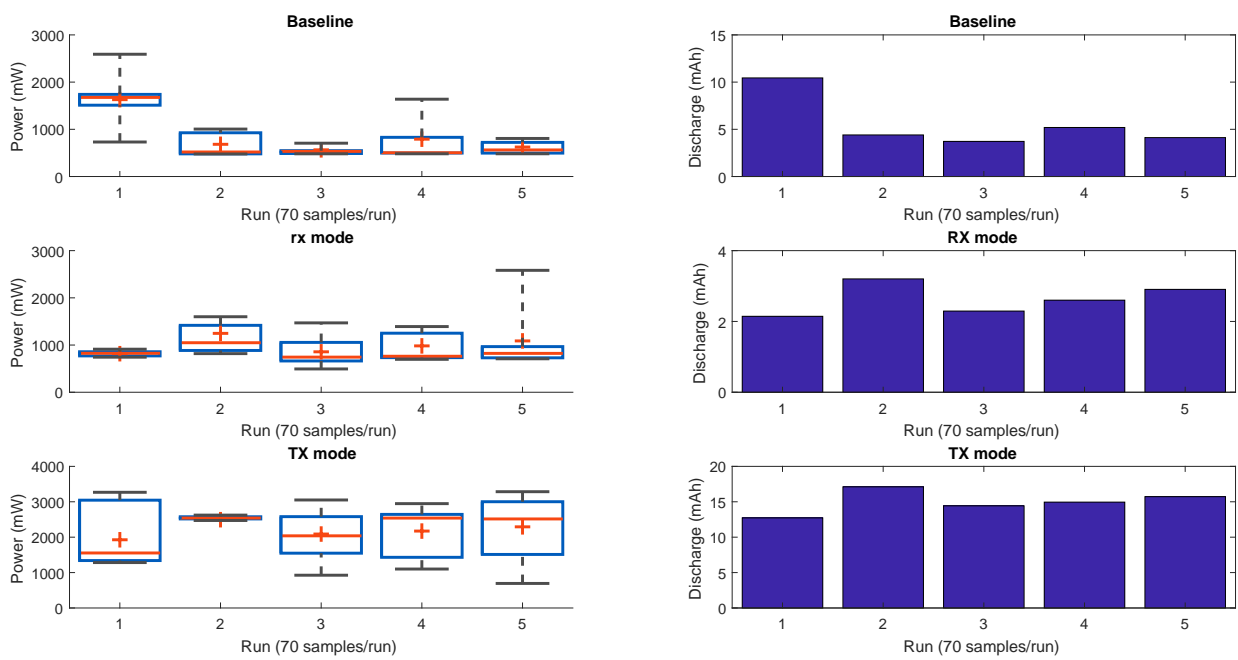


Figura A.18: Resultados del experimento sin límite de ancho de banda para el bq Aquaris X5 Plus

Apéndice B

Requisitos para realizar los experimentos

Los requisitos para poder realizar los experimentos son:

- Dispositivo Android.
 - ✧ Versión mínima: Lollipop 5.0 (API 21).
 - ✧ Permisos de Superusuario.
- Ordenador con Linux.