

**SISTEMA DE LECTURA NUMÉRICA DE MEDIDORES DE ENERGÍA
ELÉCTRICA PARA LA PRUEBA DE DOSIFICACIÓN DE LA
COMPAÑÍA ENERGÉTICA DE OCCIDENTE**



**Juan Pablo Arcos Merchán
Christian David Inga Calvache**

**Universidad del Cauca
Facultad de Ingeniería Electrónica y Telecomunicaciones
Departamento de Electrónica, Instrumentación y Control
Ingeniería en Automática Industrial
Popayán, octubre de 2016**

**SISTEMA DE LECTURA NUMÉRICA DE MEDIDORES DE ENERGÍA
ELÉCTRICA PARA LA PRUEBA DE DOSIFICACIÓN DE LA
COMPAÑÍA ENERGÉTICA DE OCCIDENTE**

**Monografía presentada como requisito parcial para optar por el
título de Ingeniero en Automática Industrial**

**Juan Pablo Arcos Merchán
Christian David Inga Calvache**

Director: Mg. Elena Muñoz España

**Universidad del Cauca
Facultad de Ingeniería Electrónica y Telecomunicaciones
Departamento de Electrónica, Instrumentación y Control
Ingeniería en Automática Industrial
Popayán, octubre de 2016**

Nota de aceptación: _____

Director: _____

Elena Muñoz España

Firma del Jurado

Firma del Jurado

Popayán, octubre de 2016.

Dedicatoria

A nuestros padres por su apoyo incondicional, por enseñarnos a nunca desfallecer en el largo camino de la vida.

A nuestros familiares que nos dan la motivación para superarnos día a día.

Es por eso que este logro se los debemos a ellos.

Agradecimientos

Un agradecimiento especial a nuestros padres, que nos dieron ese apoyo, comprensión y paciencia para sacar este proyecto adelante. Ellos nos han inculcado los valores necesarios para ser personas antes que cualquier otra cosa.

Agradecerles a nuestros familiares, que manifestaron su apoyo y colaboración, además de muestras de cariño y afecto necesarias en los momentos difíciles que pudimos pasar en el desarrollo de este proyecto.

A nuestros compañeros, amigos y colegas que estuvieron compartiendo con nosotros en los buenos y malos momentos, nos brindaron ayuda y nos llenaron de buenos recuerdos, fuera de los lazos de amistad que se reforzaron aún más.

Un agradecimiento especial a nuestra directora, la Magister Elena Muñoz España por permitirnos trabajar bajo su dirección, servirnos de guía y por tener un acompañamiento y apoyo en cada situación que se presentó a lo largo del trabajo.

Agradecimiento al Magister Juan Fernando Flórez por ofrecernos este trabajo de grado, colaborarnos con las asesorías pertinentes y aportarnos las herramientas necesarias en la ejecución de este trabajo.

Agradecimiento a la Universidad del Cauca, a la Facultad de Ingeniería Electrónica y Telecomunicaciones y especialmente al programa de Ingeniería en Automática Industrial que por medio de sus docentes y administrativos estuvieron siempre atentos y prestos a nuestro servicio.

Agradecemos a la Compañía Energética de Occidente y su jefe de Innovación de Desarrollo (Jorge Eliecer Millán) por permitir trabajar en sus instalaciones y ofrecer todo el material necesario para que todo se desarrollara de la mejor manera.

Resumen

En este proyecto se propone una solución a un problema real de la Compañía Energética de Occidente S.A.S. E.S.P. (CEOC), que necesita un método automático para apoyar la prueba de dosificación realizada en las verificaciones en sitio de los contadores de energía eléctrica, mientras incorpora los contadores inteligentes a su red de electricidad.

Se hace uso de la visión de máquina para desarrollar un sistema que realice el pre procesamiento de las imágenes del medidor y del reconocimiento óptico de caracteres (ROC) para determinar los dígitos de los odómetros. El sistema desarrollado es una aplicación móvil para teléfonos inteligentes (*Smartphone*) con sistema operativo Android que permite hacer el escalamiento de tecnología, lo cual constituye un avance tecnológico de interés en el ámbito empresarial.

La aplicación tiene la opción de obtener automáticamente las lecturas necesarias para la prueba de dosificación, así como arrojar los resultados de error y validación del equipo. Además de esto cuenta con una base de datos que permite guardar y mostrar el historial de pruebas realizadas. Las fotografías tomadas para realizar las lecturas del contador, quedan guardadas en el teléfono, esto con el fin de llevar un soporte digital de cada prueba.

La codificación de esta aplicación móvil se llevó a cabo en un entorno de desarrollo integrado conocido como “Android Studio 2.1.3.0” para la codificación y el desarrollo de la interfaz como tal, “OpenCV” para el procesamiento digital de las imágenes y “Tesseract” para el reconocimiento óptico de caracteres.

Finalmente se obtuvo el apk “SLNM”, cuyo nombre son las iniciales de “Sistema de Lectura Numérico para Medidores, que guarda una estrecha relación con la principal funcionalidad de la aplicación y del trabajo de grado.

Con esta aplicación se consigue reducir tiempos operativos y eliminar los errores humanos que son indicadores claves para evaluar la eficiencia de las brigadas y por consiguiente la de toda la compañía, además de hacer un aporte en el escalamiento de la tecnología que impacta a nivel económico a la empresa.

Palabras claves: ROC, *Smartphone*, tecnología, digitalización, carácter, apk.

Abstract

This project proposes a solution to a real problem of the Compañía Energética de Occidente S.A.S. E.S.P. (CEOC), you need an automated method to support dosing test conducted on-site verification of electrical energy meters, while incorporating smart electricity grid to its accountants.

Using machine vision to develop a system that performs the image processing pre meter and optical character recognition (OCR) to determine digits odometers done. The developed system is a mobile application for smartphones Android operating system that allows scaling technology, which is a technological breakthrough of interest in the business.

The application has the option to automatically obtain the necessary readings for the test dosage and throwing error results and validation of equipment. Besides this has a database that can store and display the history of tests. Photographs taken for meter readings, are stored in the phone, this in order to bring a digital support each test.

The coding of this mobile application was carried out in an integrated development environment known as "Android Studio 2.1.3.0" for coding and interface development as such, "OpenCV" for digital image processing and "Tesseract "for optical character recognition.

Finally the SLNM apk, whose name stands for "System Numerical meter reading, which is closely related to the core functionality of the application and degree work was obtained.

With this application it is possible to reduce operating time and eliminate human errors that are key indicators to assess the efficiency of the brigades and therefore the entire company, in addition to making a contribution in scaling technology impacting economic level the company.

Keywords: ROC, *Smartphone*, technology, scanning, character, apk.

Contenido

Capítulo 1. Introducción	1
1.1 Planteamiento del problema.....	1
1.2 Estado del arte.....	2
1.3 Objetivos.....	6
1.3.1 Objetivo general	6
1.3.2 Objetivos específicos.....	6
1.4 Organización de la memoria	7
Capítulo 2. Conceptualización.....	9
2.1 Visión de Máquina.....	9
2.1.1 Principio de la visión humana	10
2.1.2 Modelos de color	10
2.1.3 Escala de grises	12
2.1.4 Histograma de intensidad	12
2.1.5 Algoritmo de medios no locales de eliminación de ruido.....	13
2.1.6 Binarización.....	13
2.1.7 Operaciones morfológicas	14
2.2 Medidores de energía eléctrica	15
2.2.1 De acuerdo con la construcción:.....	15
2.2.2 De acuerdo a la energía que miden:.....	15
2.2.3 De acuerdo a la exactitud.	16
2.2.4 De acuerdo con la conexión a la red y según su construcción (elementos de tensión y corriente)	16
2.3 Verificación de medidores de energía eléctrica	18
2.3.1 Verificación en sitio.....	19
2.3.2 Prueba de dosificación	23
2.4 Resumen	25
Capítulo 3. Análisis de tecnologías	26
3.1 Sistema operativo móvil	26
3.1.1 Sistema operativo Android.....	27
3.1.2 Sistema operativo iOS	27

3.1.3	Sistema operativo Windows Phone	28
3.2	Bibliotecas de procesamiento de imágenes	28
3.2.1	OpenCV.....	28
3.2.2	Image Processing Toolbox	28
3.2.3	Leptónica.....	29
3.3	Motores OCR.....	29
3.3.1	Tesseract OCR.....	29
3.3.2	JavaOCR.....	31
3.4	Resumen	32
Capítulo 4.	Diseño	33
4.1	Selección de tecnologías	33
4.1.1	Selección del sistema operativo móvil	33
4.1.2	Selección de bibliotecas de procesamiento de imágenes	34
4.1.3	Selección del motor OCR	34
4.2	Requisitos de la aplicación.....	34
4.2.1	Requisitos no funcionales.....	34
4.2.2	Requisitos funcionales.....	35
4.2.3	Plan de dispositivos.....	36
4.2.4	Definición de la arquitectura	38
4.3	Diseño.....	38
4.3.1	Navegación entre pantallas	39
4.3.2	Metodología de procesamiento de las imágenes.....	41
4.4	Resumen	44
Capítulo 5.	Implementación.....	46
5.1	Desarrollo	46
5.2	Navegación de la aplicación.....	47
5.3	Funcionalidad de los botones.....	50
5.4	Identificación de la lectura numérica	53
5.4.1	Selección ROI.....	53
5.4.2	Procesamiento en escala de grises y binarización.....	57
5.4.3	Re escalado y rotación	58
5.4.4	Segmentación individual de los dígitos	59
5.4.5	Identificación de centésimas y milésimas.....	60
5.4.6	Aplicación del motor OCR.....	61

5.4.7 Entrenamiento OCR	62
5.5 Resumen	64
Capítulo 6. Resultados	65
6.1 Toma de datos de las pruebas de dosificación CEOC	65
6.1.1 Toma de datos brigadas PQR.....	66
6.1.2 Toma de datos brigadas MACRO	69
6.1.3 Toma de datos laboratorio CEOC.....	71
6.2 Pruebas de la aplicación	75
6.2.1 Pruebas en ambiente controlado.	76
6.2.2 Primera fase de pruebas.....	77
6.2.3 Segunda fase de pruebas	80
6.2.4 Pruebas de la aplicación para valores máximos de rotación y distancia	83
6.3 Interpretación de resultados.....	87
Conclusiones y trabajos futuros	91
Conclusiones	91
Trabajos futuros.....	91
Referencias bibliográficas	93
Anexo 1.....	97
Anexo 2.....	109
Anexo 3.....	115

Lista de Figuras

Figura 1. Representación de colores con el modelo de color RGB [http://www.xaraxone.com/webxealot/workbook35/page_5.htm]	11
Figura 2. Representación del modelo de color HSV [https://es.wikipedia.org/wiki/Modelo_de_color_HSV#/media/File:HSV_cone.jpg]	11
Figura 3. Actividades realizadas en la verificación en sitio [39]	19
Figura 4. Conexión de los elementos de potencia [39]	21
Figura 5. Conexión de los elementos y su terminal de mando [40]	22
Figura 6. Interfaz del SYSMOVIL en la TGO [40]	22
Figura 7. Interfaz de la prueba de dosificación en SYSMOVIL [40]	23
Figura 8. Diagrama de bloques de reconocimiento de palabras en Tesseract OCR [50]	30
Figura 9. Posibles contornos que puede tener el número 1 al 9 [Fuente Propia]	31
Figura 10. Logotipo aplicación desarrollada [Fuente propia]	39
Figura 11. Diagrama HTA de la aplicación SLNM [Fuente Propia]	40
Figura 12. Pantallas de la aplicación SLNM [Fuente Propia]	41
Figura 13. Pantallas de toma de lectura a través de SLNM [Fuente Propia]	41
Figura 14. Registrador del medidor con odómetro decimal en transición numérica de 2 a 3 [Fuente Propia]	44
Figura 15. Proporción de transición de valores numéricos en el odómetro decimal [Fuente Propia]	44
Figura 16. Medidor ISKRA y Medidor TECUN [Fuente Propia]	47
Figura 17. Pantalla inicial de la aplicación SLNM [Fuente Propia]	48
Figura 18. Pantalla de registro de la aplicación SLNM [Fuente Propia]	48
Figura 19. Pantalla de información de la aplicación SLNM [Fuente Propia]	49
Figura 20. Pantalla principal de la aplicación SLNM [Fuente Propia]	50
Figura 21. Pantalla de listar de la aplicación SLNM [Fuente Propia]	50
Figura 22. Habilidad de los botones “Lectura inicial” y “INICIAR” [Fuente Propia] ..	51
Figura 23. Habilidad de los botones “Lectura Final” y “LISTAR” [Fuente Propia] ...	52
Figura 24. Cuadro de dialogo para confirmar la salida de la prueba de dosificación [Fuente Propia]	52
Figura 25. Imagen digital de un medidor de energía eléctrica [Fuente Propia]	53
Figura 26. Identificación y selección de color rojo de una imagen de los odómetros [Fuente Propia]	54
Figura 27. Obtención de la ROI (color azul) en diferentes imágenes de un medidor [Fuente Propia]	55
Figura 28. Rectángulos identificados según el color rojo (odómetro o franja) [Fuente Propia]	56
Figura 29. Diferentes tipos de odómetros de medidores [Fuente Propia]	56
Figura 30. Función crop de Android para la selección manual del ROI [Fuente Propia]	57

Figura 31. Procesamiento en grises y binario en imagen segmentada [Fuente Propia]	57
Figura 32. Resultado del procesamiento en escala de grises y binarización en un ambiente con nivel de luminosidad alto [Fuente Propia]	58
Figura 33. Resultado de la metodología de morfología en grises y binaria y la metodología de detección de blancos con el modelo HSV [Fuente Propia]	58
Figura 34. Dígito segmentado individualmente con ruido, antes y después de determinar que la región más pequeña debe ser eliminada [Fuente Propia]	59
Figura 35. Dígitos en transición antes y después de determinar la región de mayor tamaño [Fuente Propia]	59
Figura 36. Transición de 0 a 1 en el odómetro decimal [Fuente Propia]	60
Figura 37. Dígitos segmentados individualmente con el color invertido [Fuente Propia]	62
Figura 38. Imágenes usadas para el entrenamiento del motor OCR [Fuente Propia]	63
Figura 39. Posibles contornos del dígito 8 después del entrenamiento [Fuente Propia]	63
Figura 40.. Posición del celular para medir luminosidad [Fuente Propia]	66
Figura 41. Operador PQR realizando prueba de dosificación [Fuente Propia]	67
Figura 42. Operador MACRO realizando prueba de dosificación [Fuente Propia]	69
Figura 43. Laboratorista realizando prueba de dosificación [Fuente Propia]	72
Figura 44. Contadores expuestos a diferentes ambientes lumínicos [Fuente Propia]	75
Figura 45. Representación de las pruebas de la aplicación SLNM para valores máximos de rotación y distancia [Fuente Propia]	83
Figura 46. Representación de los ejes en la prueba de distancia [Fuente Propia]	84
Figura 47. Imagen digital obtenida en la prueba de distancia [Fuente Propia]	84
Figura 48. Imagen digital obtenida a una distancia superior a 26 centímetros [Fuente Propia]	85
Figura 49. Representación de ejes de rotación en las pruebas de variación de ángulos [Fuente Propia]	85
Figura 50. Imagen digital al rotar el eje Y [Fuente Propia]	86
Figura 51. Imagen digital al rotar el eje Z [Fuente Propia]	86
Figura 52. Imagen digital al rotar el eje X [Fuente Propia]	86
Figura 53. Pruebas de la aplicación SLNM [Fuente Propia]	88
Figura 54. Pruebas de la aplicación SLNM para valores críticos [Fuente Propia]	88
Figura 55. Proporciones de transición en el odómetro decimal [Fuente Propia]	89

Lista de Tablas

Tabla 1. Resumen de la clasificación de medidores [36].....	17
Tabla 2. Rango de medida de los medidores [36]	18
Tabla 3. Elementos para la verificación en sitio.....	20
Tabla 4. Resolución del medidor vs Energía dosificada [39]	24
Tabla 5. Ventas de teléfonos inteligentes en el mundo relacionado con los sistemas operativos en el primer cuarto de 2016 y primer cuarto de 2015 [42]	26
Tabla 6. Comparación de dispositivos robustos	37
Tabla 7. Datos obtenidos en brigadas PQR	68
Tabla 8. Datos brigadas Macro	70
Tabla 9. Datos pruebas Laboratorio de Calibración y Ensayo	72
Tabla 10. Pruebas de la aplicación en la Laboratorio de la CEOC	73
Tabla 11. Comparación de lecturas entre el operador del Laboratorio y la aplicación.	74
Tabla 12. Comparación de tiempo de registro entre el operador del Laboratorio y la aplicación.....	74
Tabla 13. Pruebas de la aplicación en un ambiente controlado.....	76
Tabla 14. Resultados prueba de la aplicación primera fase	79
Tabla 15. Resultados segunda fase de pruebas de la aplicación en el medidor TECUN.	82
Tabla 16. Comparación de tiempos de registro de lectura	90

Capítulo 1. Introducción

1.1 Planteamiento del problema

La Compañía Energética de Occidente S.A.S., E.S.P. (CEOC) desde 2010 asumió la gestión de actividades para la prestación de los servicios de distribución y comercialización de energía eléctrica en el Departamento del Cauca. Las peticiones, quejas y reclamos (PQR) que realizan las personas (naturales y jurídicas) sobre los medidores de energía eléctrica y el cobro del consumo, más las estadísticas que se interpretan como posibles fallos en los medidores de la red eléctrica (pérdidas de energía) hace necesaria la intervención de las brigadas técnicas, las cuales son las encargadas de examinar y evaluar el correcto funcionamiento y la medición precisa del medidor, para garantizar el cobro de las tarifas de energía eléctrica y eliminar o reducir pérdidas de energía eléctrica.

Los instrumentos utilizados por las brigadas técnicas para realizar las pruebas en campo a los medidores de energía eléctrica, son muy robustos y de tecnología desarrollada hace mucho tiempo, actualmente usan una terminal de gestión de ordenes (TGO), la cual les brinda a los operadores una interfaz gráfica donde se especifica los procedimientos que se deben realizar y los valores que deben ingresar para obtener resultados de las pruebas, es muy parecido a un *Smartphone*, pero mucho más robusto a los golpes. Esta TGO es el corazón de las pruebas que se realizan, ya que les indica a los operadores paso a paso los procedimientos que deben realizarse, además de encargarse de los cálculos y resultados de las pruebas con los valores que son ingresados por los operadores. Esta TGO requiere mucha inversión económica para el mantenimiento, además está muy enfocado en la robustez a los golpes, pero el procesamiento es muy bajo, ya que deja de funcionar por intervalos de tiempo en las pruebas, haciendo que estas tomen mucho más tiempo. Así surge la necesidad de realizar la transferencia de tecnología a dispositivos móviles con mayor procesamiento y económico mantenimiento o reemplazo para mejorar la agilidad de las pruebas.

Dentro de las pruebas en campo que realiza la brigada técnica a los medidores, existe una en particular que se llama “Prueba de Dosificación”, esta consiste en comprobar que el registrador del medidor (odómetros) funciona correctamente, para esto se comprueba que el valor registrado por el medidor es igual a la energía dosificada en un lapso de tiempo. Un operario de la brigada toma una foto a los dígitos del medidor e ingresa el valor manualmente de lectura del medidor a la TGO, después se conecta la carga (se sabe su valor en Vatios) por cierto tiempo, después se dosifica la energía (dependiendo de la resolución del contador), se desconecta la carga y se vuelve a tomar la foto al medidor y se anota su valor actual, la TGO determina el porcentaje de error dependiendo del valor registrado por el medidor respecto a la carga patrón a través de una ecuación matemática que realiza internamente, si el porcentaje de error es superior o inferior a $\pm 5\%$ se determina que el medidor no cumple con la prueba y debe ser retirado y cambiado por un medidor calibrado y nuevo.

Según lo mencionado en el párrafo anterior, la exactitud de la prueba de Dosificación está ligada con la percepción visual del operario de la brigada, si este llega a interpretar mal un valor o se equivoca en la anotación del valor, existirá error en la prueba y se verá afectada la confiabilidad de los resultados. Además, el operador solo está en la capacidad visual de reconocer hasta las centésimas de la lectura del medidor, y en casos solo hasta las decimas dependiendo de la agudeza visual del operario que realice las pruebas.

De acuerdo a lo anterior, se ve la necesidad de desarrollar una tecnología que permita tomar el valor de la medición de una manera más sencilla y exacta durante la prueba de dosificación, además, debe ser desarrollada en un sistema operativo móvil; ya que la CEOC tiene la meta de optimizar las pruebas técnicas de verificación en sitio a través de la utilización de un teléfono inteligente. Esto conduce a plantear el siguiente interrogante ¿Cómo un sistema de lectura numérica basado en procesamiento de imágenes y reconocimiento óptico de caracteres (OCR) puede mejorar la precisión de la lectura de los medidores de energía eléctrica en la prueba de dosificación, respecto a la metodología usada actualmente por las brigadas técnicas de la CEOC?

1.2 Estado del arte

El medidor de energía eléctrica es un instrumento técnico (mecánico o electrónico) para medir la energía suministrada a cada cliente de las organizaciones encargadas de la distribución de energía eléctrica. De acuerdo a la lectura del medidor, el periodo de tiempo de consumo (mes) y la tarifa establecida por la CREG (Comisión de Regulación de Energía y Gas), se define el costo del consumo del mes del usuario.

Los medidores de energía eléctrica deben ser revisados por la CEOC cuando el usuario lo solicite o cuando la CEOC requiera verificar su correcto funcionamiento. Si en las brigadas técnicas se supera el 5% de error en las pruebas, los medidores deben ser retirados y enviados para su inspección al laboratorio de calibración y ensayo de la CEOC, este último da el parte oficial del estado y funcionamiento del medidor, y así se puede concluir si el medidor debe ser destruido.

En las últimas décadas el avance tecnológico va a pasos agigantados, y las telecomunicaciones no son ajenas a ellos. El teléfono móvil nació como herramienta para mejorar la necesidad de la comunicación de las personas, pero mientras más pasa el tiempo, más complejos y completos se vuelven; pasaron de ser herramientas que se limitaban a llamar inalámbricamente o enviar mensajes de texto, a ser herramientas con funciones de computadores personales.

Los *Smartphone* cada día tienen mayor capacidad de almacenamiento y procesamiento, y complementándolo con los sensores que traen incorporados (luminosidad, proximidad, humedad relativa, acelerómetros, giróscopos, magnetómetros, etc.), estos pueden ser utilizados en actividades o herramientas para mejorar la calidad de vida de las personas en eventos cotidianos [1] como recorrer una ciudad nueva o para ayudar a personas con discapacidad visual [2] traduciendo texto

que no está escrito en Braille. Para las organizaciones gubernamentales o privadas los *Smartphone* son herramientas que pueden ser aprovechados en actividades casi ilimitadas, cada vez son más usados para complementar las funciones de las personas dentro de la organización [3] [4] para agilizar y obtener mejores resultados en sus actividades.

Muchos trabajos académicos proponen prototipos para la lectura de medidores a través de procesamiento de imágenes en computadores (servidores web) con plataformas Windows o GNU/Linux [5] [6] [7] [8] [9] [10] obteniendo la imagen a través de un dispositivo móvil y enviándolos inalámbricamente a través de la red telefónica, el procesamiento de imágenes se hace en aplicaciones Java [9] [10] o en programas como Matlab; con su *Toolbox* de visión artificial [11] [12] complementado con su entorno de programación visual Simulink [13], en dispositivos móviles incorporados con cámara como los *SmartGlass* [14] y en *Smartphone* [6] [7] [8] [15].

En el mercado actual existen muchas marcas de *Smartphone*, cada una mejor que la otra en diferentes características, pero, además existen diferentes tipos de sistemas operativos, el sistema operativo Android de Google se encuentra en el 84.7% de los *Smartphone* del mundo, mientras que el iOS de Apple está en el 11,7% de los *Smartphone*. En [16] se realiza una aplicación para el reconocimiento de la lectura de medidores de gas, esta aplicación es probada en los sistemas operativos Android e iOS arrojando que la aplicación en iOS es 92% precisa, mientras que la misma aplicación en Android es tan solo 87%, las pruebas incluyen situaciones particulares, como lugares oscuros y reflejos de luz sobre el medidor. Los resultados de precisión son menores en el sistema operativo Android debido a que el sistema operativo Android se encuentra en una cantidad muy heterogénea de *Smartphone*, los cuales incluyen *Smartphone* de bajo presupuesto que tienen capacidades de procesamiento bajo y cámaras con poca resolución, en comparación con el iOS que se encuentra en una marca específica de *Smartphone*, iPhone.

Antes de poder utilizar un OCR (reconocimiento óptico de caracteres) para obtener la lectura del medidor, es necesario realizar procesamientos a las imágenes para poder obtener una imagen binaria y segmentada, de esta manera se usa el OCR solo donde se encuentran los dígitos que se desea saber el valor, así el OCR será mucho más preciso y no tendrá tanto consumo de procesamiento, actuando de una manera más rápida y ágil. Se han propuesto algunas secuencias de funciones [5] [6] [17] para el procesamiento de las imágenes obteniendo al final solo la imagen de la zona donde se encuentran los dígitos (región de interés) de la lectura del medidor [6] [7] [9] [13] [15] [18], o imágenes donde se encuentra en cada una de estas los dígitos por separado de la lectura [5] [6] [12] [19].

Las metodologías definidas para el procesamiento de las imágenes antes de aplicar el OCR varían muy poco, es claro que primero se debe realizar un procesamiento básico (escala de grises y blanco y negro), luego un filtrado para mejorar la imagen, seguido de la segmentación a través de diferentes métodos de región de interés (ROI) y finalizando con operaciones morfológicas para mejorar la imágenes, así cuando el OCR sea aplicado, este pueda ser más preciso en sus resultados; los prototipos de las

aplicaciones están basados en esta metodología, pero cada uno define una, donde son poco los cambios agregados [11] o no utilizados, generalmente se cambia en el orden [5] [6] pero obteniendo al final la respuesta deseada, la lectura del medidor. Existen algoritmos para la binarización de las imágenes mediante un umbral o histogramas de la imagen, pero saber cuál es el más efectivo para la lectura de los medidores es un gran reto, en [5] se realiza una comparación de algunos algoritmos de binarización comparándolos de acuerdo a la precisión y el tiempo de ejecución de este, así se puede saber a ciencia cierta cuál obtiene el mejor resultado siendo más ágil la aplicación que se desea desarrollar. En el procesamiento básico se puede utilizar el modelo de color HSV (tono, saturación y valor) [20], este método es muy ventajoso en la lectura debido a la combinación de colores.

Existen métodos para la binarización de imágenes digitales donde el fondo se deja negro ("0") y el texto es dejado como blanco ("1") [21] [22] [23] [24], de esta manera cuando se usa el OCR sobre estas imágenes binarizadas, el OCR actúa de una manera más eficiente y ágil. Las metodologías de binarización varían dependiendo de los resultados esperados o el tipo de documento que se desea digitalizar, hay proyectos específicos destinados a la binarización de periódicos [25] mejorando la precisión de la binarización aplicando y evaluando diferentes metodologías. Los odómetros solo traen valores numéricos, pero pueden ser considerados como texto, siendo posible aplicar metodologías para obtener solo el texto de una imagen digital. El método expuesto en [21] determina el umbral que debe ser aplicado para cada componente (pixel) de la imagen, teniendo la capacidad de procesar imágenes donde el fondo tiene varias intensidades y tonos, o imágenes donde el texto tiene diferentes tamaños o formas sin la necesidad de realizar alguna sintonización manual y puede ser usado en diferentes ambientes. En [22] se desarrolla un método de binarización basándose en la percepción humana del texto (grupo perceptualmente significativo de objetos atómicos), para esta metodología usan clasificadores secuenciales en el modelo de color HSV. En otras metodologías desarrolladas se plantea el problema de detección de los caracteres como una sección secuencial desde un conjunto de regiones extrémales [23] [24], incluyendo nuevas regiones extrémales como proyecciones de la magnitud de los gradientes de la imagen [23] y un selector de regiones extrémales máximamente estables [26], que explora la topología de región realizando una búsqueda exhaustiva en el espacio de todas las secuencias de caracteres. Existen proyectos académicos donde se entrenan lenguajes para motores OCR desde cero [26] [27] basados en diccionarios de lenguajes definidos por los desarrolladores de los motores, para mejorar la precisión y el resultado.

También se propone utilizar redes neuronales que puedan entregar automáticamente el ROI [9] [18]. Este método de segmentación a través de redes neuronales es capaz de detectar el área de interés y el dígito de la lectura en 0,8 segundos [18], este tiempo puede ser menor, pero puede ser menos preciso [9] ya que la ROI entregada es más grande que el área ocupada por los dígitos de la lectura del medidor; pero es una metodología más rápida y utiliza menos recursos computacionales. Además, se ha desarrollado un método para la segmentación de la ROI, el cual de acuerdo a una serie de procedimientos y aplicando filtros para mejorar la imagen, este es 5 a 9 veces más

rápido [12] que la extracción de bordes (Sobel), así la aplicación final procesa la información mucho más rápido optimizando el procesamiento del dispositivo móvil; este método de segmentación no se usó para diferentes condiciones de luminosidad sobre el medidor, pero se concluyó que el método podría servir para estos casos. Otros métodos para la segmentación del ROI utilizan técnicas de corrección de inclinación computacionalmente eficientes [21]; experimentos demuestran que la técnica es eficiente y aplicables a dispositivos móviles de gama media, logrando una precisión de la segmentación de 97.48% en 1,1 segundo para imágenes de 3 Mega-píxeles.

Los contadores de energía tienen un odómetro para representar la medida decimal de la lectura, este dígito está marcado generalmente con un color rojo, así, se crea la necesidad de usar algoritmos para la localización de rojo de forma que la aplicación pueda reconocerla como un valor más de la lectura. La metodología de localización de rojos utilizada en [10] se basa en calcular el porcentaje de rojo sobre verde y azul para decidir si un píxel es rojo, tiene un 85% de precisión; esta metodología no es tan precisa porque es muy sensible a la luminosidad del ambiente, por ejemplo, las imágenes son tomadas con el flash del celular, de esta manera es casi irreconocible el color rojo de las lecturas y la aplicación no lo reconoce como dígitos de la lectura, obviando las décimas y las centésimas.

En el ámbito científico, muchas aplicaciones y propuestas usan OCR precisos y óptimos cuando se trata de bajo consumo de procesamiento y resultados, como el caso del OCR de las librerías OpenCV [9] [10], el Tesseract OCR [8] [14] desarrollado por Google y el OCR del *Toolbox* de visión artificial de Matlab [11] [14], pero además de estos OCR, muchas investigaciones proponen sus propios OCR [5] [15] [17] [21] para la lectura del consumo de energía eléctrica. Se proponen OCR usando redes neuronales [5] [21] entrenándolas para las diferentes formas de los dígitos de los medidores, también se propone un OCR dividiendo la imagen de cada dígito en 4 partes (cuartiles) [17], luego se cuenta cuantos píxeles blancos contiene cada cuartil, luego es comparado con una pequeña base de datos y se le da un valor de acuerdo a la referencia más cercana; este algoritmo logró una precisión del 96.49% para cada dígito y 85.71% para la lectura del medidor. Ya que los *Smartphone* entre más capacidad de procesamiento más costoso son, en [15] se ve la necesidad de crear un OCR de bajo consumo de procesamiento, para poder ser usado en *Smartphone* que no tengan una gran capacidad de procesamiento; este tipo de OCR de bajo consumo comparado con el Tesseract OCR de Google es 92.74% preciso en una evaluación de 100 imágenes, pero con mucho menos recursos de procesamiento, volviendo más eficiente al dispositivo móvil.

Como los contadores de energía eléctrica muestran el consumo a través de odómetros, existe un problema muy grande cuando se va a cambiar de un dígito a otro: por ejemplo, cuando la lectura va de 1 a 2, en el odómetro se verá la parte superior del 1 y la parte inferior del 2 entre más sea el consumo. Los trabajos académicos prácticos toman estos casos como error [16] [19], porque el OCR no tiene la capacidad de diferenciar el dígito, ya que se ven los dos dígitos (1 y 2 para el ejemplo), el artículo [21] describe un algoritmo con la capacidad de resolver este tipo de problema, los experimentos demuestran una tasa de reconocimiento del 99.9%, aun cuando los

dígitos están en la transición de cambio (rotación del odómetro) y el tiempo de reconocimiento es de máximo 31ms. El algoritmo consiste en evaluar que dígito ocupa el mayor espacio en el odómetro, aplicando el OCR en el dígito que ocupe el mayor espacio, después de un tiempo se vuelve a aplicar el algoritmo comparándolo con el resultado anterior, se supone que se debe obtener el mismo dígito, ya que, si se ha consumido energía, el odómetro debió haber girado y el dígito ocupará más espacio en el odómetro.

Se encontró que muchos trabajos investigativos proponían algoritmos para el procesamiento de las imágenes, o proponían OCR para obtener el valor de la lectura del medidor, pero estos prototipos solo eran definidos teóricamente y no eran desarrollados [6] [7] [8] [15], o eran evaluados en laboratorio en condiciones ambientales controladas [20] y diferentes distancias entre la cámara [5] y el medidor, otras evaluaciones se hacían en el mismo tipo de contador pero realizando diferentes lecturas [11] [13] [14] [17] [21]. En los trabajos mencionados las aplicaciones no son evaluadas en condiciones reales donde interfieren las condiciones ambientales y humanas de los operadores que usan la herramienta, impidiendo saber con resultados que tan óptimo y preciso son los prototipos propuestos.

1.3 Objetivos

1.3.1 Objetivo general

- Proponer un sistema de lectura numérica de precisión para los medidores de energía eléctrica en la prueba de dosificación de las brigadas técnicas de la CEOC.

1.3.2 Objetivos específicos

- Diseñar una aplicación móvil que a través del procesamiento digital de imágenes obtenga los dígitos numéricos de las lecturas requeridas de los medidores de energía eléctrica (electrónicos y mecánicos) en la prueba de dosificación.
- Implementar la aplicación bajo condiciones ambientales controladas y acorde a los procedimientos usados por las brigadas de CEOC en la prueba de dosificación
- Evaluar la exactitud y precisión de los resultados del sistema desarrollado respecto a la metodología usada actualmente por las brigadas técnicas y concluir si la aplicación móvil es una herramienta tecnológica que satisface las necesidades de la CEOC.

1.4 Organización de la memoria

La memoria está distribuida en seis capítulos, claramente diferenciados, en los que se detalla tanto el planteamiento del problema, seguido de la elaboración del proyecto como el desarrollo del mismo. Además, se incluye un anexo con el fin de proporcionar aspectos de interés relativos a la propia aplicación.

CAPITULO 1: INTRODUCCIÓN.

En este capítulo se contextualiza al lector sobre el proyecto desarrollado a través del resumen, el planteamiento del problema que motivó el trabajo desarrollo y el estado del arte sobre la temática del proyecto, además, se presentan los objetivos y la organización de la memoria.

CAPITULO 2: CONCEPTUALIZACIÓN.

Se explican y se analizan los conceptos teóricos en los que está basada la aplicación desarrollada. Se comienza con una introducción a la visión artificial y sus aplicaciones, después se continúa con temas relativos al procesado digital de imágenes y su uso en la aplicación. Se describe las clases de medidores que se encuentra en el mercado, se expones y se detalla la prueba de verificación en sitio realizada por la CEOC y los instrumentos que se utilizan.

CAPÍTULO 3: ANÁLISIS DE TECNOLOGÍAS.

Se analiza información relacionada con los motores OCR y plataformas de desarrollo en el sistema operativo Android. También se recopila información de librerías de procesamiento de imágenes de licencia libre para realizar el procesamiento digital de imágenes para conseguir extraer e identificar los caracteres existentes dentro de imágenes digitalizadas.

CAPÍTULO 4: DISEÑO.

En este capítulo se realizan las labores necesarias para definir una solución viable a la problemática planteada, tales como: la selección de la tecnología tanto software como hardware con que se puedan suplir los requisitos funcionales y no funcionales de la aplicación, la definición la arquitectura que se va a trabajar y la creación de la interfaz.

CAPÍTULO 5: IMPLEMENTACIÓN.

En este capítulo de describe la creación de la aplicación, la navegabilidad de pantallas, así como la funcionalidad de sus botones. Además, se explica la forma de realizar la lectura numérica como también el entrenamiento del OCR para un óptimo desempeño.

CAPÍTULO 6: RESULTADOS

Consta de un capítulo en el que se exponen los resultados obtenidos en las pruebas del software desarrollado y se evalúan estos resultados en comparación a la metodología actualmente utilizada por la CEOC. Se enumeran las conclusiones que se han obtenido del trabajo realizado y se proponen posibles líneas de trabajo o de mejora

Capítulo 2. Conceptualización

Se investiga y se analizan los conceptos teóricos y términos utilizados en este proyecto, además, se citan las normas técnicas que se deben implementar en las pruebas de verificación en sitio realizadas por las brigadas técnicas de la CEOC y los instrumentos utilizados.

Debido a que el proyecto se encuentra en la línea de investigación de la visión de máquina, se explica el concepto y como se presentan las imágenes digitales, cuál es su contenido y los métodos utilizados en la aplicación.

2.1 Visión de Máquina

La Visión de Máquina; llamada también Visión por Computadora o Visión Artificial, es una disciplina científica que utiliza métodos para detectar datos del mundo físico ("real") a partir de imágenes para su posterior interpretación por un computador [28]. Mediante la utilización de técnicas adecuadas, la visión de máquina permite la obtención, procesamiento y análisis de información a partir de imágenes digitales tomadas desde cámaras de video o de un escáner medico; por ejemplo, en las aplicaciones médicas.

Una imagen digital puede ser definida como una función de dos dimensiones $f(x, y)$, donde x y y son coordenadas espaciales, y la amplitud de f en cualquier par de coordenadas (x, y) se llama intensidad o nivel de gris de la imagen en ese punto [28]. Cuando x y y , y el valor de la amplitud de f son todos finitos, se llama una imagen digital. Una imagen digital, se compone de un número finito de elementos, cada uno de los cuales tiene una ubicación y valor particular. *Píxel* es el término más utilizado para denotar los elementos de una imagen digital.

El procesamiento digital de imágenes es el conjunto de técnicas que se aplican a las imágenes digitales con el objetivo de mejorar la calidad de estas, para su posterior utilización o interpretación. El procesamiento digital de imágenes puede ser dividido [29] en tres pasos: codificación, extracción de características y la segmentación.

- **Codificación:** La codificación es la representación de las imágenes, comprende:
 - **Adquisición:** La adquisición es la transformación de las imágenes "reales" a imágenes digitales para ser utilizadas en el computador.
 - **Comprensión:** La comprensión es la modificación de la representación de la imagen; puede ser la transformación de un modelo de color a otro para aplicaciones específicas.
 - **Síntesis:** La síntesis es la transformación de la imagen digital en representaciones simbólicas como números para realizar operaciones matemáticas.

- **Pre procesamiento:** Se mejora la calidad de la imagen para resaltar características y eliminar ruido o defectos que pueda contener, de esta manera se extraen los parámetros necesarios o deseados.
- **Segmentación:** La segmentación es la división de la imagen e regiones o zonas, bajo un criterio de selección de acuerdo al objetivo o fin del procesamiento.

Los elementos electrónicos que se encuentran en las cámaras electrónicas encargadas de hacer la captura de la “imagen”, son llamados sensores de imagen. Estos sensores son circuitos integrados formados por componentes sensibles a la luz, cuando son expuestos a la luz proyectada por un objeto, generan una señal eléctrica proporcional a la intensidad de la luz, que luego son analizadas para crear la imagen, que posteriormente puede ser almacenada [30]. La capacidad de resolución de estos sensores depende de las células fotoeléctricas que estas tengan, entre más células fotoeléctricas (píxel) tenga el sensor, mayor nitidez tendrá la imagen final.

El Dispositivo de Carga Acoplada “CCD” (*charge-coupled device*) es un sensor con millones de diminutas células fotoeléctricas, estas células producen una carga eléctrica proporcional a la intensidad luminosa que incide sobre ellos, grabando los tres colores básicos o fundamentales (Rojo, Verde, Azul (RGB)) [31]. El sensor de píxeles activos “CMOS” está formado por millones de fotodiodos (uno para cada píxel) que producen una carga eléctrica dependiendo de la luz que recibe, tiene el mismo funcionamiento que un CCD solo que cada fotodiodo tiene un amplificador de señal eléctrica, generando la ventaja que la electrónica de la cámara puede leer la señal directamente del fotodiodo [31].

2.1.1 Principio de la visión humana

En la retina del ojo (tejido sensible a la luz) existen millones de células que detectan las longitudes de onda procedentes del entorno; se llaman conos y bastoncillos. Hay tres tipos de conos que son sensibles a diferentes espectros de luz, un tipo de cono es sensible a las ondas cortas, otro a las ondas medias y otro a las ondas largas, transformándolos en impulsos eléctricos que son interpretados por el cerebro, creando la sensación del color [32].

Por cada tipo de cono existe una sensación de color, llamados Colores Fundamentales (Cf). Cuando los conos que reaccionan a las ondas cortas, ondas medias y ondas largas son estimulados, existe la sensación del Cf Azul-Violeta, Cf Verde y del Cf Rojo-Naranja respectivamente. Debido a que existen los tres tipos de células foto-receptoras (conos) son necesarias y suficientes solo tres componentes numéricas para describir cualquier color [33].

2.1.2 Modelos de color

Los modelos de color son modelos matemáticos que describen como se representan los colores según la aplicación que se desea realizar. Estos modelos describen los colores asociándolo a un vector numérico un elemento del espacio de color definido por el modelo de color.

- **Modelo de Color RGB**

Las imágenes digitales en color, básicamente están conformadas por las componentes espectrales de cada canal del modelo de color RGB (Red, Green, Blue). Cada canal está conformado por una matriz, cada elemento de la matriz (x, y) (píxel) tiene un valor numérico entre un valor determinado, por ejemplo, entre 0 y 255 para imágenes de 8 bits. Para representar un color se realiza una mezcla entre los 3 canales, esto se realiza por la acción aditiva de las tres matrices de los componentes [29]. Así el color de cada píxel de la imagen depende de los valores de ese píxel en cada componente. Por ejemplo, en la Figura 1, el color del píxel blanco, significa que en el modelo de color RGB los tres canales tiene su máximo valor mientras en el color oro (*Gold*) cada canal del modelo de color tiene un valor diferente.

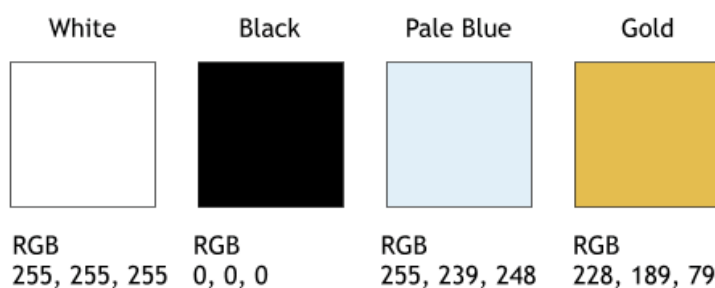


Figura 1. Representación de colores con el modelo de color RGB
[http://www.xaraxone.com/webxealot/workbook35/page_5.htm]

- **Modelo de color HSV**

El modelo de color HSV (*Hue, Saturation, Valor*) representa el color en tres dimensiones, en sus tres componentes Matiz, Saturación y Valor [29]. En la Figura 2 se observa que el modelo de color HSV puede ser representado como un cono, el eje central va desde el blanco en la parte superior hasta el negro en la parte inferior. El ángulo del eje representa el matiz del color, la distancia desde el eje representa la saturación y la distancia a lo largo del eje, representa el valor.

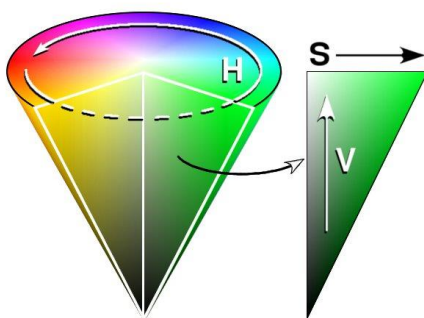


Figura 2. Representación del modelo de color HSV
[https://es.wikipedia.org/wiki/Modelo_de_color_HSV#/media/File:HSV_cone.jpg]

El modelo de color HSV es comúnmente usado para elegir un color de las imágenes, ya que el matiz se representa como el ángulo del eje central del cono (0° - 360°), cada rango de grados representa un color, por ejemplo, el color rojo va desde 0° hasta 60° , de 60° hasta 120° es el amarillo y así sucesivamente hasta 360° . De esta manera, si se desea obtener solo los objetos de color rojo de una imagen se puede convertir la imagen digital al modelo de color HSV y seleccionando solo los píxeles cuyo matiz se encuentren entre 0° a 60° se pueden obtener solo los objetos cuyo matiz se encuentra en cualquier rango de color rojo.

2.1.3 Escala de grises

Una imagen digital en escala de grises (*Grayscale*) es una imagen de un solo canal que cada elemento está representado por un valor numérico en un rango determinado, para las imágenes de 8 bits de 0 a 255. Siendo 0 negro absoluto y 255 el blanco absoluto, de esta manera cada píxel representa su luminancia o intensidad y tiene la sensación de un valor de gris [28].

2.1.4 Histograma de intensidad

El histograma de una imagen representa el número de píxeles para cada valor de la intensidad. Generalmente el histograma se representa como una distribución donde las intensidades (nivel de gris) están representadas en el eje X, y el número de ocurrencias (veces que se repite) para cada intensidad, se representa en el eje y.

• Ecuación del histograma

Cuando se ecualiza un histograma de una imagen se transforma la imagen para obtener una nueva imagen, pero su histograma tendrá una distribución uniforme de intensidades, de esta manera existirá (teóricamente) un número igual de píxeles para cada nivel de intensidad [34]. La fórmula para ecualizar un histograma es:

$$equist[i] = \left[\frac{[(L * hist_{cf}[i]) - N^2]}{N^2} \right] \quad (1)$$

donde:

- i : valor de gris
- L : número de niveles de grises
- $hist_{cf}$: histograma
- N : número de píxeles

Durante la transformación de la nueva imagen, los píxeles de un nivel de intensidad tendrán un nuevo valor de intensidad, así, el histograma estará distribuido uniformemente en el rango de intensidades. La ecualización es utilizada para maximizar el contraste de una imagen sin perder información de esta.

2.1.5 Algoritmo de medios no locales de eliminación de ruido

Cualquier imagen digital puede presentar perturbaciones o ruidos las cuales se debe a la naturaleza aleatoria del proceso de recuento de fotones en cada sensor de las cámaras digitales, además, el ruido puede ser amplificado en los sensores gracias a las operaciones software que pueda tener la cámara, como aumentar el contraste o mejorar la definición de la imagen [35].

En un principio, los primeros métodos de eliminación de ruido eran bastantes simples, se sustituía el color del píxel con el promedio de los colores de los píxeles cercanos. La ley de la varianza, asegura que, si se promedian nueve colores de píxeles cercanos, la desviación estándar del ruido se divide por tres [35]. De esta manera, si se quiere eliminar el ruido de un píxel, se promedia el color de nueve píxeles que se asemejen a ese color, así el ruido se divide por tres o por cuatro si se promedia el color de 16 píxeles similares.

La eliminación del ruido de las imágenes puede hacerse con el algoritmo de medios no locales de eliminación de ruido (“*NLMDenoising*”, non local means denoising) [28] con variaciones computacionales. Los píxeles que más se parecen al píxel que se quiere eliminar el ruido, generalmente no están cerca a este, así que se debe escanear toda la imagen para encontrarlos. Con el algoritmo *NLMDenoising*, la eliminación del ruido se realiza calculando la media de píxeles que más se asemejan, con la variación que el parecido se evalúa mediante la comparación con una ventana alrededor de cada conjunto de píxeles. Este algoritmo se escribe:

$$N Lu(p) = \frac{1}{C(p)} \int f(d(B(p), B(q)))u(q) dq \quad (2)$$

donde:

- $(B(p), B(q))$ es la distancia euclideana entre los píxeles de la imagen centrada a p y q respectivamente.
- f es una función decreciente.
- $C(p)$ es el factor de normalización.

2.1.6 Binarización

La binarización es una técnica de procesamiento de imágenes donde los únicos valores numéricos que puede tener cada píxel son 1 y 0 (Blanco y negro). Para determinar qué valor tendrá cada píxel, se necesita un valor numérico que se llama umbral. Los píxeles cuyos valores de intensidades (escala de grises) que se encuentren por debajo de este umbral tomaran un nuevo valor numérico de 0 (negro) y los valores de intensidades que estén por encima de este umbral tomaran valores de 1 (blanco) [28]. La ventaja de esta técnica es la de poder especificar el valor del umbral cuando estamos buscando objetos con colores específicos, de esta manera se pueden separar objetos o regiones de interés del fondo.

- **Método Otsu**

El método de binarización Otsu, es una técnica para determinar el valor del umbral óptimo de la imagen maximizando la varianza (dispersión de los niveles de intensidad) de las clases (diferentes valores de intensidades). El método calcula el valor umbral de tal manera que la dispersión sea lo más alta posible entre las clases [28].

Una de las ventajas que tiene este método de binarización es que no es necesario el ingreso de valores manualmente para su correcto funcionamiento, ya que sus cálculos están basados en el histograma de la imagen [28], además, este método tiene excelentes resultados para ambientes los cuales no son controlados (luminosidad y ruido aleatorio).

2.1.7 Operaciones morfológicas

Las operaciones morfológicas en las imágenes digitales se basan en la teoría de conjuntos, simplifican y mejoran la imagen conservando sus principales características y eliminando distorsiones y ruidos. Las operaciones morfológicas son ampliamente usadas en el pre-procesamiento (eliminar ruido y simplificación de formas), pero también pueden ser utilizadas en la detección de objetos y descripción de objetos (área, perímetro, tamaño, etc.) [28].

Las operaciones morfológicas más utilizadas son la erosión y dilatación [28], a partir de estas dos operaciones se crean las operaciones de apertura y cierre, las cuales son muy utilizadas para lograr la representación de formas, la descomposición y la extracción de características.

- **Dilatación**

La dilatación es una operación morfológica donde se expande la imagen de acuerdo al tamaño del elemento estructurante [28]. Dada una imagen **A** y un elemento estructurante **B**, la dilatación se define como la siguiente operación matemática de conjuntos:

$$A \oplus B = \{x | (B)_x \cap A \neq \emptyset\} \quad (3)$$

- **Erosión**

La operación morfológica de erosión es la dual de la dilatación, en pocas palabras se puede describir a la erosión como la reducción de la imagen de acuerdo al tamaño del elemento estructurante [28]. Dada una imagen **A** y un elemento estructurante **B**, la erosión se define como la siguiente operación matemática de conjuntos:

$$A \ominus B = \{x | B_x \subseteq A\} \quad (4)$$

- **Apertura y cierre**

Como se aprecia en los puntos anteriores, la operación de dilatación expande la imagen y la operación de erosión la reduce. A partir de utilizar estas dos operaciones de acuerdo a cuál se aplica primero, nacen las operaciones de apertura y cierre.

La operación morfológica de apertura suaviza los contornos de la imagen, esta operación utiliza primero la operación de erosión para eliminar ruidos (regiones más pequeñas que el elemento estructurante) y luego aplica la dilatación con el mismo elemento estructurante para recuperar la información de los objetos o regiones [28].

La operación de cierre elimina pequeños huecos que tenga la imagen, aplicando primero la operación de dilatación para rellenar espacios más pequeños que el elemento estructurante y luego aplicando la operación de erosión para eliminar la información adicional de las regiones u objetos [28].

2.2 Medidores de energía eléctrica

Un medidor de Energía Eléctrica es el conjunto de elementos electromecánicos o electrónicos que se utilizan para el registro del consumo de energía eléctrica, tanto activa como reactiva, y en algunos casos su demanda máxima [36]. Existen varios tipos de medidores [36] dependiendo de la construcción, tipo de energía que miden, clase de precisión y conexión a la red eléctrica.

2.2.1 De acuerdo con la construcción:

- **Medidor de Inducción o Electromagnético**

Es un medidor en el cual las corrientes en las bobinas fijas reaccionan con las inducidas en el elemento móvil o disco, haciéndolo mover. El principio de funcionamiento es muy similar al de los motores de inducción y se basa en la teoría de la relación de corriente eléctrica con los campos magnéticos.

- **Medidores Estáticos o Electrónicos**

Medidores en los cuales la corriente y la tensión actúan sobre elementos de estado sólido (electrónicos) para producir pulsos de salida y cuya frecuencia es proporcional a los Vatios-hora ó Var-hora. Están contruidos con dispositivos electrónicos y son de mayor precisión que los electromagnéticos.

2.2.2 De acuerdo a la energía que miden:

- **Medidores de Energía Activa.**

Miden el consumo de energía activa en kilovatios – hora.

- **Medidores de Energía Reactiva**

Miden el consumo de energía reactiva en kilovares – hora.

Los medidores electrónicos integran estas dos clases de medida, es decir que miden tanto la energía activa como la energía reactiva.

2.2.3 De acuerdo a la exactitud.

De acuerdo a la Norma NTC 2288 y 2148 [36], los medidores se dividen en las siguientes clases: 2, 1, 0.5 y 0.2 Estos valores significan los límites de error porcentual admisible para todos los valores de corriente entre el 10% nominal y la corriente máxima con un factor de potencia igual a 1.

- **Medidores clase 2.**

Se incluye medidores monofásicos, bifásicos para medir energía activa en casas oficinas, locales comerciales y pequeñas industrias con cargas inferiores a 45 kVA.

- **Medidores clase 1.**

Incluye los medidores trifásicos para medir energía activa y reactiva de grandes consumidores. Para cargas mayores a 45 kVA Se exige que sean medidores electrónicos.

- **Medidores clase 0.5**

Se utilizan para medir a grandes consumidores. Cuando el usuario es no regulado o la tarifa es horaria, el medidor debe tener un puerto de comunicación o modem para enviar la información a través de la línea telefónica.

- **Medidores clase 0.2**

Se utilizan para medir la energía activa suministrada en bloque en punto de frontera con otras empresas electrificadoras o grandes consumidores alimentados a 115 kV

Los medidores electrónicos de energía activa, deben cumplir con la Norma NTC 2147 “Medidores Estáticos de Energía Activa. Especificaciones Metrológicas para clase 0.2S y 0.5S” y NTC 4052 “Medidores Estáticos de Energía Activa para corriente alterna clase 1 “ [36]. El índice de clase es el número que expresa el límite de error porcentual admisible para todos los valores de corriente entre 0.1 veces la corriente básica y la corriente máxima ó entre 0.05 veces la corriente nominal y la corriente máxima con un $\text{Cos } \Phi = 1$.

2.2.4 De acuerdo con la conexión a la red y según su construcción (elementos de tensión y corriente)

- **Monofásico bifilar.**

Se utiliza para el registro de consumo en una acometida que tenga un solo conductor activo o fase y un conductor no activo o neutro. Es el medidor de uso más frecuente es en las instalaciones residenciales. Este compuesto por una bobina de tensión y una de corriente. Su capacidad normalmente es de entre 15 y 60 A.

- **Monofásico trifilar.**

Se utiliza para el registro del consumo de una acometida monofásica de fase partida (120/240 V) donde se tienen dos conductores activos y uno no activo o neutro.

- **Medidor bifásico trifilar.**

Se utiliza para el registro del consumo de energía de una acometida en B.T (Baja Tensión) de dos fases y tres hilos, alimentadas de la red de B.T de distribución trifásica tetrafilar. Se usa para medir la energía consumida por aparatos que requieran para su funcionamiento dos fases a 220 voltios, como por ejemplo motores de menos de 10 HP o aires acondicionados hasta 12000 BTU/H.

- **Medidor trifásico trifilar.**

Se utiliza para registrar el consumo de energía de una acometida trifásica de tres fases sin neutro.

- **Medidor trifásico tetrafilar.**

Se utiliza para registrar el consumo de una acometida trifásica en B.T. de tres fases y cuatro hilos. Se utiliza para medir la energía consumida por aparatos que requieran funcionar con tres fases a 220 voltios, como por ejemplo motores de más de 10 HP. Están compuestos por tres (3) bobinas de tensión y tres (3) bobinas de corriente.

El resumen de la clasificación de los medidores de energía eléctrica de acuerdo a sus características se encuentra en la Tabla 1 y el rango de medición de los medidores de energía eléctrica se observa en la

Tabla 2.

Tabla 1. Resumen de la clasificación de medidores [36]

Grupo	Clasificación
Según el principio de funcionamiento	Medidores de Inducción (Electromecánicos)
	Medidores Estáticos (Electrónicos)
Según su construcción (elementos de tensión y corriente)	Medidor Monofásico Bifilar
	Medidor Bifásico Trifilar
	Medidor Trifásico Tetrafilar
Según la energía a medir	Energía Activa
	Energía Reactiva
Según su conexión a la red	Conexión Directa
	Conexión Semidirecta
	Conexión Indirecta
De acuerdo a la exactitud	Clase 2
	Clase 1
	Clase 0.5
	Clase 0.2

Tabla 2. Rango de medida de los medidores [36]

Servicio	Valores enteros
Monofásico bifilar 120V ó 127 V	Entre 1 y 3 kVA
Bifásico trifilar 220/127 V	Entre 3 y menor 9 kVA
Trifásico tetrafilar 220/127 V *	Entre 9 kVA y menor a 30 kVA
Semidirecta B.T (TC's)**	Entre 30 kVA y menor a 225 kVA
Indirecta M.T. y A.T. (TC's y TP's)	Igual o superior a 225 kVA

2.3 Verificación de medidores de energía eléctrica

La verificación en sitio es una tarea que debe realizar todas aquellas empresas que presten el servicio de suministro eléctrico y tengan elementos de medida con la necesidad de cuantificar el consumo de sus clientes o usuarios [37]. Esta aporta evidencia objetiva de que un elemento satisface los requisitos especificados tanto por el fabricante como por la misma empresa. Para realizar las inspecciones a los equipos de medida existen varias pruebas tales como: La prueba de consumo instantáneo en el medidor, la prueba de corrientes en la acometida, la prueba de relación de corriente, la prueba de relación de potencial, la prueba tiempo potencia, la prueba de exactitud y la prueba de integración o dosificación.

Las verificaciones que realiza la CEOC se dividen en:

- **Verificación en laboratorio:** es aquella que se realiza en las instalaciones de CEOC con equipos especializados y ambientes controlados [38].
 - **Verificación inicial:** se realiza a un medidor antes de ser instalado en una red eléctrica (medidores nuevos)
 - **Verificación posterior:** se realiza a un medidor retirado de su sitio de instalación para comprobar el estado en el cual venía funcionando (medidores usados)
- **Verificación en sitio:** es aquella que se realiza a un medidor en su sitio de instalación, con el fin de validar sus condiciones de funcionamiento físico y metrológico [37].
 - **Verificación inicial:** se realiza luego de 6 meses de usos para los medidores electromecánicos y 12 meses para medidores estáticos (electrónicos)

- **Verificación posterior:** 2 años para medidores de clase 0.2s, 4 años para medidores de clase 0.5s y 10 años para medidores de clase 1, 2 y 3.

2.3.1 Verificación en sitio

Las empresas que brindan el servicio de suministro eléctrico se deben regir por normas y estándares que garanticen la veracidad de este proceso. La más relevante en este caso es la norma técnica colombiana NTC 5900 “Verificación en sitio de equipos para medición de energía eléctrica” [37]. Como lo expresaba el jefe de Innovación y Desarrollo de la CEOC (Jorge Eliecer Millán) y el Coordinador del Laboratorio de Calibración y Ensayo de la CEOC (Freddy Carmona Aguirre), cuando los clientes o la empresa sospechan del incorrecto funcionamiento de uno de sus medidores, se realizan las pruebas técnicas de verificación en sitio. Cuando los clientes realizan una PQR (Peticiones, quejas y reclamos) a la CEOC sobre el mal funcionamiento del medidor, las brigadas técnicas que verifican el medidor se llaman Brigadas PQR. Cuando la CEOC a través de variables estadísticas determina que un medidor no funciona correctamente (perdidas de energía), las brigadas técnicas que verifican estos medidores se llaman Brigadas MACRO, las cuales son programadas en los sitios donde se sospecha que hay pérdidas de energía. Cuando se va a atender un PQR o están en campañas MACRO, la compañía envía una brigada técnica conformada por dos operarios y un supervisor. Las pruebas técnicas que se realizan en la verificación en sitio se pueden ver en la Figura 3.

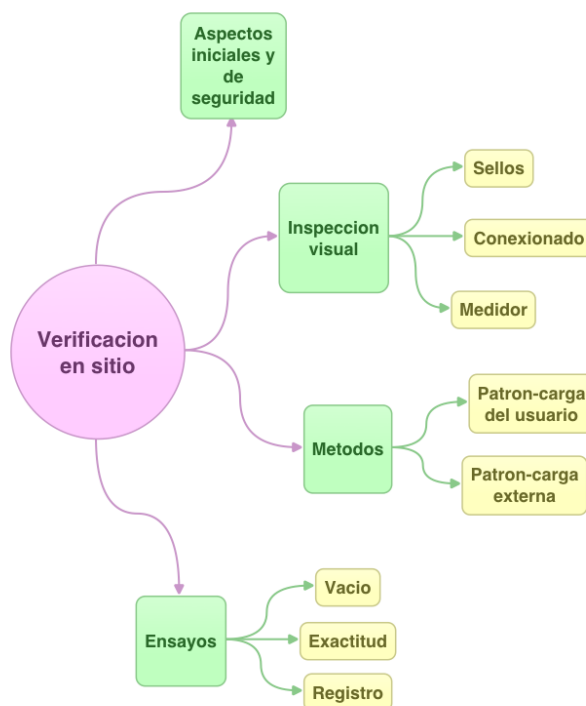


Figura 3. Actividades realizadas en la verificación en sitio [39]

Los métodos de ensayo son lo que validan el buen funcionamiento del equipo. La prueba de interés para el proyecto es la prueba de registro, también conocida por los nombres de: prueba de verificación de la constante o prueba de dosificación.

En la verificación en sitio, actualmente se utilizan los elementos presentados en la Tabla 3.

Tabla 3. Elementos para la verificación en sitio

	
<p>Analizador verificador de medidores (AVM): el equipo principal, el cual contiene todos los módulos de las pruebas a realizar y al cual se conectan los otros elementos.</p>	<p>Carga fantasma: este equipo simula un consumo controlado de alta, media y baja tensión para las pruebas de exactitud y dosificación.</p>
	
<p>TGO: este elemento se conecta vía Bluetooth con el AVM y le provee una interfaz gráfica de las funcionalidades que este trae.</p>	<p>Multímetro: Instrumento electrónico portátil para medir directamente magnitudes eléctricas activas dentro de la prueba.</p>
	

<p>Probador de fase: se usa para medir alta tensión en cables. Esto incluye fase contra fase y fase contra tierra.</p>	<p>Cámara digital: se usa para tomar las fotos en la prueba de dosificación y para llevar el registro fotográfico del proceso.</p>
	
<p>Sensor de pulsos: se usa para contar los pulsos que emite un led del contador en la prueba de exactitud. La velocidad de intermitencia es según el tipo de contador.</p>	<p>Impresora portátil: se usa para imprimir los comprobantes del proceso tanto para la CEOC, la Unión de Trabajadores de la Industria Energética Nacional (UTEN) y el usuario.</p>

El proceso de verificación en sitio se realiza de la siguiente manera:

1. Se toman los datos personales del usuario (Nombre, cedula, numero de vivienda, dirección, etc.).
2. Se acondiciona el lugar y se hace la debida señalización de seguridad.
3. Se procede a desconectar la carga de la vivienda o el sitio donde está el suministro.
4. Se realiza la conexión de los elementos de verificación como se indica en la Figura 4.

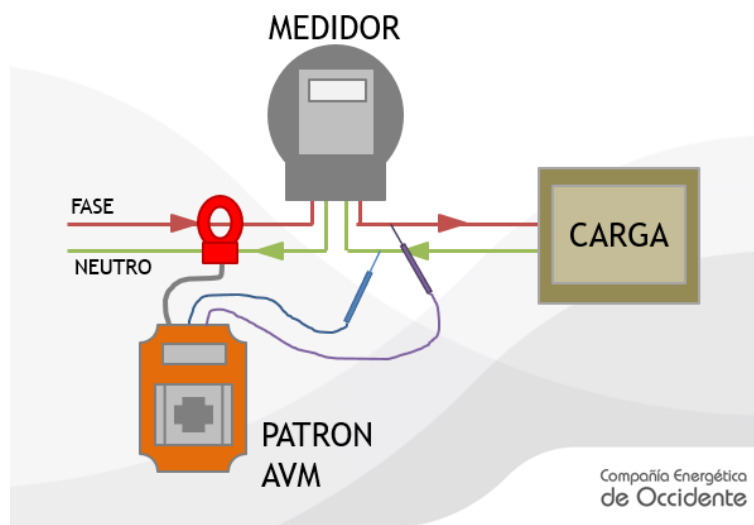


Figura 4. Conexión de los elementos de potencia [39]

- Se realiza la conexión y sincronización del AVM con su terminal TGO como se puede observar en la Figura 5.



Figura 5. Conexión de los elementos y su terminal de mando [40]

- La TGO cuenta con una interfaz y un sistema operativo Windows. La aplicación que maneja para la toma de datos y las distintas pruebas del proceso se llama sistema SYSMOVIL y su interfaz se puede observar en la Figura 6.

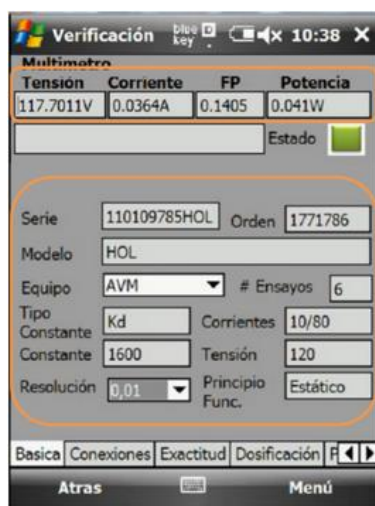


Figura 6. Interfaz del SYSMOVIL en la TGO [40]

- Se realiza la toma de datos y las pruebas según el orden que SYSMOVIL tiene en los menús inferiores de izquierda a derecha, comenzando con la opción básica.
- Una vez finalizada la verificación, el brigadista detalla lo que se realizó e imprime los comprobantes tanto para la empresa como para el usuario.

2.3.2 Prueba de dosificación

Para continuar se debe recordar la diferencia entre calidad de la medida y verificación de la medida. La calidad de la medida se realiza en un laboratorio certificado que reúne unas condiciones especiales para comprobar que los medidores se encuentran dentro del rango para el que fueron fabricados. La verificación de la medida se realiza en el terreno sin las condiciones ideales con las que cuenta el laboratorio, además interviene el factor humano y por tanto lo que se establece es que el medidor se encuentre dentro de un rango establecido por cada comercializadora, generalmente entre el $\pm 10\%$ de error [41]. Teniendo como base la explicación anterior, cabe resaltar que lo que se realiza en el terreno es la verificación de la medida para identificar anomalías evidentes o medidores fuera de rango para llevarlos al laboratorio para que allí entreguen un veredicto final.

La prueba de dosificación consiste en comparar la potencia en la acometida o red (KW) contra los KWh registrados por el medidor en un intervalo de tiempo.

Una prueba de tiempo potencia puede resultar dentro del rango, pero no garantiza que se esté integrando en forma correcta el consumo. Por ejemplo, en un medidor electromecánico la piñonera (odómetros) puede estar manipulada o en los medidores electrónicos tener una falla de fabricación o también una alternación voluntaria.

En la Figura 7 se puede observar la interfaz de la prueba de dosificación en SYSMOVIL.



Figura 7. Interfaz de la prueba de dosificación en SYSMOVIL [40]

Para realizar la prueba de dosificación se requiere de lo siguiente:

- El medidor debe contar con dos decimales en el odómetro.
- Se debe medir el factor de potencia a través del AVM

- La carga en un lapso de tiempo debe ser estable. Se recomiendan 120 segundos
- Se debe tomar la lectura inicial y la hora precisa de inicio de la inspección y luego la lectura final y la hora de terminación.

Actualmente, la prueba de dosificación se realiza después de la prueba de exactitud con los equipos mencionados anteriormente [40], así:

- Conectar el AVM a la carga fantasma.
- Tomar la lectura inicial del medidor: esto se realiza con una foto y el brigadista ingresa los números de la imagen a la TGO en la opción de “Lecturas”, en el recuadro “Inicial”.
- Se conecta el cable de dosificación de energía: este cable permite dosificar la energía necesaria para la prueba.
- Se aplica una cantidad de energía conocida al medidor (Ver Tabla 4): esto se hace con la carga fantasma y el cable de dosificación de energía. Según la resolución del medidor se debe dosificar la cantidad de energía especificada.

Tabla 4. Resolución del medidor vs Energía dosificada [39]

Resolución	Energía mínima dosificada (KWh)
0.001	0.02
0.01	0.2
0.1	2

- La energía dosificada se muestra en el recuadro de la pantalla de la TGO, que lleva el nombre “Energía Dosificada” y está dada en KWh.
- Se inicia la prueba desde la TGO: el brigadista presiona el botón “Iniciar” para comenzar con la prueba, en ese momento se activa el cronometro en el recuadro de la pantalla de la TGO con el nombre “Tiempo”.
- En el recuadro de la TGO con el nombre “Energía” se puede observar la cantidad de energía que queda por dosificarse.
- Luego de un intervalo de tiempo, que depende de la clase del medidor, la prueba finaliza.
- El brigadista toma una nueva foto para verificar la medida del contador. Este valor es ingresado en la opción de “Lecturas”, en el recuadro “final”.
- La aplicación arroja el porcentaje de error del medidor en el recuadro de la pantalla de la TGO “% de error” rigiéndose por la siguiente ecuación:

$$E_p = \frac{E_m - E_{Pat}}{E_{Pat}} \quad (5)$$

donde:

- E_p : error porcentual
- E_m : energía en el medidor
- E_{Pat} : energía en el equipo patrón

- Dependiendo el porcentaje de error se verifica en el recuadro de la pantalla de la TGO “Resultado” si el medidor cumple o no cumple con la prueba.

2.4 Resumen

Se realiza una revisión literaria para extraer y recopilar los temas necesarios para el correcto desarrollo del proyecto y se explica la teoría fundamental del proyecto de acuerdo al planteamiento del problema.

De acuerdo a la explicación del funcionamiento y objetivos de la prueba de verificación y sobre todo la prueba de dosificación, se delimita el problema que el proyecto va a solucionar y se entiende correctamente la función y resultados que se esperan al desarrollar el proyecto.

Capítulo 3. Análisis de tecnologías

En este capítulo se analiza la información relacionada con los motores OCR, y plataformas de desarrollo en sistemas operativos móviles. También se recopila información de librerías de procesamiento de imágenes de licencia gratuita para conseguir extraer e identificar los caracteres existentes dentro de imágenes digitalizadas.

3.1 Sistema operativo móvil

Desde la formulación del problema de la CEOC, el funcionario encargado del departamento de I+D+i definió que requería hacer una transferencia de tecnología para realizar la prueba de verificación en sitio con los *Smartphone*, se sabe que la nueva generación de *Smartphone* trae muchas actualizaciones hardware y software que pueden ser muy útiles tanto en su funcionamiento como en la economía de los procesos que se deben hacer.

Después de definir la tecnología hardware que se va a utilizar (*Smartphone*) se procede a investigar qué sistema operativo es el mejor para un entorno industrial, teniendo en cuenta las condiciones donde se realizan las pruebas de verificación en sitio de los medidores.

En la Tabla 5 se encuentran los datos de las ventas de teléfonos inteligentes en el mundo, relacionado con los sistemas operativos móviles en el primer cuarto de 2016 y primer cuarto de 2015 [42].

Tabla 5. Ventas de teléfonos inteligentes en el mundo relacionado con los sistemas operativos en el primer cuarto de 2016 y primer cuarto de 2015 [42]

Sistema Operativo	1Q16 (Unidades)	1Q16 (Porcentaje del mercado)	1Q15 (Unidades)	1Q15 (Porcentaje del mercado)
Android	293771,2	84,1	264941,9	78,8
iOS	51620,5	14,8	60177,2	17,8
Windows	2399,7	0,7	8270,8	2,5
BlackBerry	659,9	0,2	1325,4	0,4
Otros	791,1	0,2	1582,5	0,5
Total	349242,4	100	336297,8	100

Según el estudio realizado por la empresa consultora Gartner Inc. [42] actualmente existen muchos sistemas operativos móviles con diferentes características y para diversos tipos de celulares, los sistemas operativos que se encuentran en la mayoría de *Smartphone* son:

- Android, se encuentra en el 84,1% de los *Smartphone* del mercado.
- iOS se encuentra en el 14,8% de los *Smartphone*.
- Windows Phone en el 0,7%.

El sistema operativo Android es *Open Source* y actualmente funciona en una gran cantidad de *Smartphone* de diferentes marcas y características, caso contrario de iOS y Windows Phone que su sistema operativo se encuentra solo en un *Smartphone* específico, iPhone para iOS y Microsoft Lumia para Windows Phone.

3.1.1 Sistema operativo Android

Android es un sistema operativo desarrollado por Google, de código abierto basado en GNU/Linux. El software es libre, la fuente está completamente disponible, gracias a que se liberó el SDK (Kit de desarrollo de software). Las aplicaciones para el sistema operativo se escriben en el lenguaje Java, aunque se pueden utilizar API's (interfaz de programación de aplicaciones) para realizar aplicaciones en diferentes tipos de lenguaje [43].

El modelo de seguridad de Android, presenta ciertas limitaciones, muchos de los aspectos de la seguridad en Android dependen de la correcta construcción de las aplicaciones por parte de los desarrolladores, pero estos no cuentan con una documentación precisa para poder interactuar correctamente con el modelo de seguridad, por lo que dichas aplicaciones pueden llegar a ser inseguras [44]. Además, existe la vulnerabilidad que pueden surgir a partir de las características introducidas por los "pending intents", estos crean la posibilidad de que una aplicación le conceda su propia identidad a otra para realizar una operación determinada [45].

• Android Studio

Android Studio es un IDE (entorno de desarrollo integrado) para la realización de aplicaciones para la plataforma Android. Android Studio desde el 2014 reemplazó a Eclipse como IDE oficial de Android. Android Studio es gratuito ya que se encuentra en el mercado bajo la licencia Apache, este IDE se encuentra para sistemas operativos como Microsoft Windows, MAC OS y GNU/Linux (Ubuntu) [43].

3.1.2 Sistema operativo iOS

IOS es un sistema operativo propietario, desarrollado por Apple Inc, este sistema operativo es usado solo por los *Smartphone* iPhone [46]. Una ventaja de este sistema operativo respecto al Android es que funciona solo en una marca de *Smartphone* con características muy potentes de hardware (cámara y procesamiento), así que cuando se realizan aplicaciones para este sistema operativo se puede aprovechar estas características, caso contrario a Android, ya que se encuentra en una gran variedad de *Smartphone*, si se realiza una aplicación que se necesita de una cámara con una muy buena resolución, esta aplicación no serviría en *Smartphone* cuya resolución de cámara sea muy baja, así se tenga el Sistema Operativo Android [16].

- **Xcode**

Xcode es un IDE desarrollado por Apple Inc. para el desarrollo de aplicaciones para iOS, este IDE es propietario, pero se encuentra gratuito en el sistema operativo MAC OS. Este IDE permite la programación de aplicaciones en diferentes lenguajes de programación como JAVA, C, C++, Swift, y AppleScript [46].

3.1.3 Sistema operativo Windows Phone

Windows Phone (WP) es un sistema operativo móvil desarrollado por Microsoft, se encuentra bajo la licencia de software propietario. Actualmente este sistema operativo se encuentra en la marca de *Smartphone* Microsoft Lumia (Windows Phone y Nokia) teniendo la misma ventaja de iOS respecto a Android [47].

- **Visual Studio**

Visual Studio es un IDE desarrollado por Microsoft para sistema operativo Microsoft Windows, soporta lenguajes de programación como C++, C#, Visual Basic, .NET, F#, Java, Python, Ruby y PHP. Descargando el SDK de Windows Phone es posible crear aplicaciones para este sistema operativo móvil [47].

3.2 Bibliotecas de procesamiento de imágenes

Para la realización de aplicaciones de visión de máquina, es necesario la utilización de bibliotecas que contengan funcionalidades de procesamiento de imágenes digitales. Existe gran cantidad de bibliotecas propietarias y libres para diferentes sistemas operativos móviles, siendo OpenCV una de las mejores para aplicaciones comerciales y el Toolbox de visión de máquina de Matlab para aplicaciones académicas.

3.2.1 OpenCV

OpenCV es una biblioteca de visión artificial desarrollada por Intel, funciona bajo la licencia BSD la cual le permite ser libre para aplicaciones tanto comerciales como académicas. Contiene más de 500 funciones para el procesamiento de imágenes, creando aplicaciones para visión artificial y visión robótica. El lenguaje de programación para OpenCV es C, C++, Python y Java, funciona en las plataformas GNU/Linux, Mac OS, iOS, Microsoft Windows y Android, para este último no trae todas sus funciones, pero si la mayoría. Esta biblioteca tiene más de 47mil personas en la comunidad OpenCV, y sus descargas son superiores a los 9 millones [48].

3.2.2 Image Processing Toolbox

Matlab es una herramienta software matemático bajo la licencia de software propietario; desarrollada por MathWorks, tiene su propio IDE y su propio lenguaje de programación llamado Lenguaje "M", está disponible para las plataformas GNU/Linux, Microsoft Windows, MAC OS y Unix. Matlab puede expandir sus capacidades matemáticas a través de herramientas software llamadas Toolbox [28].

Image Processing Toolbox proporciona a Matlab un gran conjunto de algoritmos y funciones para el procesamiento de imágenes. Este Toolbox puede analizar, segmentar, reducir ruido, transformar geométricamente las imágenes y muchas funciones más de visión de máquina. Académicamente este Toolbox es una herramienta muy importante para aprender sobre la visión de máquina, ya que gracias a sus tutoriales, aplicaciones y ejemplos es posible comprender paso a paso como se aplica la visión de máquina en ambientes académicos, cotidianos e industriales y sus resultados [28].

3.2.3 Leptónica

Leptónica es una biblioteca de código abierto orientado a la pedagogía de la visión artificial gracias a sus aplicaciones de ejemplo que trae. Es ampliamente utilizado en aplicaciones académicas con procesamiento y análisis de imágenes muy básico. Trae funciones básicas como transformaciones de imágenes, operaciones morfológicas binarias y operaciones aritméticas [49].

3.3 Motores OCR

El reconocimiento óptico de caracteres (OCR), consiste en la identificación automatizada de símbolos o caracteres a partir de una imagen real, de esta manera se puede interactuar con el texto reconocido de la imagen sin la necesidad de transcribirlo digitalmente [6]. Antes de identificar los caracteres de la imagen, se debe realizar una serie de pasos donde se aplican funciones de procesamiento digital de imágenes para poder segmentar las imágenes y trabajar solo con la sección binaria de la imagen que contiene los caracteres.

Después de haber encontrado la Región de Interés (ROI), el motor OCR separa individualmente los caracteres y los compara con imágenes o datos patrón y aplica una hipótesis probabilística para identificar el carácter de la imagen [26]. Para poder mejorar la hipótesis probabilística, algunos motores OCR pueden ser entrenados para un conjunto de caracteres, números, símbolos, letras o idioma.

3.3.1 Tesseract OCR

Tesseract OCR es un motor OCR desarrollado por Google desde el 2006, escrito en lenguaje C y C++ para las plataformas Windows, Linux y MAC OS. Tiene licencia Apache, así que puede ser usado gratuitamente para proyectos académicos y comerciales. Tesseract OCR es considerado uno de los motores OCR más preciso de código abierto disponibles actualmente [27].

En sus inicios Tesseract solo funcionaba para reconocer caracteres del idioma inglés, actualmente y con las nuevas actualizaciones, Tesseract está disponible para más de 100 idiomas, incluido el español. Tesseract compara matemáticamente los caracteres identificados con una base de datos de caracteres, cuando encuentra características idénticas a los patrones Tesseract define que la imagen corresponde a ese carácter. Tesseract usa el contorno del carácter para realizar polígonos aproximados, con estos

polígonos aproximados se extraen características para ser comparadas, aunque de esta manera es más ágil realizar las comparaciones, los polígonos aproximados es un arma de doble filo, ya que tanto el ruido como pequeña información del carácter es eliminado [26].

Para reconocer las palabras de una imagen, Tesseract procesa las palabras dos veces. En el primer paso, las palabras de éxito (palabras que están en un diccionario y no son peligrosamente ambiguas) se pasan a un clasificador adaptativo para obtener información. Tan pronto como el clasificador adaptativo tiene muestras suficientes, puede proporcionar el resultado de la clasificación, incluso en la primera pasada. En el segundo paso, las palabras que no eran lo suficientemente buenas, se procesan por segunda vez y ya que el clasificador adaptativo ha ganado más información desde la primera pasada obtendrá la suficiente información para proporcionar el resultado de la clasificación [50]. El proceso de reconcomiendo de palabras de Tesseract se puede observar en la Figura 8.

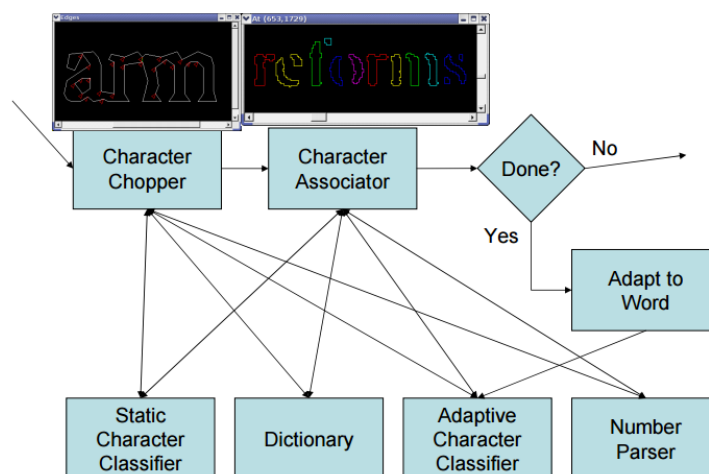


Figura 8. Diagrama de bloques de reconocimiento de palabras en Tesseract OCR [50]

Tesseract obtiene los contornos de la imagen (palabra) y luego los compara con todos los posibles contornos que puede llegar a tener un dígito y determina que dígito es (el más aproximado). En la Figura 9, se puede observar todos los posibles contornos que puede llegar a tener el número "1" al "6" para el diccionario (tessdata) inglés (*English*).

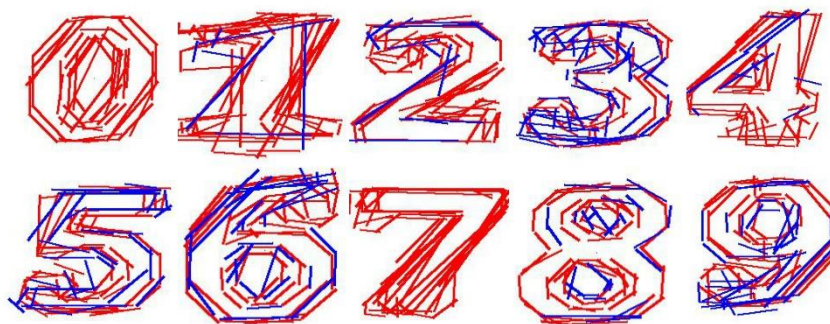


Figura 9. Posibles contornos que puede tener el número 1 al 9 [Fuente Propia]

En la Figura 9 se puede observar los posibles contornos que puede tener el número 1 al 9, de esta manera Tesseract compara los contornos de la imagen con los contornos de la base de datos y determina que dígito es (el más aproximado).

Aunque internamente Tesseract realiza un procesamiento a las imágenes que le son ingresadas, se debe hacer una segmentación a la imagen para mejorar la exactitud del motor OCR, las imágenes ingresadas no deben ser inferiores a 20 píxeles, por lo tanto, imágenes pequeñas se deben cambiar de tamaño, cualquier rotación que tenga el texto también será un problema para Tesseract, así que también deberá ser corregido, los bordes también deberán ser retirados o el motor lo reconocerá como un carácter más [14].

Tesseract OCR está escrito en el lenguaje C, así que se requiere utilizar el NDK de Android para ser programado en lenguaje Java para realizar aplicaciones para el sistema operativo móvil Android. Existen proyectos libres, donde se ha realizado una biblioteca Tesseract para Java, incluyendo una biblioteca de procesamiento de imágenes básica llamada Leptónica, de esta manera es posible realizar aplicaciones Android con Tesseract en lenguaje Java.

- Entrenamiento Tesseract

Para entrenar un nuevo lenguaje en Tesseract se debe definir sobre qué idioma se va a entrenar este lenguaje; si no se selecciona un lenguaje, Tesseract no logrará identificar los caracteres, ya que no sabrá qué forma deben tener. Por ejemplo, si se define el lenguaje hebreo y entrenamos un nuevo lenguaje con los caracteres de los dígitos segmentados, Tesseract no identificará estos números porque los dígitos en el idioma hebreo son muy diferentes y no se podrá entrenar el nuevo lenguaje [26].

3.3.2 JavaOCR

JavaOCR es una biblioteca de Java para el procesamiento de imágenes y reconocimiento óptico de caracteres; esta licenciado bajo software libre. Al estar escrita en lenguaje Java es posible crear aplicaciones Android sin ningún inconveniente. JavaOCR funciona como la mayoría de motores OCR, realiza procesamiento matemático a las imágenes segmentadas para encontrar coincidencias

en sus patrones comparándolas con caracteres y características de caracteres patrón [52].

Además de ser motor OCR, JavaOCR trae consigo funciones básicas de procesamiento de imágenes, como filtros, inversión de formatos, operaciones matemáticas y operaciones morfológicas binarias.

3.4 Resumen

En este capítulo se describieron las tecnologías software que deben ser utilizadas para el desarrollo del proyecto. Se mencionan sus diferentes tipos de uso y desarrolladores, además de su licencia de distribución y sus ventajas frente a otras tecnologías.

Tesseract OCR es el motor OCR más preciso de licencia gratuita en el mercado, además, cuenta con más de 100 diccionarios (tessdata) para el reconocimiento de caracteres de múltiples lenguajes. Su metodología de procesamiento de palabras le permite reconocer caracteres con poca información (contornos).

El sistema operativo Android es el sistema operativo móvil más usado en los dispositivos en el mundo. Ya que es software libre, muchos fabricantes adoptan este sistema operativo para sus equipos, así se obtiene una gran variedad de teléfonos inteligentes con múltiples características de resolución, procesamiento o cámaras. Además, existen actualmente muchos proyectos académicos desarrollados en este tema operativo que pueden ser adaptados o mejorados para proyectos específicos.

Capítulo 4. Diseño

En este capítulo del proyecto se escogen y se definen las tecnologías y metodologías usadas para el desarrollo del proyecto y la aplicación. Las tecnologías software se seleccionan de acuerdo a sus funciones, ventajas y licencias de distribución.

Las metodologías definidas para el procesamiento de las imágenes digitales se seleccionan bajo los procedimientos descritos por el motor OCR para obtener los mejores resultados posibles.

Existen varias metodologías para el desarrollo de aplicaciones móviles [53], los cuales se deben realizar en diferentes fases. Una de las primeras fases del diseño y desarrollo de una aplicación es la de selección y conocimiento de la tecnología.

4.1 Selección de tecnologías

La selección de tecnologías se realiza bajo el criterio de las características que sean más beneficiosas para la CEOC de acuerdo a su coste, fácil capacitación a operadores y resistente a ambientes industriales. También se analiza el criterio del tipo de licencias de distribución, académicas o licencias libres para investigaciones y comercialización.

4.1.1 Selección del sistema operativo móvil

El sistema operativo móvil seleccionado es Android, ya que este es un software libre y de código abierto que se encuentra en una gran variedad de dispositivos móviles con muy buenas características técnicas de cámara y capacidad de procesamiento.

El IDE seleccionado para desarrollar la aplicación en el sistema operativo Android, es el IDE Android Studio, ya que se encuentra con licencia Apache (permite al usuario del software dar la libertad de usar el software para cualquier propósito, para distribuirlo, modificarlo y distribuir versiones modificadas del software, sin preocuparse de las regalías [54]), es posible usarlo para propósitos comerciales y académicas de manera gratuita, además, este IDE puede ser usado en el sistema operativo Microsoft Windows.

El sistema operativo Android se encuentra en diversos modelos de dispositivos con diferentes capacidades, entre los más destacados, los Samsung Mobile que gozan de una reputación muy buena debido a sus características hardware tanto en *Smartphone* de gama media y gama alta, Huawei; aunque tiene *Smartphone* de gama media, sus *Smartphone* de gama alta tienen buenas características técnicas, los *Smartphone* Xiaomi y LG con nuevos lanzamientos al mercado muy competitivos gracias a sus características hardware.

4.1.2 Selección de bibliotecas de procesamiento de imágenes

La biblioteca seleccionada para el desarrollo de la aplicación móvil es la biblioteca OpenCV, ya que contiene una licencia BSD que permite realizar proyectos académicos y comerciales de manera gratuita. Esta biblioteca es una de las más poderosas en el ámbito académico, aunque para el sistema operativo móvil Android no contiene todas sus funcionalidades, contiene las necesarias para realizar una excelente aplicación.

Se decide trabajar con su versión 3.1 (21/12/2015) ya que es la versión más actualizada que tiene OpenCV para Android, aunque sus funcionalidades “Core” no trabajan muy bien en esta versión, está actualizada para soportar las últimas versiones Android, como por ejemplo Android 5 (Lollipop). El proceso de instalación de OpenCV en Android se encuentra documentado en el Anexo 2.

4.1.3 Selección del motor OCR

El motor OCR seleccionado es Tesseract OCR desarrollado por Google actualmente (primero Hewlett Packard), Tesseract OCR es considerado uno de los mejores motores OCR de software libre en cuanto a la precisión.

Aunque Tesseract OCR utiliza el lenguaje nativo C y C++, utilizando el NDK (Kit de desarrollo nativo) es posible implementar lenguajes de código nativo C y utilizar este motor de OCR en Android Studio. El proceso de instalación del motor OCR Tesseract se encuentra documentado en el Anexo 2.

Aunque se puede hacer uso del NDK para usar Tesseract en Android, se usará un proyecto de software libre llamado “Tess-Two” [55] el cual esta creado con *Tesseract-Android-tools* que es un conjunto de herramientas diseñadas para complicar el motor OCR Tesseract y la librería Leptónica en la plataforma Android [55].

4.2 Requisitos de la aplicación

Para el diseño del proyecto es necesario conocer los requisitos funcionales y no funcionales que debe tener la aplicación, para así cumplir con las necesidades de los operadores técnicos de las pruebas de verificación en sitio, los cuales serán los encargados de utilizar la aplicación.

4.2.1 Requisitos no funcionales

Para determinar los requisitos no funcionales que debe tener la aplicación, se analiza al usuario final (operadores técnicos) y su ambiente de trabajo para determinar que necesidades debe cumplir la aplicación.

Uno de los primeros análisis que se hace durante las visitas a las brigadas de verificación en sitio, es el ambiente donde se desarrollan, es un ambiente difícil para una aplicación de visión de máquina, ya que la luminosidad es completamente variable y no puede ser manipulada, debido a que la verificación en sitio se puede hacer en cualquier hora dentro del horario laboral, se hacen en hogares e industrias, se realizan

en ambientes urbanos (edificios) y ambientes rurales (arboles). De esta manera se determina que la aplicación debe ser completamente robusta a todo tipo de ambientes lumínicos.

Debido a que la TGO (corazón de la verificación en sitio) es lenta en cuanto al procesamiento, esto deriva a que la verificación en sitio tarda más de lo debido, se determina que, para hacer una excelente transferencia de tecnología para mejorar el proceso, la aplicación debe ser muy rápida, intuitiva y robusta en cuanto al hardware. Como se trata de una aplicación industrial, se determina; junto a los operadores técnicos, que el tamaño mínimo para el *Smartphone* utilizado debe ser de mínimo 4.7" y máximo 7", esto debido a que si el *display* es más grande, es más fácil manipular el *Smartphone*, pero si es demasiado grande, se hace más complicado su manipulación ya que se debería usar las dos manos para sostenerlo, el *Smartphone* debe ser robusto en su hardware, porque se considera un ambiente industrial, así que debe resistir a temperaturas elevadas (días soleados), a caídas y largos tiempo de uso.

4.2.2 Requisitos funcionales

La determinación de los requisitos funcionales de la aplicación se basa en la Normas Técnicas Colombianas NTC 4856 "VERIFICACIÓN INICIAL Y POSTERIOR DE MEDIDORES DE ENERGÍA ELÉCTRICA" [38] y NTC 5900 "VERIFICACIÓN EN SITIO DE EQUIPOS PARA MEDICIÓN DE ENERGÍA ELÉCTRICA" [37]. La norma técnica colombiana NTC 4856 establece el conjunto de operaciones que se deben realizar a los medidores de energía eléctrica para propósitos de verificación inicial y verificación posterior de los medidores en funcionamiento y la norma técnica colombiana NTC 5900 establece el procedimiento para la verificación en sitio de medidores de energía eléctrica en conexión directa clase 1 y clase 2.

Dentro de las operaciones que establece la norma técnica NTC 5900, existe la prueba de dosificación (llamada prueba de verificación de la constante en la NTC 4856 o prueba de registrador entre los operarios), esta prueba determina que el registrador del medidor funcione correctamente en los medidores ciclométricos (medidores con odómetro) y es la prueba que se desea automatizar basándose en la aplicación realizada en este proyecto.

La norma técnica NTC 5900 establece que las pruebas de verificación en sitio, solo se realizan a medidores de energía eléctrica ciclométricos, a los medidores de *display* digital no se realizan las pruebas. Además de la norma NTC 5900, las directrices de la CEOC establecen que los medidores que contengan registrador de manecillas, registrador de solo enteros y puentes de tensión externa no es necesario realizar las pruebas de verificación en sitio y deben ser cambiados bajo el causal "Avance tecnológico".

Basándose en las normas técnicas citadas y los criterios de las directrices de la CEOC, se establece que la aplicación es dirigida para medidores de energía eléctrica ciclométricos que tengan una resolución mínima de 0,01 KWh. Además, la aplicación debe ser desarrollada de tal manera que ingresando los valores característicos de los medidores (resolución, clase y potencia) que establece la NTC 5900, la aplicación

debe informar de una manera clara y concisa el resultado de la prueba de dosificación (cumple o no cumple) [37].

Ya que la aplicación desarrollada se define como una aplicación industrial, el color del fondo de la pantalla de la aplicación es basado en el desarrollo de HMI industriales, donde se establece que el fondo de pantalla de los HMI debe ser de color Arena o gris plata para no distraer o cansar la vista de los operadores e indicarles visualmente las variables críticas con colores primario [56]. La pantalla principal donde es ingresada y desplegada la información de la prueba de dosificación debe considerar el movimiento del ojo del operario que va de arriba a abajo y de izquierda a derecha [56], así que la información más importante debe ir arriba y la navegación o pasos para la realización de la prueba debe ir en orden de izquierda a derecha o de arriba abajo. La navegación no debe ser un obstáculo a las acciones del operador para realizar la prueba, además, el área de los botones debe ser lo suficientemente grande para que la aplicación sea fácil de usar y el texto que describe la función o procedimiento debe ser corto y conciso [56].

La información textual presentada al usuario no debe tener más de tres fuentes en la interfaz, no debe tener más de tres tamaños diferentes, el tamaño de la fuente debe ser seleccionado de tal manera que pueda ser leído a cierta distancia y el color del texto debe contrastar con el fondo de pantalla [56].

4.2.3 Plan de dispositivos

De acuerdo a los requisitos funcionales y no funcionales expuestos anteriormente, se definen los dispositivos finales que se le recomienda a la organización (CEOC) para usar en las pruebas de verificación en sitio. Estos dispositivos llamados *Smartphone Rugged* (Robusto), son instrumentos destinados a ambientes industriales, donde la temperatura o humedad tiene valores elevados, y debido a este entorno, pueden soportar caídas de gran altura o mal manejo de los operarios.

Se definen estos dispositivos debido a que cumplen con la norma internacional IEC 60529 (*Degree of Protection*) [57], de esta manera, de acuerdo al grado de protección IP de la norma anterior, soportan ambientes con polvo o partículas en el ambiente y son resistentes al agua (salpicaduras, lluvia o inmersión en el agua), también soportan temperaturas elevadas, resistentes a caídas accidentales y su batería permite más horas de uso respecto a los dispositivos convencionales.

En la Tabla 6 se realiza una comparación de los dispositivos robustos recomendados a la CEOC.

Tabla 6. Comparación de dispositivos robustos

Dispositivo	Pantalla táctil		Sistema				Comunicación			Características			Especificación				
	Tamaño	Resolución	Procesador	RAM	ROM	SO	Conexión	Wifi	Bluetooth	Cámara	Micro SD	Batería	Protección	Dimensiones	Peso	Temperatura	Precio
RugGear RG730	5"	720X1280	1.3GHz QudCore	2 GB	16 GB	Android 5.1	4G LTE	IEEE 802.11	4.0	13MP 5MP	64 GB	3020 mAh	IP68	156X82X14mm	245 g	-20°C +60°C	1782831COP
APES PRO RG700	5.3"	960X540	1.0GHz DualCore	1 GB	8 GB	Android 4.1.3	4G HSPA	IEEE 802.11	4.0	8MP 2MP	32 GB	2900 mAh	IP68	155X82X15mm	245 g	-20°C +60°C	1854285COP
CAT S60	4.7"	1280X720	1.5GHz Qualcomm	3 GB	32 GB	Android 6.0	4G LTE	IEEE 802.11	4.1	13MP 5MP	128 GB	3800 mAh	IP68	147X73X12mm	223 g	-25°C +55°C	1711617COP
CAT S40	4.7"	540X960	1.1GHz Qualcomm	1 GB	16 GB	Android 5.1	4G LTE	IEEE 802.11	4.1	8MP 2MP	64 GB	3000 mAh	IP68	144X74X12mm	185 g	-25°C +55°C	1064503COP
MTT Master 4G	4.97"	1920X1080	1.7GHz OctaCore	2 GB	16 GB	Android 5.1	4G LTE	IEEE 802.11	4.0	13.2MP 5MP	64 GB	2800 mAh	IP68	152X80X12mm	234 g	-25°C +55°C	1585430COP
MTT Performance	5"	1280X720	1.2GHz QuadCore	1 GB	8 GB	Android 5.1	4G LTE	IEEE 802.11	4.0	8MP 2MP	32 GB	3000 mAh	IP67	147X76X12mm	217 g	-25°C +55°C	967435COP

4.2.4 Definición de la arquitectura

Para el desarrollo de aplicaciones móviles, existen muchas opciones de arquitecturas [58], escoger la adecuada para la aplicación es un paso muy importante para aprovechar al máximo el uso de las tecnologías. Las arquitecturas existentes se pueden definir en 3 categorías, Fuera de línea, en línea y sincronizados [58].

- La arquitectura “Fuera de línea” o aplicaciones nativas son las aplicaciones que no necesitan que el *Smartphone* esté conectado a Wifi o a una red de datos móviles. Significa que la aplicación funcionará normalmente así no esté conectado a Internet.
- La arquitectura “en línea” son las aplicaciones que necesitan estar conectadas a Internet para su correcto funcionamiento. Estas aplicaciones están dentro de las aplicaciones de servidores web, donde la aplicación debe estar en continua comunicación con servidores para compartir información necesaria de la aplicación. Una de las ventajas de esta arquitectura es que el *Smartphone* solo debe enviar la información necesaria al servidor, no debe correr algoritmos pesados o tediosos, este trabajo queda encargado en el servidor cuyas características técnicas sean superiores al del *Smartphone*.
- Las aplicaciones de sincronización son aquellas que funcionan cuando están o no conectados a internet. Estas aplicaciones suelen ser llamadas clientes inteligentes, porque cuando el *Smartphone* está conectado a internet descarga los datos que necesite para funcionar correctamente, así cuando el cliente no está conectado a internet con los datos anteriormente descargados puede funcionar normalmente, no a su máxima capacidad, pero si sus funcionalidades básicas.

La arquitectura seleccionada para el desarrollo de la aplicación de este proyecto, es la arquitectura nativa o “fuera de línea”, esto debido a que no es necesario estar conectado a una red móvil de datos para su funcionamiento adecuado, así cuando las brigadas estén en una verificación de sitio en una zona rural, no se deben preocupar por la red móvil del *Smartphone*, ya que no es necesario. Con esta arquitectura se debe aprovechar toda la capacidad hardware del dispositivo móvil para que la aplicación funcione es su máxima capacidad.

4.3 Diseño

El diseño de cualquier aplicación móvil; ya sea industrial o no, debe ser de tal manera que sea muy sencilla de utilizar, funcionamiento muy simple e intuitiva. La aplicación debe ser diseñada de tal manera que el usuario pueda acceder al contenido primordial o importante en un lapso de tiempo muy corto, sin navegación entre pantallas complejas o confusas [53].

El nombre que tendrá la aplicación es “SLNM” (Sistema de lectura numérico de medidores); basándose en el nombre del proyecto y su logotipo se puede observar en la Figura 10.



Figura 10. Logotipo aplicación desarrollada [Fuente propia]

El logotipo de la aplicación se diseñó de tal manera que se simulara un medidor. Las letras simulan una pantalla LCD representando los dígitos del medidor (lectura) y la flecha circular guía al usuario como se lee el nombre “SLNM” y representa el disco del medidor que gira de acuerdo el consumo del usuario, siendo este disco lo más representativo de los medidores de energía eléctrica.

4.3.1 Navegación entre pantallas

La navegación entre pantallas fue diseñada de tal manera que los operarios técnicos de la prueba de verificación en sitio puedan acceder a la interfaz y datos de la prueba de dosificación de una manera muy intuitiva. Para poder usar la aplicación, los operadores deberán iniciar sesión con un usuario y una contraseña, la aplicación se diseñó para que los operadores no puedan crear usuarios, asegurando que personas ajenas a la compañía usen la aplicación, así que, para usar la aplicación, la CEOC deberá asignar usuarios y contraseñas a los operadores con anterioridad, de esta misma manera solo la CEOC puede crear usuarios y contraseñas, pero no desde la aplicación.

Para realizar la navegación y funcionalidad de las diferentes pantallas de la aplicación móvil, se utilizaron diagramas y prototipos de desarrollo Android, tales como:

- **Análisis jerárquico de tareas (HTA)**

Los diagramas HTA explicitan en qué orden realizar las acciones. En cada paso se pueden dar condiciones que permitan ir a un paso o a otro en función de las acciones del usuario [59].

Incluye dividir una tarea en tareas más simples, de forma recursiva. Éstas se agrupan como planes que especifican cómo se deben llevar a cabo esas tareas en la práctica [60].

HTA se focaliza en las acciones observables y físicas, e incluye acciones no relacionados con el software o con el dispositivo de interacción [61]. El proceso para desarrollar el diagrama HTA es el siguiente:

- Comenzar con un objetivo de usuario que se examina de forma que se puedan identificar las tareas principales que hay que realizar para lograrlo.

- Las tareas se subdividen en sub tareas: el proceso puede proseguir mediante refinamientos progresivos hasta llegar al grado de granularidad requerido.

El diagrama HTA para la aplicación tiene como usuario objetivo los Brigadistas Técnicos de la CEOC y la funcionalidad es secuencial, empezando en la pantalla de bienvenida hasta la de guardar los datos de la prueba como se puede observar en la Figura 11.

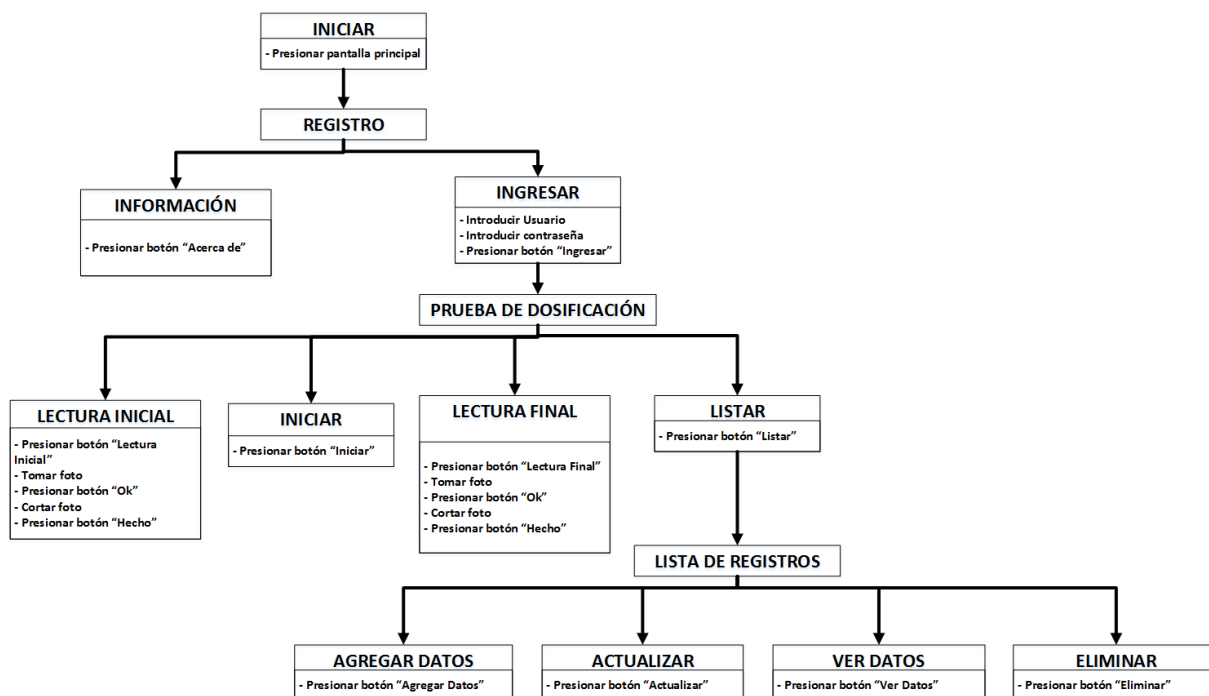


Figura 11. Diagrama HTA de la aplicación SLNM [Fuente Propia]

- **Prototipo de papel + mapa navegaciones.**

Permite hacerse una idea visual del orden en que se ejecutan las pantallas del prototipo. Puede sugerir nuevos caminos de ejecución. Este mapa en particular es muy sencillo ya que permite intuir la interacción entre las diferentes pantallas que ofrece la aplicación [62] [63]. En algún momento debe decirse el nivel de detalle máximo o mínimo que se quiere alcanzar.

El mapa de navegación general de la aplicación muestra las diferentes pantallas a las cuales el usuario puede acceder, y la jerarquía que existe entre dichas pantallas (Ver Figura 12).



Figura 12. Pantallas de la aplicación SLNM [Fuente Propia]

La aplicación tiene una pantalla principal en la cual se realiza toda la prueba de dosificación. Esta pantalla cuenta con una navegación específica que permite cumplir los pasos que establece la CEOC para llevar a cabo esta prueba. En la Figura 13 se presenta la secuencia para llevar a cabo el proceso de registro de la lectura a través de la aplicación SLNM.

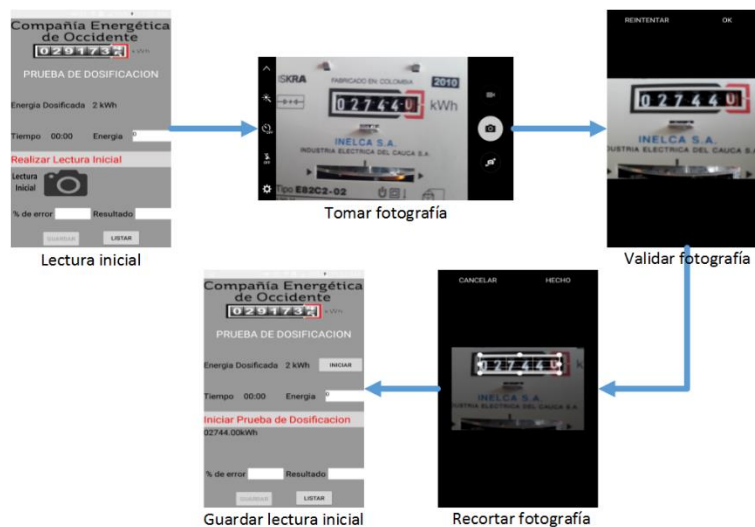


Figura 13. Pantallas de toma de lectura a través de SLNM [Fuente Propia]

4.3.2 Metodología de procesamiento de las imágenes.

El objetivo principal del proyecto es la identificación precisa del valor de la lectura del medidor. La norma NTC 5900 exige que mínimo se deben tener lecturas de dos valores decimales, esto significa que se debe registrar una lectura hasta las centésimas. En los medidores aptos para la prueba de dosificación existe solo hasta el odómetro de valor decimal.

Antes de ingresar la imagen del medidor al motor OCR para lograr identificar la lectura, se debe segmentar la imagen de tal manera que solo se encuentran los valores numéricos de los odómetros para ser identificados, de esta manera cuando se aplica el OCR este logrará resultados más rápidos y precisos.

Antes de definir los pasos y metodologías que usara la aplicación, cabe destacar que en el capítulo siguiente se presentarán los resultados obtenidos al aplicar las metodologías que se expondrán en este capítulo del proyecto.

Tesseract realiza varias operaciones de procesamiento de imágenes internamente (utilizando la librería Leptónica) antes de aplicar el OCR real. Por lo general, hace un muy buen trabajo de esto, pero inevitablemente habrá casos en que no es lo suficientemente bueno, lo que puede resultar en una reducción significativa en la precisión del resultado. La documentación de Tesseract [51] define una serie de pasos que se deben realizar a la imagen para obtener el mejor resultado posible del reconocimiento óptico.

- **Re escalado:** Tesseract funciona mejor en imágenes que tienen al menos 300dpi (puntos por pulgada), así que se debe cambiar el tamaño de las imágenes cuando no cumplen esta condición.
- **Binarización:** Tesseract realiza esta operación internamente, pero el resultado puede no ser el más óptimo, sobre todo si el fondo de la página es desigual. Se debe realizar una binarización de tal modo que se emita todo o la mayoría del fondo.
- **Eliminación de ruido:** El ruido es la variación aleatoria del brillo o del color de una imagen, que hace que el texto de la imagen sea difícil de leer. Ciertos tipos de ruido no pueden ser removidos por Tesseract en la etapa de binarización, esto causa resultados erróneos, por lo tanto, el ruido debe ser eliminado en su totalidad.
- **Rotación:** Una imagen sesgada es cuando una página o un conjunto de caracteres se encuentra torcido. La calidad de la segmentación de Tesseract se reduce significativamente si una página es demasiado sesgada, lo que afecta gravemente la calidad. Para hacer frente a esto, la imagen debe ser girada de tal modo que las líneas del texto sean horizontales.
- **Eliminación de bordes:** Las imágenes de texto a menudo tiene bordes oscuros alrededor de ellos, estos bordes pueden ser interpretados erróneamente como caracteres adicionales. Se debe eliminar los bordes que tenga la imagen para asegurar un correcto reconocimiento.

Basados en los pasos sugeridos por Tesseract [51], se define el procedimiento que usara la aplicación para segmentar la imagen individualmente para cada dígito, y ser ingresado al motor OCR para obtener resultados más ágiles y precisos. La metodología define un procedimiento que se hará al odómetro decimal para calcular el valor de las centésimas y milésimas de la lectura.

- **Selección ROI:** La aplicación tendrá la capacidad de identificar y seleccionar el mínimo rectángulo que encierra los odómetros del medidor. Esto se realiza

basándose en el modelo de color HSV para lograr identificar el color rojo que diferencia al odómetro decimal, luego se genera el mínimo rectángulo que encierra este odómetro. Encontrado el mínimo rectángulo se genera una trayectoria basándose en el ángulo de este, hasta llegar al primer odómetro (6.8 veces el ancho del mínimo rectángulo). Al final se obtiene las coordenadas del mínimo rectángulo que encierra los odómetros del medidor, estas coordenadas son utilizadas para recortar la imagen original y obtener solo la región de interés.

- **Procesamiento de imágenes en escala de grises:** El procesamiento de imágenes en escala de grises se utiliza para mejorar el contraste de la imagen, pero primero se debe realizar la eliminación de ruido a través de los algoritmos de medios no locales de eliminación del ruido, esto es muy importante, ya que permite eliminar las componentes que no contienen información que al aplicarlos al OCR obtendrían resultados equivocados. Después de eliminar el ruido de la imagen, se mejora el contraste de esa ecualizando el histograma
- **Binarización de la imagen:** Para binarizar la imagen en escala de grises, se debe seleccionar un valor de umbral de tal manera que se logre identificar solo los valores numéricos. Se debe aplicar una técnica de binarización para que el color rojo del odómetro decimal sea seleccionado como fondo; gracias a la ecualización del histograma el color rojo pierde intensidad y es más fácil la binarización a través de técnicas estadísticas de distribución como la técnica de Otsu. El ruido que todavía este presente es eliminado con operaciones morfológicas como la operación de apertura, la cual aplicándose en dos interacciones puede eliminar regiones más pequeñas que el elemento estructurante, obteniendo solo los valores numéricos de los odómetros.
- **Re escalado:** La librería OpenCV permite conocer el tamaño de la imagen, si esta es tiene una resolución menor a 300dpi se debe cambiar su tamaño a través de las funciones de OpenCV (*Resize*).
- **Rotación:** En la selección del ROI cuando se define el mínimo rectángulo que encierra los odómetros, es posible conocer el ángulo que tiene este. A través del valor del ángulo, la imagen debe ser rotada de tal manera que este ángulo sea cero.
- **Segmentación individual de los dígitos:** En este paso se debe separar los dígitos de los odómetros. Esto se realizará a través de la identificación de las regiones y ya que está la imagen binarizada y libre de ruido, las regiones identificadas serán solo los dígitos numéricos del medidor.
- **Identificación de transición:** Después de obtener las ROI donde se encuentran los dígitos numéricos, se procede a determinar si los odómetros tienen valores numéricos que están en transición (ver Figura 14), para esto se usará la identificación de regiones en el ROI. Si se encuentra que en alguna de las ROI existen más de una región, se interpreta que el dígito está en transición de valor y si contiene solo una región significa que no está en transición.



Figura 14. Registrador del medidor con odómetro decimal en transición numérica de 2 a 3 [Fuente Propia]

- Identificación de centésimas y milésimas:** Este procedimiento es el aporte investigativo más importante del proyecto, ya que en el estado del arte no se encuentra documentación al respecto, se decide abordar este problema a través de métodos estadísticos y procesamiento digital de imágenes. Se hará solo a los dígitos que se encuentran en el odómetro decimal. Se define un método para reconocer el valor de las centésimas y milésimas basado en el método estadístico de media ponderada. El promedio ponderado o media ponderada es el promedio que se obtiene asignando distintos pesos (grado de importancia) a los valores estadísticos [64]. Cuando se obtiene la proporción de tamaño (ver Figura 15) de los dígitos a través de las propiedades de las regiones, esta proporción será equivalente al peso o grado de importancia. Si en el odómetro decimal no existe transición (100/0) significa que no hay valores de centésimas y milésimas (XXXXX, 000).

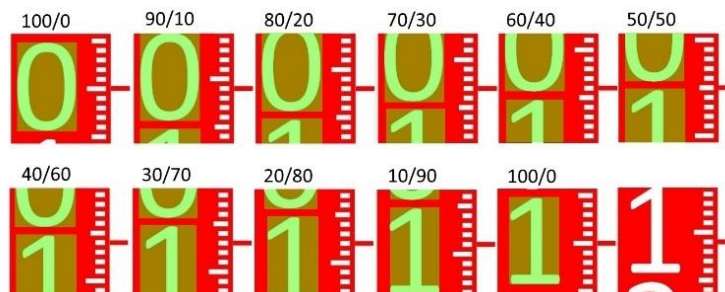


Figura 15. Proporción de transición de valores numéricos en el odómetro decimal [Fuente Propia]

- Aplicación del motor OCR:** Este será el último paso para la identificación de los valores numéricos de los odómetros. Se ingresarán las imágenes segmentadas individualmente (cada dígito) al motor OCR para la identificación.

4.4 Resumen

Las tecnologías seleccionadas (bajo criterios funcionales de la CEOC) son tecnologías con licencias de distribución gratuitas, de esta manera la aplicación que se cree a partir de este proyecto puede ser distribuido internamente por la CEOC sin ningún tipo de

problema, además puede ser modificado y mejorado en un futuro con nuevos parámetros o metodologías de desarrollo.

La metodología definida para la obtención de la lectura precisa del medidor de energía eléctrica, están basadas en metodologías o pasos definidos por el motor OCR seleccionado y las librerías usadas. Se define una nueva metodología para la lectura de los valores numéricos que se encuentran en transición y los valores de centésima.

Capítulo 5. Implementación

En este capítulo se describe y se presentan los pasos y metodologías del desarrollo de la aplicación móvil; como es la interfaz y la navegabilidad se detallan la implementación de los algoritmos utilizados para el procesamiento de imágenes para determinar el valor de la lectura del medidor incluyendo las centésimas y milésimas.

Se presentan los resultados obtenidos en las pruebas hechas en el laboratorio de la Universidad del Cauca y el laboratorio de la CEOC, además, se presentan las correcciones a los algoritmos basados en los resultados de laboratorio y las limitaciones que tendrá la aplicación en las pruebas de verificación en sitio.

5.1 Desarrollo

Toda aplicación móvil comercial o industrial debe ser desarrollada de tal manera que la navegación sea muy sencilla de entender, no se deben ser navegaciones confusas para llegar a la interfaz principal de la aplicación. La interfaz desarrollada para la aplicación del proyecto se diseñó para que el usuario final (operadores técnicos) lleguen a la interfaz principal de una manera muy intuitiva y rápida, dentro de la interfaz principal los botones son habilitados de acuerdo a la actividad que se esté desarrollando en ese momento de la prueba de dosificación. Además, en la interfaz principal se encuentra un *TextView* donde se describe y se presenta la actividad que debe ser desarrollada, basándose en la interfaz de la prueba de dosificación que actualmente están usando los operadores técnicos, así será más fácil la aceptación y adaptación de los operadores a esta nueva aplicación.

El objetivo principal de la aplicación a desarrollar es la obtención automática de la lectura numérica de los medidores de energía eléctrica, para esto se hará el uso del motor OCR Tesseract. Antes de ingresar la imagen al motor OCR, esta se debe segmentar de tal manera que solo se encuentren los valores numéricos de los odómetros, de esta manera el OCR reconocerá la lectura de una manera más ágil y además se asegurará que solo realiza la lectura de los valores numéricos y no otros caracteres como el fabricante, marca, características, etc.

En el mercado y en la ciudad de Popayán, existe una gran cantidad de medidores en funcionamiento en la zona rural y urbana. La norma técnica colombiana usada para la verificación en sitio y directrices de la CEOC especifican que solo a los medidores ciclométricos (electromecánicos y eléctricos) que presentan valor decimal (resolución mínima de 0,1 KWh) se les debe realizar la prueba de dosificación, pero, bajo estas características específicas sigue existiendo una gran cantidad de medidores en Popayán.

Se definen dos medidores de energía eléctrica, un medidor electromecánico y un medidor eléctrico, para la realización de la prueba de dosificación por parte de la aplicación del proyecto, bajo estos resultados en laboratorio y en campo se realizarán

las conclusiones pertinentes, los medidores seleccionados son (de izquierda a derecha como se observa en la Figura 16):

- Medidor Monofásico Bifilar clase 2 ISKRA E82C2-02 de resolución 0,01.
- Medidor Monofásico Bifilar clase 2 TECUN DDS994 de resolución 0,01.

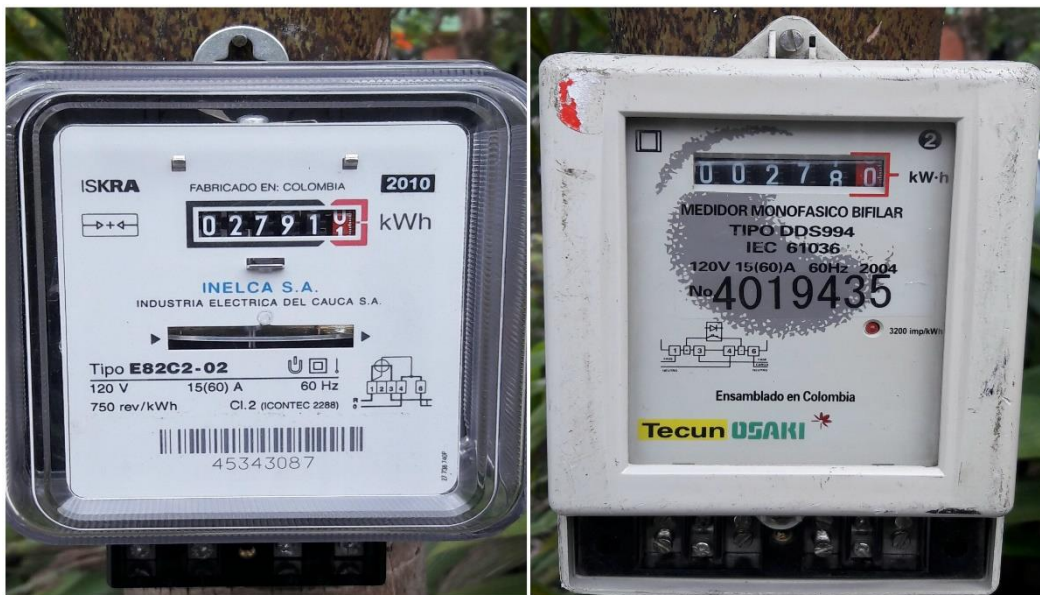


Figura 16. Medidor ISKRA y Medidor TECUN [Fuente Propia]

Estos medidores encajan en las características especificadas por la norma técnica colombiana NTC 5900 [37] y la CEOC para la prueba de dosificación, además, estos contadores se encuentran en mayor cantidad en la zona urbana de Popayán, así que la aplicación desarrollada puede usarse en una mayor proporción de medidores.

5.2 Navegación de la aplicación

La navegación entre pantallas se desarrolló para un usuario final específico, los cuales son los operadores de las brigadas técnicas de la CEOC, por tal razón debería ser fácil y rápida de utilizar (ver Anexo 1). Con base en lo anterior se describirá un poco cada pantalla:

- **Pantalla inicial:** esta pantalla contiene una imagen de publicidad de la CEOC como se observa en la Figura 17 y el usuario solo deberá tocar cualquier parte de la pantalla para acceder a la siguiente pantalla.



Figura 17. Pantalla inicial de la aplicación SLNM [Fuente Propia]

- **Pantalla de registro:** en esta pantalla el operario deberá iniciar sesión con un Usuario y Contraseña previamente asignados por la CEOC, si los datos son correctos podrá ingresar a la pantalla principal presionando el botón “Ingresar”, de lo contrario no podrá y tendrá la posibilidad de seguirlo intentando. La pantalla de registro se observa en la Figura 18. Además, desde esta pantalla se puede acceder libremente a la pantalla de información con tan solo presionar el botón “Acerca de”.



Figura 18. Pantalla de registro de la aplicación SLNM [Fuente Propia]

- **Pantalla de información:** esta pantalla contiene información sobre las personas involucradas en el desarrollo del proyecto, la Universidad del Cauca, así como el número de la versión de la aplicación. La pantalla de información se observa en la Figura 19.

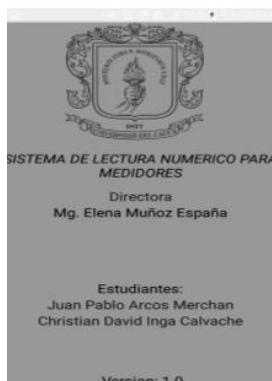


Figura 19. Pantalla de información de la aplicación SLNM [Fuente Propia]

- Pantalla principal:** esta es la pantalla que realiza la prueba de dosificación. La funcionalidad de los botones se activa secuencialmente según el orden establecido por la CEOC. Cuando se acceda a la pantalla, el usuario podrá ver que la interfaz que lo encamina al inicio de la prueba, ya que solo está activo el botón de “Lectura inicial”, sin embargo, se agregó un visor de texto que le indicará al usuario por dónde empezar. Como se dijo anteriormente, el primer paso es tomar la lectura inicial (ver Anexo 1) del medidor eléctrico, para ello el usuario deberá presionar el botón “Lectura inicial” el cual lo llevará a la cámara del *Smartphone* para que tome la foto y seleccione la región de interés. Una vez realizado esto, el usuario validará la imagen y automáticamente la aplicación le entregará la lectura en esta misma pantalla. Cuando se termine de hacer la primera lectura se habilitará únicamente el botón “Iniciar”. Cuando el operario pulse dicho botón, comenzará la dosificación de la energía establecida y el conteo del tiempo que tardara la dosificación. Cuando concluya la dosificación de energía y el tiempo de la prueba, se activará únicamente el botón de “Lectura final”. El operario tendrá que hacer los mismos pasos de la lectura inicial y la aplicación de igual forma arrojará el valor de esta lectura. Con estos dos valores la aplicación entregará un porcentaje de error de la lectura que le permitirá conocer si el medidor cumple o no cumple con la prueba. Cuando el operario termine, se dará un veredicto que se podrá observar en el visor de resultado. Además, solo hasta el final se activará el botón de “Listar”, el cual lo enviará a la base de datos de la aplicación. La pantalla principal se observa en la Figura 20.



Figura 20. Pantalla principal de la aplicación SLNM [Fuente Propia]

- Pantalla de listar:** esta pantalla contiene la base de datos de la aplicación, la cual almacenará los valores recogidos en la prueba. Aquí se listan las lecturas, tanto inicial como final, el porcentaje de error de la prueba y el resultado, el cual determina el buen o mal funcionamiento del medidor. Cuenta con los botones de guardar, actualizar, ver y borrar datos, los cuales pueden ser utilizados para realizar las acciones descritas por cada uno de ellos. El objetivo de esta pantalla es poder llevar un historial de los datos obtenidos en las pruebas para luego ser utilizados por la CEOC si fuese el caso. La pantalla de listar se observa en la Figura 21.



Figura 21. Pantalla de listar de la aplicación SLNM [Fuente Propia]

5.3 Funcionalidad de los botones

Para asegurar que la aplicación cumpla con su función principal, se han diseñado una serie de restricciones en los botones, principalmente en la pantalla donde se lleva a

cabo la prueba de dosificación. Estas restricciones hacen que los botones se habiliten de tal forma que la prueba de dosificación se realice de forma secuencial sin dar paso a errores de uso. Además de eso, la pantalla cuenta con una barra de texto, la cual le indica al usuario los pasos que debe ir haciendo a medida que avance en la prueba. Las acciones que se deben ejecutar para realizar la prueba de dosificación con la aplicación son:

- Tomar lectura inicial
- Iniciar dosificación de energía
- Tomar lectura final
- Guardar resultados

El funcionamiento de los botones es el siguiente:

En pantalla principal, el único botón que aparece habilitado es el de “Lectura Inicial” el resto están inhabilitados o invisibles. En la barra de ayuda aparece un letrero con el mensaje “Realizar lectura inicial”, indicándole al usuario que es el primer paso que debe hacer. Cuando el usuario realice la lectura inicial se habilitará el botón “Iniciar” y en la parte donde se encontraba el botón de lectura inicial aparecerá el resultado de dicha lectura. El usuario tendrá que presionar el botón “INICIAR” para empezar la dosificación de la energía, tal como se observa en la Figura 22.

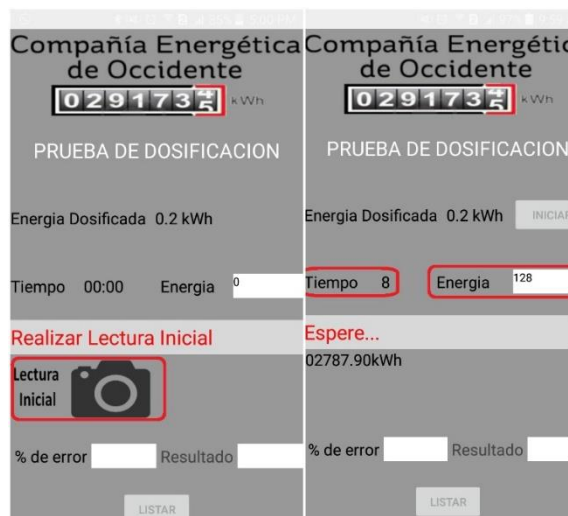


Figura 22. Habilitación de los botones “Lectura inicial” y “INICIAR” [Fuente Propia]

Luego de iniciar con la dosificación de energía, el botón “INICIAR” se deshabilita, impidiendo que el usuario reinicie la prueba. Cuando finalice la dosificación de energía, aparecerá en pantalla el botón “Lectura Final” y un mensaje “Realizar lectura final” que le indica al usuario lo que debe hacer. Cuando se realiza la lectura final, este resultado es superpuesto en el lugar donde se encontraba el botón “Lectura Final” y se habilita el botón “LISTAR” como se observa en la Figura 23.



Figura 23. Habilitación de los botones “Lectura Final” y “LISTAR” [Fuente Propia]

Como método de precaución, se implementó una función para el botón de regreso o salida que tienen todos los *Smartphone*. Cuando el operario presione de forma voluntario o involuntaria este botón (el de regreso o salida) en cualquier momento de la prueba, la aplicación abre un cuadro de dialogo preguntándole si desea salir de la aplicación. El operario podrá presionar “Cancelar” para seguir en la prueba de dosificación o “Aceptar” para salir o devolverse a la pantalla de inicio de sesión (ver Figura 24).



Figura 24. Cuadro de dialogo para confirmar la salida de la prueba de dosificación [Fuente Propia]

5.4 Identificación de la lectura numérica

Los algoritmos utilizados para el procesamiento digital de imágenes en la aplicación desarrollada permiten que el resultado final sea la identificación individual de los dígitos de los odómetros para obtener valores más precisos, ya que se logrará identificar los valores de los números en transición y el valor de centésima. A continuación, se detalla la implementación y utilización de los algoritmos.

5.4.1 Selección ROI

La región de interés (ROI) es aquella región de la imagen que contiene la mayor o toda la información necesaria para el propósito final del procesamiento. Para el caso de los medidores de energía eléctrica y el proyecto, la ROI principal es donde se encuentran los valores numéricos (odómetros).

En la Figura 25, se observa que los valores numéricos se encuentran dentro de un rectángulo formado por una franja de color negro y color rojo. También se puede observar que el odómetro de valor decimal tiene el fondo de color rojo (para indicar la décima), así, si se logra identificar el color rojo (la franja y el fondo del odómetro), se logrará segmentar los odómetros.



Figura 25. Imagen digital de un medidor de energía eléctrica [Fuente Propia]

Se desarrolló un algoritmo con la capacidad de identificar y seleccionar solo el color rojo de la imagen. Para lograr identificar solo el color rojo, el algoritmo realiza una

conversión de modelos de color, RGB a HSV, la conversión de modelo de color a HSV permite saber los tonos de matices de los colores en RGB. Para hallar el color rojo de la imagen se debe seleccionar un rango de matiz de 0° a 60° , un rango de saturación de 0% a 100% y rango de valor de 0% a 100%; se selecciona un rango de matiz de 60° ya que dentro de este rango se encuentra solo el matiz rojo, para la saturación y valor se selecciona todo el rango posible, ya que de esta manera se pueden identificar toda la posible gama de color rojo, ya sea claro u oscuro.

En la Figura 26 se observa el resultado del algoritmo cuando es usado en una imagen de los odómetros de un medidor de energía eléctrica.



Figura 26. Identificación y selección de color rojo de una imagen de los odómetros [Fuente Propia]

El algoritmo de detección de rojos, encierra con un mínimo rectángulo toda la región de píxeles encontrados, de esta manera el algoritmo puede hacer una correcta selección de ROI sin importar como se realiza la toma de la imagen al contador como se observa en la Figura 27.

Para seleccionar correctamente la ROI, se debe identificar la región de color rojo del odómetro de valor decimal, si se identifica solo esa región, se puede saber el valor del ancho que tiene y ya que este odómetro es igual a los otros 5 odómetros que le siguen, se puede seleccionar la ROI como 6.8 veces el ancho de esa región; se selecciona como 6.8 veces el ancho porque entre los odómetros existe una distancia aproximada de $\pm 0.5 \text{ cm}$.



Figura 27. Obtención de la ROI (color azul) en diferentes imágenes de un medidor [Fuente Propia]

En la Figura 27 se observa todo el funcionamiento del algoritmo desarrollado, de color magenta se observa el mínimo rectángulo que encierra a la región identificada como color rojo, de color verde se encuentra el mínimo rectángulo que encierra a la región del odómetro de color rojo y de color azul es el rectángulo que selecciona la ROI de acuerdo al ancho del rectángulo verde.

De acuerdo a la luminosidad del ambiente, el algoritmo puede sufrir variaciones, cuando la luminosidad es baja, identifica menor intensidad del color rojo del odómetro, así que el mínimo rectángulo será menor (muy oscuro), por lo tanto, la ROI final no seleccionará todo el primer odómetro, lo recortará un poco. En el caso contrario, cuando la luminosidad es mucha (luz artificial) existen regiones donde se identifican como color rojo (brillo) y el mínimo rectángulo que encierra el rojo del odómetro puede ser más ancho que el odómetro, así que la ROI final será más ancho que los odómetros. El algoritmo funciona muy bien con la luminosidad natural del sol. Después de las 4 de la tarde, de acuerdo a las zonas donde se use el algoritmo, el ambiente puede estar muy oscuro y fallar y en las noches cuando se usa la luz artificial también presenta fallos.

Este algoritmo solo debe ser usado en medidores con odómetro decimal de color rojo. Para los medidores que no tienen odómetro de color rojo, el algoritmo puede ser modificado de una manera muy sencilla. Se modifica de tal manera que identifique el color rojo de la franja que encierra los odómetros, así, de acuerdo al tamaño de la franja, puede ser calculado el mínimo rectángulo que encierra los odómetros, esto se realiza a través del cálculo de proporción del tamaño (físico) del registrador del medidor y las franjas que encierran los odómetros (ver Figura 28). Se puede realizar solo si se tiene el conocimiento de las medidas del registrador.



**Figura 28. Rectángulos identificados según el color rojo (odómetro o franja)
[Fuente Propia]**

Existen también medidores muy antiguos (en funcionamiento) que no tienen ni el odómetro decimal de color rojo, ni la franja de color rojo que encierra los odómetros (ver Figura 29), para estos medidores la ROI debe ser seleccionada de manera manual; esto es realizado con la función Android *crop*, la cual me permite recortar la imagen manualmente, pero esta función solo existe en Android 4.4 (*Kit Kat*) o superior.



Figura 29. Diferentes tipos de odómetros de medidores [Fuente Propia]

La función *crop* de Android permite que el usuario realice una selección de la ROI manualmente como se observa en la Figura 30, de esta manera se asegura una ROI correcta, pero se necesita de más tiempo (individualmente de la persona que lo use). Las ventajas de esta función es que se puede aprovechar las funciones de *zoom* para acercar la imagen y lograr una mejor segmentación del ROI si la captura de la imagen se hizo a cierta distancia. La función *crop* tiene una mejor respuesta proporcionalmente a la resolución de la cámara del teléfono inteligente.



Figura 30. Función crop de Android para la selección manual del ROI [Fuente Propia]

5.4.2 Procesamiento en escala de grises y binarización

Después de haber identificado la ROI, el siguiente paso es el procesamiento para la eliminación de ruido y obtener solo los valores numéricos en una imagen binaria 2-D (blanco y negro). El primer paso es la conversión del modelo de color, cuando se ha hallado el rectángulo donde está la ROI, se recorta la imagen RGB de acuerdo a ese rectángulo, luego se transforma la imagen de RGB a una imagen 2-D en escala de grises. La eliminación del ruido se hace a través del algoritmo de medios no locales de eliminación de ruido, luego para mejorar el contraste se ecualiza el histograma; en este punto la intensidad del color rojo del odómetro de valor decimal ha bajado, así que cuando se utiliza la técnica de Otsu para la binarización se separará mejor los valores numéricos del fondo.

En este punto la imagen binaria contiene regiones que no son valores numéricos, así que este ruido debe ser eliminado. La eliminación del ruido se realiza con morfología binaria, a través de la operación morfológica de apertura (erosión y dilatación) con un elemento estructurante elipse, el resultado obtenido se observa en la Figura 31.



Figura 31. Procesamiento en grises y binario en imagen segmentada [Fuente Propia]

El resultado de aplicar los algoritmos de morfología en grises y binaria es muy bueno, obteniendo solo los valores numéricos de los odómetros. Pero en ambientes con niveles de luminosidad altos este algoritmo no funciona de manera óptima, ya que el brillo en los odómetros es interpretado como valores numéricos, y la imagen binaria resultante tiene ruido que las operaciones morfológicas no eliminaron como se observa en la Figura 32.



Figura 32. Resultado del procesamiento en escala de grises y binarización en un ambiente con nivel de luminosidad alto [Fuente Propia]

Debido a que las brigadas técnicas realizan la prueba de dosificación en ambientes de luminosidad muy variables (altos y bajos) se desarrolló otro método de binarización basándose en el algoritmo de segmentación de imágenes a través del color rojo. Se transforma la ROI de RGB a HSV, y bajo la misma idea de detección de rojos se detecta el color blanco.

Ya que los valores numéricos están de color blanco, con el modelo de color HSV es posible segmentar la imagen con mejores resultados. Para el color blanco, el rango de matiz y saturación puede ser cualquiera, siempre y cuando el rango del valor se establezca en el máximo. Cuando se establece todo el rango de blanco posible, el algoritmo logra segmentar algo de ruido (brillo), así que el resultado no es el mejor, pero cuando el rango de blanco es seleccionado de tal manera que no identifique el blanco más intenso (brillo) el resultado es mejor que el algoritmo de morfología en grises y binaria (ver Figura 33). Para segmentar mejor la imagen, es utilizada la operación morfológica binaria de apertura, eliminando ruido (si existe) sin eliminar información de los dígitos.



Figura 33. Resultado de la metodología de morfología en grises y binaria y la metodología de detección de blancos con el modelo HSV [Fuente Propia]

En la Figura 33 se observa que la metodología de detección de blancos con el modelo HSV obtiene un mejor resultado, ya que contiene más información de los dígitos, así se logrará entrenar de una mejor manera el motor OCR.

5.4.3 Re escalado y rotación

El re escalado de la imagen binaria se realiza a través de la función de OpenCV llamada "*Resize*", para obtener 300 dpi se debe cambiar de tamaño la imagen para obtener una medida mínima de 50 píxeles, ya sea en altura o ancho.

Para saber que tan sesgada se encuentran los dígitos segmentados, se identifica el mínimo rectángulo que encierre los dígitos, y de acuerdo al ángulo que tenga este, se debe rotar la imagen este valor.

5.4.4 Segmentación individual de los dígitos

Uno de los desafíos de la aplicación, es lograr identificar y reconocer los valores numéricos de los odómetros que se encuentran en transición, para reconocer estos valores numéricos se debe identificar los dígitos individualmente. Para lograr esto, se debe determinar cuántas regiones existen en la imagen, después se determinan que regiones se encuentran en la misma (o cerca) de la coordenada x y estas regiones son interpretadas como un odómetro (un dígito).

Existen dos casos para cuando hay dos o más regiones cerca de la misma coordenada x (un odómetro), el primer caso es cuando el número está en transición; por eso hay dos regiones, el segundo y peor caso es cuando existe ruido en la imagen segmentada.

Para determinar que una región es ruido y debe ser eliminada, se deben comparar los tamaños de las regiones encontradas, si una región es 95% mayor que la otra, la de menor tamaño será interpretada como ruido y será eliminada, de esta manera solo quedará la región de mayor tamaño y esta será el dígito que será ingresado al OCR; como se puede observar en la Figura 34. Si sucede que una región es mayor en un 95% respecto a la otra y no es ruido sino un dígito en transición, pero se elimina de igual manera, el error que se ingresa al resultado del error ponderado final es de tan solo $\pm 0,0005$.



Figura 34. Dígito segmentado individualmente con ruido, antes y después de determinar que la región más pequeña debe ser eliminada [Fuente Propia]

Después de determinar que las regiones en el odómetro (segmentado individualmente) no corresponden a ruido, se determina que las regiones son dígitos numéricos, ahora se procede a determinar el porcentaje de tamaño en el odómetro. De acuerdo al tamaño de las regiones, se selecciona la del mayor tamaño y es eliminada la de menor tamaño, y así se obtiene solo un dígito que será ingresado al motor OCR como se observa en la Figura 35.

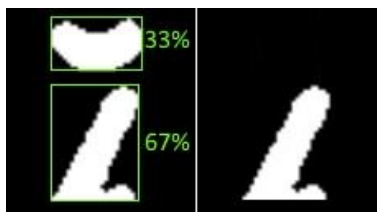


Figura 35. Dígitos en transición antes y después de determinar la región de mayor tamaño [Fuente Propia]

Al motor OCR se ingresa el dígito de mayor tamaño, porque este dígito contiene más información del número respecto al más pequeño y se asegura entregar la mayor información posible al motor OCR para lograr un mejor reconocimiento del dígito.

En un odómetro cuyo valor numérico se encuentra en transición, el valor que se debe identificar en este, siempre será el dígito que se encuentre en la parte superior. En la Figura 36 se observa el proceso de transición del dígito cero a uno, se observa que, aunque el dígito uno esté en mayor proporción respecto al cero, el valor numérico del odómetro sigue siendo cero, y será cero hasta que el dígito uno esté en una proporción de 100% (o 95% por eliminación de ruido).

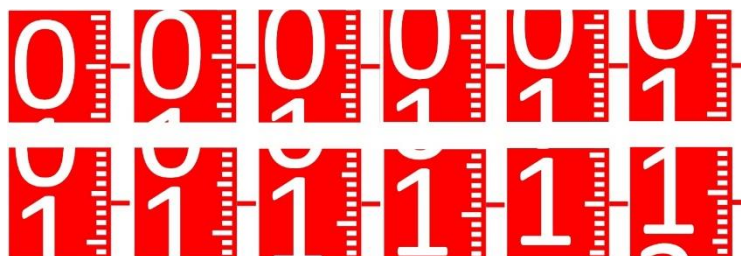


Figura 36. Transición de 0 a 1 en el odómetro decimal [Fuente Propia]

De esta manera, se debe identificar la proporción de tamaño en el odómetro y su posición. Por ejemplo, en la Figura 35 el dígito de mayor proporción es el dígito cuatro y su posición es 2 (numero abajo), por lo tanto, este dígito será ingresado al OCR y es reconocido e interpretado como el dígito cuatro, pero debido a que su posición es 2, significa que el verdadero valor numérico del odómetro es 3 ($4 - 1$). Si llegara a suceder el caso contrario, donde el dígito tres es mayor en proporción respecto al cuatro, el dígito tres será ingresado y reconocido por el OCR y ya que su posición es 1 (número arriba) el valor numérico del odómetro será de 3.

5.4.5 Identificación de centésimas y milésimas.

En algunos proyectos académicos [16] [19] definen la transición de los números como error, ya que no son reconocidos por el OCR. Existen otros proyectos académicos [21] donde es seleccionada la región (dígito) que se encuentra en mayor proporción en el odómetro y es ingresado al OCR. En el proyecto investigativo se utiliza la metodología definida en [21] para solventar la problemática de la transición de los dígitos en el odómetro, pero se descubre que esta metodología presenta un error matemático en cuanto a la exactitud de los resultados.

En la Figura 36 se observa los 11 pasos (el paso 12 empieza la transición de uno a dos) de la transición de cero "0" a uno "1". Cabe destacar que el paso 1 ("0,0") y el paso 11 ("1,0") no son considerados como números en transición, ya que en el paso 1 existe un pequeño fragmento del dígito uno, pero ya que el cero es 95% más grande que el uno, el uno será interpretado como ruido y será eliminado, y en el paso 11 se encuentra solo el dígito uno, así que estos dos pasos no son números en transición.

Hasta el paso 5 ("0,4") el método del número de mayor proporción es correcto, ya que de "0,1" a "0,4" el cero está en mayor proporción respecto al uno, así que cuando es

ingresado al motor OCR, se reconocerá como cero y el dígito decimal será cero. Pero entre el paso 7 (“0,6”) y el paso 10 (“0,9”) el uno estará en mayor proporción respecto al cero, y ya que es ingresado al motor OCR el dígito de mayor proporción, es ingresado el dígito uno y el valor decimal será uno. Se interpreta que el método del número de mayor proporción que se expone en proyecto académico [21], en el peor de los casos llega a tener un error de 0,04. Ya que el objetivo del proyecto es tener una aplicación precisa, se debe realizar otro procedimiento para la identificación de las centésimas y milésimas.

Se propone una metodología basada en el promedio ponderado, la cual consiste en determinar el valor decimal y el valor centesimal de la lectura a través del porcentaje de proporción en el odómetro decimal; esta metodología es solo aplicada en el odómetro decimal (último odómetro).

El promedio ponderado se determina a través de la siguiente ecuación:

$$\bar{x} = \frac{\sum(w * x)}{\sum w} \quad (6)$$

donde:

- \bar{x} es el Promedio o media ponderada.
- w es el peso o grado de importancia; para el proyecto es el porcentaje de proporción.
- x es el valor numérico (dígito del odómetro).

Por ejemplo, si la Figura 35 correspondiera al odómetro decimal y se aplica la metodología de promedio ponderado, la ecuación quedaría:

$$\bar{x} = \frac{(3 * 33) + (4 * 67)}{100} = 3,67$$

De acuerdo al cálculo anterior el valor de centésima sería “6” y el valor de milésima sería “7”. Es importante entender que, si en el odómetro decimal no existe transición, significa que el valor decimal y centesimal sería “0”.

5.4.6 Aplicación del motor OCR

Hasta este punto, la imagen del medidor esta segmentada de tal manera que solo se encuentran los valores numéricos de los odómetros identificados individualmente, pero hace falta un paso muy importante, generalmente las imágenes o caracteres con que ha sido entrenado Tesseract son imágenes con los caracteres de color negro y el fondo de color blanco (documentos), así que si se llegara a ingresar la imagen segmentada al motor OCR, este no reconocerá ningún dígito, ya que procesará solo los caracteres de color negro; en este caso, los valores numéricos de los odómetros están de color blanco y el fondo de color negro. De esta manera cuando se ingresa la imagen segmentada a Tesseract, este no logrará interpretar la imagen, ya que él está procesando las regiones negras, e interpretara que son caracteres irreconocibles o nuevos para él.

A la imagen segmentada se le debe invertir el color, para que los dígitos que estaban de color blanco sean transformados a color negro, así el motor OCR podrá reconocer los dígitos, como se observa en la Figura 37.



Figura 37. Dígitos segmentados individualmente con el color invertido [Fuente Propia]

Después, estos dígitos serán ingresados al motor OCR Tesseract. Tesseract es uno de los motores OCR más precisos de librería abierta, además, contiene más de 100 idiomas en su base de datos, pero cuando los dígitos segmentados son ingresados al motor no se tienen resultados muy precisos.

Debido a la segmentación del método de identificar el color blanco con el modelo de color HSV y la morfología binaria, el motor OCR interpreta algunos números como letras; como el caso del 4 lo interpreta como una "A", o algunos números son interpretados como números equivocados; como el caso del 3 que se interpretado como un "5". Debido a los resultados no precisos, se debe crear un lenguaje nuevo donde se entrene el motor OCR con los dígitos segmentados.

5.4.7 Entrenamiento OCR

Para el entrenamiento del nuevo lenguaje basado en los dígitos segmentados, se seleccionó el lenguaje español; podría funcionar el lenguaje inglés, francés, italiano, etc. Una de las características seleccionadas para este nuevo lenguaje, es el alfabeto que tendrá, serán solo los números de 0 a 9. En el Anexo 3 se encuentra documentado el proceso de entrenamiento del motor OCR.

- **Primera fase de entrenamiento**

Para la primera fase de entrenamiento, se usaron imágenes segmentadas del medidor Monofásico Bifilar Clase 2 **ISKRA E82C2-02** el cual tiene una resolución de 0,01. Se usaron 50 imágenes de diferentes lecturas sin transición. La selección de las lecturas se hizo de tal manera que tuviera los números de 0 a 9 en repetidas ocasiones para tener un entrenamiento más robusto; esto se puede realizar porque se puede manipular los odómetros del medidor. Además, la toma de las lecturas se hizo en diferentes horas del día y diferentes valores de luminosidad para entrenarlo en diferentes ambientes. Algunas imágenes utilizadas para el entrenamiento se observan en la Figura 38.

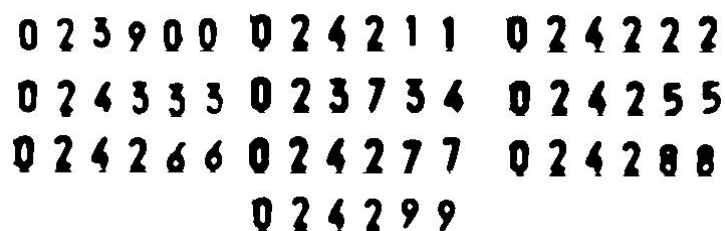


Figura 38. Imágenes usadas para el entrenamiento del motor OCR [Fuente Propia]

El proceso de entrenamiento interno de Tesseract es la identificación de caracteres del lenguaje seleccionado (en este caso español) y procesarlos para obtener una figura geométrica aproximada que encierre el carácter, luego son obtenidos todos los contornos de esta figura geométrica y guardada en una base de datos “.traindata” de acuerdo al carácter (numero) que ingresó el usuario. El proceso de entrenamiento es cada vez mejor en cuanto al número de imágenes, por ejemplo, si se entrena el número ocho “8” con 10 imágenes diferentes, Tesseract identificará los contornos de todos los ochos y los guarda en la base de datos, así cuando se use Tesseract con una imagen del ocho, se comparará los contornos con la base de datos del nuevo lenguaje y define que numero es. El tessdata generado después de la primera fase de entrenamiento define los contornos de los dígitos a través de las imágenes utilizadas, como se observa en la Figura 39.

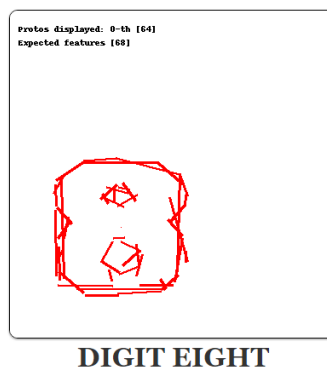


Figura 39. Posibles contornos del dígito 8 después del entrenamiento [Fuente Propia]

- **Segunda fase de entrenamiento**

En la segunda fase de entrenamiento del motor OCR se usaron 10 imágenes de diferentes lecturas del medidor **ISKRA E82C2-02** y 30 imágenes de diferentes lecturas del medidor Monofásico Bifilar Clase 2 **TECUN DDS994** de resolución 0,01, en el ambiente controlado del laboratorio de calibración y ensayo de la CEOC. Para el entrenamiento de esta fase, los odómetros de los medidores no pueden ser manipulados, debido a que los medidores deben estar tal cual llegan a la compañía para no alterar los resultados del laboratorio, así que las imágenes de las lecturas son exactamente como llegan al laboratorio, pero se realiza la toma de imágenes a

diferentes medidores de la misma referencia, así se obtiene imágenes de lecturas con dígitos completos y dígitos en transición. La segunda fase del entrenamiento del motor OCR se realiza en el laboratorio de la CEOC porque se tiene acceso a los medidores especificados. A través de las imágenes segmentadas, se comprueba la eficacia de la metodología de binarización a través del modelo de color HSV. Las imágenes obtenidas no contienen ruido y los dígitos son identificados correctamente.

- **Tercera fase de entrenamiento**

La tercera fase de entrenamiento se realiza en ambientes reales simulados en lo más preciso a las brigadas técnicas de la CEOC en las pruebas de verificación en sitio. El entrenamiento se realizó con 5 imágenes del medidor **ISKRA E82C2-02** y 15 imágenes del medidor **TECUN DDS994**. Al igual que la segunda fase de entrenamiento, los odómetros no pueden ser manipulados para obtener diferentes lecturas, así que se realiza la toma de diferentes imágenes de la misma referencia del contador, obteniendo imágenes de entrenamiento con dígitos completos y dígitos en transición. Las imágenes para el entrenamiento son tomadas en diferentes horas del día, diferentes tipos de ambiente, diferentes ambientes lumínicos y diferentes alturas para simular en lo más parecido a los ambientes donde se realizan las pruebas de verificación en sitio.

Al final de la tercera fase de entrenamiento del motor OCR, el nuevo lenguaje en Tesseract fue entrenado con 110 imágenes de diferentes lecturas de los dos medidores.

5.5 Resumen

En este capítulo se demostró el desarrollo intuitivo de la aplicación al indicar al usuario el funcionamiento de la aplicación a través de habilitación y des-habilitación de botones, de esta manera se asegura el correcto procedimiento de la aplicación de acuerdo a la norma técnica colombiana.

Para obtener resultados precisos en Tesseract, se debió entrenar un lenguaje desde cero, basado en lenguaje español y entrenado con los dígitos segmentados de las imágenes de las lecturas.

Determinar la posición del dígito de mayor proporción cuando existe transición en el odómetro es muy importante para determinar qué valor numérico realmente se debe designar a ese odómetro. La metodología del promedio ponderado propuesta para la determinación de centésimas y milésimas es una metodología más eficiente y ágil que determinar el valor centesimal y milésima a través de las franjas horizontales que existen en el odómetro decimal.

Capítulo 6. Resultados

En este capítulo del proyecto se muestran y se comparan los resultados de las pruebas de la aplicación en un ambiente simulado y de los obtenidos en el acompañamiento a las brigadas técnicas encargadas de realizar las pruebas de verificación en sitio. El ambiente se simuló de tal manera que sea lo más parecido al ambiente donde se realizan las pruebas de verificación en sitio realizado por las brigadas.

Debido a inconvenientes legales con la aseguradora de la CEOC (como estudiantes no se tiene ARL), no fue posible utilizar la aplicación en las pruebas de verificación en sitio realizadas por la CEOC y de igual manera los operadores técnicos encargados de la prueba de dosificación no podrían realizar pruebas con la aplicación, ya que no es una herramienta legalmente definida para las pruebas ni establecida por la CEOC. Se podría realizar un diseño experimental para solicitar el uso de la aplicación en las pruebas de dosificación, pero legalmente tomaba demasiado tiempo en obtener respuesta.

6.1 Toma de datos de las pruebas de dosificación CEOC

Para comparar la aplicación desarrollada con la metodología actualmente utilizada por las brigadas técnicas, se define al tiempo como la variable comparativa. Se deben tomar datos del tiempo utilizado por los operadores técnicos para la toma y registro de la lectura del medidor en la TGO y compararlo con el tiempo utilizado por la aplicación.

La comparación de la precisión de los operadores técnicos y la aplicación, se hace a través de la lectura percibida visualmente. Se compara la lectura registrada por los operadores con la lectura percibida por los estudiantes, luego, se compara la lectura percibida por los estudiantes con la lectura de la aplicación. De esta manera, se puede determinar la precisión de la aplicación respecto a los operadores técnicos.

Se tomaron los datos ambientales y de tiempo en 8 brigadas PQR en un ambiente industrial, 6 pruebas de dosificación en el laboratorio de calibración y ensayo de la CEOC y 10 brigadas MACRO en un ambiente rural. Las brigadas PQR son las brigadas técnicas de la CEOC que determinan el funcionamiento del medidor para justificar las respuestas a las peticiones, quejas y reclamos de los usuarios, para los casos de las pruebas registradas, era para justificar el correcto cobro de la energía consumida. Las pruebas en el laboratorio de calibración y ensayo de la CEOC se realizaron bajo un ambiente controlado en iluminación y comodidad para obtener datos de tiempos óptimos y mínimos. Las brigadas MACRO de la CEOC son brigadas encargadas de cambiar cables y contadores (si no funcionan) para reducir pérdidas de energía eléctrica las cuales tienen que cubrir la CEOC.

Para simular un ambiente “Real” donde utilizar la aplicación, se midió también la altura de los medidores y la luminosidad sobre los medidores en las pruebas de verificación hechas por las brigadas técnicas. La luminosidad sobre el medidor se midió a través

de una aplicación desarrollada para un *Smartphone* Motorola Moto E (segunda generación), el sensor de luminosidad con que cuenta el Moto E es un CT406 Light Sensor de la marca TAOS, tiene un rango máximo de medición de 27000 *luxes* y una resolución de 1,0. En la Figura 40 se puede observar que el Moto E se posicionaba sobre el medidor exactamente en la región de los odómetros para tener un valor de la luminosidad más precisa.

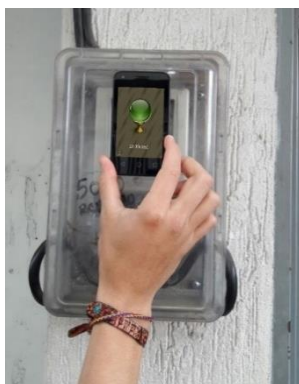


Figura 40.. Posición del celular para medir luminosidad [Fuente Propia]

Para la toma de datos, se midió el tiempo de 3 diferentes operadores técnicos, un operador técnico de PQR, un operador técnico de MACRO y un laboratorista. También se midieron variables físicas del ambiente y la altura de los medidores para recrear correctamente los ambientes.

6.1.1 Toma de datos brigadas PQR

Se acompañó a las brigadas técnicas de PQR en ocho diferentes pruebas de verificación en sitio en la zona sur de la ciudad de Popayán, los datos registrados corresponden solo a la prueba de dosificación hecha en los diferentes medidores de energía eléctrica. El ambiente donde se realizaron las pruebas de verificación en sitio es un ambiente urbano (barrios de Popayán) y ningún contador tenía exposición directa a la luz del sol, ya que los contadores al encontrarse en las paredes de las viviendas los techos daban sombra a los medidores igual que las casas de dos o tres pisos que se encontraban delante de las viviendas a las cuales se les realizaba las pruebas de verificación en sitio. En la Figura 41 se observa a un operador técnico realizando la prueba de dosificación.



Figura 41. Operador PQR realizando prueba de dosificación [Fuente Propia]

Ya que se habla de un ambiente urbano, todas las viviendas donde se realizaban las pruebas tenían un medidor monofásico bifilar de clase 2 y resolución 0,01, pero el medidor más común en esta zona es el medidor TECUN tipo DDS994.

Los operadores solo pueden tener una precisión de 0,01 para medidores de resolución 0,01 KWh, pero se determina que la lectura interpretada por el operador es exacta, ya que en muchas de las pruebas la lectura obtenida por el operador es igual a la lectura percibida por los estudiantes, en pocos casos no era igual, pero con una desviación de tan solo 0,01 KWh como se observa en la Tabla 7.

Tabla 7. Datos obtenidos en brigadas PQR

No.	Altura (cm)	Iluminancia (lx)	Medidor			Lecturas (KWh)				Tiempo (s)			Error (%)	Resultado
			Marca	Clase	Resolución	Inicial		Final		Dosificación	Registro			
						Operador	Percibida	Operador	Percibida		Inicial	Final		
1	196	3265	TECUN	2	0,01	07337,33	07337,34	07337,72	07337,72	140	49	54	0,95	Cumple
2	171	1971	TECUN	2	0,01	02562,31	02562,32	02562,96	02562,96	140	45	50	2,25	Cumple
3	184	5389	CDM	2	0,01	00741,00	00741,00	00741,82	00741,82	120	35	40	3,10	Cumple
4	179	3932	TECUN	2	0,01	06522,29	06522,29	06522,75	06522,75	140	55	130	1,30	Cumple
5	170	8475	METREX	2	0,01	07042,24	07042,25	07042,50	07042,50	170	50	55	0,30	Cumple
6	176	8320	TECUN	2	0,01	04152,81	04152,80	04153,01	04153,01	140	50	51	0,00	Cumple
7	172	5858	TECUN	2	0,01	08393,25	08393,25	08393,42	08393,42	140	37	42	-0,15	Cumple
8	178	1796	TECUN	2	0,01	02559,21	02559,22	02559,34	02559,33	140	42	50	-0,35	Cumple

Respecto al tiempo, es muy curioso observar que la lectura inicial (antes de dosificar la energía) lleva menos tiempo realizarla que la lectura final. En promedio, los operadores les llevaba **45 segundos** realizar la lectura inicial y el 50% de los registros de las lecturas finales que se realizaron en el menor tiempo, tienen un tiempo máximo de **47 segundos**.

Para determinar la tendencia central del tiempo de la lectura final, se usó la mediana, debido a que en la prueba 4 se detuvo la TGO y le llevó más tiempo al operador registrar la lectura de lo normal, y debido a que este es un extremo muy poco inusual, la media no determinaría una correcta interpretación, ya que, al contrario de la media, la mediana no es afectada por valores extremos [64].

6.1.2 Toma de datos brigadas MACRO

Se acompañó a las brigadas MACRO de la CEOC durante 10 pruebas de verificación en sitio distribuidas en dos veredas de la ciudad de Popayán, la vereda Clarete y la vereda La Sabana. En la Figura 42 se puede observar a un operador técnico realizando la prueba de dosificación a un medidor de energía eléctrica de marca ISKRA.



Figura 42. Operador MACRO realizando prueba de dosificación [Fuente Propia]

Ya que las pruebas de verificación se realizaron en zonas rurales, se encuentra un ambiente muy diferente a las pruebas realizadas por las brigadas PQR, y existen ocasiones donde el medidor se encuentra en total exposición a la luz directa del sol. Ya que las brigadas MACRO deben verificar medidores en los cuales se sospecha no funcionan correctamente (por eso existen pérdidas), en ocasiones deben verificar medidores macro, estos medidores son medidores bifásicos o trifásicos y están conectados a medidores monofásicos.

Los datos registrados en el acompañamiento de las brigadas técnicas MACRO se encuentran en la Tabla 8.

Tabla 8. Datos brigadas Macro

No.	Altura (cm)	Iluminancia (lx)	Medidor				Lecturas (KWh)				Tiempo (s)			Error (%)	Resultado
			Marca	Característica eléctrica	Clase	Resolución	Inicial		Final		Dosificación	Registro			
							Operador	Percibida	Operador	Percibida		Inicial	Final		
1	184	8223	KRISIK	Bifásico Trifilar	2	0,01	03837,12	03837,13	03837,33	03837,33	160	90	85	4,24	Cumple
2	187	2798	ISKRA	Monofásico Bifilar	2	0,01	03012,21	03012,21	03012,41	03012,42	140	45	50	3,70	Cumple
3	198	66623	DISICO	Bifásico Trifilar	2	0,01	04328,49	04328,50	04328,70	04328,70	180	35	40	4,29	Cumple
4	179	2653	ISKRA	Bifásico Trifilar	2	0,01	02812,13	02812,14	02812,34	02812,34	280	170	160	0,06	Cumple
5	175	68806	ISKRA	Monofásico Bifilar	2	0,01	15479,84	15479,84	15480,04	15480,05	140	50	55	0,07	Cumple
6	194	6325	METREX	Monofásico Bifilar	2	0,01	00047,16	00047,15	00047,36	00047,36	130	50	55	-0,10	Cumple
7	187	4617	ISKRA	Monofásico Bifilar	2	0,01	15679,71	15679,70	15679,91	15679,91	140	40	46	-0,26	Cumple
8	174	93800	ISKRA	Monofásico Bifilar	2	0,01	03856,90	03856,91	03857,10	03857,10	140	45	53	-0,10	Cumple
9	185	9118	ISKRA	Monofásico Bifilar	2	0,01	02981,62	02981,62	02981,82	02981,83	140	45	50	-0,44	Cumple
10	172	6450	ISKRA	Monofásico Bifilar	2	0,01	03292,13	03292,14	03292,34	03292,34	140	36	46	-2,09	Cumple

En tres ocasiones (pruebas 3, 5 y 8) hubo exposición directa a la luz del sol en los medidores de energía eléctrica. Estos medidores macro (1, 3 y 4) se encuentran generalmente en los postes de energía eléctrica, por lo tanto, están en condiciones de recibir la luz del sol directamente, en el caso de las pruebas 1 y 4 había árboles que generaban sombra a los medidores.

Al igual que las brigadas PQR, se puede determinar que los operadores de los macros obtienen una lectura del medidor precisa (resolución de 0,01) ya que son iguales (a veces desviación de 0,01) a las lecturas percibidas. También se puede observar que el tiempo de registro de la lectura inicial es menor al tiempo de registro de la lectura final, el 50% de las lecturas que se realizaron en un menor tiempo, tienen un tiempo máximo de **45 segundos** y el 50% de los registros de la lectura final que se realizó un menor tiempo, tenían un tiempo máximo de **51 segundos**. En la prueba 4, el operador ingresó la lectura final erróneamente a la TGO y debió repetir la prueba de dosificación de nuevo, así que el tiempo de registro de las lecturas y el tiempo de dosificación se duplicó.

En los datos obtenidos en las brigadas técnicas PQR y MACRO se puede concluir que el medidor ISKRA es el medidor más común por lo menos en las veredas Clarete y La Sabana, y el medidor TECUN es el más común por lo menos en la zona sur de Popayán.

Se determinó que los medidores seleccionados para la realización de las pruebas de la aplicación son el Medidor ISKRA E82C2-02 y el Medidor TECUN DDS994 ya que son los más comunes en la zona rural y urbana donde se tomaron los datos de la prueba de dosificación. Estos medidores son medidores ciclométricos (electromecánico y eléctrico respectivamente) de resolución 0,01 y están dentro de los medidores más comunes en Popayán.

6.1.3 Toma de datos laboratorio CEOC.

Se realiza una tercera toma de datos en el laboratorio de calibración y ensayo para tener los datos más óptimos, los cuales sería el tiempo de registro de las lecturas en un ambiente controlado y más cómodo. No se toma la altura del medidor ya que la prueba de dosificación en el laboratorio no se realiza de manera vertical (pared) sino de manera horizontal sobre una mesa. Las pruebas de dosificación no se realizan a través de una TGO, se realizan directamente con el AMV como se observa en la Figura 43. La prueba con el AVM es exactamente igual a la realizarla con la TGO, la única diferencia es que se dosifica *0,5 KWh* y no *0,2 KWh* como se haría con la TGO para medidores de resolución 0,001, así el tiempo de dosificación es mayor, pero el tiempo de registro de la lectura debe ser igual a las pruebas de PQR y MACRO.



Figura 43. Laboratorista realizando prueba de dosificación [Fuente Propia]

Se realizan tres pruebas de dosificación a un Medidor Monofásico Bifilar ISKRA E82C2-02 de clase 2 y de resolución 0,01 KWh y tres pruebas de dosificación a un Medidor Monofásico Bifilar TECUN DDS994 de clase 2 y de resolución 0,01 KWh. Los datos registrados en las pruebas en el Laboratorio se observan en la Tabla 9.

Tabla 9. Datos pruebas Laboratorio de Calibración y Ensayo

No.	Iluminancia (lx)	Medidor	Lecturas (KWh)				Tiempo (s)			Error (%)	Resultado
			Inicial		Final		Dosificación	Registro			
			Operador	Percibida	Operador	Percibida		Inicial	Final		
1	835	TECUN	03191,74	03191,73	03192,33	03192,33	140	49	54	0,18	Cumple
2	835	TECUN	03196,54	03196,54	03197,13	03197,13	140	45	50	0,18	Cumple
3	835	TECUN	03197,37	03197,37	03197,94	03197,94	140	35	40	0,14	Cumple
4	835	ISKRA	74681,00	74681,00	74681,47	74681,47	140	55	130	-0,06	Cumple
5	835	ISKRA	74681,97	74681,97	74682,43	74682,43	140	50	55	-0,08	Cumple
6	835	ISKRA	74686,57	74686,57	74687,07	74687,07	140	50	55	0,00	Cumple

Al igual que la toma de datos anterior, el tiempo de registro de la lectura final es mayor al registro de la lectura inicial. En promedio el laboratorista se tomó **47 segundos** registrando la lectura inicial y el 50% de los tiempos de registro de la lectura final que se demoraron menos, tenían un tiempo máximo de **54 segundos**. En la prueba 4 el AVM empezó a presentar fallas y se detenía cuando el laboratorista trataba de ingresar la lectura final, por esta razón en la prueba 4 se tomó casi el doble de tiempo registrando los datos.

A este punto se puede concluir que el peor de los casos cuando se realiza la prueba de dosificación sucede cuando la TGO o el AVM presentan fallas, cuando esto sucede, se duplica el tiempo de registro de las lecturas aumentando el tiempo en que se desarrolla la prueba de dosificación. Si un operador ingresa una lectura erróneamente debe realizar la prueba de dosificación de nuevo, duplicando el tiempo de desarrollo de la prueba, y si el operador ingresa mal la lectura, pero no se da cuenta, la prueba será errónea arrojando un resultado equivocado.

- Pruebas en Laboratorio de Calibración y Medición

En la toma de datos de las pruebas de dosificación hechas en el laboratorio de Calibración y Medición de la CEOC, se decidió realizar pruebas con la aplicación desarrollada, las pruebas se realizaron sobre las mismas pruebas de dosificación que realizó el operador del laboratorio, los resultados se observan en la Tabla 10.

Tabla 10. Pruebas de la aplicación en la Laboratorio de la CEOC

No.	Iluminancia (lx)	Medidor	Lecturas (KWh)				Tiempo (s)			Error (%)	Resultado
			Inicial		Final		Dosificación	Registro			
			Aplicación	Percibida	Aplicación	Percibida		Inicial	Final		
1	835	TECUN	03191,747	03191,73	03192,336	03192,33	140	12	13	0,18	Cumple
2	835	TECUN	03196,548	03196,54	03197,135	03197,13	140	12	10	0,17	Cumple
3	835	TECUN	03197,374	03197,37	03197,747	03197,94	140	17	19	-0,25	Cumple
4	835	ISKRA	74681,003	74681,00	74681,470	74681,47	140	10	11	-0,07	Cumple
5	835	ISKRA	74681,978	74681,97	74682,436	74682,43	140	13	15	-0,08	Cumple
6	835	ISKRA	74686,579	74686,57	74687,076	74687,07	140	18	17	-0,01	Cumple

Durante las pruebas, la aplicación falló en una de las pruebas donde existía transición en el odómetro decimal y el odómetro de unidad; lectura de color rojo en la Tabla 10. Casualmente, aunque en el odómetro de unidad existía transición, el motor OCR logro reconocer el valor correcto, pero en el odómetro decimal el dígito “9” lo reconoció como un “7” agregando -0,6% de error al error porcentual final, ya que es un valor agregado de muy poca magnitud, no cambia el resultado final (Cumple).

Las lecturas percibidas por el operador del Laboratorio y las lecturas reconocidas por la aplicación son comparadas en la Tabla 11.

Tabla 11. Comparación de lecturas entre el operador del Laboratorio y la aplicación.

Lecturas (KWh)			
Inicial		Final	
Operador	Aplicación	Operador	Aplicación
03191,73	03191,747	03192,33	03192,336
03196,54	03196,548	03197,13	03197,135
03197,37	03197,374	03197,94	03197,747
74681,00	74681,003	74681,47	74681,470
74681,97	74681,978	74682,43	74682,436
74686,57	74686,579	74687,07	74687,076

La aplicación demostró ser más precisa que el operador del Laboratorio de Calibración y Medición. El operador puede percibir lecturas de precisión 0,01 KWh y la aplicación logra tener una precisión de 0,001 KWh, pero los dos (operador y aplicación) logran reconocer la misma exactitud en las lecturas.

La aplicación tiene un mejor tiempo de registro de las lecturas de los medidores, observándose esta conclusión en la Tabla 12.

Tabla 12. Comparación de tiempo de registro entre el operador del Laboratorio y la aplicación.

No.	Tiempo de registro (s)			
	Inicial		Final	
	Operador	Aplicación	Operador	Aplicación
1	49	12	54	13
2	45	12	50	10
3	35	17	40	19
4	55	10	130	11
5	50	13	55	15
6	50	18	55	17

Mientras el operador gasta un promedio de *47 segundos* en el registro de la lectura inicial, el estudiante con la aplicación tarda en promedio *13 segundos*. Para la lectura final, en la prueba 4 cuando el AVM dejó de funcionar por un momento, el operador gasta en promedio *54 segundos* en registrarla, pero el estudiante con la aplicación gasta en promedio *14 segundos*, demostrando que, con la ayuda de la aplicación, las pruebas de dosificación tardarían menos tiempo en realizarse.

6.2 Pruebas de la aplicación

En la toma de datos de las pruebas de dosificación se pudo observar que los medidores se encontraban a una altura entre 1,70 *metros* y 1,96 *metros*, así que se definieron tres alturas para las pruebas de la aplicación, 1,70 *metros*, 1,85 *metros*, y 2,00 *metros*. También se puede determinar tres tipos de luminosidad, la exposición directa al sol, la cual es mayor de 10000 *luxes*, y dos tipos de luminosidad generada por la sombra, valores entre 1000 y 5000 *luxes* y valores entre 5000 y 10000 *luxes*. Los valores menores a 1000 *luxes*, como el caso de las pruebas hechas en el laboratorio de calibración y ensayo de la CEOC, son valores de iluminancia en ambientes cerrados; en el caso del proyecto, ambiente controlado.

Para realizar las pruebas a la aplicación desarrollada, se definieron tres ambientes lumínicos para cada medidor, exposición directa al sol, y dos tipos de sombra, una sombra generada por un árbol de la Facultad de Ingeniería Electrónica y Telecomunicaciones (FIET) y otra sombra generada por el techo del Instituto de Posgrado en Electrónica y Telecomunicaciones (IPET). Los diferentes ambientes lumínicos se pueden observar en la Figura 44. Además, se realizaron pruebas de la aplicación en un ambiente controlado.



Figura 44. Contadores expuestos a diferentes ambientes lumínicos [Fuente Propia]

6.2.1 Pruebas en ambiente controlado.

Antes de realizar pruebas con la aplicación en diferentes ambientes lumínicos, se realizaron pruebas de la aplicación en un ambiente controlado para evaluar el funcionamiento de la metodología de binarización en un mismo nivel lumínico.

Se realizó diez pruebas con el Medidor Monofásico Bifilar clase 2 ISKRA E82C2-02 de resolución 0,01 y diez pruebas con el Medidor Monofásico Bifilar clase 2 TECUN DDS994 de resolución 0,01 en el mismo ambiente lumínico (sala de estudio del IPET). Para simular que se dosifica 0,2 KWh al medidor TECUN, se gira el odómetro de manera manual, ya que es un medidor eléctrico y no tiene piezas mecánicas (piñones) para manipular el odómetro externamente y debe hacer directamente sobre él. Para simular que se dosifica 0,2 KWh al medidor ISKRA, se gira la piñonera del medidor una revolución, esta revolución gira el odómetro decimal $\pm 0,2$ KWh. La lectura inicial del medidor es seleccionada aleatoriamente para tener diferentes valores numéricos en las pruebas.

Los resultados de las pruebas a los medidores con la aplicación se observan en la Tabla 13.

Tabla 13. Pruebas de la aplicación en un ambiente controlado.

No.	Iluminancia (lx)	Medidor	Lecturas (KWh)				Tiempo de registro (s)		Error (%)	Resultado
			Inicial		Final		Inicial	Final		
			Aplicación	Percibida	Aplicación	Percibida				
1	594	ISKRA	02713,617	02713,61	02713,824	02713,82	17	18	0,035	Cumple
2	594	ISKRA	02717,588	02717,58	02717,874	02717,84	19	16	0,43	Cumple
3	594	ISKRA	02721,675	02721,67	02721,831	02721,83	16	18	-0,22	Cumple
4	594	ISKRA	02726,130	02726,13	02726,324	02726,31	16	13	-0,03	Cumple
5	594	ISKRA	02726,253	02726,23	02726,344	02726,46	20	18	-0,55	Cumple
6	594	ISKRA	02731,399	02731,39	02731,594	02731,59	20	19	-0,02	Cumple
7	594	ISKRA	02735,200	02735,20	02735,588	02735,57	17	18	0,94	Cumple
8	594	ISKRA	02735,417	02735,40	02735,680	02735,68	18	17	0,315	Cumple
9	594	ISKRA	02739,371	02739,37	02739,775	02739,77	13	13	1,02	Cumple
10	594	ISKRA	02739,778	02739,77	02739,590	02739,58	13	14	-1,94	Cumple
11	594	TECUN	00194,117	00194,11	00194,320	00194,32	11	13	0,015	Cumple
12	594	TECUN	00194,612	00194,61	00194,820	00194,83	18	18	0,04	Cumple
13	594	TECUN	00195,650	00195,35	00195,590	00195,59	14	15	-1,3	Cumple
14	594	TECUN	00200,891	00200,88	00201,077	00201,07	12	12	-0,07	Cumple
15	594	TECUN	00201,436	00201,42	00201,917	00201,91	12	14	1,405	Cumple
16	594	TECUN	00201,917	00201,91	00202,115	00202,11	17	20	-0,01	Cumple
17	594	TECUN	00201,938	00201,93	00202,130	00202,13	11	12	-0,04	Cumple
18	594	TECUN	00204,284	00204,29	00204,624	00204,62	10	11	0,7	Cumple
19	594	TECUN	00204,799	00204,79	00203,860	00204,96	16	17	-5,69	No Cumple
20	594	TECUN	00205,690	00205,69	00205,925	00205,93	17	16	0,175	Cumple

De las veinte pruebas, la aplicación tuvo errores en tres pruebas (5, 13 y 19), en las cuales había transición en los odómetros. En la prueba 19 se tiene un resultado de que el medidor no cumple (no funciona correctamente) ya que el error porcentual es superior al permitido por la NTC 5900 ($\pm 5\%$) [37], esto es debido a que en la lectura final el odómetro de unidad y decimal estaban en transición, entonces Tesseract reconoció mal los dos dígitos ya que no se tenía suficiente información de estos y es confundido con un valor numérico diferente, este error agrega $-4,5\%$ de error al error porcentual final, debido a esto, la aplicación determina que el medidor no funciona correctamente.

Las pruebas 9 y 10 y 15 y 16 son pruebas donde a lectura inicial se deja igual a la lectura final de la prueba anterior, esto se realiza para comprobar que la aplicación reconoce el mismo valor numérico a los mismos valores de los odómetros, fallando en la prueba 9 y 10 por $0,003 \text{ KWh}$.

La aplicación tiene un porcentaje de 100% de exactitud cuando los valores numéricos no se encuentran en transición. En valores en transición, la aplicación logra casi el 73% de exactitud, en cuanto al tiempo, con la aplicación los estudiantes tardan en promedio **15 segundos** en registra la lectura del medidor.

6.2.2 Primera fase de pruebas

La primera fase de pruebas de la aplicación se realizó con el Medidor Monofásico Bifilar clase 2 ISKRA E82C2-02 de resolución 0,01, la primera fase de pruebas fue realizada solo por un estudiante del proyecto. Se realizaron treinta pruebas de la aplicación en las instalaciones del IPET en la FIET, se realizaron diez pruebas con el medidor a una altura de $1,70 \text{ metros}$, diez pruebas a $1,85 \text{ metros}$ y diez pruebas a $2,00 \text{ metros}$. El medidor se expuso a tres diferentes condiciones de luminosidad, seis pruebas a exposición directa de la luz del sol, doce pruebas a la sombra generada por un árbol y doce pruebas debajo de un techo del IPET.

Para simular que se dosifica $0,2 \text{ KWh}$ al medidor ISKRA, se gira la piñonera del medidor una revolución, esta revolución gira el odómetro decimal $\pm 0,2 \text{ KWh}$. No se realiza una dosificación verdadera por qué se necesita una carga que consuma más o menos 40 amperios y no se cuenta con esta carga. Tampoco fue posible conseguir una carga fantasma como las utilizadas por los operadores de las brigadas técnicas. El obstáculo de no tener cargas físicas de esta potencia, realmente no se considera un problema para la simulación de la prueba de dosificación, ya que la variable a comparar es el tiempo en que se tarda una persona registrando la lectura, ya se manualmente a la TGO o por parte de la aplicación. La simulación de la dosificación de $0,2 \text{ KWh}$ se realiza solo para simular el correcto funcionamiento del medidor, algunas pruebas se sobrepasan los $0,2 \text{ KWh}$ para demostrar que la ecuación del cálculo del error es correcta.

La lectura inicial del medidor se selecciona aleatoriamente para obtener diferentes dígitos enteros y en transición. De las treinta pruebas, cinco pruebas son seleccionadas al azar, en estas cinco pruebas el odómetro es manipulado de manera consiste para simular cinco casos específicos:

- Se simula la dosificación de 3 *KWh* y 2 *KWh* para simular un medidor en mal funcionamiento.
- La lectura inicial se manipula para que el odómetro decimal se encuentre en transición.
- La lectura final se manipula para que el odómetro decimal se encuentre en transición.
- La lectura inicial y final se manipula para que el odómetro decimal se encuentre en transición.

Los resultados de las treinta pruebas con la aplicación desarrollada, se observa en la Tabla 14.

Tabla 14. Resultados prueba de la aplicación primera fase

No.	Altura (cm)	Iluminancia (lx)	Lecturas (KWh)				Tiempo de registro (s)		Error (%)	Resultado
			Inicial		Final		Inicial	Final		
			Aplicación	Percibida	Aplicación	Percibida				
1	170	3173	02754,873	02754,87	02755,000	02755,00	20	18	-0,365	Cumple
2	170	3173	02755,872	02755,87	02756,000	02756,00	19	17	-0,36	Cumple
3	170	3173	02756,024	02756,02	02756,225	02756,22	17	20	0,005	Cumple
4	170	3173	02756,300	02756,31	02756,431	02756,43	20	18	-0,345	Cumple
5	170	7072	02757,874	02577,87	02758,000	02758,00	13	14	-0,37	Cumple
6	170	7072	02758,873	02758,87	02759,000	02759,00	14	14	-0,365	Cumple
7	170	7072	02759,576	02759,57	02759,725	02759,72	12	14	-0,255	Cumple
8	170	7072	02760,210	02760,20	02760,430	02760,43	18	17	0,1	Cumple
9	170	65535	02761,050	02761,85	02762,031	02762,03	12	14	3,905	Cumple
10	170	65535	02762,428	02762,43	02762,617	02762,60	17	20	-0,055	Cumple
11	185	3173	02762,619	02762,60	02762,882	02762,88	16	18	0,315	Cumple
12	185	3173	02764,820	02764,82	02765,075	027665,07	16	18	0,275	Cumple
13	185	3173	02765,760	02767,96	02767,212	02767,20	16	15	6,26	No Cumple
14	185	3173	02771,433	02771,43	02771,638	02771,63	18	17	0,025	Cumple
15	185	7072	02772,211	02772,21	02772,400	02772,41	11	19	-0,055	Cumple
16	185	7072	02776,188	02776,17	02776,332	02776,33	18	16	-0,28	Cumple
17	185	7072	02779,026	02779,02	02779,200	02779,20	14	14	-0,13	Cumple
18	185	7072	02781,082	02781,07	02781,200	02781,21	14	17	-0,41	Cumple
19	185	65535	02782,520	02782,52	02785,400	02785,40	12	13	13,4	No Cumple
20	185	65535	02783,230	02783,23	02783,563	02783,44	14	16	0,665	Cumple
21	200	3173	02785,400	02785,40	02785,600	02785,60	12	13	-9E-13	Cumple
22	200	3173	02787,900	02787,90	02788,100	02788,14	13	11	-9E-13	Cumple
23	200	3173	02788,034	02788,03	02788,200	02788,20	14	15	-0,17	Cumple
24	200	3173	02788,200	02788,20	02788,400	02788,41	13	13	1E-12	Cumple
25	200	7072	02789,300	02789,05	02789,224	02789,22	10	11	-1,38	Cumple
26	200	7072	02790,224	02790,36	02790,700	02790,94	13	12	1,38	Cumple
27	200	7072	02790,439	02790,42	02792,919	02792,90	10	12	11,4	No Cumple
28	200	7072	02793,022	02793,02	02793,644	02793,64	12	13	2,11	Cumple
29	200	65535	02796,312	02796,30	02796,525	02796,52	11	14	0,065	Cumple
30	200	65535	02796,526	02796,52	02796,738	02796,73	12	13	0,06	Cumple

La sombra generada por el árbol, produce que el nivel de iluminación al medidor sea de tan solo *7173 luxes*, mientras que la sombra generada por el techo del IPET produce que el nivel de iluminación al medidor sea de *7072 luxes*. Cuando el medidor está expuesto directamente a la luz solar, el nivel de iluminación es de *65535*. Estos niveles

de iluminación están dentro de los rangos medidos en el acompañamiento a las brigadas PQR y MACRO.

Se observa que curiosamente el registro de la primera lectura es más rápido que el registro de la lectura final, esto sucede también en los datos obtenidos en el acompañamiento de las brigadas técnicas, realmente no se puede concluir porque sucede esta situación, se puede pensar que al ser el último valor registrado se toma un poco más de tiempo para hacer un mejor trabajo.

Al primer estudiante se le toma un promedio de *14 segundos* el registro de la lectura inicial con la aplicación y en promedio le toma *15 segundos* el registrar la lectura final con la aplicación. En cuanto a la precisión, si se compara las lecturas obtenidas por la aplicación y las lecturas percibidas visualmente, la aplicación es 0,001 más precisa que la percepción visual de la persona, y esta precisión puede llegar a ser de 0,00001 dependiendo del punto decimal que se establezca en el código fuente de la aplicación.

De las treinta pruebas realizadas a la aplicación, esta falló en seis ocasiones (color rojo en la Tabla 14), tres de las cuales se manipuló conscientemente el odómetro para generar transición (22, 25 y 26), pero las otras tres ocasiones (9, 13 y 20) también fallaron cuando había transición en los odómetros. En tres pruebas se puede observar que el error porcentual es mayor a $\pm 5\%$, de las cuales dos pruebas (27 y 19) fue manipulado conscientemente el odómetro para simular *3 KWh* y *2 KWh* respectivamente y en la otra prueba (13) el error es mayor al permitido porque la aplicación obtuvo un valor de lectura errónea, porque existía transición en odómetro decimal y el odómetro de unidad.

Las lecturas en color amarillo (ver Tabla 14) corresponden a lecturas del odómetro que no fueron cambiadas (23 y 24, y 29 y 30), se dejó el mismo valor de lectura para comprobar que la aplicación interpreta el mismo valor en dos ocasiones, fallando entre la prueba 29 y 30 por una variación de 0,01, esto puede ser debido a que la imagen se tome en un ángulo diferente.

6.2.3 Segunda fase de pruebas

La segunda fase de pruebas de la aplicación, se realizó con el Medidor Monofásico Bifilar clase 2 TECUN DDS994 de resolución 0,01. La segunda fase de pruebas fue realizada por un segundo estudiante en un día diferente a la primera fase de pruebas. Se realizaron treinta pruebas de la aplicación en las instalaciones del IPET en la FIET, al igual que la primera fase, se realizaron diez pruebas con el medidor a una altura de *1,70 metros*, diez pruebas a *1,85 metros* y diez pruebas a *2,00 metros*. El medidor se expuso a tres diferentes condiciones de luminosidad, seis pruebas a exposición directa de la luz del sol, doce pruebas a la sombra generada por el mismo árbol de la primera fase y doce pruebas bajo el techo del IPET.

Para simular que se dosifica *0,2 KWh* al medidor TECUN, se gira el odómetro de manera manual, ya que es un medidor eléctrico y no tiene piezas mecánicas (piñones) para manipular el odómetro externamente y debe hacer directamente sobre él.

La lectura inicial del medidor se selecciona aleatoriamente para obtener dígitos enteros y en transición. De las treinta pruebas, solo 1 prueba se seleccionó de manera aleatoria para manipular el odómetro y simular la dosificación de *1,5 KWh* para simular el mal funcionamiento el medidor. Otras dos pruebas fueron seleccionadas al azar para no cambiar la lectura y comprobar que la aplicación arroja la misma lectura al mismo valor numérico presente en el odómetro.

El resultado de las treinta pruebas de la aplicación para el medidor ISKRA, están contempladas en la Tabla 15.

Tabla 15. Resultados segunda fase de pruebas de la aplicación en el medidor TECUN.

No.	Altura (cm)	Iluminancia (lx)	Lecturas (KWh)				Tiempo de registro (s)		Error (%)	Resultado
			Inicial		Final		Inicial	Final		
			Aplicación	Nosotros	Aplicación	Nosotros				
1	170	4816	00211,546	00211,53	00211,734	00211,73	20	19	-0,06	Cumple
2	170	4816	00213,387	00213,38	00213,570	00213,58	15	14	-0,085	Cumple
3	170	4816	00219,100	00219,11	00219,399	00219,38	15	20	0,495	Cumple
4	170	4816	00222,547	00222,53	00222,746	00222,74	18	16	-0,005	Cumple
5	170	7492	00229,118	00229,11	00229,332	00229,32	13	15	0,07	Cumple
6	170	7492	00229,335	00229,33	00229,573	00229,57	17	16	0,19	Cumple
7	170	7492	00241,435	00241,43	00241,639	00241,63	18	19	0,02	Cumple
8	170	7492	00243,431	00243,43	00243,060	00243,64	13	19	-2,855	Cumple
9	170	88311	00244,108	00244,10	00244,376	00244,37	12	12	0,34	Cumple
10	170	88311	00247,750	00247,35	00247,574	00247,57	15	14	-1,88	Cumple
11	185	4816	00249,388	00249,37	00249,527	00249,52	12	15	-0,305	Cumple
12	185	4816	00251,836	00251,83	00252,836	00252,83	13	12	4	Cumple
13	185	4816	00256,060	00256,86	00257,072	00257,07	13	15	4,06	Cumple
14	185	4816	00262,732	00262,72	00262,915	00262,92	14	17	-0,085	Cumple
15	185	7492	00268,179	00268,17	00268,300	00268,30	17	16	-0,395	Cumple
16	185	7492	00268,300	00268,30	00268,574	00268,57	11	12	0,37	Cumple
17	185	7492	00270,950	00270,94	00271,530	00271,54	11	12	1,9	Cumple
18	185	7492	00272,777	00272,77	00272,912	00272,92	11	11	-0,325	Cumple
19	185	88311	00276,133	00276,12	00277,732	00277,72	13	12	6,995	No Cumple
20	185	88311	00277,060	00277,64	00277,831	00277,83	17	19	2,855	Cumple
21	200	4816	00279,102	00279,10	00279,329	00279,31	10	12	0,135	Cumple
22	200	4816	00285,433	00285,43	00285,680	00285,68	14	17	0,235	Cumple
23	200	4816	00285,615	00285,61	00285,887	00285,87	16	19	0,36	Cumple
24	200	4816	00287,886	00287,89	00288,088	00288,08	13	16	0,01	Cumple
25	200	7492	00288,571	00288,57	00288,734	00288,73	14	10	-0,185	Cumple
26	200	7492	00291,679	00291,77	00291,832	00291,83	14	12	-0,235	Cumple
27	200	7492	00291,832	00291,83	00292,082	00292,08	17	18	0,25	Cumple
28	200	7492	00295,282	00295,29	00295,470	00295,47	17	19	-0,06	Cumple
29	200	88311	00298,485	00298,49	00298,624	00298,62	10	13	-0,305	Cumple
30	200	88311	00299,375	00299,37	00299,571	00299,57	15	17	-0,02	Cumple

La sombra generada por el mismo árbol, pero un día después de la primera fase de pruebas, produce que el nivel de iluminación al medidor sea de *4816 luxes*, mientras que la sombra generada por el mismo techo del IPET produce que el nivel de iluminación al medidor sea de *7492 luxes*. Cuando el medidor está expuesto

directamente a la luz solar, el nivel de iluminación es de 88535, esto debido a que el día estaba más soleado. Estos niveles de iluminación están dentro de los rangos medidos en el acompañamiento a las brigadas PQR y MACRO y demuestra que la metodología de binarización funciona de una manera correcta sin importar la luminosidad del día.

Al segundo estudiante también se le toma un promedio de *14 segundos* el registrar la lectura inicial con la aplicación y en promedio le toma *15 segundos* registrar la lectura final con la aplicación, igual con lo sucedido por el primer estudiante en la primera fase de pruebas. En cuanto a la precisión, se puede observar que la aplicación es más precisa que la percepción visual de las personas, ya que la aplicación puede tener una resolución entre 0,001 y 0,00001 dependiendo de los decimales que se seleccione en el código fuente de la aplicación.

De las treinta pruebas realizadas, la aplicación falló en tres ocasiones (color rojo en la Tabla 15) en las cuales el odómetro decimal se encontraba en transición. En una prueba se puede observar que el error porcentual es mayor a $\pm 5\%$, pero esta prueba se simuló la dosificación 1,5 KWh para simular un mal funcionamiento del medidor.

Las lecturas en color amarillo de la Tabla 15 corresponden a lecturas del odómetro que no fueron cambiadas (15 y 16, y 26 y 27), se dejó el mismo valor de lectura para comprobar que la aplicación reconoce el mismo valor numérico en dos ocasiones similares.

6.2.4 Pruebas de la aplicación para valores máximos de rotación y distancia

A la aplicación desarrollada (SLNM) se le hicieron 3 pruebas con diferentes ángulos de captura de la imagen, para lograr identificar la máxima variación para obtener resultados exactos. Además, se realizó una prueba de distancia para encontrar la distancia óptima de captura de la imagen.

La Figura 45 representa la posición del teléfono inteligente para la realización de las pruebas de registro de lectura. El teléfono inteligente se encontraba de manera vertical y en paralelo al medidor ISKRA.

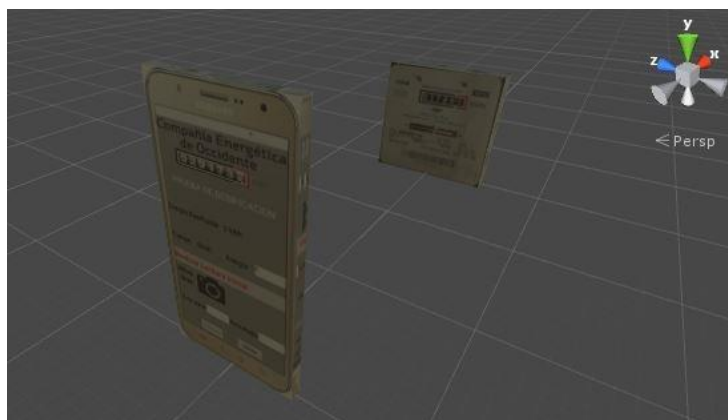


Figura 45. Representación de las pruebas de la aplicación SLNM para valores máximos de rotación y distancia [Fuente Propia]

- **Pruebas de variación de la distancia**

En la Figura 46 se observa la representación del teléfono con sus respectivos ejes. Para la prueba de distancia se variaba la distancia del teléfono inteligente pero solo en el eje x (eje de color rojo en la Figura 46).

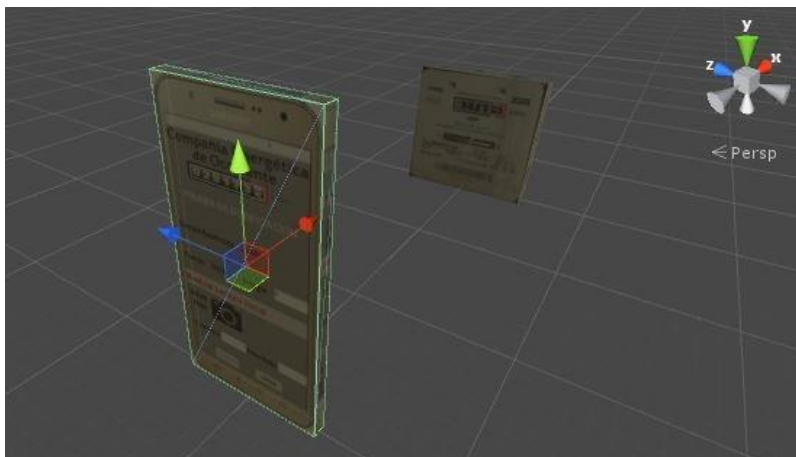


Figura 46. Representación de los ejes en la prueba de distancia [Fuente Propia]

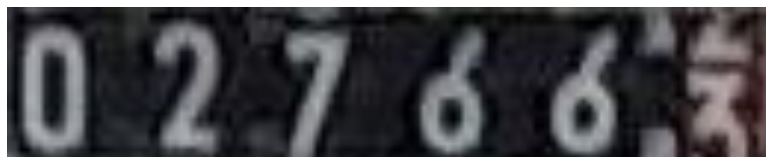
En las pruebas de distancia se obtuvieron los siguientes resultados:

Distancia mínima: Cuando el teléfono inteligente se encontraba entre 5 y 9 centímetros del medidor, la imagen digital obtenida era una imagen borrosa como se observa en la Figura 47 y la aplicación identificaba incorrectamente el valor numérico del medidor, sobre todo en el odómetro decimal. Después de los 10 centímetros de distancia, la aplicación obtenía valores numéricos correctos, de esta manera se determina que la distancia mínima para la captura de la imagen es de 10 centímetros.



Figura 47. Imagen digital obtenida en la prueba de distancia [Fuente Propia]

Distancia máxima: Cuando el teléfono inteligente se encontraba a 26 centímetros o más, la aplicación identifica una lectura numérica incorrecta, debido a que a esta distancia cuando se aplica el zoom en la función crop de Android para la selección de la región de interés, la resolución de la imagen digital resultante es muy pequeña como se observa en la Figura 48.



**Figura 48. Imagen digital obtenida a una distancia superior a 26 centímetros
[Fuente Propia]**

La distancia óptima entre el teléfono inteligente y el medidor para el registro de la lectura numérica se encuentra entre los 10 centímetros y los 25 centímetros. Se recomienda que la distancia para el registro de la lectura numérica sea de 15 centímetros para obtener resultados exactos; esta distancia fue la distancia usada para las pruebas de la aplicación en ambientes controlados y en ambientes con diferentes valores de luminosidad.

- **Pruebas de variación de ángulos**

Para determinar los ángulos críticos en el registro del valor numérico de los medidores, se realizaron tres pruebas variando los ángulos de la toma de la imagen en sus 3 ejes. En la Figura 49 se observa como una esfera en el teléfono inteligente representa la rotación de los ejes para las pruebas de variación de los ángulos.

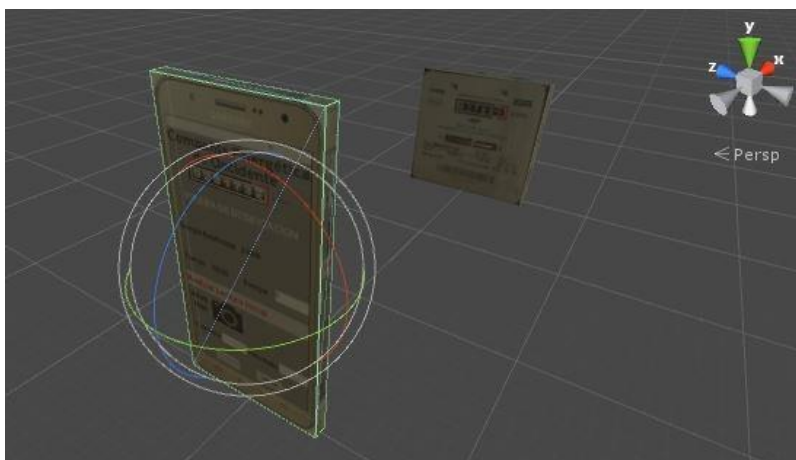


Figura 49. Representación de ejes de rotación en las pruebas de variación de ángulos [Fuente Propia]

Para identificar los ángulos de variación en el teléfono inteligente se utilizó una aplicación móvil llamada "Transportador - ON PROTRACTOR" que, a través de los acelerómetros del teléfono se identifica el ángulo que se ha rotado. La aplicación "Transportador - ON PROTRACTOR" se encuentra en la Play Store de Google.

Variación en el eje Y: Cuando se rotaba el eje Y (eje de color verde en la Figura 49) para el registro del valor numérico del medidor, se determinó que el máximo ángulo donde se puede rotar el eje y obtener una identificación exacta es de ± 20 grados, para los ángulos mayores a este valor, la identificación era incorrecta debido a que en la selección de la región de interés se seleccionan secciones no pertenecientes a los

odómetros del medidor como se observa en la figura x6 y eran identificados como dígitos adicionales, además el sesgo de la imagen era demasiado grande para que la aplicación lo corrigiera. Para los ángulos menores a ± 20 grados, el algoritmo de eliminación de sesgo (rotación de los dígitos) funcionaba correctamente y eliminaba este ángulo de rotación para una correcta identificación.



Figura 50. Imagen digital al rotar el eje Y [Fuente Propia]

Variación en el eje Z: Cuando se rotaba el eje Z (eje de color azul en la Figura 49) para el registro del valor numérico del medidor, se determinó que el máximo ángulo donde se puede rotar el eje Z y obtener una identificación exacta es de ± 30 grados, para los ángulos mayores a este valor, afecta más la resolución de la imagen que el ángulo de rotación. Cuando se rota el eje Z se varía la distancia de la cámara del teléfono al medidor como se observa en la Figura 49 y a la hora de seleccionar la región de interés, debido a que la distancia a variado bastante se debe hacer zoom a la imagen y la resolución se disminuye como se observa en la Figura 51.



Figura 51. Imagen digital al rotar el eje Z [Fuente Propia]

Variación en el eje X: La rotación de este eje es la más crítica, cuando se rota el eje X (eje de color rojo en la Figura 49) para el registro del valor numérico del medidor, se determinó que el máximo ángulo que se puede rotar el eje y obtener una identificación exacta es de ± 10 grados. Para ángulos mayores a ± 10 grados el sesgo de la imagen es muy grande como se observa en la Figura 52 y la aplicación no lo puede corregir, por lo tanto, la aplicación no puede identificar los valores numéricos. Para valores menores a ± 10 grados, la aplicación si puede corregir el sesgo de la imagen e identificar correctamente el valor numérico del medidor.



Figura 52. Imagen digital al rotar el eje X [Fuente Propia]

El valor óptimo de los ángulos para tomar la fotografía de los medidores es de 0 grados (no existe rotación) para los 3 ejes, pero con una variación de ± 5 grados (una buena fotografía) se obtienen excelentes resultados.

6.3 Interpretación de resultados

Puede concluirse que con la función de Android “*crop*” y el algoritmo de eliminación de sesgo (imagen torcida) la altura no es una variable crítica en las pruebas de dosificación, ya que, si se logra captar una buena imagen, con la función “*crop*” se puede seleccionar la región de interés (odómetros) y se elimina el sesgo. El usuario de la aplicación solo debe asegurarse en tomar la imagen en buen ángulo, donde se pueda captar la totalidad de los odómetros.

El algoritmo desarrollado para la binarización de las imágenes funciona correctamente sin importar el nivel de iluminación en el medidor, ya que para los dos medidores y los seis diferentes niveles de iluminación el algoritmo funcionó muy bien. Cabe destacar que este algoritmo funciona correctamente en medidores ciclométricos con valores numéricos de color blanco y fondo diferente al color blanco, para los medidores que no se encuentren en esta caracterización deben cambiarse los parámetros de binarización y operadores morfológicos.

El entrenamiento realizado en Tesseract para los dígitos específicos de los valores numéricos del medidor ISKRA y TECUN, fue un enteramiento bien realizado, ya que reconoció la totalidad (0 a 9) de los dígitos de los odómetros para los dos medidores y se puede concluir que, si se entrena un motor OCR para todos los medidores o por medidor individualmente, la aplicación funcionará para todos los posibles medidores que se pueden encontrar en Popayán.

Respecto a los resultados obtenidos en las pruebas realizadas a la aplicación, se puede concluir que la aplicación es más precisa que la percepción visual de las personas, en pocas palabras, la aplicación es más precisa que los operadores técnicos de las brigadas de la CEOC. Mientras los operadores técnicos pueden percibir lecturas máximas de 0,01 KWh (medidores en condiciones de realizarse las pruebas de verificación en sitio), la aplicación puede determinar lecturas numéricas de 0,001 KWh a 0,00001 KWh y así obtener valores más precisos del error porcentual del medidor.

De los 172 registros que tuvo que hacer la aplicación en las 86 pruebas que se realizaron, se observa en la Figura 55 que 157 registros fueron correctos (exactos) y 15 registros fueron erróneos, de esta manera se interpreta que la aplicación SLNM es 91.2% exacta.

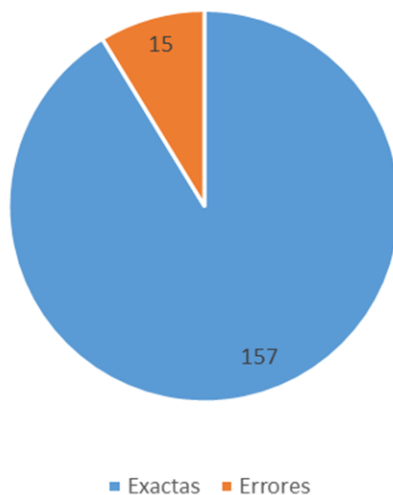


Figura 53. Pruebas de la aplicación SLNM [Fuente Propia]

De los 15 registros erróneos que tuvo la aplicación desarrollada, solo uno de esos errores fue lo suficientemente crítico para determinar que un medidor debía ser reemplazado cuando realmente funcionaba muy bien. Este error sucedió cuando había transición en el odómetro de la unidad y la aplicación identificó equivocadamente el valor numérico del medidor, por esta razón se determinó que el error del medidor superaba el $\pm 5\%$ que establece la NTC 5900 [37] para interpretar que el medidor debe ser reemplazado por uno nuevo. Bajo esta identificación, se interpreta que la aplicación tiene un porcentaje de 99.4% (ver Figura 55) de exactitud para valores críticos ($\pm 5\%$ de error del medidor).

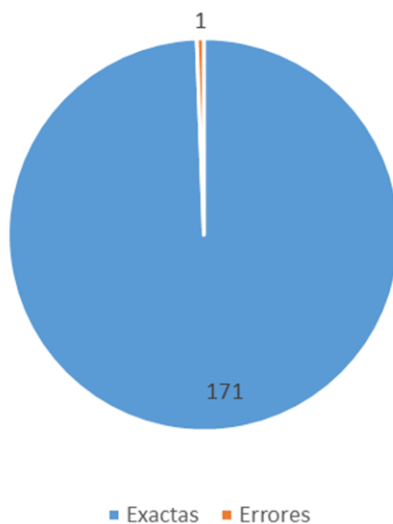


Figura 54. Pruebas de la aplicación SLNM para valores críticos [Fuente Propia]

Para lecturas numéricas del medidor, donde los dígitos no estaban en transición, la aplicación logró el 100% de pruebas correctas, mientras que, para lecturas numéricas en transición, logra solo el 72% de pruebas correctas. El peor de los casos, sucede cuando la aplicación reconoce un dígito incorrecto porque se encuentra en transición, si el dígito en transición corresponde al odómetro decimal, la aplicación puede agregar un máximo de 3.5% para medidores de resolución 0,01, siendo este valor inferior al rango del error permitido para la aceptación del medidor. Si el dígito en transición es el odómetro de la unidad, la aplicación puede agregar un error máximo de 44% para medidores de resolución 0,01 KWh, siendo este muy superior al error permitido, y para este caso particular la aplicación comete un error fatal dentro de la prueba de dosificación. Para medidores de menor resolución, el error agregado al error porcentual final será menor.

La aplicación falla en los dígitos en transición, en tres de los once posibles casos de transición. La aplicación falla cuando el porcentaje de tamaños en el odómetro es de 60/40, 50/50 y 40/60 (ver Figura 55).

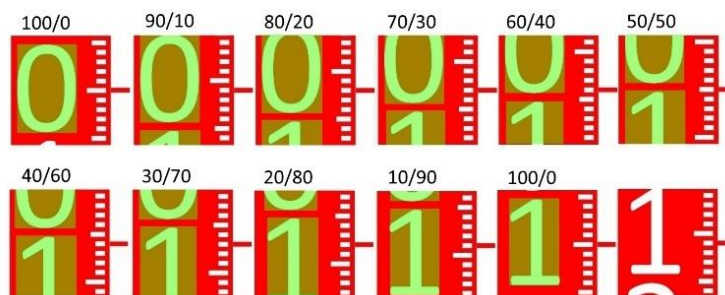


Figura 55. Proporciones de transición en el odómetro decimal [Fuente Propia]

En estos tres específicos casos (60/40, 50/50 y 40/60), la información obtenida de los números no es suficiente para que el motor OCR Tesseract determine el valor numérico correcto. Por ejemplo, en la figura 40, en la transición 60/40 el cero tiene una mayor proporción, por lo tanto, y según el algoritmo desarrollado, la región del cero será la ingresada al motor OCR, pero por las ecuaciones internas que realiza Tesseract, el contorno de esta región puede ser "0", "6" u "8" por qué se tiene muy poca información del número, y si Tesseract define que es un "8", el error agregado al error porcentual final será de 3%. En casos específicos, como el número de transición que involucra el dígito "1", por tener tan pocos contornos, no importa el valor de transición del dígito, ya que el motor OCR con muy poca información puede reconocerlo sin ningún problema.

En la Tabla 16 se presentan los resultados de tiempo de registro de la lectura inicial y la lectura final, para el operador de las brigadas PQR, el operador de las brigadas MACRO, el operador del Laboratorio de Calibración y Ensayo, el primer estudiante con la aplicación y el segundo estudiante con la aplicación.

Tabla 16. Comparación de tiempos de registro de lectura

	Tiempo (s)		
	L. Inicial	L. Final	Promedio
Operador PQR	45	47	48
Operador MACRO	45	51	
Laboratorista	47	54	
Estudiante 1	14	15	14
Estudiante 2	14	15	

En la Tabla 16 se puede observar que los operadores técnicos tardan en promedio *45 segundos* en registrar la lectura inicial con la metodología manual y *50 segundos* en promedio para registrar la lectura final. En cambio, los estudiantes con la aplicación desarrollada tardaron solo *14 segundos* en registrar la lectura inicial y solo *15 segundos* en registrar la lectura final, esto corresponde a la reducción del 70% en la reducción del tiempo necesario para registrar las lecturas. Además, con la aplicación desarrollada el operador no podrá ingresar una lectura errónea y no deberá realizar una prueba de nuevo, como sucedió con el operador MACRO. Cabe destacar que en ningún momento la aplicación sufrió detecciones o lentitud en su procesamiento como sucedía con la TGO y el AVM en las brigadas técnicas, lo que producía que el tiempo de registro aumentará.

Conclusiones y trabajos futuros

Conclusiones

- La metodología de binarización a través del modelo de color HSV demostró ser una metodología muy robusta a las variaciones de luminosidad, en las diferentes pruebas a diferentes condiciones de luminosidad el algoritmo funcionó de una manera correcta binarizando las imágenes y segmentado los dígitos numéricos de los odómetros.
- Aunque la función crop de Android es más lenta que la segmentación de los odómetros a través de la identificación de color rojo, esta función puede ser usada en cualquier tipo de contador, sin importar marca, fabricante o año de fabricación.
- La metodología propuesta para el cálculo de las centésimas y milésimas en las lecturas de los medidores, permitió alcanzar una exactitud de 0,001 KWh, siendo superior a la precisión de los operadores técnicos de las brigadas de la CEOC, además, el método de media ponderada resultó ser un procedimiento muy simple y fácil de utilizar en las aplicaciones de reconocimiento de lecturas numéricas de los medidores de energía eléctrica ciclométricos.
- Aunque la aplicación obtuvo resultados muy buenos, por el momento no es una herramienta tecnológica que satisfaga las necesidades de los operadores y la norma NTC 5900 sobre las pruebas de dosificación, esto es debido a que la aplicación fue desarrollada y entrenada para el reconocimiento de solo dos medidores (electrónico y mecánico), y solo en la ciudad de Popayán existe más de 500 diferentes medidores irreconocibles para la aplicación.

Trabajos futuros

- Para que la aplicación desarrollada pueda usarse para todo tipo de medidores, se debe crear una base datos de las características de cada medidor que se encuentra en el Departamento del Cauca, de esta manera la aplicación puede ser ajustable o adaptable automáticamente para cada medidor, ya sea en segmentación o en reconocimiento óptico de caracteres.
- Entrenar el motor OCR Tesseract individualmente para cada medidor que se encuentra en el Departamento del Cauca, de esta manera, dependiendo del medidor, la aplicación usara el lenguaje correspondiente para obtener un reconocimiento correcto.
- A los medidores que tiene resolución 1 no se les puede realizar la prueba de dosificación por que los operadores y su percepción visual no pueden lograr obtener lecturas de 0.01 como lo especifica las NTC, la aplicación puede ser modificada y mejorada para obtener valores decimales y centesimales de este tipo de medidor, de esta manera a estos tipos de medidores sería posible realizarles las pruebas de dosificación.

- En el Laboratorio de Calibración y Medición de la CEOC, en la prueba de verificación de la constante (la misma prueba de dosificación) realizada a los medidores de energía eléctrica, el registro de las lecturas de estos también es un proceso manual. La aplicación desarrollada en este proyecto puede ser adaptada o mejorada para automatizar la prueba de la verificación de la constante en el laboratorio de Calibración y Ensayo.

Referencias bibliográficas

- [1] O. M. Foong, S. Sulaiman, and K. K. Ling, "Text signage recognition in android mobile devices," *J. Comput. Sci.*, vol. 9, no. 12, pp. 1793–1802, 2013.
- [2] J. J. and others Fernández Hebrero, "Atención avanzada en emergencias con HCIS," 2015.
- [3] S. Madhan and M. Pradeep, "IMAGE PROCESSING OF ANDROID-BASED PATROL ROBOT FEATURING AUTOMATIC LICENSE PLATE RECOGNITION," vol. 3, no. March, pp. 296–301, 2015.
- [4] E. Hossain, Z. Han, and H. V Poor, "Smart Grid Communications and Networking," 2012.
- [5] J. Kotwal, S. Pawar, S. Pansare, M. Khopade, and P. Mahalunkar, "Android App for Meter Reading," vol. 4, no. 1, pp. 9853–9857, 2015.
- [6] R. Dayama, A. Chatla, H. Shaikh, and M. Kulkarni, "Android Based Meter Reading Using OCR," *Int. J. Comput. Sci. Mob. Comput.*, vol. 3, no. 3, pp. 536–539, 2014.
- [7] M. Suryawanshi, A. Nikam, A. Gajengi, and D. Waje, "Meter Reading Application for Android Based Handheld Device," *Ijarccce*, vol. 5, no. 1, pp. 101–103, 2016.
- [8] I. H. Bejarano, R. José, and S. García, "Reconocimiento de caracteres mediante imágenes en contadores de gas en entornos reales," 2015.
- [9] A. F. G. L. Aparicio Martín de Loeches, "Reconocimiento óptico de caracteres en imágenes digitales de contadores de gas," 2014.
- [10] M. V. Shinde and P. W. Kulkarni, "Automation of electricity billing process," *2014 Int. Conf. Power, Autom. Commun. INPAC 2014*, pp. 18–22, 2014.
- [11] K. Parthiban and A. M. Palanisamy, "Reading values in electrical meter using image processing techniques," *2013 Int. Conf. Intell. Interact. Syst. Assist. Technol. IISAT 2013 - Proc.*, pp. 1–7, 2013.
- [12] J. Garay, J. Palacio, A. Oliveira, S. Kofuji, and S. Barbosa, "Optical Character Recognition and Zigbee Wireless Communication for AMR: Conclusive Project," *Recent Patents Eng.*, vol. 7, no. 2, pp. 125–132, 2013.
- [13] A. Flammini and E. Sisinni, "Using smartglasses for utility-meter reading," 2015.
- [14] A. F. Mollah, N. Majumder, S. Basu, and M. Nasipuri, "Design of an Optical Character Recognition System for Camera-based Handheld Devices," *Int. J. Comput. Sci. Issues*, vol. 8, no. 4, pp. 283–289, 2011.
- [15] M. Chouiten and P. Schaeffer, "Vision based mobile Gas- - - Meter Reading

- Machine Learning method and application on real cases,” no. August, pp. 94–97, 2014.
- [16] L. A. Elrefaei, A. Bajaber, S. Natheir, N. Abusanab, and M. Bazi, “Automatic Electricity Meter Reading Based on Image Processing,” pp. 4–8, 2015.
- [17] M. Vanetti, I. Gallo, and A. Nodari, “GAS meter reading from real world images using a multi-net system,” *Pattern Recognit. Lett.*, vol. 34, no. 5, pp. 519–526, 2013.
- [18] K. Xie, “Automatic Utility Meter Reading,” 2010.
- [19] M. RODRÍGUEZ, G. BERDUGO, D. JABBA, M. CALLE, and M. JIMENO, “HD_MR: a new algorithm for number recognition in electrical meters,” *Turkish J. Electr. Eng. Comput. Sci.*, vol. 22, pp. 87–96, 2014.
- [20] a. F. Mollah, S. Basu, and M. Nasipuri, “Segmentation of Camera Captured Business Card Images for Mobile Devices,” *Int. J. Comput. Sci. Appl.*, vol. 1, no. 1, pp. 33–37, 2010.
- [21] T. Kasar, J. Kumar, and A. Ramakrishnan, “Font and background color independent text binarization,” *Second Int. Work. camera-based Doc. Anal. Recognit.*, vol. 2, no. 1, pp. 3–9, 2007.
- [22] L. Gomez and D. Karatzas, “Multi-script text extraction from natural scenes,” *Proc. Int. Conf. Doc. Anal. Recognition, ICDAR*, pp. 467–471, 2013.
- [23] L. Neumann and J. Matas, “Real-Time Lexicon-Free Scene Text Localization and Recognition,” *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 38, no. 9, pp. 1872–1885, 2016.
- [24] L. Neumann and J. Matas, “Text localization in real-world images using efficiently pruned exhaustive search,” *Proc. Int. Conf. Doc. Anal. Recognition, ICDAR*, pp. 687–691, 2011.
- [25] R. Holley, “How good can it get?: Analysing and improving OCR accuracy in large scale Historic Newspaper Digitisation Programs,” *D-Lib Mag.*, vol. 15, no. 3–4, pp. 1–13, 2009.
- [26] N. White, “Training Tesseract for Ancient Greek OCR,” *Eutypon*, vol. 28–29, pp. 1–11, 2013.
- [27] M. Nayak, “Odia Characters Recognition by Training Tesseract OCR Engine Odia Characters Recognition by Training Tesseract OCR Engine,” no. September, 2016.
- [28] R. C. Gonzalez, R. E. Woods, and S. L. Eddins, “Digital image using Matlab processing,” *Pers. Prentice Hall, Lexingt.*, 2004.
- [29] T. Acharya and A. K. Ray, “Image Processing: Principles and Applications,” *Image (Rochester, N.Y.)*, pp. 1–3, 2005.

- [30] A. Garcia and B. R. Amazaga, *Fundamentos de fotografía digital*. Universidad de Cantabria, Aula de Fotografía, 1995.
- [31] A. Thuewissen, "Image sensor architectures for digital cinematography," 2005.
- [32] M. A. Verdugo Alonso, O. Gonzalez, M. del C. M. A. Verdugo, and others, *Personas con discapacidad: perspectivas psicopedagógicas y rehabilitadoras*, no. 376. Siglo Veintiuno de España, 2005.
- [33] J. ALBERS, "La Interaccion del Color, Madrid, Edit," *Alianza Forma*, 1998.
- [34] J. C. Russ and R. P. Woods, "The image processing handbook," *J. Comput. Assist. Tomogr.*, vol. 19, no. 6, pp. 979–981, 1995.
- [35] A. Chambolle, V. Caselles, D. Cremers, M. Novaga, and T. Pock, "An introduction to total variation for image analysis," *Theor. Found. Numer. methods sparse Recover.*, vol. 9, pp. 263–340, 2010.
- [36] Electrificadora del meta S.A. E.S.P., "Capítulo 5. Medición de energía eléctrica," pp. 1–5.
- [37] I. Internacional, "Norma Tecnica Colombiana 5900," no. 571, 2011.
- [38] I. Internacional, "Norma Tecnica Colombiana 4856," no. 571, 2015.
- [39] C. E. de Occidente, "MODULO II - NTC 5900." p. 15.
- [40] C. E. de Occidente, "MÓDULO III AVM CARGA FANTASMA." p. 16.
- [41] C. E. de Occidente, "MODULO I - CONCEPTOS BÁSICOS." p. 48.
- [42] Gartner Inc., "Gartner Says Worldwide Smartphone Sales Grew 3.9 Percent in First Quarter of 2016," 2016. [Online]. Available: <http://www.gartner.com/newsroom/id/3323017>.
- [43] B. C. Zapata, *Android studio application development*. Packt Publishing Ltd, 2013.
- [44] A. Romano and C. Luna, "Descripción y análisis del modelo de seguridad de Android," *Reportes Técnicos 13-08*, 2013.
- [45] A. V. Romano, "Descripción y análisis formal del modelo de seguridad de android," 2014.
- [46] A. Allan, *Learning iPhone Programming: From Xcode to App Store*. " O'Reilly Media, Inc.," 2010.
- [47] N. Randolph and C. Fairbairn, *Professional Windows Phone 7 Application Development: Building Applications and Games Using Visual Studio, Silverlight, and XNA*. Wrox Press Ltd., 2010.
- [48] S. Kapur and N. Thakkar, *Mastering OpenCV Android Application Programming*. 2015.

- [49] D. Bloomberg, "Leptonica: An open source C library for efficient image processing, analysis and operation." 2007.
- [50] R. Smith, D. Antonova, and D. Lee, "Adapting the Tesseract Open Source OCR Engine for Multilingual OCR," *MOCR Proc. Int. Work. Multiling. OCR*, no. Cc, pp. 1–8, 2009.
- [51] R. Smith, "An overview of the tesseract OCR engine," *Proc. Int. Conf. Doc. Anal. Recognition, ICDAR*, vol. 2, pp. 629–633, 2007.
- [52] C. Ponsard, R. Ramdoyal, and D. Dziamski, "An ocr-enabled digital comic books viewer," in *International Conference on Computers for Handicapped Persons*, 2012, pp. 471–478.
- [53] M. C. G. Mantilla, L. L. C. Ariza, and B. M. Delgado, "Metodologia para el desarrollo de aplicaciones moviles," *Rev. Tecnura*, vol. 18, no. 40, pp. 20–35, 2014.
- [54] N. Kew, *Desarrollo de modulos y aplicaciones con Apache*. 2008.
- [55] R. Theis, "Tess-two," *Dispon. en <https://github.com/rmtheis/tess-two>*. *Consult. en*, vol. 17, 2016.
- [56] D. E. Pantalla and N. Introducci, "Diseño de pantalla," pp. 1–10, 2007.
- [57] ANSI-Approval-Date, "Ansi/lec 60529-2004," *Natl. Electr. Manuf. Assoc.*, p. 8, 2004.
- [58] R. Vique, "Métodos para el desarrollo de aplicaciones móviles," *Pid_00176755*, p. 66, 2006.
- [59] N. Stanton, P. M. Salmon, and L. A. Rafferty, *Human factors methods: a practical guide for engineering and design*. Ashgate Publishing, Ltd., 2013.
- [60] M. Lyons, "Towards a framework to select techniques for error prediction: supporting novice users in the healthcare sector," *Appl. Ergon.*, vol. 40, no. 3, pp. 379–395, 2009.
- [61] N. A. Stanton, "Hierarchical task analysis: Developments, applications, and extensions," *Appl. Ergon.*, vol. 37, no. 1, pp. 55–79, 2006.
- [62] C. Contreras, R. Manuel, M. Chirinos, and M. Araque, "Propuesta de diseño de una interfaz movil de reportes de trafico vehicular, aplicando el metodo agil de desarrollo de software 'Agilu,'" *Academia*, vol. 13, no. 29, pp. 45–72, 2014.
- [63] G. Iriarte, U. Gran, and M. De Ayacucho, "Diseño y Evaluación de Material Instruccional para la Enseñanza del Lenguaje Gestual a Traves de Software Educativo , para el Profesional de la Docencia," pp. 22–28, 2006.
- [64] M. F. Triola, *Estadística*. Pearson educacion, 2004.

Anexo 1

Manual de usuario: Sistema de lectura numérico para medidores (SLNM)

1. Descripción de la aplicación:



Figura 1. Logo de la aplicación.

La aplicación Sistema de lectura numérico para medidores con sus siglas SLNM es una aplicación para *Smartphone* con sistema operativo Android versión 4.4 KitKat o posterior, que permite hacer de forma automática la prueba de dosificación que realizan las brigadas de la compañía energética de occidente como parte del protocolo de las verificaciones en sitio, ya sean en las PQRS o en las campañas MACRO. La aplicación es nativa, por lo que las conexiones son rápidas y la interfaz está simplificada para el uso en el teléfono. Las funciones de la aplicación son: inicio de sesión, realizar la prueba de dosificación, listar historial de pruebas realizadas y

guardar en la base de datos, además de llevar un registro de las fotos tomadas en cada prueba, las cuales se almacenan en la galería del equipo.

2. Requerimientos



Figura 2. Smartphone Samsung Galaxy J7 4G

Los requerimientos mínimos para que la aplicación SLNM funcione correctamente, son los siguientes:

- Sistema operativo Android 5.0 (Lollipop) o mayor.
- Cámara trasera de 8 MP o superior.
- Procesador Qualcomm MSM8939 Snapdragon 615 octa-core (4xCortex-A57@1.4GHz + 4xCortex-A53@1GHz), GPU Adreno 405 / Exynos 7580 octa-core 1.5GHz, GPU Mali-T720MP2
- Memoria RAM de 1.5GB RAM
- Conectividad (3G y/o WIFI).

- Tener instalado OpenCV Manager. Nota, si no tiene instalado OpenCV Manager, al momento de iniciar la aplicación aparece una ventana emergente donde lo redirigirá al enlace de descarga de OpenCV Manager.

3. Instalación

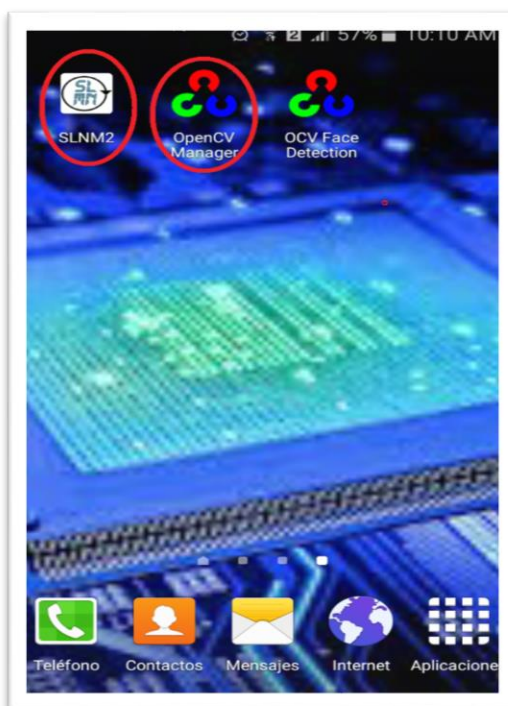


Figura 3. Pantalla en el teléfono que muestra los íconos de la aplicación SLNM y OpenCV Manager

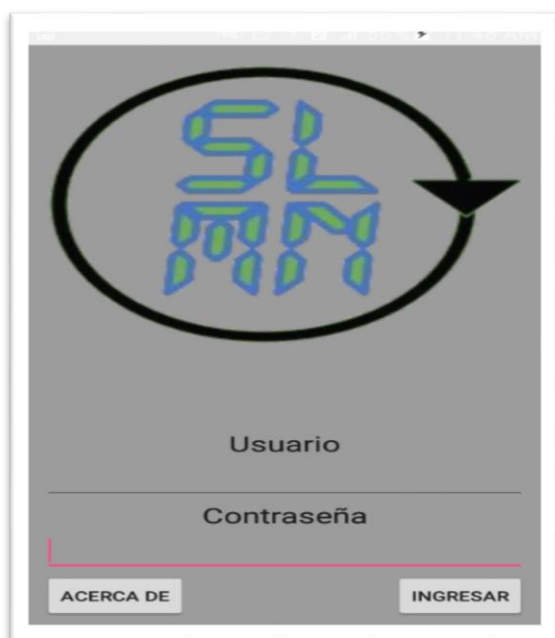
Una vez que se descarga la aplicación en el teléfono, la instalación de la misma se realiza de manera automática. Cuando la aplicación se ha instalado correctamente es preciso ubicar el ícono ejecutable (como se muestra en la Figura 3) y con solo tocarlo se comenzará a ejecutar la misma. Para hacer uso de la aplicación SLNM, es necesario instalar una aplicación complementaria, la cual es OpenCV Manager. La biblioteca OpenCV es utilizado por otras aplicaciones para la mejora de imágenes, composición panorámica, detección de objetos y reconocimiento, etc. OpenCV Manager proporciona la mejor versión de la biblioteca OpenCV para su hardware. También recibe las últimas actualizaciones de estabilidad y rendimiento para la biblioteca. Sólo resta tocar el icono de la aplicación para abrirla y empezar a trabajar con ella.

4. Navegación de la aplicación



4.1. Pantalla inicial:

Al ubicar el ícono de la aplicación SLNM en el menú de aplicaciones o en la pantalla del teléfono presionar sobre la aplicación para que se inicie de manera automática. Después de esto se desplegará esta pantalla, la cual contiene una animación con el logo y eslogan de la Compañía Energética, así como un dibujo animado. Aquí se tendrá que volver a tocar la pantalla para que te envíe a la pantalla de inicio de sesión.



4.2. Pantalla de inicio de sesión

Para utilizar la aplicación el operario debe tener un usuario y una contraseña, las cuales serán proporcionadas por la Compañía Energética de Occidente. Si los valores ingresados son correctos, el operario podrá ingresar a la pantalla principal presionando el botón "INGRESAR", de lo contrario aparecerá un mensaje de advertencia y el operario podrá intentarlo de nuevo.

Además, en esta pantalla el usuario podrá elegir entre ir a la pantalla de información o ingresar a la prueba de dosificación. Para ingresar a la pantalla de información, no es necesario el inicio de sesión, solamente se deberá presionar el botón **“ACERCA DE”**. Para ingresar a la pantalla principal el usuario es “admin” y la contraseña es “12345”.

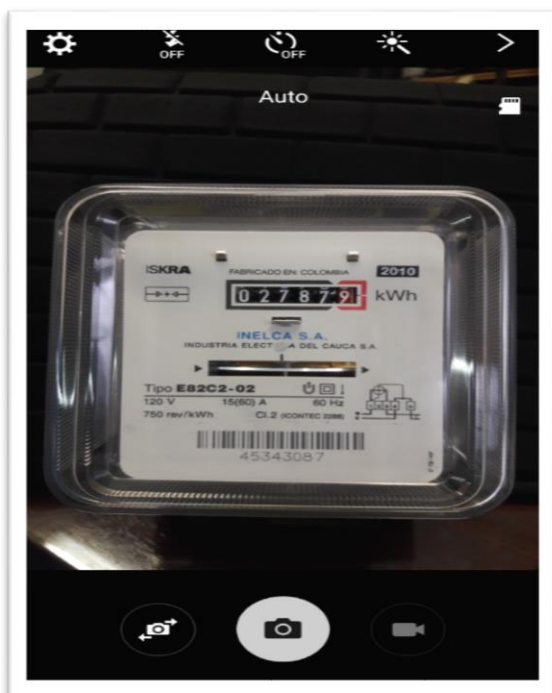
4.3. Pantalla de información

En esta pantalla el usuario tendrá información acerca de los creadores de la aplicación, así como el nombre y la versión de la misma. El logo de la universidad de cauca por ser la entidad educativa que formó a estos estudiantes



4.4. Pantalla principal

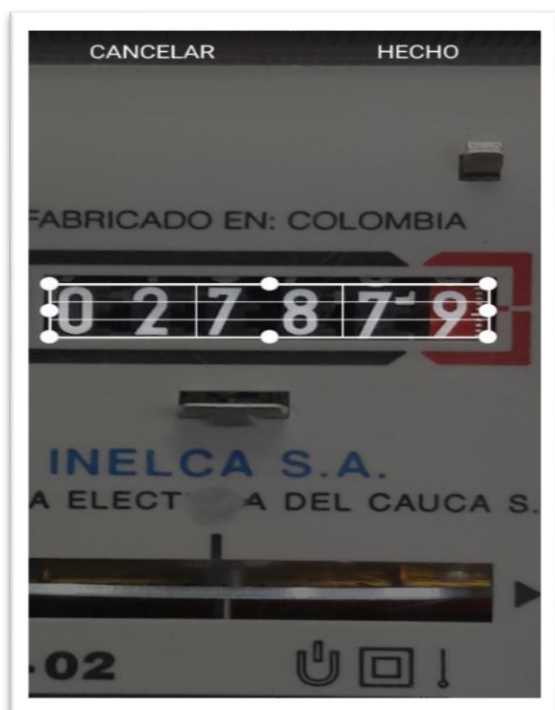
Se denomina pantalla principal porque es aquí donde se realiza la prueba de dosificación. La interfaz se diseñó de tal manera que el operario no pueda equivocarse en los pasos que se deben seguir a la hora de realizar esta prueba. Además existe una barra con letras en rojo, ubicada en el medio de la pantalla, la cual indica los pasos que se deben ir realizando. Cuando apenas se ingresa, sale la pantalla que está a la izquierda con solo un botón habitado, el cual es **“LECTURA INICIAL”**, por esta razón solo se permite realizar esta acción. Se deberá presionar para poder ingresar a la cámara y tomar la correspondiente foto.



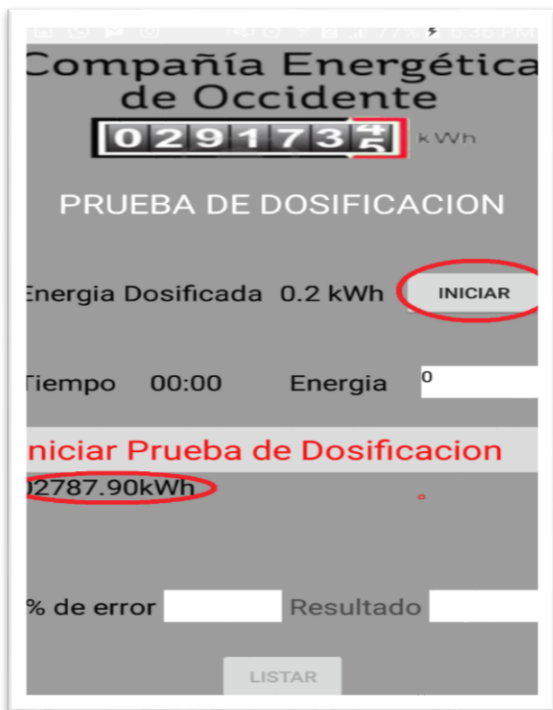
Al presionar el botón **“LECTURA INICIAL”**, la aplicación abrirá la cámara, tal como aparece en la imagen de la izquierda, y el operario tendrá que buscar una buena posición para tomar la foto, de esto dependerá el éxito del reconocimiento de los números.



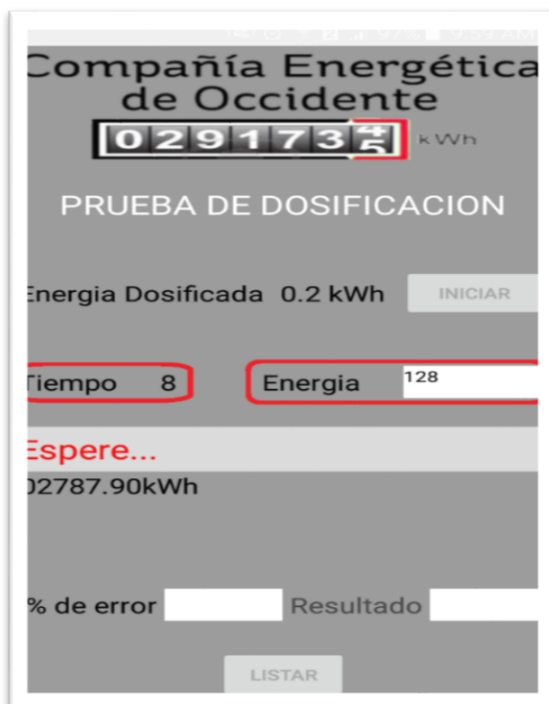
Una vez tomada la foto, el operario podrá revisar si está conforme con la captura y presionar el botón **“OK”** que aparece en la parte superior izquierda para poder continuar con el proceso, de no ser así y se desea tomar otra foto, el operario deberá presionar el botón **“REINTENTAR”**



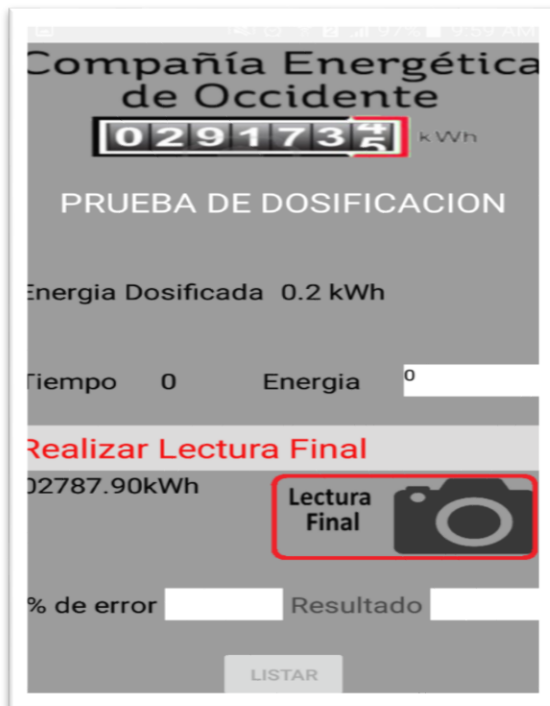
Después de haber validado la foto, el operario tendrá que escoger la región de interés, la cual es el odómetro. Para eso la aplicación cuenta con un recuadro ajustable y además de eso se podrá acercar o alejar la imagen deslizando los dedos en forma de expansión o contracción para elegir una mejor región de interés. Luego de esto el operario validará esta acción con el botón **“HECHO”** el cual se encuentra en la parte superior izquierda, de lo contrario podrá retroceder con el botón **“CANCELAR”**.



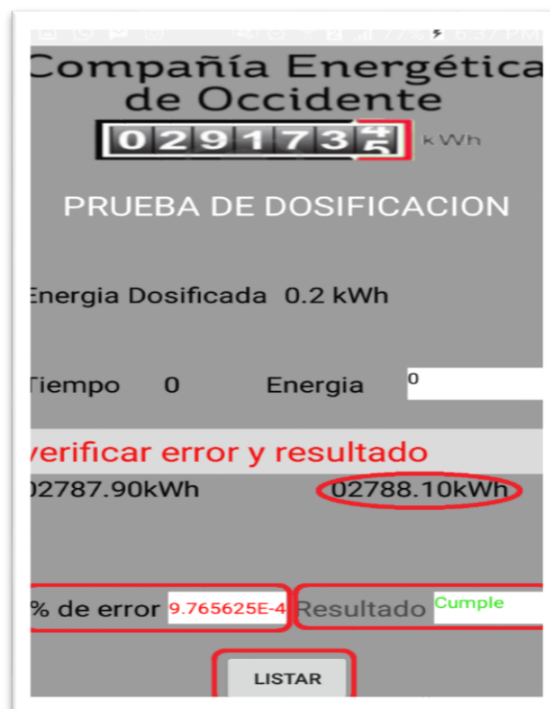
Después de tomar, validar y recortar la foto inicial, la aplicación enviará la región de interés elegida a la parte donde se realizara de forma automática todo el pre procesamiento y reconocimiento óptico de caracteres. Cuando termine el proceso, la aplicación enviará el número reconocido a la pantalla principal y se mostrará cómo se señala en la figura de la izquierda. Además de eso se habilitara el botón **"INICIAR"** con el cual se dará la orden para que empieza la dosificación de la energía.



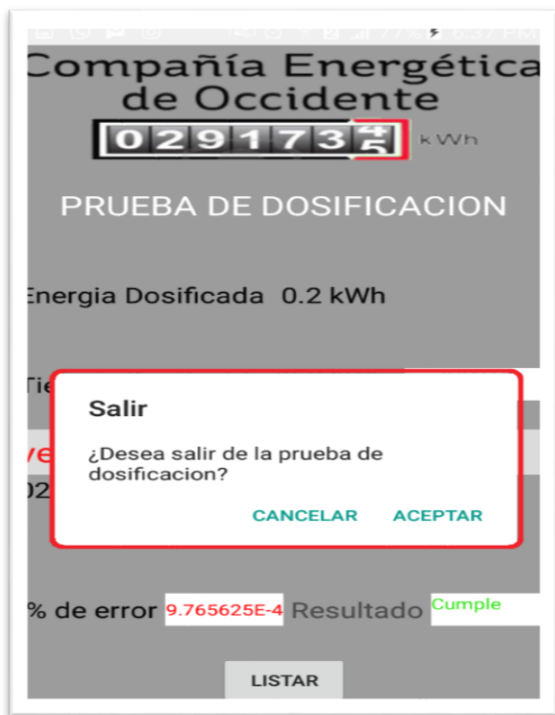
Luego de presionar el botón **"INICIAR"**, el tiempo y la energía empezaran a decrementar simulando el consumo energético por parte del medidor eléctrico. Se puede observar que la imagen de la izquierda que también se inhabilita este mismo botón, esto restringe un reinicio de la prueba.



Cuando finalice la dosificación de la energía y el tiempo estimado para esto, desaparece el botón “INICIAR” y aparece el botón “LECTURA FINAL”. Este botón tiene la misma funcionalidad que el botón “LECTURA INICIAL”, por lo que se tendrán que repetir los pasos mencionados en esta lectura.



Después de realizar todos los pasos para obtener la lectura final, la aplicación tendrá las 2 lecturas necesaria para mostrar el error y el resultado del equipo. Además de esto se habilitara el botón “LISTAR”, el cual nos enviara a una pantalla que maneja la base de datos



La pantalla principal debe tener prevención a la hora de salir de la prueba, por ello cuenta con una funcionalidad que le pregunta al operario si desea salir de la prueba en cualquier momento. Si el usuario por algún motivo pulsa el botón de atrás que tiene todo *Smartphone*, aparecerá el mensaje de la figura de la izquierda, con dos opciones, “**CANCELAR**” para continuar en la prueba de dosificación y “**ACEPTAR**” para confirmar la salida.



4.5. Pantalla de base de datos.

En esta pantalla se listan de forma ordenada los valores obtenidos en la prueba de dosificación y quedan listos para ser agregados a la base de datos



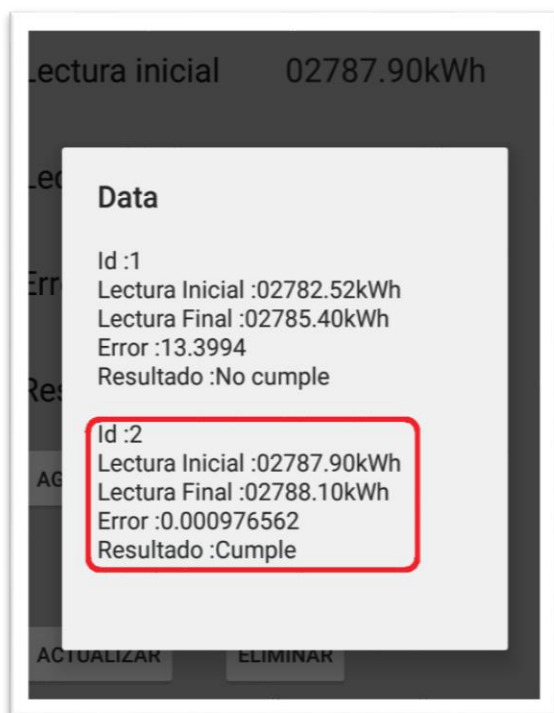
Al presionar el botón “**AGREGAR DATOS**”, los resultados de “**LECTURA INICIAL**”, “**LECTURA FINAL**”, “**ERROR**” y “**RESULTADO**” que aparecen al lado izquierdo de la pantalla, se guardaran en una base de datos en el teléfono, y además de eso se mostrará el mensaje “**Datos Insertados**” el cual validará la acción anterior.



El botón “**ACTUALIZAR**” sirve para reingresar los datos obtenidos en la prueba y renovarlos en la base de datos. Al presionarlo parecerá el mensaje “**Datos Actualizados**” para validar la acción anterior.



El botón “**ELIMINAR**” borrará de la base de datos los valores que aparecen actualmente en la pantalla. Al presionarlo aparecerá el mensaje “**Datos Borrados**” validando la acción anterior.



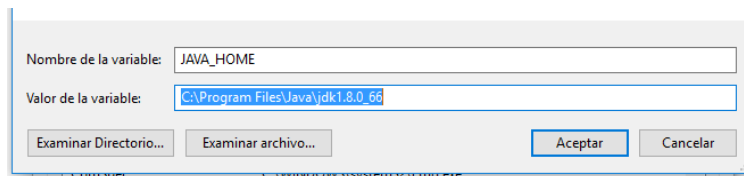
El botón “**VER DATOS**” tiene la función de mostrar los datos que hasta el momento se han ingresado a la base de datos. Al presionarlo nos aparecerá en la pantalla los datos de cada prueba ordenados por un identificador, el cual hace referencia al orden que lleva la base de datos.

Anexo 2

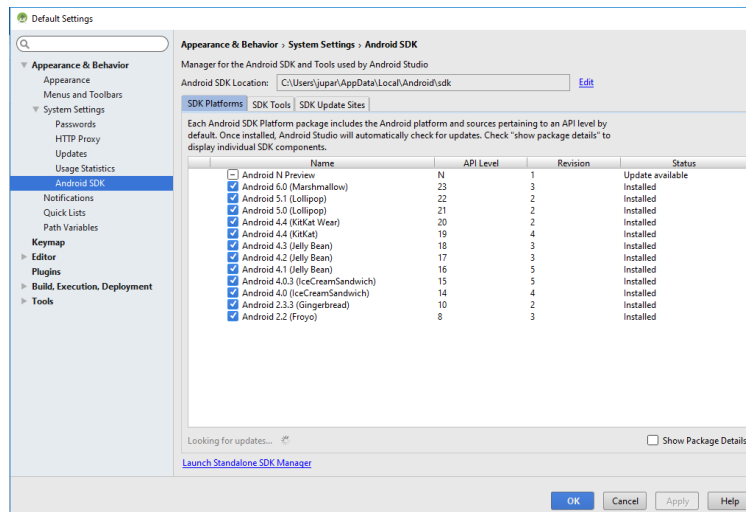
Instrucciones de instalación Android Studio, OpenCV y Tesseract OCR

Instalar Android Studio en Windows 10

1. Ingresar a <https://developer.android.com/studio/index.html> y clicar en el enlace de descarga de Android Studio 2.2.1.0.
2. Ingresar a <http://www.oracle.com/technetwork/java/javase/downloads/jdk8-downloads-2133151.html>, descargue e instale el “Java SE Development Kit 8u101” para su versión de Windows (x86 o x64).
3. Seleccione **Panel de control > Sistema y seguridad > Sistema > Configuración avanzada del sistema > Variables de entorno**. En variables de sistema click en “Nueva”, y añadir **JAVA_HOME**, y debe seleccionar la carpeta donde instaló java, como, por ejemplo:

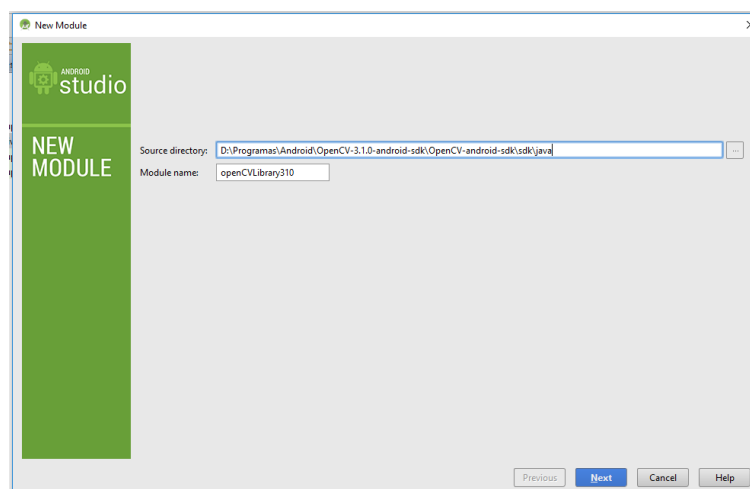


4. Abrir Android Studio, en la pantalla de inicio ir a “Configure” luego a “SDK Manager”, se debe descargar todas las plataformas SDK (SDK Platforms) y las herramientas SDK (SDK Tools) como se observa en la siguiente imagen:



Agregar OpenCV a Android Studio

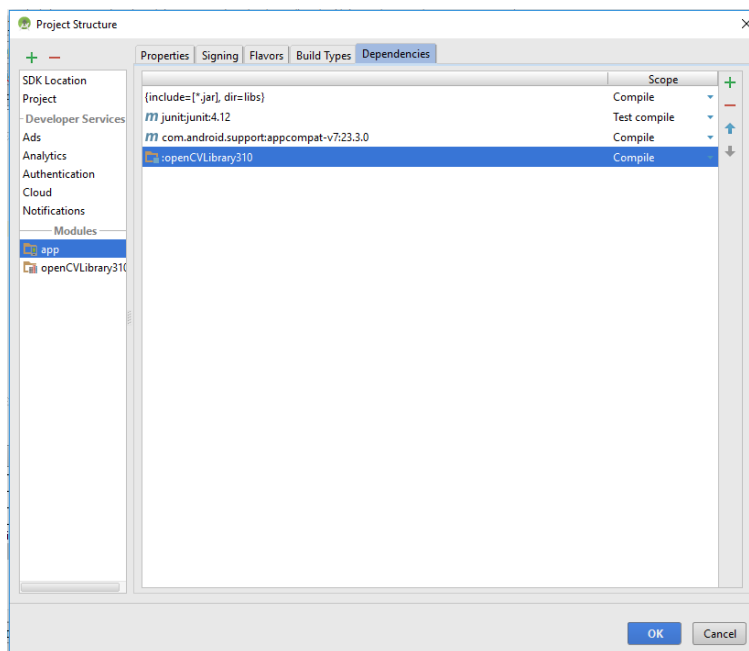
1. Ingresar a <http://opencv.org/>, descargar y descomprimir “OpenCV for Android”.
2. Abrir Android Studio, en la pantalla de inicio dar click en “Start a new Android project”, “Next”, “Next”, “Next” y “Finish”.
3. Para importar las librerías de OpenCV se debe ir a **File > New > Import Module...** y seleccionar a carpeta donde descomprimieron OpenCV, se debe buscar la carpeta **sdk/java**. “Next” y Finish”.



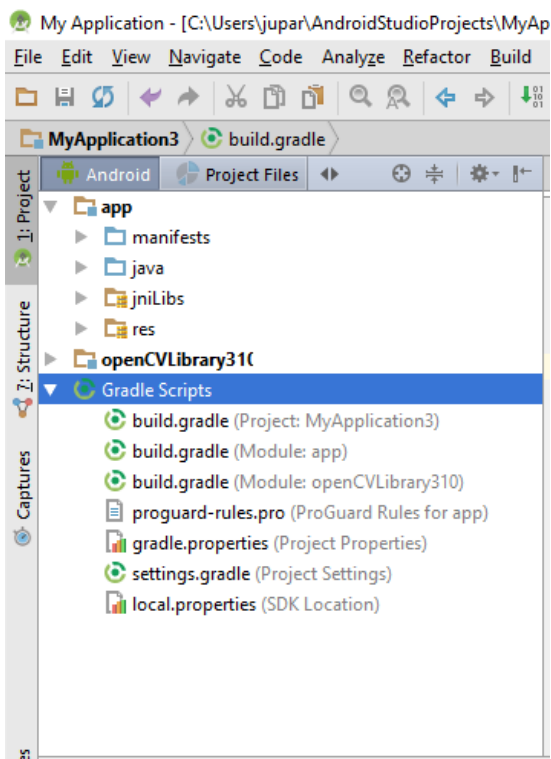
4. Ahora se debe crear un módulo de dependencias, para esto se debe ir a “Project Structure” (Ctrl + Alt + Mayús + S) o el símbolo:



5. Ahora se debe ir a “Modules” y seleccionar “app”, dar click en el agregar (“+”), “Module dependencies”, seleccionar “openCVLibrary310” y “OK”.



6. En este punto ya se ha agregado las librerías OpenCV, pero en Android Studio aparecerán errores de compilación, para corregir esto se debe ir a “Project” (superior izquierda), “Android” y “Gradle Scripts”.



7. Se debe dar doble click en *build.gradle (Module: app)* y en *build.gradle (Module: openCVLibrary310)*. En *openCVLibrary310* se debe verificar que

compileSdkVersion, buildToolsVersion, minSdkVersion y targetSdkVersion son iguales a los valores en *app*, de no ser así, se deben digitar los mismos valores.

```

android {
    compileSdkVersion 23
    buildToolsVersion "23.0.2"

    defaultConfig {
        minSdkVersion 15
        targetSdkVersion 23
    }

    buildTypes {
        release {
            minifyEnabled false
            proguardFiles getDefaultProguardFile('proguard-android.txt'), 'proguard-rules.txt'
        }
    }
}

```

```

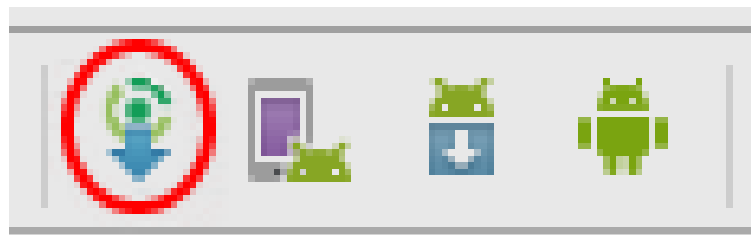
android {
    compileSdkVersion 23
    buildToolsVersion "23.0.2"

    defaultConfig {
        applicationId "com.example.jupar.myapplication"
        minSdkVersion 15
        targetSdkVersion 23
        versionCode 1
        versionName "1.0"
    }

    buildTypes {
        release {
            minifyEnabled false
            proguardFiles getDefaultProguardFile('proguard-android.txt'), 'proguard-rules.pro'
        }
    }
}

```

- Ahora se debe compilar el proyecto, para esto se debe dar click en “Sync Project with Gradle Files” o al símbolo:



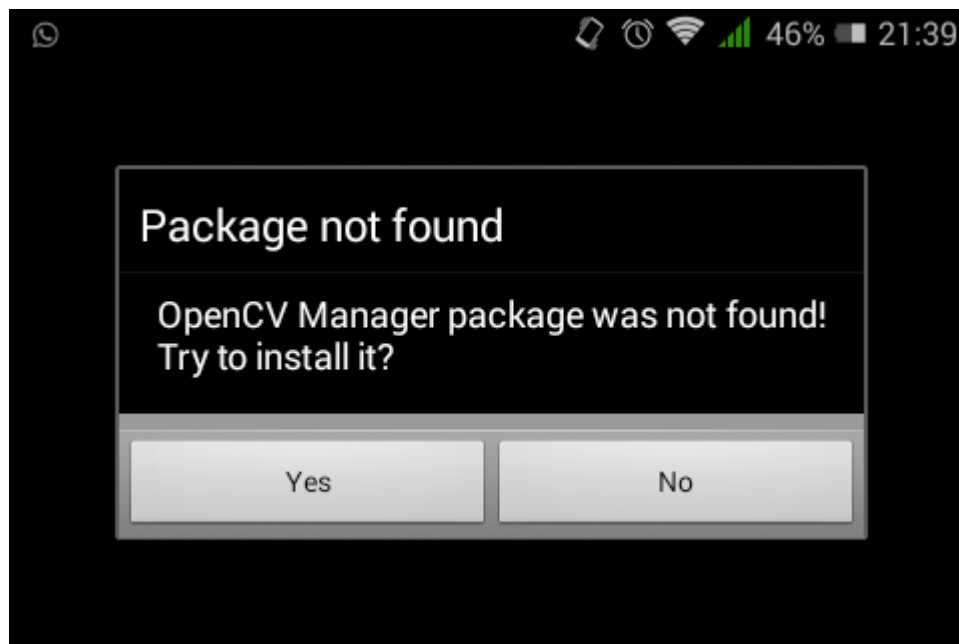
- Después de compilado no debe existir ningún tipo de error, esto se puede comprobar en “Messages”:

```

Messages Gradle Build
:openCVLibrary310:generateDebugAndroidTestResources
:openCVLibrary310:mergeDebugAndroidTestResources
:openCVLibrary310:processDebugAndroidTestResources
:openCVLibrary310:generateDebugAndroidTestSources
BUILD SUCCESSFUL
Total time: 16,147 secs
0 errors
0 warnings
See complete output in console

```

- Cuando descarga y ejecuta esta aplicación (Hello Word) en un teléfono inteligente con sistema operativo Android, aparecerá una ventana emergente donde le informará que debe descargar e instalar OpenCV Manager. OpenCV Manager es una aplicación que permite utilizar la librería OpenCV en Android.



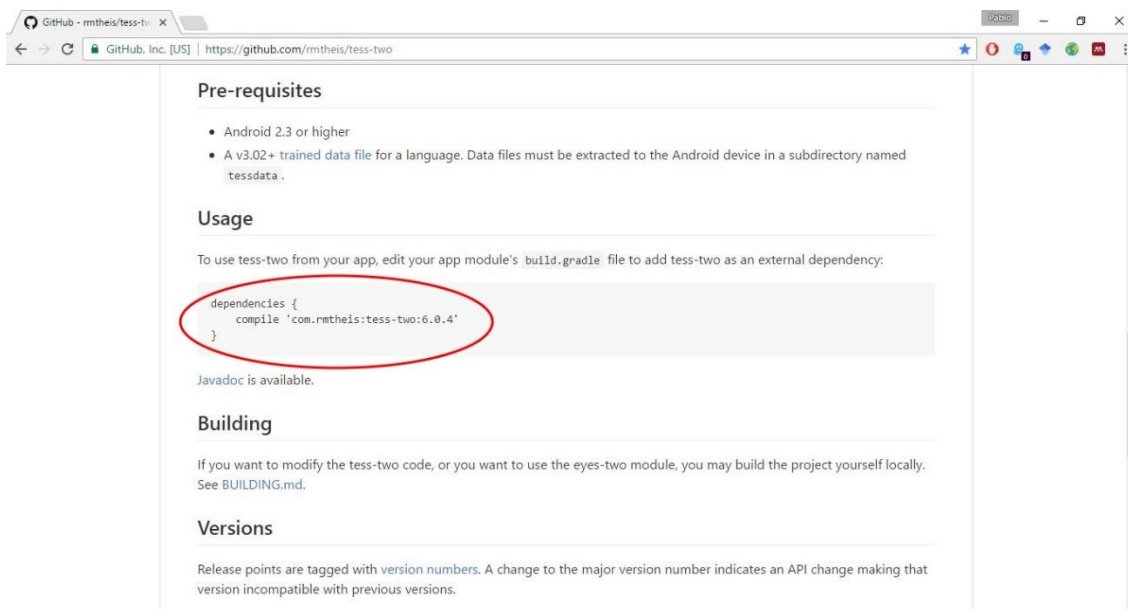
11. La ventana emergente lo redirigirá a la cuenta de Google Play donde podrá descargar e instalar OpenCV Manager. Ya que sin esta aplicación no se puede usar OpenCV en Android, es indispensable tener suficiente memoria para descargarlo.



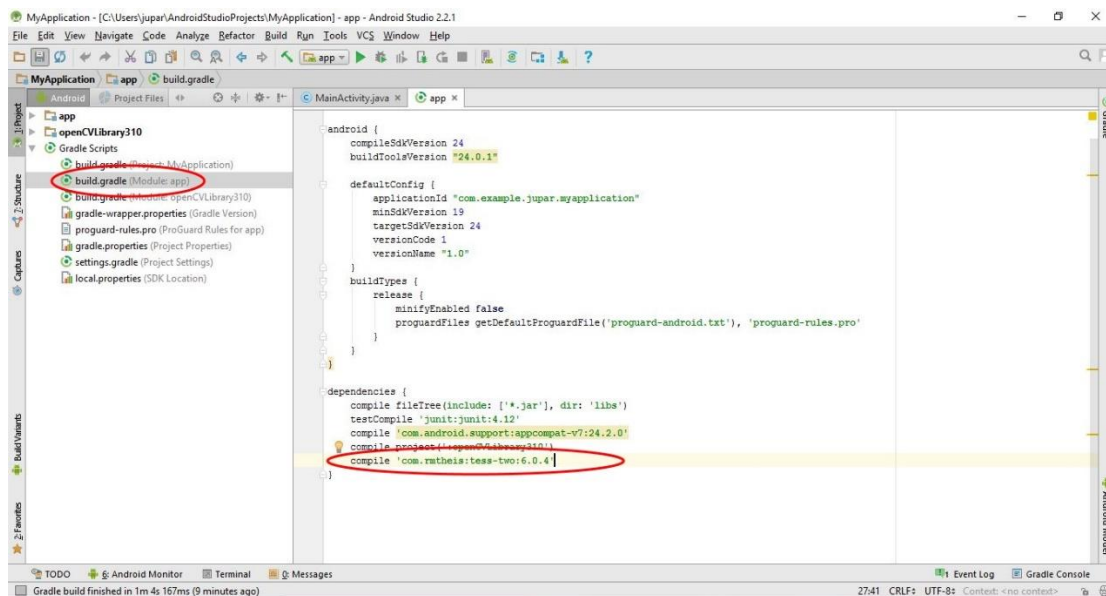
12. En este punto ya se puede empezar a realizar la programación de la aplicación. Para cerrar el proyecto se debe ir a **File > Close Project**.

Agregar Tesseract OCR en Android Studio

1. Antes de agregar el motor Tesseract OCR en Android Studio se debe verificar la última versión disponible de este en la página del proyecto Tess-Two: <https://github.com/rmtheis/tess-two>.



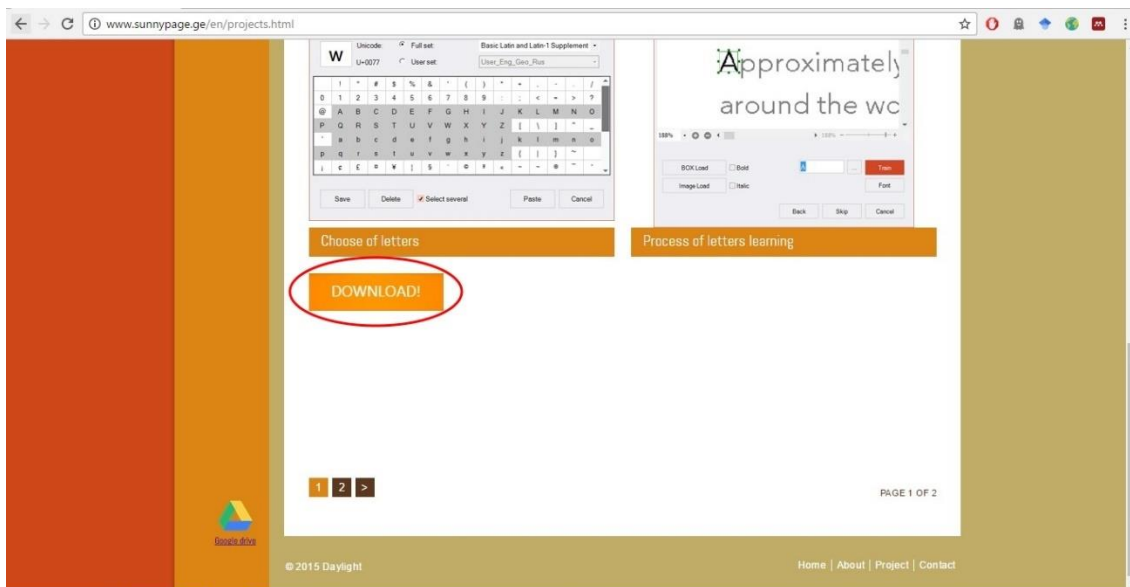
2. En Android Studio, en el proyecto donde desea agregar Tesseract debe ir a "build.gradle (Module:app)" y en dependencias (dependencies) debe agregar el texto de la última versión encontrada en la página anteriormente referenciada. Para este caso, el texto de la última versión era: `compile 'com.rmtheis:tess-two:6.0.4'`. Al sincronizar el proyecto, Tesseract OCR se instalará.



Anexo 3

Instrucciones de entrenamiento del motor OCR Tesseract.

1. Ingresar a <http://www.sunnypage.ge/en/projects.html>, descargar e instalar "SunnyPage 2.7".



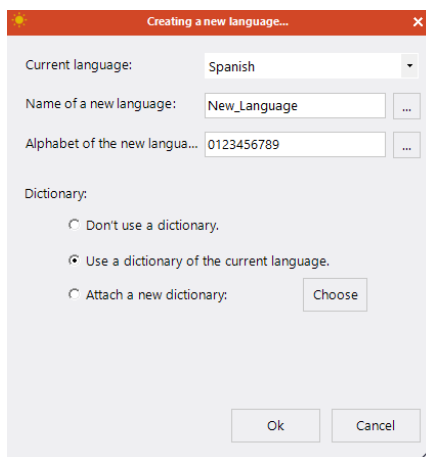
2. Abrir SunnyPage. SunnyPage es gratuito temporalmente, después de un tiempo aparecen ventanas emergentes informando que debe comprar la licencia, solo se debe dar click en "Registration" y luego en "Trial". Esta versión Trial permite realizar el entrenamiento del Motor OCR Tesseract.
3. En Sunny Page, click en "Open File" y seleccionar la imagen con que va a ser entrenado el Motor OCR, esta imagen debe ser con los números en negro y el fondo blanco como se ve en la siguiente imagen.



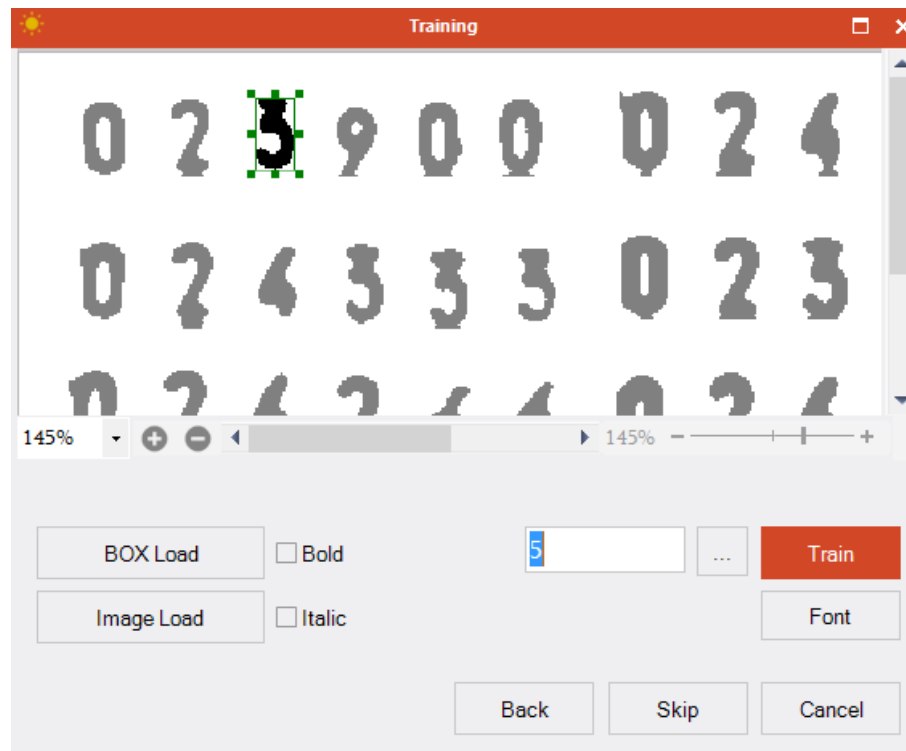
4. Si va a entrenar un nuevo lenguaje en Tesseract deberá seleccionar la opción “Creating a new language based on the current language” (Color amarillo oscuro). Si va a mejorar el entrenamiento a un lenguaje ya creado deberá seleccionar “Improvement of the currently language” (Color Verde) y no olvidar en la opción “Current language” (Elipse color rojo) seleccionar el lenguaje previamente creado.



5. Para crear un nuevo lenguaje, en las opciones deberá seleccionar el lenguaje español, darle un nuevo nombre al lenguaje que va a crear, en el alfabeto escribir solo los números ya que vamos a entrenar el lenguaje para reconocer los dígitos de los medidores eléctricos y en la opción diccionario seleccionar la opción de usarlo. Las opciones deben ser seleccionadas como aparecen en la siguiente imagen:



6. Después aparecerá una imagen donde se identifica el dígito según la región de la imagen, deberá comprobar que el dígito identificado corresponde con el valor numérico, si el valor es correcto deberá dar click en “train” si el valor no es el corrector de sebera digitar el valor correcto y click en “train”. Eso se deberá repetir para todos los dígitos identificados.



7. El archivo “tessdata” generado a partir del entrenamiento se encuentra en la carpeta de instalación de SunnyPage, en este caso se encuentra en “C:\Program Files (x86)\Daylight\SunnyPage 2,7\tessdata”.

