



**ESPE**  
UNIVERSIDAD DE LAS FUERZAS ARMADAS  
INNOVACIÓN PARA LA EXCELENCIA

**DEPARTAMENTO DE CIENCIAS DE LA COMPUTACIÓN**

**CARRERA DE INGENIERÍA EN TECNOLOGÍAS DE LA INFORMACIÓN  
(SISTEMAS E INFORMÁTICA)**

**TRABAJO DE TITULACIÓN, PREVIO A LA OBTENCIÓN DEL TÍTULO  
DE INGENIERO EN SISTEMAS E INFORMÁTICA**

**TEMA: SISTEMA PARA CONTEO DE PASAJEROS DEL TRANSPORTE  
PÚBLICO URBANO EN TIEMPO REAL UTILIZANDO CÁMARAS DE  
VIDEO, TECNOLOGÍAS Y ALGORITMOS OPEN SOURCE.  
CASO DE ESTUDIO: EMPRESA OPTIMOVILIDAD**

**AUTORA: CUEVA VERA, JESSICA PAOLA**

**DIRECTOR: ING. PARRA MOREIRA, PABLO XAVIER**

**SANGOLQUÍ**

**2018**

## CERTIFICADO TUTOR



DEPARTAMENTO DE CIENCIAS DE LA COMPUTACIÓN  
CARRERA DE INGENIERÍA EN TECNOLOGÍAS DE LA  
INFORMACIÓN (SISTEMAS E INFORMÁTICA)

### CERTIFICACIÓN

Certifico que el trabajo de titulación, **“SISTEMA PARA CONTEO DE PASAJEROS DEL TRANSPORTE PÚBLICO URBANO EN TIEMPO REAL UTILIZANDO CÁMARAS DE VIDEO, TECNOLOGÍAS Y ALGORITMOS OPENSOURCE. CASO DE ESTUDIO: EMPRESA OPTIMOVILIDAD”** fue realizado por la señorita **Cueva Vera, Jéssica Paola** el mismo que ha sido revisado en su totalidad, analizado por la herramienta de verificación de similitud de contenido; por lo tanto cumple con los requisitos teóricos, científicos, técnicos, metodológicos y legales establecidos por la Universidad de Fuerzas Armadas ESPE, razón por la cual me permito acreditar y autorizar para que lo sustente públicamente.

Sangolquí, 16 de febrero del 2018

Firma:

Ing. Parra Moreira, Pablo Xavier

C.C.: .....1802733012.....

## AUTORÍA DE RESPONSABILIDAD



### DEPARTAMENTO DE CIENCIAS DE LA COMPUTACIÓN

### CARRERA DE INGENIERÍA EN TECNOLOGÍAS DE LA INFORMACIÓN (SISTEMAS E INFORMÁTICA)

#### AUTORÍA DE RESPONSABILIDAD

Yo, **Cueva Vera, Jéssica Paola**, declaro que el contenido, ideas y criterios del trabajo de titulación: **“Sistema para conteo de pasajeros del transporte público urbano en tiempo real utilizando cámaras de video, tecnologías y algoritmos OpenSource. Caso de Estudio: Empresa Optimovilidad”** es de mi autoría y responsabilidad, cumpliendo con los requisitos técnicos, metodológicos y legales establecidos por la Universidad de Fuerzas Armadas ESPE, respetando los derechos intelectuales de terceros y referenciando las citas bibliográficas.

Consecuentemente el contenido de la investigación mencionada es veraz.

Sangolquí, 16 de febrero del 2018

---

Cueva Vera Jéssica Paola  
C.C. 1719690610

## AUTORIZACIÓN



DEPARTAMENTO DE CIENCIAS DE LA COMPUTACIÓN  
CARRERA DE INGENIERÍA EN TECNOLOGÍAS DE LA INFORMACIÓN  
(SISTEMAS E INFORMÁTICA)

### AUTORIZACIÓN

Yo, **Cueva Vera Jéssica Paola**, autorizo a la Universidad de las Fuerzas Armadas ESPE publicar el trabajo de titulación: ***“Sistema para conteo de pasajeros del transporte público urbano en tiempo real utilizando cámaras de video, tecnologías y algoritmos OpenSource. Caso de Estudio: Empresa Optimovilidad”*** en el Repositorio Institucional, cuyo contenido, ideas y criterios son de mi responsabilidad.

Sangolquí, 16 de febrero del 2018

---

Cueva Vera Jéssica Paola  
C.C. 1719690610

## DEDICATORIA

Al motor de mi vida, al ser máspreciado que Dios puso en mi camino, a mi pequeña Melita, este logro es nuestro, porque cada mañana al despertar con tú sonrisa me motiva para seguir adelante y no caer, a pesar de todos los problemas. Este proyecto de titulación te lo dedico con mucho amor a ti hija mía.

A mis padres, porque nunca dejaron de confiar en mí, por cada consejo brindado, por la motivación diaria de seguir adelante y culminar con mi carrera universitaria. Lo logramos papis.

A ti papito, por ser un gran hombre y ejemplo a seguir, por nunca dejarme caer, y siempre motivarme a caminar con la frente en alto hacia adelante, porque a pesar de todos mis tropiezos nunca desconfiaste en mí y me apoyaste en todo momento, por ser un segundo padre para mi Melita y brindarme tanto cariño y cuidados, te lo mereces todo papi, esto te lo quiero dedicar a ti con mucho amor.

A mi querida hermana Daniela, por ser mi alma gemela y acompañarme con todas las locuras, por el apoyo que me brindaste. Te amo ñañaíta.

## **AGRADECIMIENTO**

Agradezco en primer lugar a Dios y a la Virgen, por nunca dejarme caer en los momentos difíciles. Gracias por toda la sabiduría y la salud depositada en mí todos estos años de estudio, para culminar con éxito un logro más en vida.

A la Universidad de las Fuerzas Armadas ESPE, por la oportunidad brindada de estudiar y ser una profesional.

A mi director de proyecto de investigación, Ing. Pablo Parra, gracias por su esfuerzo y dedicación en este tiempo, pues con su experiencia y conocimiento manifestado.

Al Ing. Édison Yáñez y todos quienes conformar la empresa Optimovilidad un agradecimiento especial por la gran acogida en su institución y el apoyo brindado para el presente proyecto.

A mis padres y hermanos, gracias por apoyarme en todo momento, por ser el pilar fundamental en mi vida y un ejemplo a seguir siempre, gracias por todos los consejos que aportaron en la culminación de mi carrera.

Y a todas las personas que en estos años de estudio compartieron gratos momentos a mi lado y aportaron para culminar esta carrera universitaria.

## ÍNDICE

CARÁTULA	
CERTIFICADO TUTOR.....	i
AUTORÍA DE RESPONSABILIDAD.....	ii
AUTORIZACIÓN.....	iii
DEDICATORIA.....	iv
AGRADECIMIENTO.....	v
Resumen.....	xii
Abstract.....	xiii
CAPÍTULO 1.....	1
ASPECTOS GENERALES.....	1
1.1    Introducción.....	1
1.2    Planteamiento del problema.....	2
1.3    Justificación.....	3
1.4    Objetivos.....	4
CAPÍTULO 2.....	5
ESTADO DEL ARTE.....	5
2.1    Tecnologías para conteo de personas.....	5
2.1.1    Video cámaras.....	5
2.1.2    Sensor Infrarrojo.....	6
2.1.3    Sensores de movimiento infrarrojos.....	6
2.2    Algoritmos de conteo de personas.....	6
2.2.1    Conteo LOI.....	7
2.2.2    Conteo ROI.....	9
2.3    Sistemas inteligentes de transporte público.....	14
2.3.1    Sistemas de Ayuda a la Explotación (SAE).....	14
2.3.2    Sistemas Integrados de Recaudo (SIR).....	15
2.3.2    Sistemas de Información al Usuario (SIU).....	16
2.4    Raspberry Pi 3.....	16
2.5    Cámaras.....	18
2.6    Software OpenCV.....	18
2.6.1    Módulos de OpenCV.....	21

2.7	Lenguaje de programación .....	22
2.7.1	Python.....	22
2.7.2	PHP .....	24
2.8	Base de datos MySQL.....	26
2.9	Metodología de desarrollo ágil SCRUM .....	2
2.8.1	Roles.....	3
2.8.1	Fases .....	4
CAPÍTULO III.....		8
ANÁLISIS DEL ALGORITMO DE CONTEO Y REQUISITOS DEL SISTEMA.....		8
3.1	Análisis de métodos y algoritmos de conteo de personas.....	8
3.2	Especificación de los requerimientos. ....	39
3.2.1	Requisitos funcionales .....	40
3.2.1	Requisitos no funcionales .....	41
3.3	Descripción de los requerimientos.....	41
3.4	Especificación y descripción de roles del sistema. ....	46
3.5	Personal involucrado .....	46
CAPÍTULO IV .....		48
DISEÑO, DESARROLLO E IMPLEMENTACIÓN DEL SISTEMA DE CONTEO .....		48
4.1.	Desarrollo del algoritmo de conteo .....	48
4.1.1	Diagrama de flujo del algoritmo de conteo.....	48
4.1.2	Captura del video .....	49
4.1.3	Detección y sustracción de fondo .....	50
4.1.4	Generación de blobs .....	51
4.1.5	Corrección de blobs utilizando procesos morfológicos .....	53
4.1.6	Seguimiento de blobs.....	55
4.1.7	Conteo de entradas y salidas.....	57
4.2.	Instalación y configuración de OpenCV en la Raspberry Pi 3 .....	60
4.3.	Modelo conceptual de la base de datos del sistema .....	62
4.3.	Modelo lógico de la base de datos del sistema .....	63
4.4.	Diagrama de clases.....	63
4.5.	Diagrama de secuencia.....	64
4.8.	Arquitectura del sistema .....	67
4.8.1	Diagrama de integración de hardware .....	67



4.8.2 Diagrama de arquitectura del sistema.....	68
CAPÍTULO V .....	69
PRUEBAS DEL ALGORITMO DE CONTEO Y DEL SISTEMA.....	69
5.1. Casos de prueba del algoritmo de conteo .....	69
5.1.1 Escenario 1: Valor del umbral adecuado para la detección de objetos .....	70
5.1.2 Escenario 2: Pruebas del conteo realizado por el algoritmo. ....	78
CONCLUSIONES Y RECOMENDACIONES.....	83
6.1. Conclusiones.....	83
6.2. Recomendaciones.....	85
BIBLIOGRAFÍA.....	87
ANEXOS.....	

## Índice de Tablas

<b>Tabla 1</b> <i>Características de Matlab y OpenCV</i> .....	20
<b>Tabla 2</b> <i>Rango de comparativa</i> .....	27
<b>Tabla 3</b> <i>Comparativa de gestores de base de datos</i> .....	2
<b>Tabla 4</b> <i>Fases de la metodología Scrum</i> .....	4
<b>Tabla 5</b> <i>Características de RUP, XP y Scrum</i> .....	6
<b>Tabla 6</b> <i>Métodos y algoritmos de la categoría LOI</i> .....	35
<b>Tabla 7</b> <i>Métodos y algoritmos de la categoría ROI</i> .....	37
<b>Tabla 8</b> <i>Comparativa de algoritmos y métodos de conteo</i> .....	39
<b>Tabla 9</b> <i>Requisitos funcionales</i> .....	40
<b>Tabla 10</b> <i>Requisitos no funcionales</i> .....	41
<b>Tabla 11</b> <i>Historia de Usuario HU-01</i> .....	42
<b>Tabla 12</b> <i>Historia de Usuario HU-02</i> .....	43
<b>Tabla 13</b> <i>Historia de Usuario HU-03</i> .....	44
<b>Tabla 14</b> <i>Historia de Usuario HU-04</i> .....	45
<b>Tabla 15</b> <i>Historia de Usuario HU-05</i> .....	45
<b>Tabla 16</b> <i>Roles del sistema web</i> .....	46
<b>Tabla 17</b> <i>Personal involucrado</i> .....	47
<b>Tabla 18</b> <i>Resultados de la detección de objetos</i> .....	71
<b>Tabla 19</b> <i>Resultados de detección de objetos con nuevo valor de umbral</i> .....	75
<b>Tabla 20</b> <i>Resultados del conteo final</i> .....	79
<b>Tabla 21</b> <i>Resultado final del sistema conteo</i> .....	81

## Índice de Figuras

<b>Figura 1</b> Agrupamiento bayesiano .....	8
<b>Figura 2</b> Resultados de la detección.....	10
<b>Figura 3</b> Procesamiento de imagen del algoritmo.....	13
<b>Figura 4</b> Sistema de ayuda a la explotación .....	15
<b>Figura 5</b> Raspberry Pi 3.....	17
<b>Figura 6</b> Funciones de OpenCV .....	19
<b>Figura 7</b> Proceso de ejecución de PHP .....	25
<b>Figura 8</b> Interacción de una base de datos.....	26
<b>Figura 9</b> Proceso Metodología Scrum .....	6
<b>Figura 10</b> Diagrama de flujo del algoritmo de conteo .....	48
<b>Figura 11</b> Captura de frames a) con personas. b) sin personas.....	50
<b>Figura 12</b> Sustracción de fondo en OpenCV .....	51
<b>Figura 13</b> Detección de contornos .....	52
<b>Figura 14</b> Generación de blobs.....	53
<b>Figura 15</b> Procesos morfológicos .....	54
<b>Figura 16</b> Seguimiento de blobs .....	57
<b>Figura 17</b> Conteo de entradas y salidas .....	59
<b>Figura 18</b> Almacenamiento de datos MySQL .....	60
<b>Figura 19</b> Modelo conceptual de la base de datos .....	62
<b>Figura 20</b> Modelo lógico de la base de datos .....	63
<b>Figura 21</b> Diagrama de clases .....	63
<b>Figura 22</b> Diagrama de secuencia – Reporte del conteo.....	64
<b>Figura 23</b> Diagrama de secuencia – Gestionar usuarios .....	65
<b>Figura 24</b> Diagrama de secuencia – Gestionar unidades.....	66
<b>Figura 25</b> Diagrama de arquitectura de hardware .....	67
<b>Figura 26</b> Diagrama de arquitectura del sistema .....	68
<b>Figura 27</b> Sistema de conteo implementado.....	69

<b>Figura 28</b> Detección de movimiento salidas a) Captura normal b) Captura con substracción de fondo.....	70
<b>Figura 29</b> Detección de movimiento entradas a) Captura normal b) Captura con substracción de fondo.....	71
<b>Figura 30</b> Resultado capturado del video .....	73
<b>Figura 31</b> Resultado del sitio web.....	73
<b>Figura 32</b> Detección de movimiento a) Captura normal b) Captura con substracción de fondo.....	75
<b>Figura 33</b> Nuevo resultado capturado del video .....	77
<b>Figura 34</b> Nuevo resultado del sitio web .....	77
<b>Figura 35</b> Entrada de dos personas.....	78
<b>Figura 36</b> Resultado final en el procesamiento de video .....	80
<b>Figura 37</b> Resultado final en el sitio web .....	80
<b>Figura 38</b> Porcentaje de error y efectividad del conteo.....	82

## Resumen

En el presente trabajo se implementó un sistema de conteo de pasajeros en tiempo real utilizando el lenguaje de programación y librerías de código abierto, además se investigaron y analizaron una serie de algoritmos de conteo seleccionando el más adecuado y que cumpla con las características necesarias del escenario de una unidad del transporte público donde la concurrencia de personas no es muy elevada en horas valle. A continuación, se desarrolló y codificó el algoritmo adecuado donde el diagrama de flujo consistió en las siguientes actividades: la captura del video; sustracción de fondo de cada frame o cuadro capturado; detección y seguimiento de cada persona; finalizando con el conteo que permite determinar si es una entrada o salida. Adicionalmente, se desarrolló la interfaz de usuario web, la misma que permite visualizar los datos numéricos del conteo procesado por el algoritmo en tiempo real, y mediante reportes con fechas personalizadas cada socio puede consultar el número de entradas y salidas de su unidad de transporte terrestre. Finalmente, se realizaron las pruebas de: configuración del valor de umbral y porcentaje de error del algoritmo de conteo, colocando localmente en la oficina de la empresa Optimovilidad una cámara USB y para el procesamiento del mismo una Raspberry Pi.

### Palabras clave

- **ALGORITMO DE CONTEO**
- **PYTHON**
- **OPENCV**
- **SUSTRACCIÓN DE FONDO**

## **Abstract**

In the present work, a real-time passenger counting system was developed and implemented using the programming language and open source libraries. In addition, a series of counting algorithms were investigated and analyzed, selecting the most appropriate and complying with the necessary characteristics. of the scenario of a public transport unit where the concurrence of people is not very high during off-peak hours. Next, the appropriate algorithm was developed and codified where the flow chart consisted of the following activities: the capture of the video; background subtraction of each captured frame or frame; detection and monitoring of each person; ending with the count that allows to determine if it is an entry or exit. Additionally, the web user interface was developed, which allows to visualize the numerical data of the count processed by the algorithm in real time, and by means of reports with personalized dates, each partner can check the number of inputs and outputs of their land transport unit. Finally, the tests were performed: configuration of the threshold value and percentage of error of the counting algorithm, placing a USB camera locally in the office of the company Optimovilidad and for processing it a Raspberry Pi.

## **Keywords**

- **COUNTING ALGORITHM**
- **PYTHON**
- **OPENCV**
- **BACKGROUND SUBTRACTION**

# CAPÍTULO 1

## ASPECTOS GENERALES

### 1.1 Introducción

Actualmente, las herramientas tecnológicas usadas para el conteo en tiempo real de objetos y personas han ido evolucionando día tras día, logrando de esta manera que su implementación y manejo se lo realice en diferentes lugares, inclusive en el transporte público. Entre los beneficios del conteo de personas se tiene el de obtener información en línea de: datos del tráfico de personas que utiliza un bus urbano; cantidad de personas que utilizan terminales terrestres; personas que se encuentran en paradas o estaciones de transferencia de sistemas integrados; cantidad de personas que visitan u observan un almacén de un centro comercial; número de peatones que cruzan las intersecciones para diseñar semaforización, entre otros. (Counterest, 2017)

Estos datos facilitarán la toma de decisiones de los socios y del diseño de soluciones de tráfico, como las conocidas zonas 30, o ciudades amigables con el medio ambiente. (Urdaneta, 2016)

La Comisión de Movilidad del Municipio de la ciudad de Quito reguló la implementación de Sistemas Inteligentes en el Transporte Público de la ciudad, estos consisten en los siguientes: Sistema de Ayuda a la Explotación (SAE) el cual realiza la gestión integral del servicio de transporte público de pasajeros basado en tecnologías de localización (GPS) para mejorar y optimizar la regularidad del servicio de transporte; Sistema de Información al Usuario (SIU), herramienta tecnológica que brinda información a los usuarios de manera oportuna, confiable y adecuada mediante paradas, sitios web, teléfonos celulares, entre otros; y el Sistema Integrado de Recaudo (SIR) que tiene por objetivo implementar un procedimiento de confiabilidad y seguridad en la recaudación de pasajes, en este tipo de Sistema, los buses del transporte urbano hacen uso de

contadores con electrónica infrarroja que permite a los socios conocer el número exacto de pasajeros, sin embargo los mismos son manipulados por el personal que labora en la unidad obteniendo información errónea. (Movilidad, 2015)

La empresa de Sistemas de Movilidad Inteligente Optimovilidad S.A., es una compañía ecuatoriana dedicada a la provisión de soluciones tecnológicas para el sector del transporte terrestre en sus distintas modalidades, en especial del transporte público (urbano, interparroquial, intercantonal, interprovincial). Los principales objetivos empresariales son: Atender las necesidades del mercado de transporte terrestre productivo; Permitir a los socios del sector contar con herramientas informáticas y tecnológicas; Optimizar sus inversiones y mejorar su productividad; Investigar y desarrollar nuevos servicios y productos tecnológicos orientados al transporte público, entre otros. (Optimovilidad, 2017)

## **1.2 Planteamiento del problema**

Actualmente la empresa Optimovilidad provee sus servicios de control y operación de flota en las ciudades de Quito, Ambato, Riobamba, Machala y Salinas utilizando un sistema de conteo de pasajeros con circuitos infrarrojos denominado contador INS50, el cual se tiene operativo en las cooperativas de transporte Translatinos, Quitumbe, Tumbaco, 21 de Julio, Vencedores, Victoria, Unión Ambateña, Oroconti, Via Flores y Libertadores, este sistema de conteo se encuentra instalado en las barreras laterales de la puerta delantera y trasera del bus, la misma que cuenta los ingresos y salidas de los pasajeros respectivamente.

El principal problema que se tiene con este sistema y que se pretende solucionar con este proyecto es el alto porcentaje de evasión económica por parte de choferes y ayudantes con los propietarios de los buses, equivalente al 30% de recaudación diaria. Se estima recuperar al menos un 50% de la evasión de los ingresos. El personal que labora en la unidad manipula los equipos contadores para que los mismos muestren información incoherente y errónea a los propietarios de la unidad, los informes de trabajo



elaborados por los técnicos de la empresa Optimovilidad menciona que los problemas comunes para evasión son los siguientes: golpear en las barras laterales causando daños a los circuitos de conteo; el ayudante se para delante de las barras mientras los pasajeros ingresan; mojar las barras laterales para causar daños y problemas a los circuitos de conteo; y colocar bufandas, prendas, entre otros delante de las barras laterales para originar bloqueos y obtener datos erróneos. Esto genera inconformidad en los socios por lo que se necesita plantear una serie de políticas y procesos de control extras a implementar, creando más trabajo, además de los reclamos continuos al personal de soporte técnico de la empresa. (Optimovilidad, 2017)

Finalmente, el conteo en los terminales o paradas de integración en los sistemas de transporte integrados como Trolebús, Metrovía o cualquier otro medio de transporte masivo es hasta el momento muy complicado ya que no se conoce de manera precisa la cantidad de viajes y transbordos que se realiza. (Movilidad, 2015) En Ecuador es difícil implementar sistemas tecnológicos de pago de pasaje por el manejo político y no técnico que tiene el sector. El proyecto que se está diseñando es una propuesta que contribuirá a solucionar estos problemas de forma eficaz y eficiente.

### **1.3 Justificación**

El sistema para conteo de pasajeros en tiempo real es una solución innovadora y eficiente basada en algoritmos que permiten realizar el conteo bidireccional simultáneo de los pasajeros que suben y bajan del bus. La empresa Optimovilidad con su capacidad de innovación pretende atender las nuevas necesidades del mercado del transporte público terrestre solucionando los problemas que los socios de las unidades enfrentan a diario para mejorar su productividad.

Las cooperativas de transporte público urbano del país actualmente buscan herramientas tecnológicas para el recaudo integrado de todas las unidades. El sistema para conteo que se propone tiene como finalidad garantizar la información del número de pasajeros que utilizan el transporte público a un costo accesible a la realidad del país, así

el socio maximizará la inversión y minimizará las pérdidas que se tienen por la evasión en conductores y ayudantes.

En el desarrollo del sistema para conteo en tiempo real se aplicarán los conocimientos adquiridos en la carrera de Tecnologías de la Información (Sistemas e Informática), haciendo uso de herramientas, algoritmos tecnológicos OpenSource y recopilación de toda la información necesaria que permita desarrollar un sistema confiable y seguro a los propietarios de las unidades de transporte sobre el número de pasajeros de su bus para alcanzar de esta manera la fidelización de los mismos con la empresa de servicios Optimovilidad.

## **1.4 Objetivos**

### **Objetivo General**

Desarrollar un sistema para el conteo de personas del transporte público urbano en tiempo real utilizando cámaras de video, tecnologías y algoritmos OpenSource para la empresa Optimovilidad garantizando de esta manera seguridad y efectividad en los datos.

### **Objetivos Específicos**

- Realizar una revisión sistemática de literatura de métodos y algoritmos empleados en el reconocimiento y movimiento de objetos.
- Realizar una elicitación de requerimientos del sistema para conteo de pasajeros.
- Diseñar, desarrollar e integrar hardware y software para el sistema de conteo de pasajeros utilizando la metodología de desarrollo ágil Scrum.
- Implementar y realizar las pruebas funcionales y no funcionales del sistema para conteo.

## **CAPÍTULO 2**

### **ESTADO DEL ARTE**

#### **2.1 Tecnologías para conteo de personas**

El conteo de personas es un tema ampliamente estudiado y comercialmente explotado debido a que en muchas ocasiones es necesario considerar y determinar el número aproximado de personas que se tienen por día en ciertos lugares. (Rojas Cuevas, 2013) A continuación, se mencionan algunas tecnologías que son utilizadas en la actualidad para efectuar el conteo de personas.

##### **2.1.1 Video cámaras**

Ciertos autores mencionan como una opción para el conteo de personas (y localización) el uso de múltiples cámaras de vídeo, las mismas que mediante la extracción del tamaño y los patrones de movimiento de las personas que pasan, y el uso de histogramas de movimiento basados en imágenes de trama diferenciada que clasifican los movimientos detectados se realiza el conteo. Los resultados que son captados por las cámaras se unen para formar un vector de movimiento para cada individuo reconocido. Por el contrario, se propone una solución basada en una sola cámara montada en el techo, que identifica a las personas mediante la extracción en segundo plano de la imagen de la cámara.

El sistema alcanza una exactitud que bordea el 98,5%. La principal desventaja de un sistema basado en cámaras es que requiere una fuente de luz ambiental y recursos informáticos relativamente potentes para realizar el procesamiento de imágenes. (Hemangi & Nikhita, 2016)

### **2.1.2 Sensor Infrarrojo**

Las matrices IR<sup>1</sup> combinan una serie de sensores IR para formar una matriz de detectores. Como su nombre sugiere, las señales del sensor se proporcionan como una matriz, donde cada elemento de la misma corresponde a un sensor IR. Las matrices de infrarrojos proporcionan una solución rentable y también funcionan sin ninguna fuente de luz ambiental. Los algoritmos de reconocimiento de patrones permiten detectar personas que se mueven a través de la vista del sensor a una precisión del 95%. Las matrices IR son ampliamente utilizadas en sistemas comerciales. (Hemangi & Nikhita, 2016)

### **2.1.3 Sensores de movimiento infrarrojos**

En el sistema de conteo de personas basado en detectores de movimiento PIR<sup>2</sup>, para cada paso supervisado, se instalan tres sensores PIR a una distancia de 0,8 m. Los sensores están conectados a un coordinador mediante un enlace RF inalámbrico. Los sensores detectan eventos de movimiento y envían estos datos al coordinador. El coordinador infiere un recuento de personas para correlacionar el número, la fase y la diferencia de tiempo de los picos encontrados en la señal. El sistema logra una tasa del 100% para detectar la dirección del movimiento, y detecta con precisión el 89% del número de personas que pasan. Sin embargo, el costo y el esfuerzo de emplear múltiples nodos de sensores para cada punto de entrada/salida es una desventaja de la tecnología. (Hemangi & Nikhita, 2016)

## **2.2 Algoritmos de conteo de personas**

Para establecer el conteo de personas utilizando video, se ha dividido en dos extensas categorías: Conteo a través de una línea de detección en una determinada duración de tiempo o línea de interés (LOI) y estimación del total de números de personas de alguna

---

<sup>1</sup> IR: Infrared Sensor, sensor infrarrojo

<sup>2</sup> PIR: Passive Infrared, infrarrojo pasivo

región en un determinado tiempo o región de interés (ROI). (Cong, Chun Zhu, & Tang, 2009)

### **2.2.1 Conteo LOI**

El conteo LOI se puede desarrollar usando técnicas de conteo de rastreo de características, a continuación, se mencionan ciertos algoritmos empleados para este tipo de técnicas.

#### **A. Basado en características de trayectoria**

Consiste en una metodología que requiere abordar el problema de segmentar objetos en movimiento en videos de multitud de personas, utilizando una versión altamente paralelizada del rastreador KLT para procesar el video en un conjunto de trayectorias de características.

Este conjunto de trayectorias proporciona un sustrato para el análisis del movimiento, sin embargo, se presentan dificultades para su procesamiento por sus longitudes desiguales y su naturaleza fragmentada. (Rabaud, 2006)

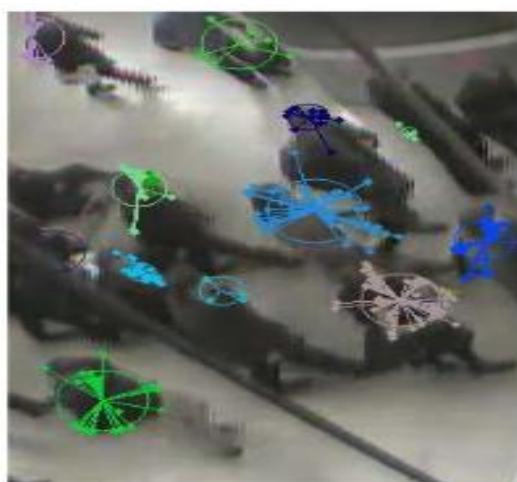
#### **B. Utilizando direcciones de movimiento**

Se basa en el análisis de área y color, para proponer un contador bidireccional de la circulación peatonal que pasa por una puerta. Las personas que pasan se cuentan aproximadamente con el área de personas proyectadas en una imagen capturada por una cámara de video, para reconocer la dirección de movimiento del peatón se realiza un análisis de histograma HSI de cada de patrón de persona reconocido. (Chen T.-h. , 2006)

### C. Basado en características de rastreo

Describe un algoritmo de agrupamiento bayesiano basado en datos que tiene como objetivo principal la detección de entidades individuales. Se rastrea funciones de imagen simples para agruparlas de manera probabilística en clústeres que representan entidades que se mueven independientemente. El nuevo enfoque de este algoritmo consiste en que la proximidad espacial y la coherencia de la trayectoria a través del espacio de la imagen, se utilizan como el único criterio probabilístico para la agrupación. (Brostow & Cipolla, 2006)

La figura 1, muestra el resultado del algoritmo de agrupamiento bayesiano propuesto.



**Figura 1** Agrupamiento bayesiano

Fuente: (Brostow & Cipolla, 2006)

### D. Basado en análisis y seguimiento de blob<sup>3</sup>

Este método solo obtiene el reflejo de la línea láser en los peatones utilizando un filtro IR, por lo que la adquisición de imágenes de objetos en movimiento es extremadamente simple y confiable. El número de peatones se obtiene utilizando

---

<sup>3</sup> Blob: Binary Large Object (Objeto binario grande)

análisis simple de blob y las direcciones de movimiento se obtienen de las pendientes de los blobs. Los experimentos muestran que este método es capaz de contar la cantidad de peatones con una precisión de aproximadamente del 95%. (Lee, Kim, & Yoon, 2008)

#### E. Usando líneas múltiples

Este sistema de conteo de personas en tiempo real, utiliza una cámara colocada en la parte superior que cuenta la cantidad de personas que entran y salen de un área observada. Se maneja un algoritmo basado en el análisis de una zona de imagen compuesta por un conjunto de líneas de recuento virtuales. (Barandiaran, Murguia, & Boto, 2008)

#### F. Modelado y sustracción de fondo basado en el conteo de personas

Consiste en implementar un modelado y sustracción de fondo Sigma-Delta para segmentar personas de la región que provee blobs como resultado con el componente conectado. Estos blobs son utilizados para rastrear y contar la entrada y salida de personas. (Kumar, Parashar, & Verma, 2012)

### **2.2.2 Conteo ROI**

El conteo ROI se puede desarrollar usando dos técnicas como se menciona a continuación con sus respectivos algoritmos:

#### **2.2.2.1 Técnicas basadas en detección**

Se enfoca en detectar personas individualmente y contar las mismas. Se utiliza varios de los procesos de seguimiento como: diferenciación de fondo; movimiento y apariencia en segmentación conjunta; método de reconocimiento de siluetas o formas y método de reconocimiento de objetos estándar. (Sridhar & Kowcika, 2015)

### A. Detección y seguimiento basado en la región cabeza/hombro

Mediante una cámara fija se utiliza un método de detección de la cabeza basada en la geometría del cuerpo para detectar y rastrear personas en movimiento. Este algoritmo consiste en encontrar la cabeza de las personas definiéndose como Rectángulo de Característica Personal (BFR). Los resultados experimentales manifiestan que el método propuesto puede alcanzar el 94% de exactitud con número de fusión de personas inferior a seis. (Hu, Zhou, & Zhou, 2011)

En la figura 2, se puede observar los resultados del método propuesto y la efectividad del mismo.



**Figura 2** Resultados de la detección

Fuente: (Hu, Zhou, & Zhou, 2011)

### B. Basado en la fusión de información de forma y movimiento

Consiste en un método directo de conteo por detección, basado en la fusión de la información espacial recibida de un algoritmo adaptado del Histograma de Gradientes Orientados (HOG) con información temporal mediante la explotación de características distintivas de movimientos de diferentes partes del cuerpo humano. (Patzold, Heras, & Sikora, 2010)



### C. Basado en combinación de nueva detección estática y detección dinámica

Es un método de conteo rápido y efectivo que se fundamenta en dividir el área de monitoreo primero en bloques, para luego reconocer a las personas en cada bloque y combinar la detección estática con la detección dinámica. En la detección estática se utiliza el algoritmo de covarianza para evitar el ajuste artificial del umbral, y mediante el uso de la detección dinámica se pueden excluir de manera efectiva interferencias de objetos no humanos. (Zhou, Xu, & Wang, 2012)

#### 2.2.2.1 Técnicas basadas en características y píxeles de regresión

Consiste en extraer las características como píxeles de primer plano y puntos de interés. Los vectores son formados con estas características y utilizan algoritmos de aprendizaje automático para disminuir el número de personas. Algunas características comunes son: bordes; coeficientes wavelet<sup>4</sup>; redes neuronales, regresión lineal, procesos de regresión gaussiano y clasificadores discretos. (Sridhar & Kowcika, 2015)

##### A. Basado en múltiples características locales

Enfoque que utiliza características locales para contar el número de personas en cada segmento de blob de primer plano, de tal manera que la estimación de multitud total sea la suma de los tamaños de grupo. Este algoritmo de conteo es empleado fácilmente para estimar la densidad de una multitud total de personas a lo largo de diferentes regiones de la escena, además puede ser utilizado en un entorno de cámaras múltiples. (Ryan, Denman, Fookes, & Sridharan, 2009)

##### B. Conteo de personas en grupos

El modelo tiene como primer paso la segmentación de primer plano y después de los diferentes blobs que se proyectan sobre la cabeza y el suelo plano. Las

---

<sup>4</sup> Wavelet: Corresponde a una oscilación similar a una onda con una amplitud que comienza en cero, aumenta y luego disminuye de nuevo a cero.

proyecciones posteriores son utilizadas para estimar el número de personas en un grupo y las estimaciones de conteo se combinan con la información de seguimiento para obtener una estimación de conteo suave. (Fehr, y otros, 2009)

#### C. Método robusto para contar personas

Este método cuenta el número de personas en espacios interiores complejos utilizando una sola cámara a través de cuatro módulos: pre procesamiento de imágenes; procesamiento morfológico; marcado de imágenes y módulo de conteo de personas, con el fin de dominar la información del interior para aumentar la eficiencia y la utilización de las instalaciones del edificio. (Ye Q. , 2010)

#### D. Basado en autoaprendizaje

El método se encuentra basado en autoaprendizaje que cuenta en entornos concurridos utilizando una sola cámara. La principal diferencia entre este método y los tradicionales es que se adopta blobs separados como la entrada del estimador del número de personas. Primero, los blobs se seleccionan según sus características después de la estimación de fondo y calibración mediante seguimiento. A continuación, cada blob seleccionado en la escena es entrenada para predecir el número de personas en el blob.

Finalmente, el estimador de número de personas se conforma combinando sub estimadores entrenados de acuerdo a una regla predefinida. (Ye & Zhong, 2007)

#### E. Basado en la detección de bordes

El algoritmo de conteo incluye tres pasos: movimiento de personas detectando, rastreando y contando. Para esto, se construye un Modelo de Borde de Primer Plano y Fondo a partir de seriales de marcos y recuperando el borde de primer plano, por lo que se obtiene el cuadro limitador de las personas en movimiento.

Además, el proceso de conteo propuesto se describe en detalle, y el fenómeno de combinación supera el problema de las personas que pasan juntas. (Yu, Chen, Sun, & Xie, 2008)

#### F. Recuento bidireccional automático de personas

Para obtener un conteo de personas más preciso, se utiliza una segmentación de dos etapas para extraer a cada persona de una multitud. Primero, una multitud se segmenta mediante la técnica de diferencia de trama seguida por el procesamiento morfológico y crecimiento de la región, tal y como se puede apreciar en la figura 3.

A continuación, se utiliza el método de etiquetado de componentes genera varios patrones de personas individuales de la multitud segmentada. Las características de la imagen como área, altura y ancho de cada patrón individual de personas. Finalmente, cada persona segmentada se rastrea hasta tocar la línea de base y luego se cuenta. (Chen, Chen, Wang, & Chen, 2012)



**Figura 3** Procesamiento de imagen del algoritmo

Fuente: (Chen, Chen, Wang, & Chen, 2012)

## **2.3 Sistemas inteligentes de transporte público**

El transporte público urbano tiene una gran influencia en el bienestar de los ciudadanos que hacen uso del mismo, por lo cual, los atascos de tráfico afectan directamente en las personas que necesariamente requieren de este servicio para su movilización.

La Secretaría de Movilidad del Distrito Metropolitano de Quito tiene como uno de sus objetivos, disponer de un sistema de última generación que permita la integración de los servicios de transporte de la ciudad y de esta manera gestionar el recaudo y localización de la flota. Además de generar información al usuario de la oferta de transporte público a través de medios tecnológicos de accesibilidad universal para lograr una adecuada operación del servicio. (Secretaría de Movilidad, 2017)

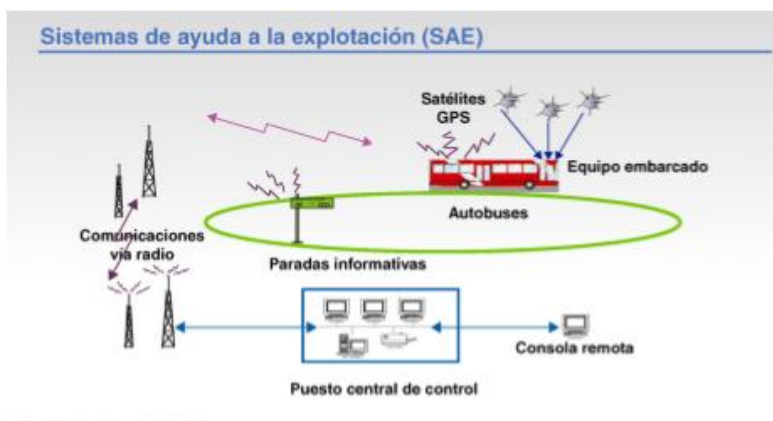
### **2.3.1 Sistemas de Ayuda a la Explotación (SAE)**

Los SAE son sistemas específicos de gestión basado en la utilización de tecnologías de información y comunicación para la gestión del tráfico. La aplicación de estos sistemas comienza en Europa en los años noventa, como una alternativa sostenible al problema creado por la creciente demanda de movilidad, especialmente en áreas urbanas ofreciendo una oportunidad para una movilidad razonable. En la figura 4 se puede observar de manera esquematizada el funcionamiento de un SAE. (Pablos, Pérez, & Montes, 2011)

El proceso de funcionamiento de un SAE consiste en: mediante el uso de un GPS se localiza de forma continua el autobús; el sistema por radiofrecuencia envía la posición del mismo al puesto central de operación. El puesto central de operación, es el encargado de realizar la regulación y explotación de toda la red de autobuses. (Pablos, Pérez, & Montes, 2011)

Por lo tanto, el funcionamiento básico del sistema consiste en los siguientes pasos:

- Localización
- Comunicación
- Regulación
- Información



**Figura 4** Sistema de ayuda a la explotación

**Fuente:** (Pablos, Pérez, & Montes, 2011)

### 2.3.2 Sistemas Integrados de Recaudo (SIR)

El sistema integrado de recaudo consiste en un sistema de pago electrónico de pasajes el cual persigue los siguientes objetivos: permitir un rápido y cómodo acceso de los usuarios al servicio de transporte; mejorar el tiempo de expendio de pasajes gracias a que la información se queda registrada en el sistema; otorgar seguridad al proceso de recaudo de pasajes y; acceder a una integración tarifaria entre distintos operadores de transporte, es decir, un sistema de caja común. (Pérez, 2007)

Actualmente, las unidades del transporte público de Quito utilizan equipos contadores, los mismos que permiten a los operadores realizar la gestión de recaudo de pasajes. Las empresas dedicadas a este negocio, utilizan circuitos electrónicos con sensores infrarrojos que permiten detectar entradas y salidas de personas.

Existen nuevas tecnologías para conteo, el presente proyecto plantea un sistema basado en cámaras de video que garanticen efectividad en el número de pasajeros que utilizan el transporte público.

### **2.3.2 Sistemas de Información al Usuario (SIU)**

Es una herramienta de operación del transporte público encargada de proporcionar, en tiempo real, información efectiva, de manera visual y auditiva a los pasajeros, tanto a bordo como en espera en las paradas. (indra, 2017)

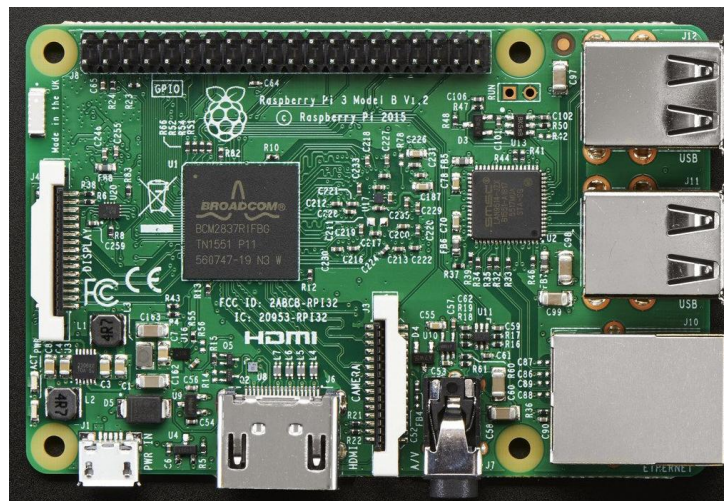
Se plantean soluciones para móviles y sitios web que proporcionen al usuario del transporte público información de las rutas de operación, horarios y frecuencias en tiempo real, logrando de esta manera la satisfacción en el pasajero y la mejora en el servicio de las flotas.

## **2.4 Raspberry Pi 3**

Según (Raspberry, 2017) Raspberry Pi es una computadora de bajo costo y su tamaño aproximado al de una tarjeta de crédito, el cual se conecta a un monitor de computadora o TV, y utiliza un teclado y un ratón estándar. Es un pequeño dispositivo que permite la programación en lenguajes como Scratch y Python, además permite explorar la informática y aprender a programar en estos idiomas adecuadamente.

Este dispositivo permite realizar todo lo de una computadora de escritorio, desde navegar por Internet y reproducir video de alta definición hasta el procesamiento de textos y juegos. Además, Raspberry Pi tiene la capacidad de interactuar con el mundo exterior y se ha utilizado en una amplia gama de proyectos de creación digital como: máquinas de música; detectores de padres; estaciones meteorológicas; cámaras de videovigilancia; entre otros.

La Fundación Raspberry Pi ha revelado su última innovación en el mercado (ver figura 6), una computadora Pi 3 habilitada para Wi-Fi con el doble de rendimiento que la generación anterior, manteniendo su costo, pero con hardware mucho más potente. El Raspberry Pi 3 funciona a 1.2 GHz, en comparación con el Pi 2 de 900MHz, además tiene un sistema de potencia actualizado, y los mismos cuatro puertos USB. Este dispositivo en lugar de utilizar un disco duro para su almacenamiento, emplea una tarjeta micro SD para esto soportando hasta 64 GB de información.



**Figura 5** Raspberry Pi 3

**Fuente:** (Raspberry, 2017)

Una característica adicional de la Pi 3 consiste en su conectividad inalámbrica incorporada, la nueva Raspberry Pi se encuentra claramente posicionada como un equipo de bajo costo para dispositivos de IoT (Internet de las cosas), o como la base flexible y de bajo costo de los nuevos tipos de dispositivos.

Las especificaciones técnicas de la Raspberry Pi 3 consiste en lo siguiente:

- CPU: cuatro núcleos y 64 bits.
- GPU: 400 MHz VideoCore IV multimedia
- Memoria: 1GB LPDDR2-90 SDRAM
- 4 puertos USB

- Salidas de video: HDMI, video compuesto PAL y NTSC
- Red: Ethernet 10 y LAN inalámbrica 802.11n
- Bluetooth: 4.1
- Fuente de alimentación: 5V a través de MicroUSB o GPIO
- Tamaño: 85.60 mm \* 56.5 mm
- Peso: 45g (1.6 oz)

## 2.5 Cámaras

Una cámara es un dispositivo con la finalidad de capturar imágenes proyectadas a través de un sensor convirtiéndolas en señales eléctricas, para transferir estas señales a un sistema electrónico.

Las cámaras utilizadas para efectuar el conteo de personas deben contar con un estándar alto de calidad y recursos analíticos de video configurables, pues muchas veces el funcionamiento de un algoritmo depende en gran medida de la calidad de la resolución de la imagen sobre la que se trabaja, debido a que facilita y acelera el proceso computacional de detección de los algoritmos. Al tener imágenes con una resolución aceptable, no existiría demasiado ruido en las mismas y minimizando de esta manera el porcentaje de error en el conteo. (Rivas, 2013)

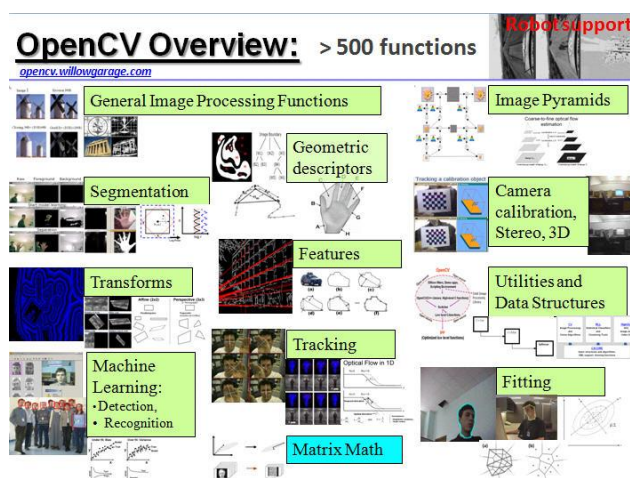
## 2.6 Software OpenCV

Open CV (Open Source Computer Vision Library) es una biblioteca de software de visión artificial y de aprendizaje automático. OpenCV fue construido para proporcionar una infraestructura común para aplicaciones de visión computarizada y para acelerar el uso de la percepción de la máquina en los productos comerciales. (Rodríguez, 2015)



La biblioteca cuenta con más de 2500 algoritmos optimizados, que incluye un amplio conjunto de clásicos y últimos algoritmos de visión artificial y de aprendizaje automático. (NVIDIA, 2017)

Los algoritmos pueden tener diferentes usos como: detectar y reconocer rostros; identificar objetos; clasificar acciones humanas en videos; buscar imágenes similares de una base de datos de imágenes; realizar un seguimiento de movimientos de cámara; rastrear objetos en movimiento; extraer modelos 3D de objetos; unir imágenes para producir una imagen de alta resolución de una escena completa; quitar los ojos rojos de las imágenes tomadas con flash; seguir los movimientos de los ojos; entre otras aplicaciones. (OpenCV, About, 2017)



**Figura 6 Funciones de OpenCV**

Fuente: (Rodríguez, 2015)

Junto con compañías bien establecidas como Google, Yahoo, Microsoft, Intel, IBM, Sony, Honda, Toyota que emplean la librería, hay muchos startups como Applied Minds, VideoSurf y Zeitera, que hacen uso extensivo de OpenCV. Entre los usos desarrollados de según (OpenCV, About, 2017) se tiene:

- Abarcar desde la costura de imágenes de streetview.
- Detectar intrusiones en video de vigilancia en Israel

- Supervisar equipos de minas en China.
- Ayudar a robots a navegar y recoger objetos en Willow Garage
- Detectar accidentes de ahogamiento en piscinas en Europa, España y Nueva York.
- Inspeccionando las etiquetas de los productos de las fábricas de todo el mundo para detectar rápidamente los rostros en Japón.

Entre las características principales de OpenCV, que la diferencia de Matlab para el procesamiento de imagen, en el sitio web de (Thakkar, 2012), se menciona en la siguiente tabla:

**Tabla 1**  
*Características de Matlab y OpenCV*

	<b>OpenCV</b>	<b>Matlab</b>
Lenguaje que fue desarrollado	Lenguaje C	Lenguaje Java
Velocidad	Lenguaje más rápido en tiempo de ejecución. Procesa 30 fotogramas por segundo	Mas demora en la ejecución. Se obtiene de 3 a 4 cuadros por segundo
Recursos necesarios	Requiere solo de 70MB de RAM para ser ejecutados en tiempo real.	Utiliza muchos recursos del sistema, se requiere más de un GB de memoria RAM para que pueda ser ejecutado a través de video.
Costo	Licencia de código abierto, es gratuito	Licencia comercial y de usuario único, valor aproximado de USD 2150

**CONTINÚA** 

Portabilidad	Windows, Linux, MacOS y cualquier dispositivo que ejecute C.	Windows, Linux y MacOS.
--------------	--	-------------------------

### 2.6.1 Módulos de OpenCV

OpenCV como librería se encuentra distribuida en una estructura modular, donde cada módulo tiene su importancia y función específica que cumplir para que la librería pueda compilar adecuadamente. Los módulos según (OpenCV, About, 2017) son detallados a continuación:

- **core:** Módulo básico de OpenCV, es un módulo compacto que define las estructuras de datos básicos, funciones básicas de procesamiento de imágenes y para los demás módulos.
- **imgproc:** Módulo de procesamiento de imágenes que contiene: filtrado de imágenes lineales y no lineales; transformaciones geométricas de imágenes; histogramas; conversión del espacio de color; entre otras.
- **video:** Es un módulo de análisis de video que incluye la estimación de movimiento, sustracción de fondo y algoritmos de seguimiento de objetos.
- **highgui:** Interfaz fácil de usar para la captura de video, códecs de imágenes y video, capacidades simples de interfaz de usuario.
- **calib3d:** Algoritmos básicos de geometría de vista múltiple, estimación de pose de objetos, elementos de reconstrucción en 3D y calibración de cámara única y estéreo.
- **objdetect:** Detección de objetos e instancias de clases predefinidas como: cara; ojos; tazas; personas; autos; entre otros.
- **features2d:** detectores de características sobresalientes y descriptores.

## 2.7 Lenguaje de programación

Un lenguaje de programación es considerado un programa de lenguaje especial, utilizado por programadores para desarrollar aplicaciones, programas software u otras instrucciones para ser ejecutados en los computadores.

Para el desarrollo del presente proyecto se utilizaron los lenguajes de programación open source Python y PHP, los cuales son descritos a continuación.

### 2.7.1 Python

Python es un lenguaje de programación interpretado, multiparadigma, de alto nivel con un tipado dinámico, dotado de una gestión automática de los recursos, de un alto grado de introspección y de un sistema de gestión de excepciones. (Chazallet, 2016)

Es libre y gratuito, funciona sobre todas las plataformas, apareció en 1990 y posee varias implementaciones, entre ellas CPython, Jython, IronPython y PyPy. (python, 2017)

Su licencia es la Python Software Foundation License, comparativamente cercana a la licencia BSD y compatible con la licencia GPL. La sintaxis que emplea es minimalista, explícita, clara, sencilla y muy cercana al lenguaje natural lo cual permite que un algoritmo se comprenda tras su primera lectura, es decir, la elaboración de un algoritmo compuesto por palabras, se declina de forma prácticamente natural, siendo esta una de la principal ventaja del lenguaje de programación. (Chazallet, 2016)

Python es un lenguaje de programación orientado a una filosofía con directrices muy claras, dando mucho espacio a que los desarrolladores no se encuentren impuestos a una única manera de hacer las cosas, por lo tanto, el desarrollador sabe que se encuentra realizando correctamente un algoritmo de alto nivel, el cual se procesará de la mejor forma posible, gestionando la memoria y los demás recursos correctamente mejorando la productividad considerablemente. (Chazallet, 2016)

A continuación, se enumeran las características principales del lenguaje de programación Python según (Herrera & Sánchez, 2013):

- Es un lenguaje de programación de alto nivel, es decir permite una rápida implementación de aplicaciones con muy poco código, además es ideal para prototipado rápido permitiendo el desarrollo de grandes aplicaciones.
- Python es un lenguaje interpretado por una máquina virtual, no compilado, por lo que es multiplataforma.
- Tiene estructuras de datos de alto nivel, como las listas, tuplas y diccionarios
- Permite la programación basada en objetos, programación funcional y la programación estructurada.
- Es un lenguaje dinámico, es decir se puede añadir nuevas funciones y clases a un objeto existente, inclusive en tiempo de ejecución.
- Python contiene librerías científicas maduras y con gran capacidad.
- Python al permitir procedimientos más rápidos y eficientes para desarrollo de algoritmos matemáticos, pueden programarse las partes más sensibles en C y realizar lo demás de la aplicación en Python
- Tiene un gran conjunto de librerías para extender su funcionalidad, como: acceso a base de datos; almacenamiento de datos; procesamiento de imágenes; sonido; entre otros.
- El código Python se escriben en archivos con extensión .py. Estos archivos se pueden agrupar en módulos, e importar los elementos de los módulos a otros archivos.
- El código es extraordinariamente sencillo de comprender.
- Actualmente, existe una comunidad grande de usuarios y debido a la filosofía abierta, los desarrolladores son mucho más aficionados a compartir su código con otros.

Python es un lenguaje de programación de propósito general, el cual no fue pensado a sus inicios para aplicaciones científicas, al continuar de los años y persiguiendo la

filosofía GNU, se han ido desarrollando módulos científicos, los cuales permiten realizar numerosas tareas de tratamiento de datos, visualización, aplicaciones científicas específicas y cálculo simbólico. Entre los módulos principales que posee Python se tiene a los siguientes:

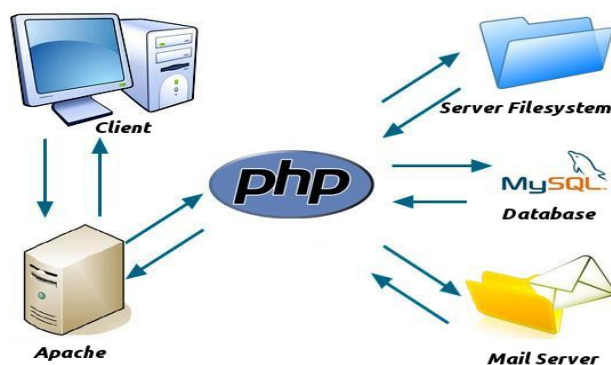
- Numpy; para la generación de tipos de datos científicos.
- Matplotlib; Gráficas en 2D y 3D.
- Scipy; Contiene funciones científicas de uso general.
- Mayavi; Visualización de datos tridimensionales.
- Opencv; Es una biblioteca open source de visión artificial cuyas librerías se han adaptado a Python.
- Meep; cálculo de diferencias finitas en el dominio del tiempo.

### **2.7.2 PHP**

PHP es un lenguaje de programación de código abierto que representa a un preprocesador de hipertexto. Es un potente lenguaje scripting por el lado del servidor, donde su función primordial es la de escribir páginas web dinámicamente.

El proceso paso a paso que de PHP consiste en: el cliente envía una solicitud HTTP/HTTPS al Servidor Apache; el motor PHP ejecuta esos comandos; el servidor envía la salida HTML al cliente; finalmente el sistema del cliente HTML se ejecuta en la aplicación del mismo y muestra el resultado.

En la figura 3, se muestra el proceso de ejecución del lenguaje de programación PHP



**Figura 7** Proceso de ejecución de PHP

**Fuente:** (Republic, 2017)

PHP puede ser integrado con una gran cantidad de bases de datos populares, tales como MySQL, PostgreSQL, Oracle, Sybase y Microsoft SQL Server.

Existen diversos lenguajes que permiten la programación del lado del servidor como ASP.NET o JSP, sin embargo, según, (Republic, 2017) existen varias ventajas para elegir PHP como principal. A continuación, se mencionan las mismas:

- Fácil de aprender

PHP es un lenguaje fácil de aprender y manejar. Para los programadores principiantes que empiezan en el desarrollo web, PHP es a menudo considerado como el mejor y preferible lenguaje de programación de opción.

- Código abierto

PHP es un lenguaje de programación de código abierto, es desarrollado y mantenido por una comunidad mundial de desarrolladores, los cuales hacen que su código fuente este libremente disponible para su descarga y uso. No existen costos asociados con el uso de PHP para proyectos individuales o comerciales, distribuido bajo una licencia que autoriza la modificación y redistribución.

- Portabilidad

PHP se ejecuta en distintas plataformas tales como: Microsoft Windows, Linux, Mac OS, entre otros, además es compatible con la mayoría de los servidores que se utilizan actualmente, como Apache, IIS, entre otros.

- Rendimiento rápido

Las líneas de código escritas en PHP generalmente se ejecutan de manera más rápida que los escritos en otros lenguajes de scripting como ASP.NET o JSP.

## 2.8 Base de datos MySQL

MySQL es un software que permite la gestión de bases de datos. En la actualidad, existen algunas formas diferentes de organizar datos en una base de datos, conocidos como modelos de bases de datos. El modelo relacional es uno de los más populares, y MySQL, al igual que PostgreSQL, Oracle, Microsoft SQL Server hacen uso de este modelo. En consecuencia, MySQL es conocido como un sistema de administración de bases de datos relacionales, o RDBMS.

En la figura 8 se muestra cómo una base de datos, el DBMS y el código de un sitio web interactúan entre sí.



**Figura 8** Interacción de una base de datos

Fuente: (Doyle, 2011)



MySQL tiene algunas ventajas para los desarrolladores web en comparación con algunos otros sistemas para la gestión de bases de datos:

- Es de código abierto, lo que representa que es gratis para su uso y modificación.
- Para trabajar con bases de datos, MySQL es un software fácil de usar, configurar y trabajar, es relativamente sencillo.
- Su funcionalidad con PHP es efectiva. A partir de la versión 5.3, PHP tiene un controlador MySQL nativo que se encuentra estrechamente relacionado con el motor PHP, por lo que es una buena opción para programadores de PHP.
- MySQL es considerado como un software de base de datos muy rápido. Esta velocidad es respaldada por la gran cantidad de pruebas comparativas realizadas.
- Existen varias interfaces de usuario convenientes para administrar un servidor MySQL.

La siguiente tabla comparativa, indica las debilidades y fortalezas de MySQL con respecto a otras bases de datos relacionales existentes en el mercado, motivo por el cual fue seleccionada para el desarrollo del presente proyecto.

**Tabla 2**  
*Rango de comparativa*

---

*	Malo
**	Aceptable
***	Bueno
****	Muy Bueno
*****	Excelente

---

**Tabla 3**  
*Comparativa de gestores de base de datos*

	<b>Seguridad</b>	<b>Velocidad</b>	<b>Escalabilidad</b>	<b>Cant. de información</b>	<b>Interfaz con el usuario</b>
Oracle	*****	***	***	**	****
Sybase	***	*****	*****	****	*****
Genexus	**	**	*****	**	*****
Visual FoxPro	*	****	***	*****	****
MySQL	*****	****	*****	****	*****

Fuente: (Lemus, García, & Salgado, 2016)

## 2.9 Metodología de desarrollo ágil SCRUM

Es una metodología que parte de la esencia del desarrollo ágil de software. Es un proceso de gestión y control que reduce la complejidad en el desarrollo para satisfacer las necesidades del cliente. Scrum es un marco de trabajo simple que promueve la colaboración de todos quienes conforman el equipo de trabajo para lograr desarrollar productos complejos. (Francia, 2017)

El ciclo de vida de la metodología Scrum es incremental e iterativa, y cada iteración es denominada como sprint con una duración aproximada de 2 y 4 semanas dando como resultado una versión del software con funcionalidades listas a ser utilizadas. En cada script nuevo, se va ajustando la funcionalidad construida y añadiendo nuevos requerimientos que aporten valor al proyecto.

### 2.8.1 Roles

El equipo de Scrum se focaliza en construir sistemas de calidad. A continuación, se detalla los roles que conforman este equipo:

- **Scrum master**

Es la persona que lidera al equipo, responsable de que los procesos de la metodología y los entregables se cumplan.

Es el encargado de gestionar la reducción de obstáculos en el proyecto.

Entre las actividades principales que realiza el Scrum Master se tiene:

- Asegurar que los objetivos y el alcance del producto sean entendidos por todo el equipo Scrum.
- Asistir al equipo Scrum a comprender los elementos de los productos.
- Coaching del equipo de desarrollo, ayudando a crear productos de alto valor.
- Eliminar impedimentos para el progreso del equipo de desarrollo.
- Ayudar a los empleados y partes interesadas a comprender el proceso de Scrum y desarrollo de nuevos productos.

- **Product owner**

Aquel que representa al cliente o interesados del software a desarrollar, es el encargado de la parte del negocio que entrega los requisitos y gestiona la prioridad de los mismos.

Entre sus actividades se encuentra:

- Expresar de manera clara los ítems del producto backlog.
- Ordenar cada ítem en el producto backlog para alcanzar los objetivos.
- Optimizar el valor del trabajo que realiza el grupo de desarrollo.
- Asegurar que la lista del producto backlog sea transparente, visible y claro.

- **Team**

Es el grupo de profesionales con habilidades necesarias que se encuentran involucrados en el desarrollo del proyecto, y llevan a cabo las historias de usuario comprometidas al inicio de cada sprint.

El Team Scrum tiene las siguientes características:

- Son auto organizados, el equipo busca la manera de convertir los atrasos de sus productos en incrementos potenciales.
- El equipo es multifuncional, posee las habilidades necesarias para incrementar la producción.
- Scrum no reconoce títulos ni subgrupos para los miembros del equipo.

### 2.8.1 Fases

La metodología se encuentra compuesta por diversas fases, como las que se mencionan a continuación:

**Tabla 4**  
*Fases de la metodología Scrum*

<b>Fase</b>	<b>Características</b>
Recogida de requisitos	<ul style="list-style-type: none"> <li>• Generar una lista de requisitos funcionales y no funcionales</li> <li>• Priorizar y proceder como un plan de proyecto entregado por el cliente o propietario del producto al equipo de desarrollo</li> </ul>
Gestión del backlog	<ul style="list-style-type: none"> <li>• Conjunto de tareas y funcionalidades a ejecutar</li> <li>• Cada requisito tiene aproximadamente un tiempo de desarrollo y una prioridad.</li> </ul>

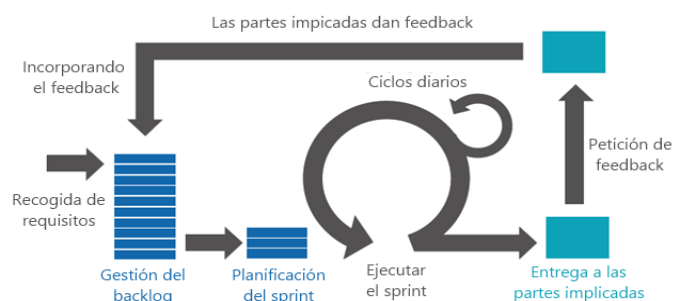
**CONTINÚA** 

---

	<ul style="list-style-type: none"><li>• El equipo de trabajo Scrum y el dueño del producto elaboran esta lista con la priorización de cada requisito</li><li>• Antes de iniciar la primera iteración, el cliente debe tener definido la meta del proyecto y la lista de requisitos</li></ul>
Planificación de la iteración	<ul style="list-style-type: none"><li>• Al inicio de cada sprint se lleva a cabo una reunión de planificación para analizar el trabajo a realizar.</li><li>• El sprint se encuentra dividido en dos partes como se menciona a continuación:<ul style="list-style-type: none"><li>○ Selección de requisitos Se consultan las dudas existentes y se priorizan los requisitos para que el equipo de desarrollo inicie por los más prioritarios.</li><li>○ Planificación de la iteración Elaborar una lista de tareas necesarias para desarrollar los requisitos.</li></ul></li></ul>
Ejecución de la iteración	<ul style="list-style-type: none"><li>• El proyecto es ejecutado en bloques cortos y fijos denominados sprint.</li><li>• Duración máxima de 4 semanas</li><li>• No debe sobrepasar este periodo de tiempo</li></ul>
Inspección y adaptación	<ul style="list-style-type: none"><li>• El último día de la iteración se realiza una reunión para la revisión de la misma.</li><li>• El equipo de desarrollo presenta al cliente los requisitos terminados.</li><li>• Se analiza la manera de trabajo del equipo de desarrollo durante la iteración para un proceso de mejora continua.</li></ul>

---

En la figura 4, se puede considerar de mejor manera el proceso de la metodología de desarrollo Scrum.



**Figura 9** Proceso Metodología Scrum

Fuente: (Leal, 2017)

Para el desarrollo del presente proyecto, se realizó un análisis previo de metodologías de desarrollo ágil siendo Scrum la más apropiada por los beneficios que presenta (ver tabla 5).

**Tabla 5**

*Características de RUP, XP y Scrum*

	<b>RUP</b>	<b>XP</b>	<b>Scrum</b>
<b>Fases del ciclo de vida</b>	Comienzo; Elaboración; Construcción; Transición	Exploración; Planeación; Iteración de lanzamiento; Producción.	Planeación; Puesta en marcha; Desarrollo; Lanzamiento.
<b>Principios y valores principales</b>	<ul style="list-style-type: none"> <li>• Atacar los riesgos importantes de manera temprana y continua.</li> <li>• Asegurar valor al cliente.</li> <li>• Mantener el software ejecutable.</li> </ul>	<ul style="list-style-type: none"> <li>• Proceso de planeación</li> <li>• Pequeños lanzamientos</li> <li>• Diseño simple</li> <li>• Pruebas</li> <li>• Refactorización</li> </ul>	<ul style="list-style-type: none"> <li>• Procesos y herramientas individuales e interactivos</li> <li>• Colaboración del cliente</li> </ul>

**CONTINÚA**



- 
- Adaptar el cambio oportuno en el proyecto.
  - Construir el sistema en componentes.
  - Trabajar juntos como un equipo.
  - Programación en parejas
  - Integración continua
  - Cliente en el sitio
  - Estándares de codificación
  - Responde al cambio en el siguiente sprint
  - Atención
  - Franqueza
  - Respeto
  - Valor agregado

---

**Fuente:** Traducido de (Cane, 2013)

## **CAPÍTULO III**

### **ANÁLISIS DEL ALGORITMO DE CONTEO Y REQUISITOS DEL SISTEMA**

#### **3.1 Análisis de métodos y algoritmos de conteo de personas.**

Para un mejor análisis de los algoritmos de conteo expuestos en el capítulo 2 del presente proyecto, la tabla 6 y 7 que se muestran a continuación describen de manera simplificada las características principales, número estimado de personas a conteo y el porcentaje de error de cada uno de los métodos y algoritmos planteados.

El conteo LOI consiste en estimar el número de personas que atraviesan una línea de detección o línea de interés de conteo en el video, para lo cual se establecen los métodos y algoritmos mencionados en la siguiente tabla:



**Tabla 6**  
Métodos y algoritmos de la categoría LOI

	Nombre	Problema a solucionar	Característica	N° personas	% de error
	Basado en características de trayectoria	Segmentar objetos en multitud de personas	Utiliza una versión de rastreador KTL	Multitudes	Errores de procesamiento
<b>C</b>	Utilizando direcciones de movimiento	Bi direccionamiento de personas por una puerta	Análisis de histograma HSI	2 a 3 personas	-
<b>O</b>	Basado en características de rastreo	Detección de entidades individuales	Algoritmo de agrupamiento bayesiano	Multitudes, animales	Detección falsa del 22%
<b>N</b>	Basado en análisis y seguimiento de blob	Adquisición de imágenes de objetos en movimiento	Análisis simple de blob y direcciones de movimiento	Peatones	Error del 5%
<b>T</b>	Usando líneas múltiples	Personas que entran y salen de un área observada	Análisis de una zona compuesta por líneas de recuento virtuales	Pocos peatones	5% de error
<b>E</b>	Modelado y sustracción de fondo basado en el conteo de personas	Contar la entrada y salidas de personas	Modelado y sustracción de fondo Sigma-Delta	Pocos peatones	Error del 1%

Un conteo ROI estima el número de personas de un área de interés de una imagen en particular, esta región es definida por el usuario para establecer el conteo, para su efecto se mencionan en la tabla a continuación:

**Tabla 7**  
Métodos y algoritmos de la categoría ROI

	<b>Nombre</b>	<b>Problema a solucionar</b>	<b>Característica principal</b>	<b>Número personas</b>	<b>% de error</b>
<b>C O N T E O R O I</b>	Detección y seguimiento basado en la región cabeza/hombro	Detectar y rastrear personas en movimiento	Método de detección de la cabeza basada en la geometría del cuerpo	Inferior a 6 personas	6% de error
	Basado en la fusión de información de forma y movimiento	Detección basada en información temporal	Algoritmo adaptado del HGO	Multitudes	-
	Basado en combinación de nueva detección estática y dinámica	Excluir interferencia de objetos no humanos	Algoritmo de covarianza.	Multitudes	-
	Basado en múltiples características locales	Entornos con cámaras múltiples	Contar el número de personas en cada segmento de blob	Multitudes	1,22% de error
	Conteo de personas en grupos	Estimar el número de personas en grupos	Estimación de conteo suave	Grupos	-
	Método robusto para contar personas	Conteo en espacios interiores complejos	Utiliza 4 módulos de procesamiento	Espacios pequeños	Conteo preciso

**CONTINÚA** 

Basado en autoaprendizaje	Conteo de entorno concurridos	Cada blob es entrenado para estimar el número de personas	Multitudes	-
Basado en la detección de bordes	Personas que pasan juntas	Modelo de borde de primer plano y fondo	Multitudes	-
Recuento bidireccional automático de personas	Extraer personas de multitudes	Segmentación de dos etapas	Multitudes	10% de error

Para el desarrollo del presente proyecto se descartaron los algoritmos y métodos desarrollados para multitudes debido a que el número de personas aproximado que suben por la puerta de ingreso de una unidad de transporte público es de 1 a 2 como máximo. Por tal motivo se realizó una clasificación de todos los algoritmos con esta característica, como puede observarse en la siguiente tabla:

**Tabla 8**  
*Comparativa de algoritmos y métodos de conteo*

<b>Métodos y algoritmos</b>	<b>Porcentaje de error</b>
Utilizando direcciones de movimiento	Sin registro
Basado en análisis y seguimiento de blob	5%
Usando líneas múltiples	5%
Modelado y sustracción de fondo basado en el conteo de personas	3%
Detección y seguimiento basado en la región cabeza/hombro	6%

Para la selección del algoritmo adecuado se consideró el porcentaje de error que alcanza en el conteo, donde el método de modelado y sustracción de fondo con un porcentaje menor, es estimado como el más adecuado para el desarrollo del proyecto.

### **3.2 Especificación de los requerimientos.**

Después de las diversas reuniones con los involucrados en el desarrollo del sistema para conteo de pasajeros en tiempo real se establecieron los diferentes requisitos funcionales y no funcionales.

### 3.2.1 Requisitos funcionales

Los requisitos funcionales conforman todas las funcionalidades que se requiere para el sistema web propuesto, a continuación, en la tabla 9 se enumeran los mismos:

**Tabla 9**  
*Requisitos funcionales*

<b>Código</b>	<b>Descripción</b>
<b>SW01</b>	El sistema permitirá al socio de la unidad ingresar al sistema previa su autenticación.
<b>SW02</b>	El sistema para su inicio de sesión al socio asignará la contraseña de 00000.
<b>SW03</b>	El sistema manejará dos perfiles de usuario: Administrador y Socio.
<b>SW04</b>	El sistema no permitirá al socio visualizar información de otras unidades.
<b>SW05</b>	El sistema permitirá al socio generar reportes con fechas personalizadas del número de pasajeros que reporto su unidad.
<b>SW08</b>	El sistema no permitirá al socio crear nuevos socios y unidades.
<b>SW09</b>	El algoritmo de conteo de pasajeros se ejecutará almacenando en la base de datos el número de entradas y salidas de personas.
<b>SW10</b>	El sistema permitirá al administrador crear nuevos socios con un usuario y contraseña para su autenticación.
<b>SW11</b>	El sistema permitirá al administrador crear nuevas unidades y asignará la unidad creada a un socio registrado.
<b>SW12</b>	El sistema manejará módulos y perfiles de usuario.
<b>SW13</b>	El sistema le permitirá al socio cambiar su clave.

Nota: Código utilizado es <<SW>> Requisito Funcional del sistema.

### 3.2.1 Requisitos no funcionales

Los requisitos no funcionales, permiten la operatividad oportuna y adecuada del sistema. La tabla 10 indica los requisitos no funcionales del proyecto propuesto:

**Tabla 10**  
*Requisitos no funcionales*

<b>Código</b>	<b>Descripción</b>
<b>NSW01</b>	El sistema web será compatible para cualquier navegador web
<b>NSW02</b>	El sistema web deberá estar disponible siempre.
<b>NSW03</b>	El sistema web deberá ser desarrollado bajo software libre
<b>NSW04</b>	El sistema web deberá ser documentando en lenguaje español
<b>NSW05</b>	El sistema web utilizará protocolos de autenticación de usuarios.
<b>NSW06</b>	El sistema web será intuitiva y fácil para el uso del usuario.
<b>NSW06</b>	El usuario necesitará acceso a internet para acceder al sistema.

Nota: Código utilizado es <<NSW>> Requisito No Funcional del sistema.

### 3.3 Descripción de los requerimientos

Posteriormente, se analizaron todos los requerimientos para proceder a generar las historias de usuario de manera detallada, según lo expuesto por la metodología de desarrollo Scrum.

El sistema tiene por objetivo permitir a los socios de las unidades previa su autenticación y permisos visualizar el conteo de su unidad en tiempo real, además con permisos de administrador, permitirá gestionar socios y unidades.

**Tabla 11**  
*Historia de Usuario HU-01*

<b>Historia de Usuario</b>	
<b>Número:</b> HU-01	<b>Usuario:</b> Administrador, Socio
<b>Nombre historia:</b> Inicio de sesión al sistema	
<b>Prioridad en negocio:</b> Alta	<b>Riesgo en desarrollo:</b> Medio
<b>Puntos estimados:</b>	
<b>Programador responsable:</b> Equipo de trabajo	
<b>Descripción:</b> Como usuario Administrador se requiere gestionar los socios, unidades, reportes y la visualización del conteo. Como usuario Socio se requiere únicamente la visualización del conteo en tiempo real.	
<b>Validación:</b>	
<ul style="list-style-type: none"><li>• Se permitirá el acceso únicamente a usuarios registrados en el sistema como Administrador y Socio.</li><li>• El usuario accede al sistema una vez que su usuario y contraseña sean validados a los registrados.</li><li>• El usuario Socio iniciará sesión con la contraseña que por defecto será creada como 00000.</li></ul>	



**Tabla 12**  
*Historia de Usuario HU-02*

---

**Historia de Usuario**

**Número:** HU - 02      **Usuario:** Administrador

**Nombre historia:** Registro de usuarios

**Prioridad en negocio:** Alta      **Riesgo en desarrollo:** Baja

**Puntos estimados:**

**Programador responsable:** Equipo de trabajo

**Descripción:** Como usuario Administrador se necesita registrar nuevos usuarios con su respectivo perfil, los cuales podrán visualizar el conteo de personas en tiempo real.

La información que se requiere almacenar es:

- Nombres y apellidos.
- Cédula de identidad
- Correo electrónico
- Nombre de usuario
- Contraseña
- Perfil: Socio o Administrador.

**Validación:**

- El nuevo usuario será registrado exitosamente.
- No se realizará el registro del nuevo usuario si existen campos vacíos.
- El sistema web validará la información registrada.

**Tabla 13**  
*Historia de Usuario HU-03*

---

**Historia de Usuario**

**Número:** HU-03                      **Usuario:** Administrador

**Nombre historia:** Registrar unidades

**Prioridad en negocio:** Alta                      **Riesgo en desarrollo:** Baja

**Puntos estimados:**

**Programador responsable:** Equipo de trabajo

**Descripción:** Como usuario Administrador se necesita registrar nuevas unidades para que sean asignados a sus socios, para que sea posible la visualización del conteo.

La información que se requiere almacenar es:

- Número de la unidad
- Placa
- Modelo
- Fecha de registro
- Socio

**Validación:**

- La nueva unidad será registrada exitosamente.
- No se realizará el registro de la nueva unidad si existen campos vacíos.
- El sistema web validará la información registrada.

**Tabla 14**  
*Historia de Usuario HU-04*

<b>Historia de Usuario</b>	
<b>Número:</b> HU-04	<b>Usuario:</b> Administrador
<b>Nombre historia:</b> Conteo de personas	
<b>Prioridad en negocio:</b> Alta	<b>Riesgo en desarrollo:</b> Alta
<b>Puntos estimados:</b>	
<b>Programador responsable:</b> Equipo de trabajo	
<b>Descripción:</b> Capturado el video de la cámara de video se procesa el algoritmo de conteo desarrollado, donde se empieza a contabilizar entradas y salidas de personas enviando los datos a la base de datos.	
<b>Validación:</b>	
<ul style="list-style-type: none"> <li>• La cámara web debe procesar video a la Raspberry.</li> <li>• El algoritmo de conteo debe encontrarse programado en la Raspberry para procesar el video capturado, y enviar los datos de entradas y salidas.</li> </ul>	

**Tabla 15**  
*Historia de Usuario HU-05*

<b>Historia de Usuario</b>	
<b>Número:</b> HU-05	<b>Usuario:</b> Administrador, Socio
<b>Nombre historia:</b> Visualización de reportes	
<b>Prioridad en negocio:</b> Media	<b>Riesgo en desarrollo:</b> Media
<b>Puntos estimados:</b>	
<b>Programador responsable:</b> Equipo de trabajo	
<b>Descripción:</b> Se debe visualizar en el sistema web un reporte del conteo realizado en la unidad.	
<b>Validación:</b>	
<ul style="list-style-type: none"> <li>• El usuario debe haber iniciado sesión.</li> </ul>	

**CONTINÚA** 

- Se debe presentar una opción para la opción de reportes.
- El usuario deberá seleccionar un rango de fechas para generar el reporte.

### 3.4 Especificación y descripción de roles del sistema.

En la tabla 16 se realiza una breve descripción de los roles del sistema de conteo web.

**Tabla 16**

*Roles del sistema web*

<b>Roles</b>	<b>Descripción</b>
<b>Administrador</b>	<ul style="list-style-type: none"> <li>• Control total de los módulos del sistema web</li> <li>• Asignación de roles</li> </ul>
<b>Socio</b>	<ul style="list-style-type: none"> <li>• Generar reportes personalizados.</li> </ul>

### 3.5 Personal involucrado

El equipo de personas que desarrollarán el proyecto de titulación propuesto se encuentra formado por profesionales de la empresa Optimovilidad y del Departamento de Ciencias de la Computación. La tabla detalla los roles que cumplirán cada miembro del equipo de trabajo.

**Tabla 17**  
*Personal involucrado*

<b>Rol</b>	<b>Encargado</b>	<b>Área</b>
<b>Product Owner</b>	Ing. Édison Yáñez	Optimovilidad – Gerente General
<b>SCRUM Master</b>	Ing. Pablo Parra	Director del proyecto de titulación
<b>Equipo de trabajo</b>	Srta. Paola Cueva	Desarrolladora Documentadora

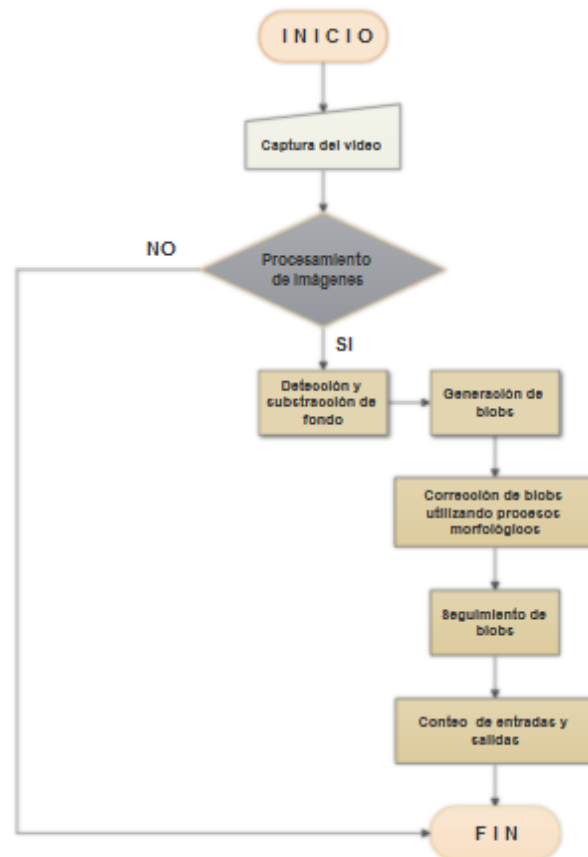
## CAPÍTULO IV

### DISEÑO, DESARROLLO E IMPLEMENTACIÓN DEL SISTEMA DE CONTEO

#### 4.1. Desarrollo del algoritmo de conteo

El algoritmo de modelado y sustracción de fondo seleccionado para el desarrollo del presente proyecto permitirá realizar el conteo de ingresos y salidas de los pasajeros en una unidad de transporte utilizando una cámara, para determinar de esta manera el número de pasajeros total que hacen uso del servicio de transporte.

##### 4.1.1 Diagrama de flujo del algoritmo de conteo



**Figura 10** Diagrama de flujo del algoritmo de conteo

La figura 10 muestra el proceso de conteo a partir de la captura de video, procesamiento de imágenes hasta el conteo de entradas y salidas de personas.

Para el desarrollo del algoritmo se utilizó la biblioteca libre OpenCV, la cual fue seleccionada por su compatibilidad con diversos sistemas operativos, además su consumo en cuanto a recursos computacionales es muy eficiente y no requiere de mucha memoria para su procesamiento, entre otros. En cuanto a las funciones para visión se dispone de aproximadamente 500 funciones que permiten de manera óptima realizar la detección, seguimiento y conteo de personas.

A continuación, se explica de manera detallada el desarrollo del algoritmo en el lenguaje de programación Python, el cual fue seleccionado por ser de código abierto además poseer módulos científicos y su fácil integración con la librería OpenCV para la visualización de imágenes y videos.

#### **4.1.2 Captura del video**

La captura de video consiste en una secuencia de frames o imágenes. La imagen capturada requiere ser procesada para identificar aspectos importantes en el conteo.

*VideoCapture('Direccionamiento donde se encuentra el video almacenado');* en los paréntesis se coloca la dirección y el nombre del video que se requiere realizar el conteo.

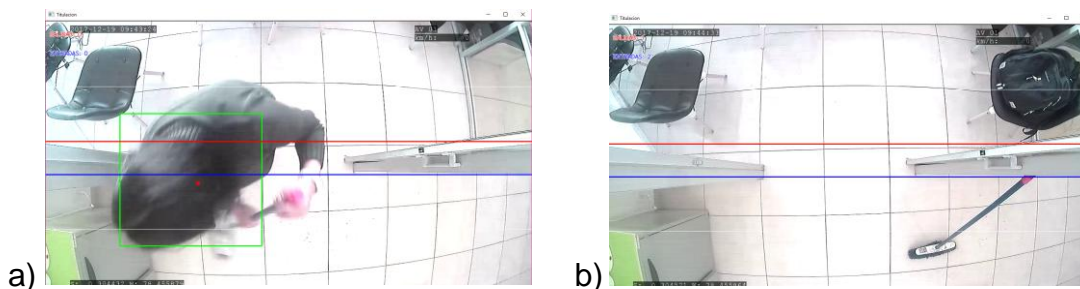
Para la captura del video en tiempo real se utiliza la función:

*VideoCapture(0);* donde 0 representa al dispositivo cámara web, el cual deberá estar conectado a una computadora mediante cable USB.

Finalmente, para la captura mediante streaming de cualquier video con una dirección IP se utiliza la función:

`VideoCapture('user:password/http://DIRECCIÓN_IP:8080/video');` donde DIRECCION\_IP corresponde a la IP que se desea obtener el video, y en el caso de requerir autenticación se adiciona el usuario y la contraseña.

En la figura 11 se puede visualizar un ejemplo de captura de video.



**Figura 11** Captura de frames a) con personas. b) sin personas.

Una vez realizada la captura del video se procede a realizar la detección y sustracción de fondo de cada frame o imagen de la secuencia del video.

#### 4.1.3 Detección y sustracción de fondo

La sustracción del fondo permite identificar el primer plano y el fondo de una imagen. Un fondo es considerado como algo constante en una serie de imágenes, es decir que permanezca estática, mientras que el primer plano es todo aquello que cambia y se mueve.

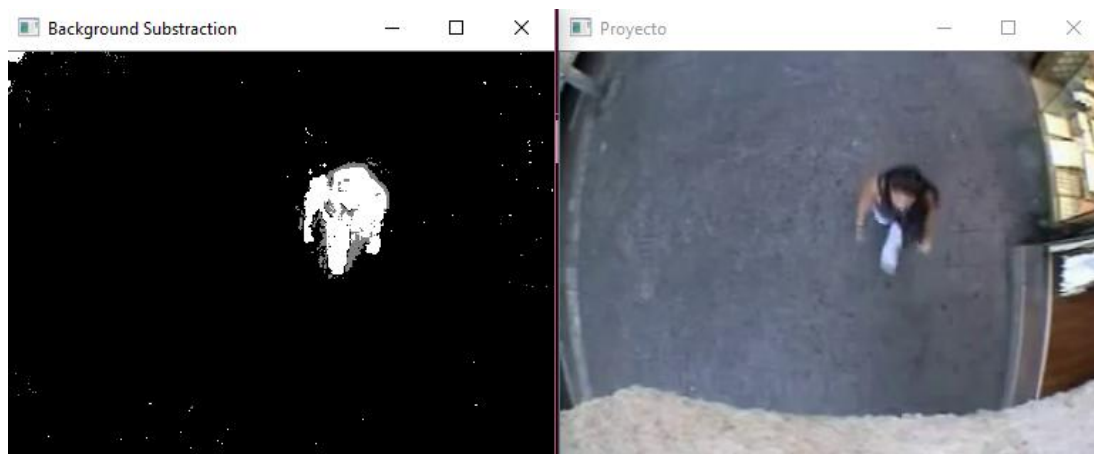
OpenCV ha implementado tres funciones para realizar la sustracción de fondo, para el desarrollo del proyecto se utilizó la función **BackgroundSubtractorMOG2**, el cual es un algoritmo de segmentación de fondo basado en la combinación gaussiana. La característica importante del algoritmo consiste en que selecciona el número apropiado de distribución gaussiana para cada pixel, proporcionando una mejor adaptabilidad a escenas variables por el cambio de iluminación, entre otros. (OpenCV, 2015)

Para su uso es necesario crear un objeto de fondo con la función: **`cv2.createBackgroundSubtractorMOG2(detectShadows = True);`** donde



detectShadows por defecto es True y representa que las sombras serán detectadas y marcadas en color gris.

La figura 12, a continuación, muestra el proceso de sustracción de fondo realizada al video.



**Figura 12** Sustracción de fondo en OpenCV

#### 4.1.4 Generación de blobs

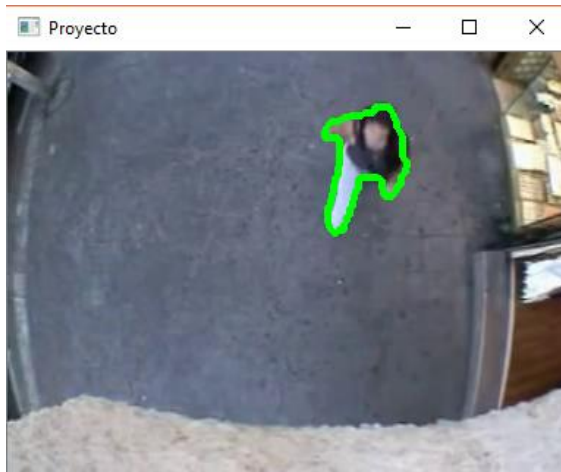
Para generar los blobs es necesario en primer lugar detectar los contornos de los frames utilizando la función **findContours** de OpenCV, para ejecutar esta función es importante que el objeto se encuentre en blanco y el fondo en negro. La función es sustancial para este proyecto pues es útil en el análisis de formas y detección de objetos, en este caso para personas.

La función final para efectuar la detección de contornos consiste en la siguiente: `cv2.findContours(mask,cv2.RETR_EXTERNAL,cv2.CHAIN_APPROX_SIMPLE)`; donde

- Mask; contiene el video con la función de sustracción de fondo.

- `cv2.RETR_EXTERNAL`; significa que únicamente se detectarán los contornos externos, es decir, los contornos dentro de los contornos no serán considerados.
- `cv2.CHAIN_APPROX_SIMPLE`; es el algoritmo utilizado para realizar el contorno.

En la figura 13 se puede visualizar el uso de la función para la detección de contornos en un video, se puede apreciar que se dibuja aproximadamente el contorno de la persona detectada.



**Figura 13** Detección de contornos

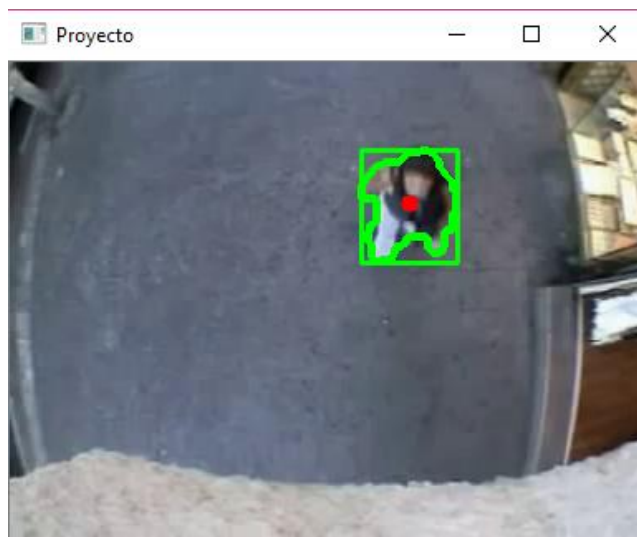
Una vez finalizada la detección de contornos, se procede a clasificar los mismos como personas y de esta manera establecer los blobs. Para lo cual se realiza lo siguiente:

- Definir un área mínima.
- Encontrar contornos.
- Para cada contorno obtener el área
- Configurar el umbral, este valor no es universal por lo que depende de la transmisión de video y requiere ser probada con valores diferentes para su

adecuado funcionamiento. OpenCV aplica la técnica de umbralización<sup>5</sup> con la función ***threshold***.

Para obtener la detección de blob que muestra la figura 14 se utilizó la función: `cv2.threshold(fgmask,200,255,cv2.THRESH_BINARY)`; donde

- `fgmask`: frame aplicado la función de sustracción de fondo.
- `200`: valor del umbral utilizado, para ejecutar la operación. Este valor es variable y depende mucho del video capturado, por lo cual no se puede establecer un valor universal.
- `250`: máximo valor utilizado con las operaciones de umbral.
- `THRESH_BINARY`: operación utilizada.



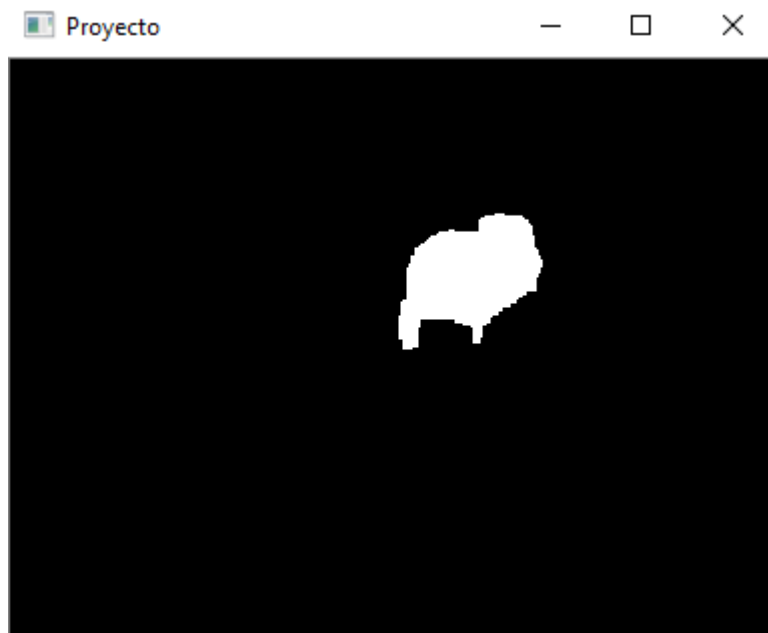
**Figura 14** Generación de blobs

#### 4.1.5 Corrección de blobs utilizando procesos morfológicos

Estos procesos son utilizados para eliminar el ruido de las imágenes captadas logrando de esta manera exaltar la silueta de la persona como se puede observar en la figura 15, además permite eliminar las sombras de color gris generadas.

---

<sup>5</sup> Umbralización: método de segmentación para separar las regiones de la imagen a analizar



**Figura 15** Procesos morfológicos

Para lograr esto se utiliza las funciones:

- Erosión

Esta operación calcula el mínimo local sobre el área del kernel, a medida que este se escanea sobre la imagen, el valor del píxel mínimo superpuesto por B se calcula y reemplaza la imagen debajo del punto de anclaje con ese valor mínimo.

- Dilatación

La operación de dilatación hace que el objeto en blanco sea más grande, es decir, consiste en convolucionar una imagen A con el kernel B, a medida que este se escanea sobre la imagen, es calculado el valor máximo de píxeles superpuesto por B. Esta operación de maximización hace que las regiones brillantes de la imagen crezcan por ende aumenta su tamaño.

- Apertura

Se obtiene mediante la erosión de una imagen seguida de una dilatación, y es usada para eliminar objetos pequeños, por ejemplo, objetos brillantes en un plano oscuro. Para el proyecto se utilizó la función:

`cv2.morphologyEx(imBin2, cv2.MORPH_OPEN, kernelOp)` donde

*imBin2*: imagen a ser procesada

*cv2.MORPH\_OPEN*: operación de apertura

*kernelOp*: elemento estructurado.

- Cierre

Función inversa a la apertura. Se consigue mediante la dilatación de una imagen seguida de una erosión, y es empleada para eliminar agujeros pequeños. Para el proyecto se utilizó la función:

`cv2.morphologyEx(mask2, cv2.MORPH_CLOSE, kernelCl)` donde

*mask2*: imagen a ser procesada

*cv2.MORPH\_CLOSE*: operación de cierre.

*kernelCl*: elemento estructurado.

#### 4.1.6 Seguimiento de blobs

Adicionalmente, es necesario realizar el seguimiento al blob detectado para conocer el direccionamiento del mismo, la dirección de interés para establecer si el conteo es una entrada o una salida

El proceso utilizado consiste en el siguiente: en el primer fotograma se detecta el objeto para proceder a darle una identificación y almacenar en memoria la posición inicial en la imagen. A continuación, se necesita realizar el seguimiento de la persona mediante el ID asignado por primera vez, para lo cual se debe hacer coincidir el contorno de la persona identificada con los cuadros siguientes, además de seguir almacenando en memoria las otras coordenadas.

Finalmente, una vez que la persona detectada atraviesa un límite asignado en la imagen se procede a evaluar todas las posiciones almacenadas para determinar si se mueve hacia arriba o abajo.

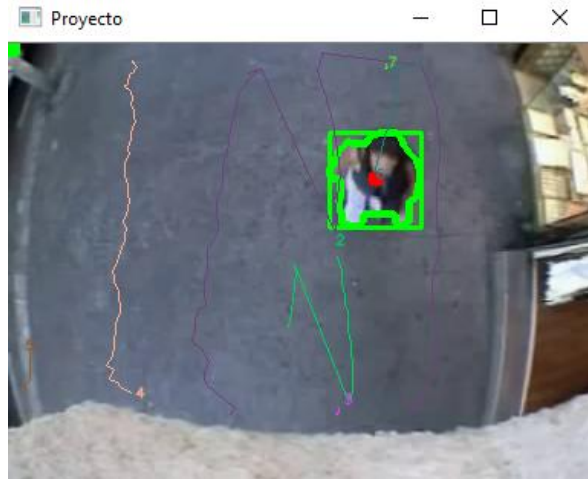
Para manejar el ID y el almacenamiento de las coordenadas de un objeto que fue detectado se utilizó una clase Persona que permitirá el almacenamiento en memoria del rastreo de posición de coordenadas.

Cuando el objeto se encuentra cerca de uno que ya fue detectado con anterioridad se utiliza el método:

```
def modificarCoords(self, xn, yn):  
    self.age = 0  
    self.tracks.append([self.x,self.y])  
    self.x = xn  
    self.y = yn
```

donde xn y yn corresponde a los parámetros que recibe las coordenadas del objeto detectado almacenándolo en un array de coordenadas y se pretende unir el contorno detectado con la persona previamente revelada.

Como se puede observar en la figura 16, la persona es detectada y además se dibuja el seguimiento realizado a la misma.



**Figura 16** Seguimiento de blobs

#### 4.1.7 Conteo de entradas y salidas

Finalmente, al realizar el seguimiento de las personas detectadas se procede a ejecutar el conteo de las mismas, para lo cual se utilizan dos líneas límites que permitirán determinar el sentido de movimiento del objeto detectado, el cual puede ser subida o bajada. Además, mediante los métodos *cont\_entrada(a, b)* y *cont\_salida(a,b)* donde a y b reciben las dos líneas creadas denominadas arriba y abajo, el método evalúa si el objeto cruzó la línea de arriba y retorna *verdadero* incrementando el contador de salidas.

En el caso de que el método cruce la línea de abajo se retorna *verdadero* y el contador de entradas incrementa. De esta manera se realiza el conteo de las personas detectadas que entran y salen de la unidad.

Método para el conteo de salidas realizadas:

```
def cont_salidas(self, abajo, arriba):
    if len(self.tracks) >= 2:
        if self.state == '0':
            #Validación que determina si cruzó la línea de arriba para contabilizar las salidas.
            if self.tracks[-1][1] < arriba and self.tracks[-2][1] >= arriba:
```

```

        state = '1'
        self.dir = 'arriba'
        return True
    else:
        return False
else:
    return False

```

Método para el conteo de entradas realizadas:

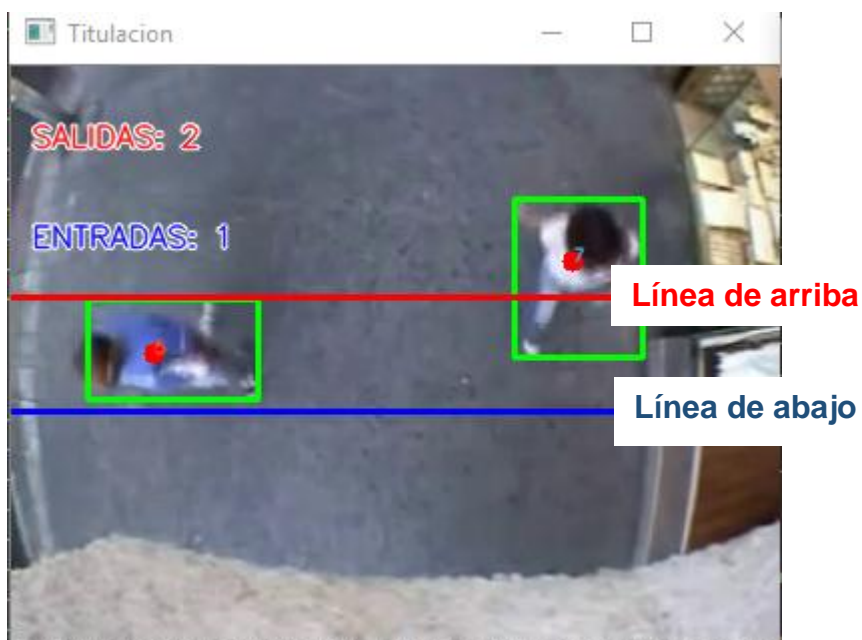
```

def cont_entradas(self,abajo,arriba):
    if len(self.tracks) >= 2:
        if self.state == '0':
            #Validación que determina si cruzó la línea de abajo para registrar las entradas.
            if self.tracks[-1][1] < abajo and self.tracks[-2][1] >= abajo:
                state = '1'
                self.dir = abajo
                return True
            else:
                return False
        else:
            return False

```

En la figura 17 se puede visualizar las líneas para el conteo, además, el número acumulado de entradas y salidas realizadas por las personas detectadas en el video.





**Figura 17** Conteo de entradas y salidas

Finalmente, para enviar los datos a la base de datos y los mismo puedan ser visualizados en el sitio web por el socio se utiliza el siguiente código de programación en Python:

```
userData = {"idUnidad": idUnidad, "entrada":1,"salida":0,"fecha": time.strftime("%y-
%m-%d %H:%M:%S")}
resp = requests.post('http://localhost/conteo/conteoFinal.php', params=userData)
```

Donde:

userData contiene toda la información a ser enviada y almacenada en la base de datos.

conteoFinal.php tiene el código SQL que realiza el almacenamiento de la información.

La figura 18, muestra el almacenamiento de los datos en la base de datos MySQL.

		ID_CONTEO	ID_UNIDAD	ENTRADAS	SALIDAS	FECHA
<input type="checkbox"/>	 Editar  Copiar  Borrar	28	1	0	1	2018-01-17 08:56:33
<input type="checkbox"/>	 Editar  Copiar  Borrar	29	1	0	1	2018-01-17 08:56:34
<input type="checkbox"/>	 Editar  Copiar  Borrar	30	1	1	0	2018-01-17 08:56:34
<input type="checkbox"/>	 Editar  Copiar  Borrar	31	1	0	1	2018-01-17 08:56:36
<input type="checkbox"/>	 Editar  Copiar  Borrar	32	1	0	1	2018-01-17 08:57:13
<input type="checkbox"/>	 Editar  Copiar  Borrar	33	1	0	1	2018-01-17 08:57:44
<input type="checkbox"/>	 Editar  Copiar  Borrar	34	1	0	1	2018-01-17 08:57:45
<input type="checkbox"/>	 Editar  Copiar  Borrar	35	1	1	0	2018-01-17 08:57:47
<input type="checkbox"/>	 Editar  Copiar  Borrar	36	1	0	1	2018-01-17 08:57:49
<input type="checkbox"/>	 Editar  Copiar  Borrar	37	1	0	1	2018-01-17 08:57:50
<input type="checkbox"/>	 Editar  Copiar  Borrar	38	1	0	1	2018-01-17 08:57:50
<input type="checkbox"/>	 Editar  Copiar  Borrar	39	1	0	1	2018-01-17 08:57:51
<input type="checkbox"/>	 Editar  Copiar  Borrar	40	1	0	1	2018-01-17 08:57:51
<input type="checkbox"/>	 Editar  Copiar  Borrar	41	1	0	1	2018-01-17 08:57:52
<input type="checkbox"/>	 Editar  Copiar  Borrar	42	1	1	0	2018-01-17 08:57:52
<input type="checkbox"/>	 Editar  Copiar  Borrar	43	1	0	1	2018-01-17 08:57:53
<input type="checkbox"/>	 Editar  Copiar  Borrar	44	1	0	1	2018-01-17 08:57:53
<input type="checkbox"/>	 Editar  Copiar  Borrar	45	1	0	1	2018-01-17 08:57:57
<input type="checkbox"/>	 Editar  Copiar  Borrar	46	1	0	1	2018-01-17 08:57:57
<input type="checkbox"/>	 Editar  Copiar  Borrar	47	1	1	0	2018-01-17 08:57:57

**Figura 18** Almacenamiento de datos MySQL

#### 4.2. Instalación y configuración de OpenCV en la Raspberry Pi 3

A continuación, se detalla la instalación y configuración de OpenCV para Python 3 en la Raspberry Pi 3 con sistema operativo Raspbian Stretch, proceso importante para el adecuado funcionamiento del algoritmo de conteo.

Es importante instalar todas las dependencias de OpenCV, para lo cual primero se debe actualizar cualquier paquete existente en la Raspberry Pi utilizando los siguientes comandos:

```
sudo apt-get update && sudo apt-get upgrade
```

A continuación, utilizando los siguientes comandos se puede instalar: herramientas de desarrollador que permitan configurar el proceso de compilación OpenCV; paquetes de entrada y salida de imagen que admitan cargar varios formatos de archivos de imágenes como jpeg, png entre otros; paquetes de entrada y salida de video que permitan leer distintos formatos de archivos de video y trabajar con transmisiones directas de video; y un módulo GUI utilizado para mostrar imágenes en pantalla.

```
sudo apt-get install build-essential cmake pkg-config
sudo apt-get install libjpeg-dev libtiff5-dev libjasper-dev libpng12-dev
sudo apt-get install libavcodec-dev libavformat-dev libswscale-dev
sudo apt-get install libv4l-dev
sudo apt-get install libxvcore-dev libx264-dev
sudo apt-get install libgtk2.0-dev libgtk-3-dev
```

Seguidamente, se procede a descargar el archivo de instalación de OpenCV de su página web oficial, para lo cual se tiene el siguiente comando:

```
wget -O opencv.zip https://github.com/ltseez/opencv/archive/3.3.0.zip
```

Antes de empezar a compilar OpenCV en la Raspberry Pi 3, se requiere instalar pip, que consiste en un administrador de paquetes de Python:

```
wget https://bootstrap.pypa.io/get-pip.py
sudo python get-pip.py
sudo python3 get-pip.py
```

Finalmente, se puede compilar e instalar OpenCV mediante los siguientes comandos:

```
cd ~ / opencv-3.3.0 /
mkdir build
cd build
cmake -D CMAKE_BUILD_TYPE = RELEASE \
      -D CMAKE_INSTALL_PREFIX = /usr/local \
```

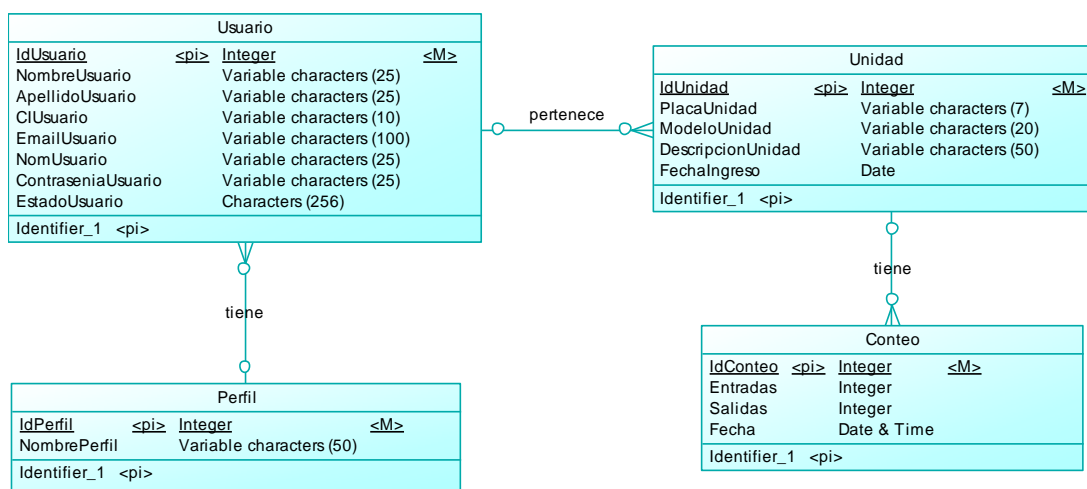
```
-D INSTALL_PYTHON_EXAMPLES = ON \
-D OPENCV_EXTRA_MODULES_PATH = ~ / opencv_contrib-3.3.0 / modules \
-D BUILD_EXAMPLES = ON ..
```

Para verificar el proceso de instalación en el terminal de la Raspberry Pi 3 se ejecuta los comandos source y workon para importar las librerías de Python y OpenCV:

```
source ~/.profile
workon cv
python
>>> import cv2
>>> cv2.__version__
'3.3.0'
>>>
```

#### 4.3. Modelo conceptual de la base de datos del sistema

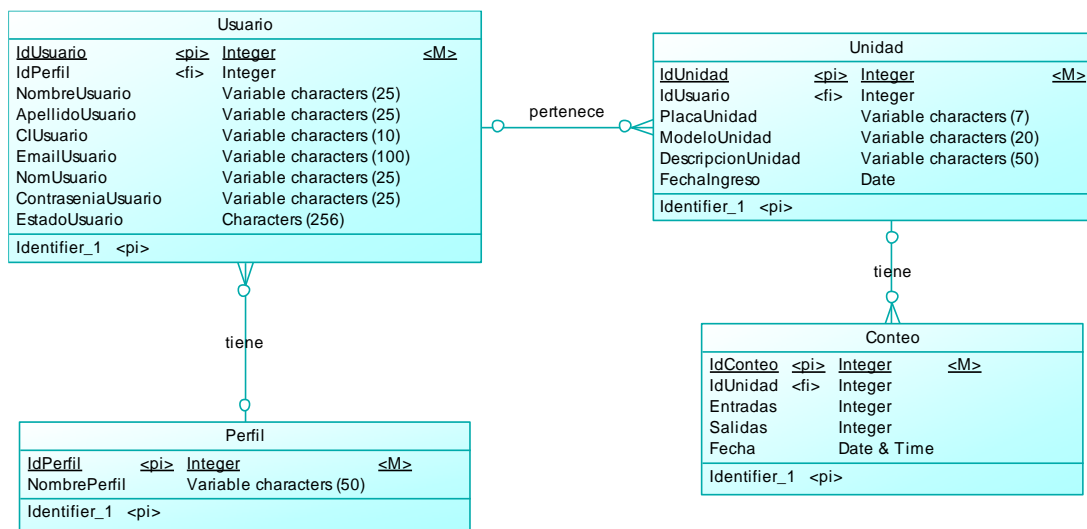
En la figura 19 se puede apreciar el modelo de la base de datos del sistema de conteo propuesto.



**Figura 19** Modelo conceptual de la base de datos

### 4.3. Modelo lógico de la base de datos del sistema

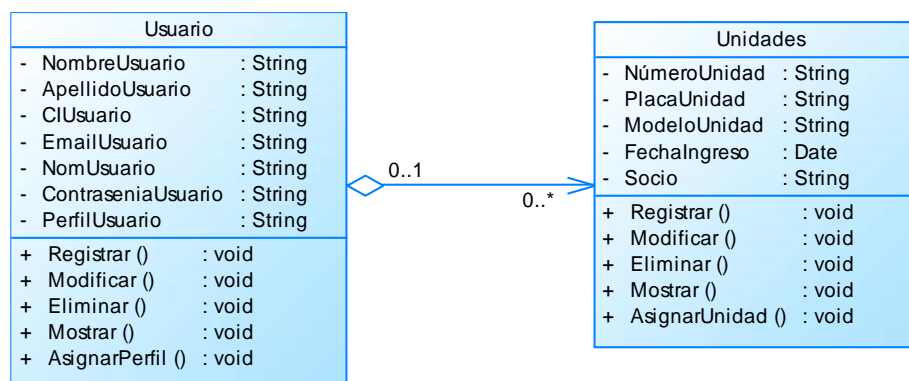
El modelo lógico de la base de datos del sistema de conteo propuesto en el presente proyecto de investigación puede ser visualizado en la figura 20.



**Figura 20** Modelo lógico de la base de datos

### 4.4. Diagrama de clases

En la figura 21 se puede visualizar el diagrama de clases para el desarrollo del sistema web.



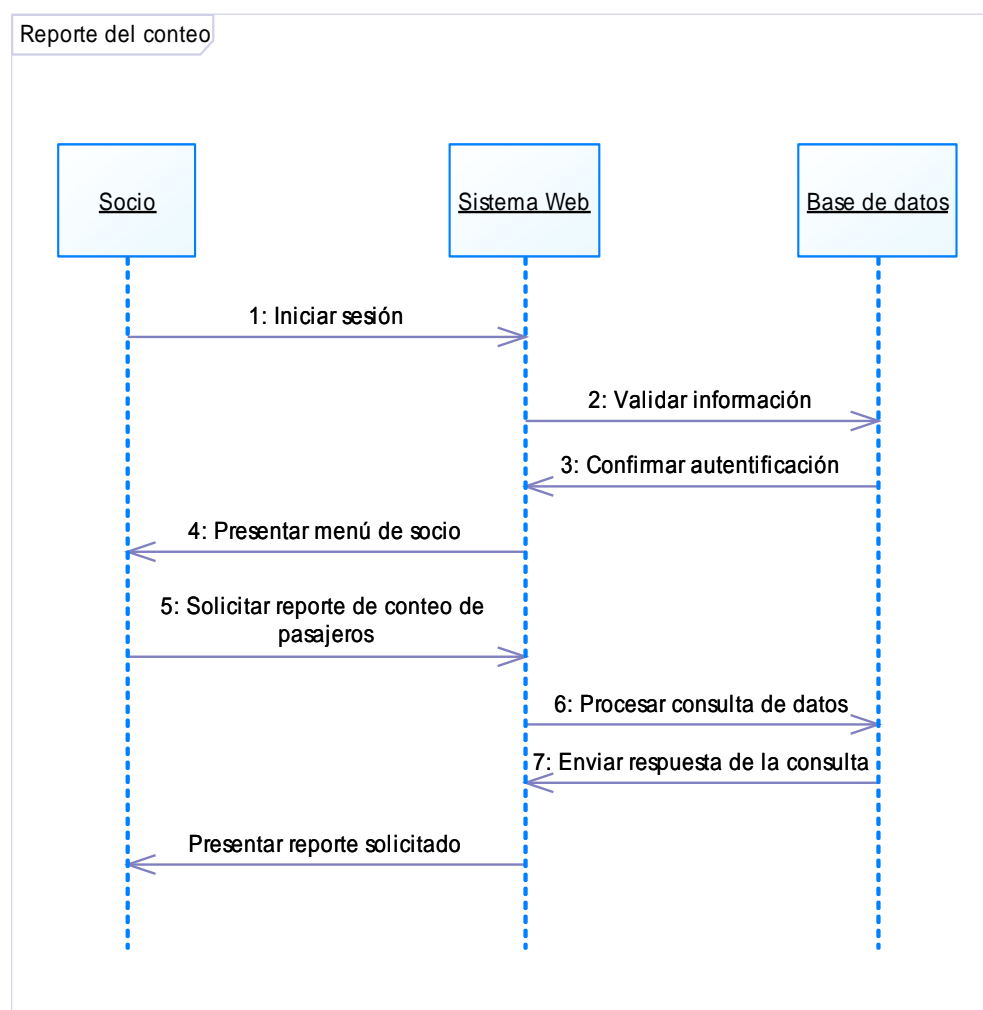
**Figura 21** Diagrama de clases

## 4.5. Diagrama de secuencia

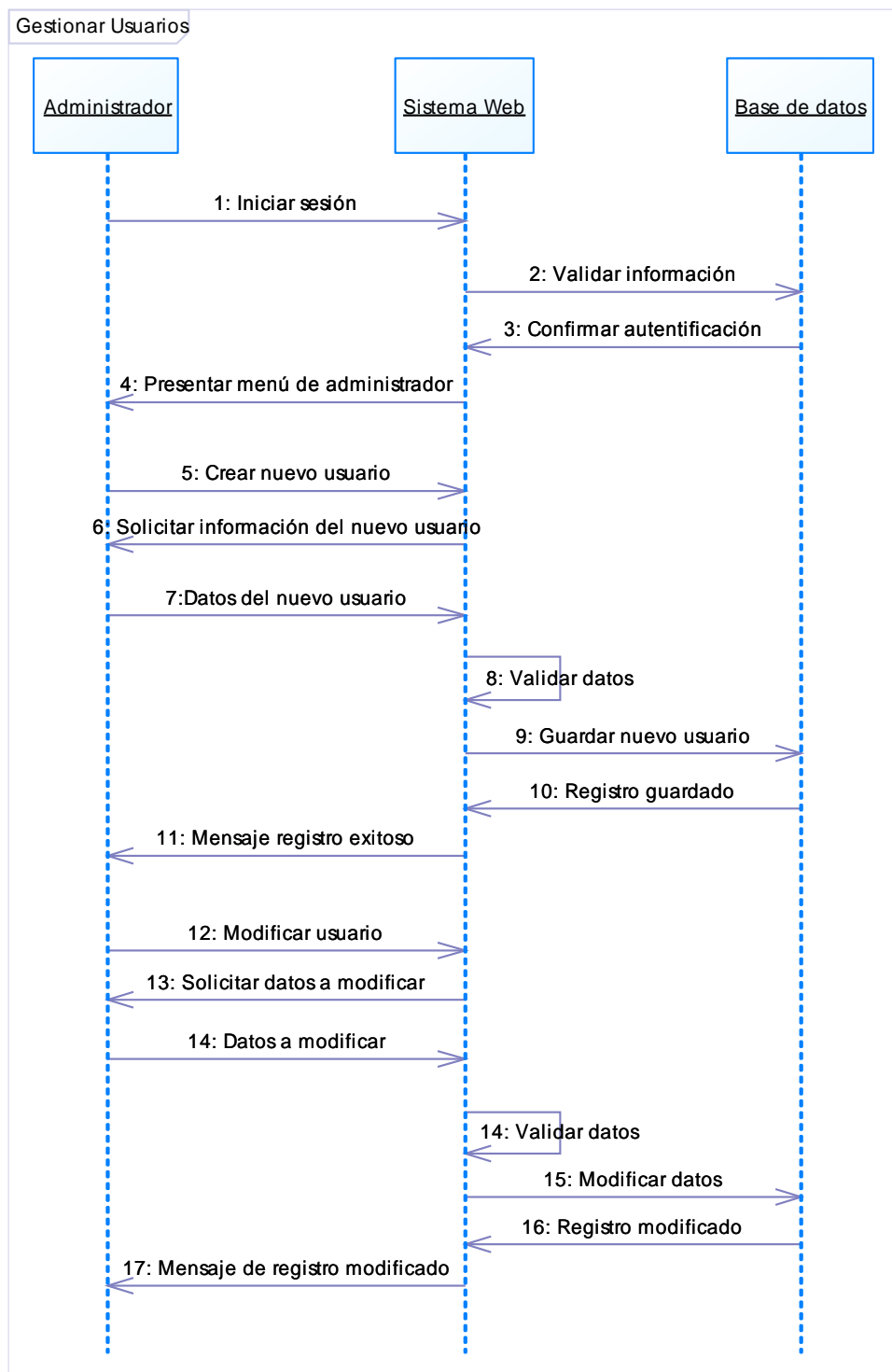
Un diagrama de secuencia permite mostrar la interacción entre objetos de una aplicación a través del tiempo.

En la figura 22 se puede observar la secuencia a seguir para que el usuario socio pueda acceder al reporte del conteo de pasajeros en tiempo real.

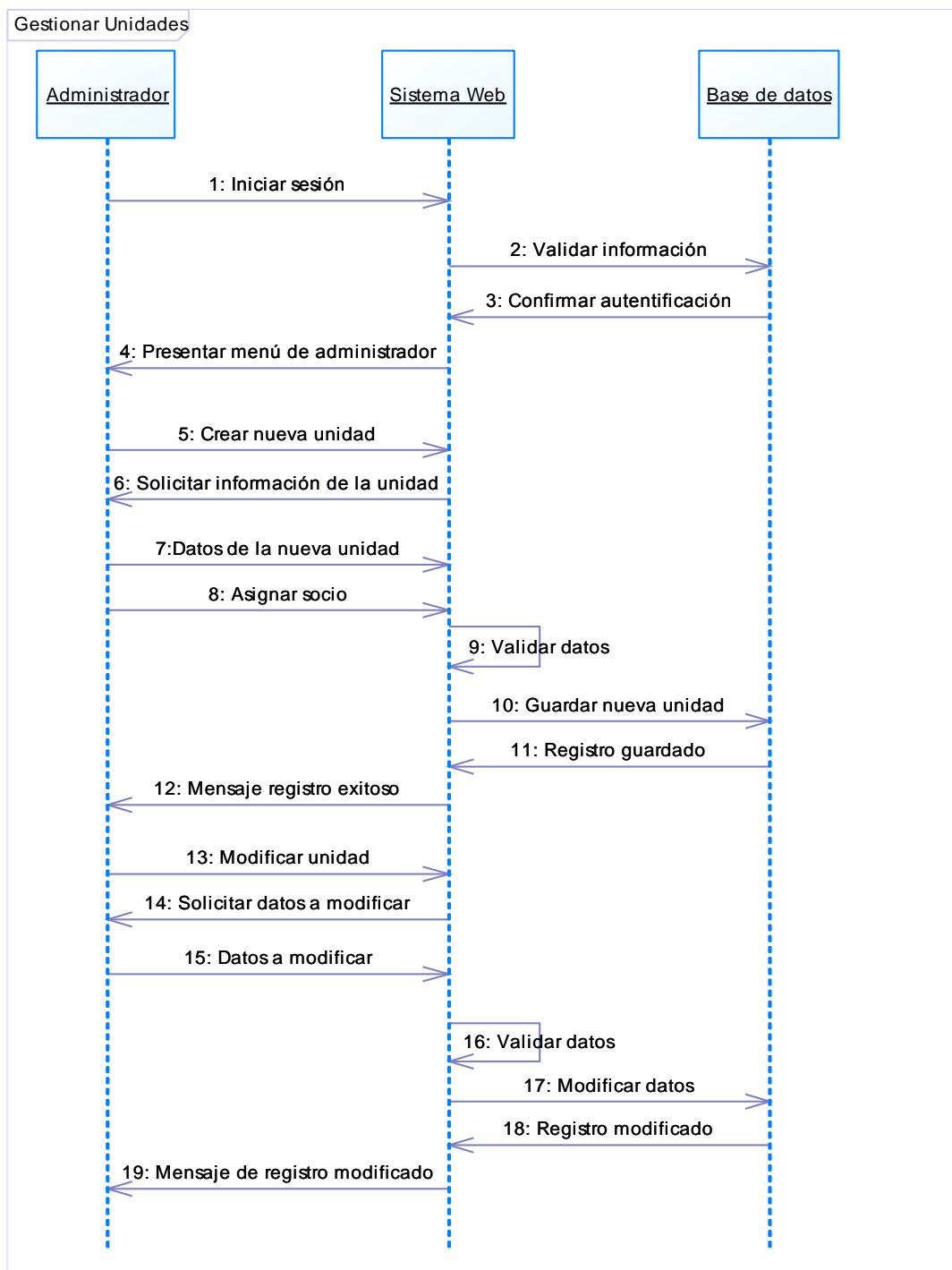
En la figura 23 y 24 se ilustra la secuencia del usuario administrador para gestionar usuarios y gestionar unidades respectivamente.



**Figura 22** Diagrama de secuencia – Reporte del conteo



**Figura 23** Diagrama de secuencia – Gestionar usuarios



**Figura 24** Diagrama de secuencia – Gestionar unidades

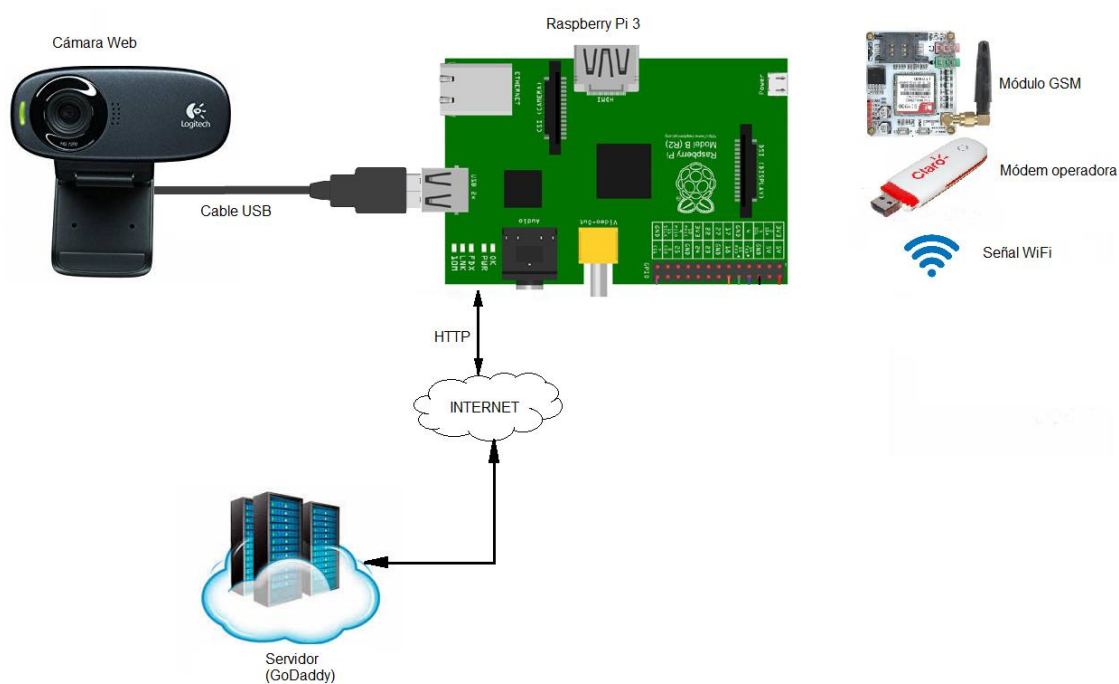


## 4.8. Arquitectura del sistema

A continuación, se describe la arquitectura del sistema de conteo y el diagrama de arquitectura de hardware de la infraestructura utilizada.

### 4.8.1 Diagrama de integración de hardware

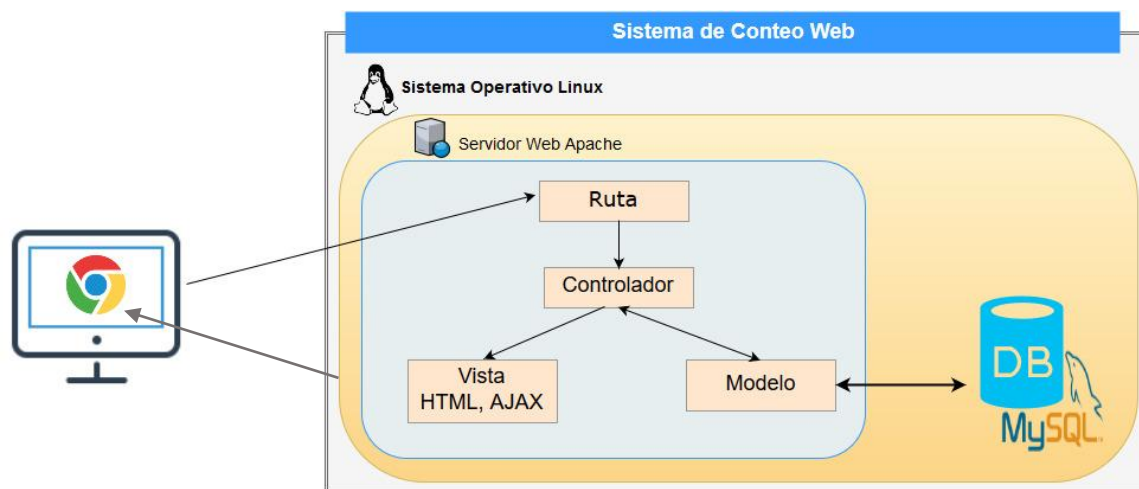
En la figura 25 se puede visualizar el diagrama de integración del hardware utilizado para el sistema de conteo.



**Figura 25** Diagrama de arquitectura de hardware

## 4.8.2 Diagrama de arquitectura del sistema

La figura 26, muestra el diagrama de arquitectura del sistema.



**Figura 26** Diagrama de arquitectura del sistema

## CAPÍTULO V

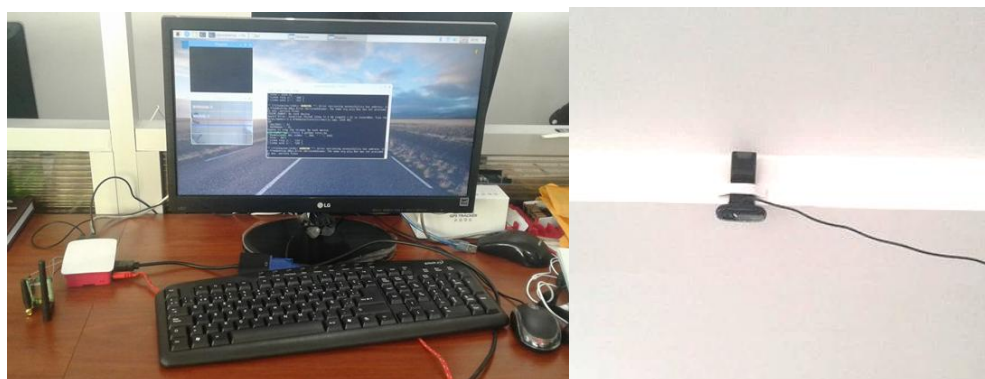
### PRUEBAS DEL ALGORITMO DE CONTEO Y DEL SISTEMA

En el apartado anterior se realizó el análisis, desarrollo e implementación del sistema de conteo tanto del ingreso como las salidas de las personas. El algoritmo desarrollado primero realiza la detección de la persona y a través de su movimiento se establece su entrada o salida.

El este capítulo mediante pruebas experimentales se validarán los datos del algoritmo de conteo para establecer de esta manera un porcentaje de error.

#### 5.1. Casos de prueba del algoritmo de conteo

Las pruebas realizadas para determinar el porcentaje de error en el conteo del algoritmo se realizaron en la oficina de la empresa colocando la cámara y la Raspberry Pi 3 como se puede observar en la figura 27. Se establecieron dos escenarios: el primero consiste en definir el valor del umbral adecuado que realice la detección del objeto adecuado; el segundo escenario radica en efectuar un conteo manual y comparar con los datos proporcionados por el algoritmo, luego realizar una comparativa de los mismos para lograr establecer el porcentaje de error del sistema.



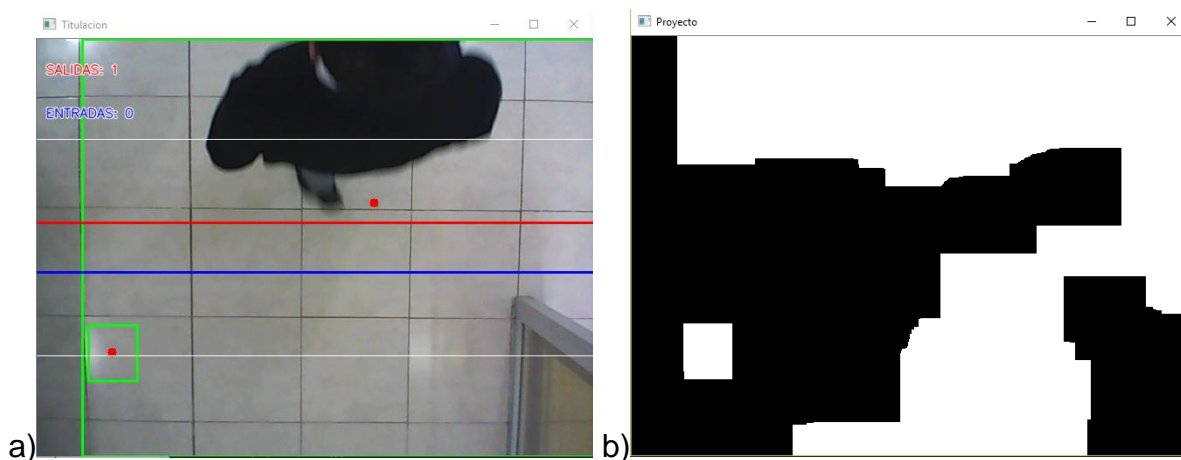
**Figura 27 Sistema de conteo implementado**

### 5.1.1 Escenario 1: Valor del umbral adecuado para la detección de objetos

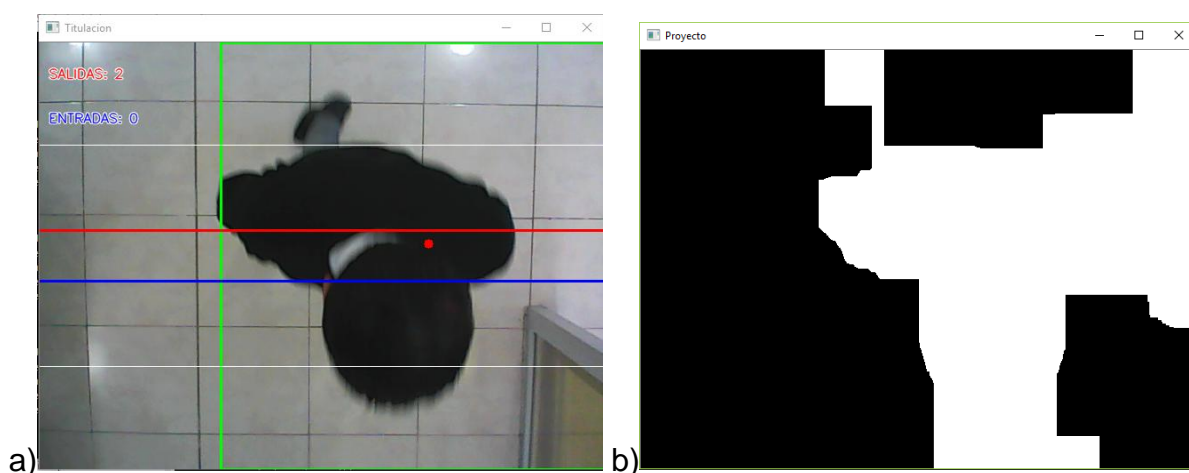
a) Para la primera prueba, se utilizó los siguientes parámetros de configuración:

- En el método `createBackgroundSubtractorMOG2` el valor `varThreshold = 15`
- El método `threshold` se colocó como valor de umbral = 10

Como puede observarse en las figuras 28(b) y 29(b), la imagen capturada una vez procesada para eliminar su ruido con los parámetros de umbral establecidos al no ser los adecuados detecta manchas en el piso considerándolo como objetos, por lo tanto, esto ocasionará problemas y datos erróneos en el conteo.



**Figura 28** Detección de movimiento salidas a) Captura normal b) Captura con substracción de fondo



**Figura 29** Detección de movimiento entradas a) Captura normal b) Captura con substracción de fondo

Para determinar el porcentaje de error con estos parámetros, se simularán el ingreso y salidas de 15 personas, luego se comparará con el resultado que se muestra del conteo, y finalmente se establecerá un porcentaje de error. Los resultados pueden apreciarse en la tabla 18:

**Tabla 18**  
*Resultados de la detección de objetos*

Fecha y hora	N° de entradas	N° de salidas
19/1/2018 16:43	0	1
19/1/2018 16:43	0	1
19/1/2018 16:44	1	0
19/1/2018 16:44	0	1
19/1/2018 16:44	1	0
19/1/2018 16:44	0	1
19/1/2018 16:44	1	0
19/1/2018 16:44	0	1
19/1/2018 16:44	1	0
19/1/2018 16:44	0	1

CONTINÚA 

19/1/2018 16:44	0	1
19/1/2018 16:44	1	0
19/1/2018 16:44	1	0
19/1/2018 16:44	0	1
19/1/2018 16:44	1	0
19/1/2018 16:45	1	0
19/1/2018 16:45	0	1
19/1/2018 16:45	1	0
19/1/2018 16:45	0	1
19/1/2018 16:45	0	1
19/1/2018 16:45	1	0
19/1/2018 16:45	1	0
19/1/2018 16:45	0	1
19/1/2018 16:45	1	0
19/1/2018 16:45	1	0
19/1/2018 16:45	0	1
19/1/2018 16:45	0	1
19/1/2018 16:45	1	0
19/1/2018 16:45	1	0
19/1/2018 16:45	1	0
19/1/2018 16:45	0	1
19/1/2018 16:45	1	0
19/1/2018 16:45	0	1
19/1/2018 16:45	0	1
19/1/2018 16:45	1	0
19/1/2018 16:45	0	1
19/1/2018 16:46	0	1
19/1/2018 16:46	1	0
19/1/2018 16:46	1	0
19/1/2018 16:46	1	0
19/1/2018 16:46	0	1
19/1/2018 16:46	1	0
<b>TOTAL</b>	<b>21</b>	<b>18</b>

En la figura 30 y 31 se puede visualizar el resultado capturado en el video procesado y en el sitio web desarrollado respectivamente, siendo este el mismo dato total de la tabla anterior.



**Figura 30** Resultado capturado del video

25	2018-01-19 16:45:39	1	0
26	2018-01-19 16:45:40	0	1
27	2018-01-19 16:45:44	0	1
28	2018-01-19 16:45:44	1	0
29	2018-01-19 16:45:44	1	0
30	2018-01-19 16:45:50	1	0
31	2018-01-19 16:45:54	0	1
32	2018-01-19 16:45:54	1	0
33	2018-01-19 16:45:59	0	1
34	2018-01-19 16:46:03	0	1
35	2018-01-19 16:46:03	1	0
36	2018-01-19 16:46:03	1	0
37	2018-01-19 16:46:08	1	0
38	2018-01-19 16:46:08	0	1
39	2018-01-19 16:46:13	1	0

Total entradas: 21  
Total salidas: 18

**Figura 31** Resultado del sitio web

Para el cálculo del error se utilizó la siguiente fórmula

$$\%error = \frac{Valor\ exacto - Valor\ aproximado}{Valor\ exacto} \times 100$$

Donde:

Valor exacto: corresponde al número de entradas o salidas que se van a simular.

Valor aproximado: corresponde al número de entradas o salidas capturados por el sistema y algoritmo.

Porcentaje de error de entradas:

$$\%errorEntradas = \frac{15 - 21}{15} \times 100$$

$$\%errorEntradas = 40\%$$

Porcentaje de error de salidas:

$$\%errorSalidas = \frac{15 - 18}{15} \times 100$$

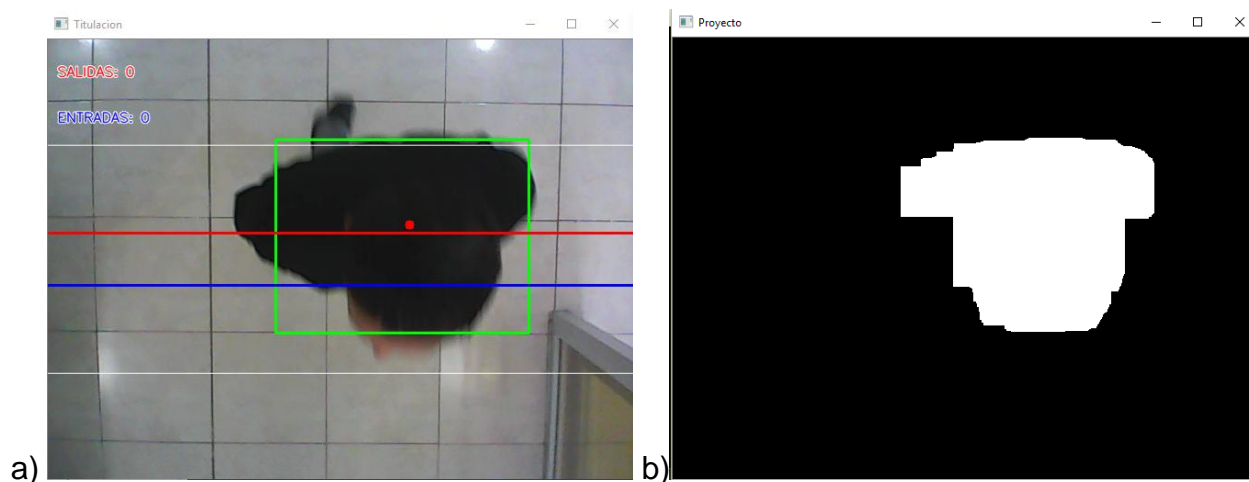
$$\%errorSalidas = 20\%$$

b) Para ejecutar la segunda prueba, se reiniciaron los valores de la base de datos y se manejó los siguientes parámetros de configuración:

- En el método `createBackgroundSubtractorMOG2` el valor de umbral `varThreshold = 65`
- El método `threshold` se colocó como valor de umbral = 200

Como puede observarse en la figura 32(b), la imagen capturada detecta de manera clara únicamente a la persona que atraviesa por la cámara eliminando en gran porcentaje el ruido de la imagen.





**Figura 32** Detección de movimiento a) Captura normal b) Captura con substracción de fondo

Para determinar el porcentaje de error con estos parámetros, se simularán el ingreso y salidas de 15 personas como en la primera prueba realizada, se comparará con los resultados que se muestran en el conteo, y finalmente se establecerá un porcentaje de error. Los resultados pueden apreciarse en la tabla 19:

**Tabla 19**

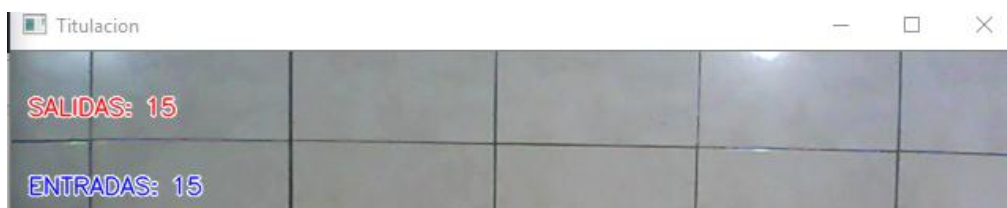
*Resultados de detección de objetos con nuevo valor de umbral*

Fecha y hora	N° de entradas	N° de salidas
19/1/2018 17:17	0	1
19/1/2018 17:17	1	0
19/1/2018 17:17	0	1
19/1/2018 17:17	1	0
19/1/2018 17:18	0	1
19/1/2018 17:18	1	0
19/1/2018 17:18	0	1
19/1/2018 17:18	1	0
19/1/2018 17:18	0	1

**CONTINÚA** 

19/1/2018 17:18	1	0
19/1/2018 17:18	0	1
19/1/2018 17:18	1	0
19/1/2018 17:18	0	1
19/1/2018 17:18	1	0
19/1/2018 17:18	0	1
19/1/2018 17:18	1	0
19/1/2018 17:18	0	1
19/1/2018 17:18	1	0
19/1/2018 17:18	0	1
19/1/2018 17:18	1	0
19/1/2018 17:18	0	1
19/1/2018 17:19	1	0
19/1/2018 17:19	0	1
19/1/2018 17:19	1	0
19/1/2018 17:19	0	1
19/1/2018 17:19	1	0
19/1/2018 17:19	0	1
19/1/2018 17:19	1	0
19/1/2018 17:19	0	1
19/1/2018 17:19	1	0
<b>TOTAL</b>	<b>15</b>	<b>15</b>

Como puede observarse en las figuras 33 y 34, el resultado capturado en el video procesado y en el sitio web desarrollado respectivamente, corresponde al mismo dato total de la tabla 18.



**Figura 33** Nuevo resultado capturado del video

 A screenshot of a web browser window titled 'Optimovilidad' showing a table of data. The table has 30 rows and 5 columns. Below the table, there are two summary statistics: 'Total entradas: 15' and 'Total salidas: 15'.
 

16	2018-01-19 17:18:39	1	0	
17	2018-01-19 17:18:43	0	1	
18	2018-01-19 17:18:46	1	0	
19	2018-01-19 17:18:50	0	1	
20	2018-01-19 17:18:54	1	0	
21	2018-01-19 17:18:59	0	1	
22	2018-01-19 17:19:03	1	0	
23	2018-01-19 17:19:06	0	1	
24	2018-01-19 17:19:09	1	0	
25	2018-01-19 17:19:12	0	1	
26	2018-01-19 17:19:15	1	0	
27	2018-01-19 17:19:18	0	1	
28	2018-01-19 17:19:22	1	0	
29	2018-01-19 17:19:25	0	1	
30	2018-01-19 17:19:28	1	0	

Total entradas: 15  
Total salidas: 15

**Figura 34** Nuevo resultado del sitio web

Al aplicar las fórmulas para calcular el porcentaje de error, se obtuvieron los siguientes resultados:

Porcentaje de error de entradas:

$$\%errorEntradas = \frac{15 - 15}{15} \times 100$$

$$\%errorEntradas = 0\%$$

Porcentaje de error de salidas:

$$\%errorSalidas = \frac{15 - 15}{15} \times 100$$

$$\%errorSalidas = 0\%$$

### **Conclusión:**

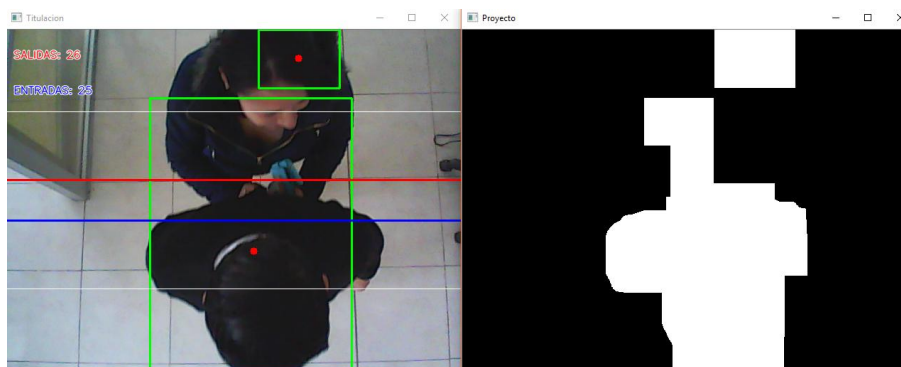
Con el porcentaje de error calculado en la prueba 1 y 2, se puede concluir que los parámetros de umbral con valores: 65 y 200 proyectaron el menor porcentaje de error en el conteo, por lo tanto, se consideraron estos datos para las pruebas del apartado 5.1.2.

#### **5.1.2 Escenario 2: Pruebas del conteo realizado por el algoritmo.**

Las pruebas del escenario 2 consisten en validar y obtener el porcentaje de error del algoritmo de conteo con los valores de umbral que proyectaron el menor porcentaje de error en el reconocimiento de objetos en el escenario 1.

Para determinar el error se simularon ingresos y salidas de 60 personas, donde se consideró el siguiente factor:

- Cuando entran dos personas muy unidas, una atrás de la otra, como se visualiza en la figura 35.



**Figura 35** Entrada de dos personas

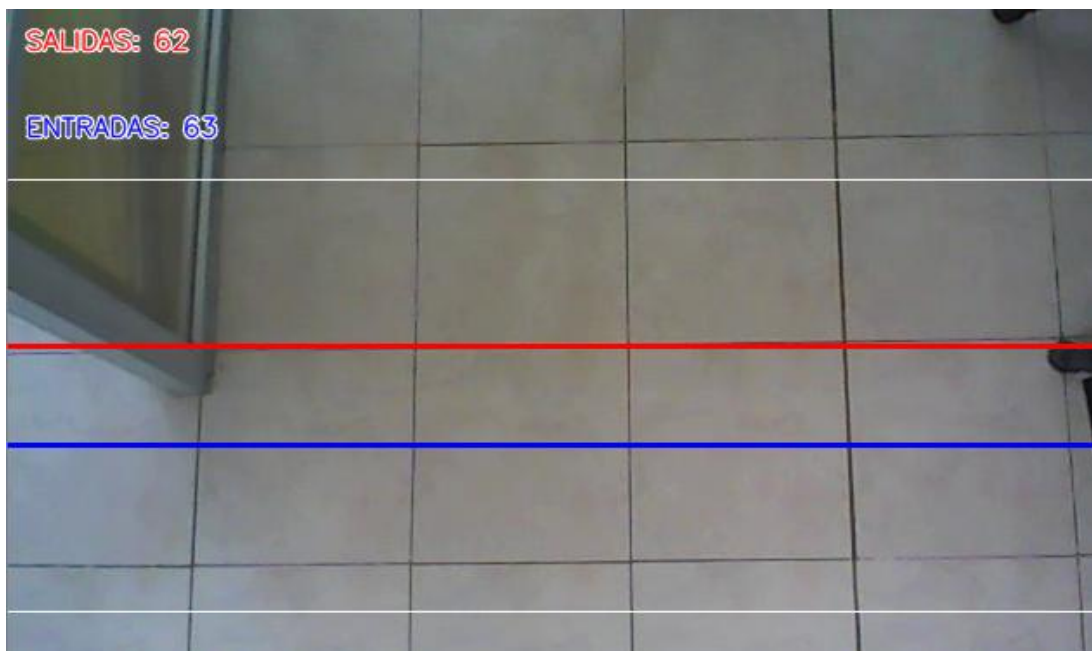
Para la muestra de los resultados se realizó una tabla (ver tabla 20) por periodos de tiempo, donde se obtienen las entradas y salidas capturadas en el sitio web del sistema desarrollado y se realiza un filtro por minutos.

**Tabla 20**

*Resultados del conteo final*

<b>Rango de tiempo en minutos</b>	<b>N° de entradas</b>	<b>N° de salidas</b>
54:49 – 57:55	9	10
58:53 – 03:59	9	7
01:08 – 03:25	3	5
04:37 – 06:48	3	3
07:06 – 10:55	10	9
11:00 – 14:41	20	18
15:20 – 18:46	4	5
19:02 – 21:03	5	5
<b>TOTAL</b>	<b>63</b>	<b>62</b>

Las figuras 36 y 37 muestran el resultado capturado directamente del video y del sitio web desarrollado, respectivamente.



**Figura 36** Resultado final en el procesamiento de video

The screenshot shows a web browser window with a table of data. The table has four columns: ID, timestamp, and two numerical values. Below the table, there are two lines of text: 'Total entradas: 63' and 'Total salidas: 62'.

111	2018-02-01 10:17:50	0	1
112	2018-02-01 10:18:06	1	0
113	2018-02-01 10:18:16	0	1
114	2018-02-01 10:18:35	1	0
115	2018-02-01 10:18:46	0	1
116	2018-02-01 10:19:02	1	0
117	2018-02-01 10:19:10	0	1
118	2018-02-01 10:19:21	1	0
119	2018-02-01 10:19:33	0	1
120	2018-02-01 10:19:49	1	0
121	2018-02-01 10:19:58	0	1
122	2018-02-01 10:20:12	1	0
123	2018-02-01 10:20:33	0	1
124	2018-02-01 10:20:47	1	0
125	2018-02-01 10:21:03	0	1

Total entradas: 63  
Total salidas: 62

**Figura 37** Resultado final en el sitio web

Con los resultados obtenidos se procede a aplicar las fórmulas para calcular el porcentaje de error, adquiriendo lo siguiente:

Porcentaje de error de entradas:

$$\%errorEntradas = \frac{60 - 63}{60} \times 100$$

$$\%errorEntradas = 5\%$$

Porcentaje de error de salidas:

$$\%errorSalidas = \frac{60 - 62}{60} \times 100$$

$$\%errorSalidas = 3,33\%$$

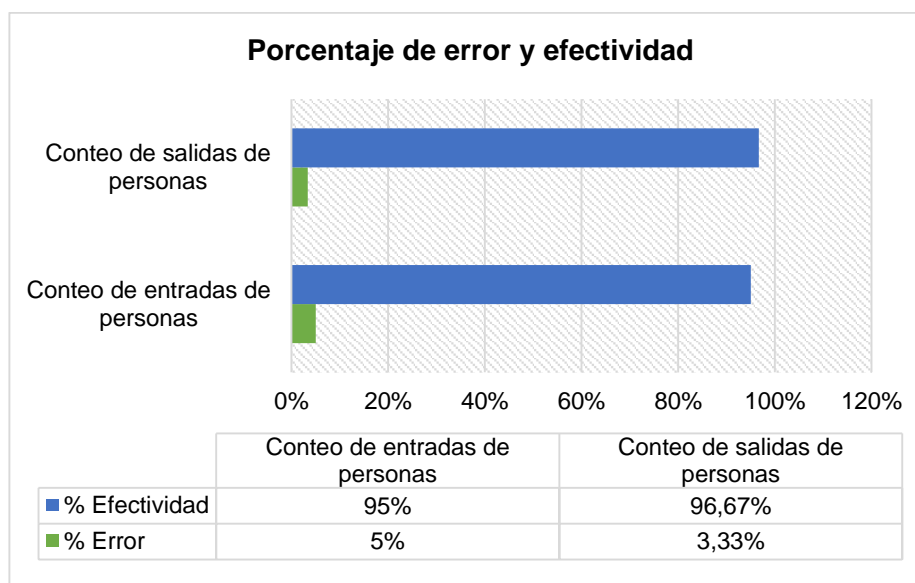
**Conclusión:**

Finalizando las pruebas, se puede determinar que existe un margen de error considerable del sistema de conteo en las entradas y salidas de personas.

**Tabla 21**

*Resultado final del sistema conteo*

<b>Característica</b>	<b>% Error</b>	<b>% Efectividad</b>
Conteo de entradas de personas	5 %	95 %
Conteo de salidas de personas	3,33 %	96,67 %
<b>PORCENTAJE TOTAL</b>	<b>4,17</b>	<b>95,84</b>



**Figura 38** Porcentaje de error y efectividad del conteo

Como puede observarse en la figura 38, el porcentaje de efectividad del sistema es del 95,8% con un margen de error del 4,2%, por lo tanto, se considera que el sistema de conteo es aceptable y puede ser implantado en las unidades.



## CONCLUSIONES Y RECOMENDACIONES

### 6.1. Conclusiones

Al concluir el presente proyecto de titulación se obtuvieron las siguientes conclusiones:

El sistema de conteo en tiempo real propuesto presenta a los socios información numérica de las personas que ingresan y salen de una unidad, además la posibilidad de visualizar el conteo con la ayuda de una pantalla con puerto HDMI y conexión directa a la Raspberry, considerando que no se realiza ninguna transmisión de video debido al alto consumo de datos que requiere ejecutar esta operación y por ende resulta demasiado costoso para los socios.

Las condiciones naturales como luz influyen en la calidad de video que se procesa para el algoritmo de conteo, por tal motivo es importante considerar este factor en el error del conteo, pues al capturar imágenes estimadas como malas, la detección de objetos y el conteo de los mismos será muy inexacto.

El algoritmo de conteo de sustracción y modelado de fondo seleccionado para la ejecución del presente proyecto, se consideró como el más apropiado por las condiciones en las que opera una unidad de transporte público, donde 2 es número de personas máximo que ingresan por las puertas debido a su tamaño, por lo tanto, cumple con la característica principal del algoritmo elegido.

OpenCV es una librería de código abierto que contiene una gran cantidad de funciones que permiten de mejor manera detectar el movimiento de las personas, es importante señalar que para el presente proyecto la cámara se debe encontrar estática y no realizar movimientos, porque la programación del algoritmo utiliza la función de sustracción de fondo que trabaja exclusivamente en estos escenarios.

El lenguaje de programación Python permitió implementar las librerías de OpenCV de manera adecuada, además dispone de la librería auxiliar request, la cual permite enviar un HTTP con la información y datos que deben ser almacenados en la base de datos, evitando de esta manera el uso de hardware adicional, pues únicamente con programación en el lenguaje es posible. Para el adecuado funcionamiento del envío de datos se requiere conexión de internet al hardware que se utilice, este puede ser un módem, módulos GSM o conexión Wifi.

El desarrollo de algoritmos utilizando videos en la actualidad es conocido como analítica de video y ha tenido una gama de aplicaciones en lo que se refiere a seguridades por los beneficios que aporta. En el presente proyecto se analizó el video en tiempo real para desarrollar un algoritmo que cuente entradas y salidas de las personas detectadas, logrando de esta manera disponer de un sistema de conteo con información en tiempo real y a un precio accesible para la sociedad.

Al finalizar las pruebas del algoritmo indican que los resultados son factibles para su implementación en las unidades de transporte público considerando que el valor de umbral requiere ser configurado para cada instalación que se requiera, pues la carrocería de cada unidad de transporte varía y por ende las dimensiones de altura de las puertas son distintas.

## 6.2. Recomendaciones

Para el presente proyecto de titulación se utilizó como infraestructura hardware una Raspberry pi 3 disponibles en la oficina de Optimovilidad, sin embargo, se recomienda para un adecuado procesamiento del video instalar y utilizar una tarjeta de memoria micro SD de categoría 10, con capacidad de almacenamiento mínima de 64 GB.

Se recomienda revisar y analizar la cámara web para la Raspberry Pi 3, debido a que algunos modelos de cámaras no son compatibles con este sistema empotrado. La cámara utilizada para el presente proyecto fue el modelo Logitech C 217 y su compatibilidad fue validada en el sitio web [https://elinux.org/RPi\\_USB\\_Webcams](https://elinux.org/RPi_USB_Webcams).

Realizar la incorporación de un módulo GPS al algoritmo de conteo desarrollado, el cual permita referenciar Geo cercas con paradas que serán especificadas por el operador de la flota, de esta manera se logrará tener un reporte de pasajeros acumulado por cada geo cerca definida.

A pesar de que el porcentaje de error obtenido en las pruebas del sistema es considerado como aceptable para su operación en las unidades, se recomienda para minorar el mismo: utilizar sistemas que detecten adecuadamente la luminosidad en las unidades de transporte; el uso de cámaras infrarrojas debido a que en la noche este sería un problema difícil de controlar; utilizar el hardware adecuado para el procesamiento del algoritmo y manejar una cámara con buena resolución de imagen.

Se recomienda realizar un estudio de los equipos MDVR que se encuentran instalados en las unidades de transporte para que, mediante la captura de este video, procesar el algoritmo de conteo desarrollado, sin tener la necesidad de adicionar otros equipos.

Como línea de trabajo futuro se recomienda, utilizando una cámara auxiliar realizar la detección de rostro de personas y mediante algoritmos de OpenCV divisar rasgos o

características que permitan identificar a personas adultas, niños y adultas mayores, logrando de esta manera que el cobro de pasajes completos y medios pasajes sea más exacto, con la finalidad de no perjudicar económicamente al personal que labora en el bus.

Como trabajo futuro se recomienda utilizar una arquitectura hardware que permita ejecutar el almacenamiento del video capturado y la transmisión en tiempo real del mismo, para no realizar únicamente el procesamiento del algoritmo, de esta manera se podrá disponer de un módulo de video vigilancia que faciliten los procesos de fiscalización en las cooperativas de transporte.

## BIBLIOGRAFÍA

- Activenanda. (19 de Marzo de 2017). *ATSS Trusted Brand*. Recuperado el 23 de Octubre de 2017, de <http://www.atss.in/what-is-mobile-dvr/>
- Barandiaran, J., Murguia, B., & Boto, F. (2008). *Real-Time People Counting Using Multiple Lines*. Ninth International Workshop on Image Analysis for Multimedia Interactive Services, 159-162.
- Brostow, G., & Cipolla, R. (2006). *Unsupervised Bayesian Detection of Independent Motion in Crowds*. Conference Computer Vision and Pattern Recognition.
- Cane, T. (12 de Diciembre de 2013). *RUP vs XP vs Scrum - Which is the next standard?* Obtenido de Teach Notes: <http://notesontechstuff.blogspot.com/2013/12/rup-vs-xp-vs-scrum-which-is-next.html>
- Chazallet, S. (2016). *Python 3. Los fundamentos del lenguaje*. Barcelona: Ediciones Eni.
- Chen, C.-H., Chen, T.-Y., Wang, D.-J., & Chen, T.-J. (2012). *A Cost-Effective People-Counter for a Crowd of Moving People Based on Two-Stage Segmentation*. Journal of Information Hiding and Multimedia Signal Processing, 12-23.
- Chen, T.-h. (2006). *An Intelligent People-Flow Counting Method for Passing Through a Gate*. Robotics, Automation and Mechatronics.
- Cong, Y., Chun Zhu, S., & Tang, Y. (2009). *Flow mosaiking: Real-time pedestrian counting without Scene-specific learning*. IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 1093-1100.
- Counterest. (2017). *7 beneficios de usar contadores de personas y analítica de pasajeros en transporte público*. Recuperado el 28 de Septiembre de 2017, de <http://counterest.net/beneficios-sobre-contadores-de-personas-y-analitica-de-pasajeros-en-transporte-publico/>
- Doyle, M. (25 de Agosto de 2011). *MySQL for Absolute Beginners*. Obtenido de Elated: <https://www.elated.com/articles/mysql-for-absolute-beginners/>

- Fehr, D., Sivalingam, R., Morellas, V., Papanikolopoulos, N., Loftallah, O., & Park, Y. (2009). *Counting people in groups*. Sixth IEEE International Conference on Advanced Video and Signal Based Surveillance, 152-157.
- Francia, J. (25 de Septiembre de 2017). *Scrum Una mejor manera de construir productos*. Obtenido de scrum.org: <https://www.scrum.org/resources/blog/que-es-scrum>
- Hemangi, B., & Nikhita, K. (2016). *People counting system using Raspberry Pi with OpenCV*. International Journal for Research in Engineering Application & Management, 1-5.
- Herrera, J., & Sánchez, L. M. (2013). *Introducción - Computación científica con Python*. Obtenido de Computación científica con Python para módulos de evaluación continua en asignaturas de ciencias aplicadas: <http://webs.ucm.es/info/aocg/python/introduccion/index.html>
- Hu, Y., Zhou, P., & Zhou, H. (2011). *A New Fast and Robust Method Based on Head Detection for People-Flow Counting System*. International Journal of Information Engineering (IJIE), 33-43.
- indra. (2017). *Sistemas de ayuda a la explotación SAE, Sistema de información a viajeros*. Obtenido de INDRA: <https://www.indracompany.com/es/informacion-viajeros-1>
- Kumar, R., Parashar, T., & Verma, G. (2012). *Background Modeling and Subtraction Based People Counting for Real Time Video Surveillance*. International Journal of Soft Computing and Engineering, 100-102.
- Leal, B. (5 de Abril de 2017). *Metodología Scrum en proyectos digitales*. Obtenido de Estrategia Digital: <https://www.ida.cl/blog/estrategia-digital/metodologia-scrum-en-proyectos-digitales/>
- Lee, G.-G., Kim, H.-k., & Yoon, J.-Y. (2008). *Pedestrian Counting Using an IR Line Laser*. International Conference on Convergence and Hybrid Information Technology.
- Lemus, A., García, A., & Salgado, E. (2016). *Maestría en Ingeniería Industrial*. Obtenido de SlydePlayer: <http://slideplayer.es/slide/3438905/>
- Movilidad. (2015). *Eje territorial, Comisión de tránsito*. Quito.
- OpenCV. (2015). *OpenCV*. Obtenido de Background Subtraction: [https://docs.opencv.org/3.1.0/db/d5c/tutorial\\_py\\_bg\\_subtraction.html#gsc.tab=0](https://docs.opencv.org/3.1.0/db/d5c/tutorial_py_bg_subtraction.html#gsc.tab=0)

- OpenCV. (2017). *About*. Obtenido de OpenCV: <https://opencv.org/about.html>
- Optimovilidad. (2017). *Quiénes Somos*. Obtenido de Optimovilidad: [www.optimovilidad.com](http://www.optimovilidad.com)
- Pablos, C., Pérez, L., & Montes, J. L. (2011). *Impacto de los sistemas de apoyo a la explotación (SAE) en la mejora de los servicios de transporte público urbano*. Cuadernos de Economía y Dirección de la Empresa, 12-24.
- Patzold, M., Heras, R., & Sikora, T. (2010). *Counting People in Crowded Environments by Fusion of Shape and Motion Information*. Seventh IEEE International Advanced Video and Signal Based Surveillance (AVSS), 157-164.
- Perez, E. (7 de Diciembre de 2011). *Lenguajes de programación*. Obtenido de Python: <http://aplicaciones-web-lenguajes-programaci.blogspot.com/2011/12/python.html>
- Pérez, I. (Diciembre de 2007). *Sistema Integral de Recaudo para el Transporte Público en el Área Metropolitana de Caracas*. Trabajo especial de grado. Caracas: Universidad Simón Bolívar.
- python. (2017). *Documentation*. Obtenido de Python Software Foundation: <https://www.python.org/doc/>
- Rabaud, V. (2006). *Counting Crowded Moving Objects*. IEEE Computer Society Conference on Computer Vision and Pattern Recognition.
- Raspberry. (2017). *Raspberry Pi Foundation*. Obtenido de <https://www.raspberrypi.org/>
- Republic, T. (2017). *PHP Introduction*. Obtenido de Tutorial Republic: <https://www.tutorialrepublic.com/php-tutorial/php-introduction.php>
- Rivas, L. (2013). *Detección y seguimiento de una persona en una habitación*. Trabajo de fin de máster. Valencia: Universidad Politécnica de Valencia.
- Rodríguez, A. (18 de Agosto de 2015). *OpenCV: Librería de Visión por Computador*. Obtenido de Universidad de la Laguna: <https://osl.ull.es/software-libre/opencv-libreria-vision-computador/>
- Rojas Cuevas, J. T. (10 de Abril de 2013). *Conteo de personas mediante videocámaras*. Mexico, México: Tesis de pregrado.
- RushRash. (2011). *Rush Rash Inc*. Obtenido de [http://rushrash.com/cctv/DVRs-Mobile-DVR-Cars-and-Van-DVR/c2\\_24/index.html](http://rushrash.com/cctv/DVRs-Mobile-DVR-Cars-and-Van-DVR/c2_24/index.html)

- Ryan, D., Denman, S., Fookes, C., & Sridharan, S. (2009). *Crowd Counting Using Multiple Local Features*. *Digital Image Computing: Techniques and Applications*, 81-88.
- Secretaría de Movilidad. (2017). *Implementación de los sistemas inteligentes de transporte (SIR-SAE-SIU)*. Obtenido de Secretaría de Movilidad Alcaldía: <http://www.secretariademovilidad.quito.gob.ec/index.php/proyectos1/politica-planeamiento-movilidad/proyecto-politicas-planeamiento-movilidad-menu/sistemas-inteligentes-sir-sae-siu.html>
- Sridhar, S., & Kowcika, A. (2015). *A literature study on crowd (people) counting with the help of surveillance videos*. *International Journal of Innovative Technology and Research*, 2353-2361.
- Thakkar, K. (21 de Noviembre de 2012). *What is OpenCV? OpenCV vs. MATLAB - An insight*. Obtenido de Karan Jitendra Thakkar: <https://karanjthakkar.wordpress.com/2012/11/21/what-is-opencv-opencv-vs-matlab/>
- Urdaneta, A. (2016). *Conoce los beneficios de utilizar contadores de personas*. Recuperado el 28 de Septiembre de 2017, de Wibo Analytics: <http://blog.wiboanalytics.com/beneficios-de-utilizar-contadores-de-personas-contador-de-personas>
- Vila, J. L. (2017). *Cómo implementar Scrum: adoptar los roles del equipo Scrum*. Obtenido de Management Plaza: <http://managementplaza.es/blog/implementar-scrum-adoptar-los-roles-del-equipo-scrum/>
- Ye, Q. (2010). *A robust method for counting people in complex indoor spaces*. *Conference on Education Technology and Computer (ICETC)*, 450-454.
- Ye, W., & Zhong, Z. (2007). *Robust people counting in crowded environment*. *International Conference on Robotics and Biomimetics*, 1133-1137.
- Yu, S., Chen, X., Sun, W., & Xie, D. (2008). *A robust method for detecting and counting people*. *International Conference on Audio, Language and Image Processing*, 1545-1549.



Zhou, X., Xu, S., & Wang, Y. (2012). *A combination of new static detection and dynamic detection in people counting*. International Conference on Measurement, Information and Control (MIC), 605-609.

