



UNIVERSIDAD TÉCNICA DE AMBATO

FACULTAD DE INGENIERÍA EN SISTEMAS

ELECTRÓNICA E INDUSTRIAL

CARRERA DE INGENIERÍA EN ELECTRÓNICA Y

COMUNICACIONES

Tema:

“SISTEMA INTELIGENTE DE RECONOCIMIENTO DE PATRONES CON
VISIÓN ARTIFICIAL PARA LA ALERTA AUTOMÁTICA DE INTRUSOS EN
LAS ÁREAS DE ALMACENAMIENTO DE LAS PYMES”

**Trabajo de Graduación. Modalidad: Proyecto de Investigación, presentado previo
la obtención del título de Ingeniero en Electrónica y Comunicaciones.**

SUBLÍNEA DE INVESTIGACIÓN: Procesamiento digital de señales e imágenes

AUTOR: López Márquez Gabriel Andrés

TUTOR: Ing. Mg. Edgar Freddy Robalino Peña

Ambato – Ecuador

Mayo – 2016

APROBACIÓN DEL TUTOR

En mi calidad de Tutor del Trabajo de Investigación sobre el Tema: **“Sistema inteligente de reconocimiento de patrones con visión artificial para la alerta automática de intrusos en las áreas de almacenamiento de las PYMES”**, del señor Gabriel Andrés López Márquez, estudiante de la Carrera de Ingeniería en Electrónica y Comunicaciones, de la Facultad de Ingeniería en Sistemas, Electrónica e Industrial, de la Universidad Técnica de Ambato, considero que el informe investigativo reúne los requisitos suficientes para que continúe con los trámites y consiguiente aprobación de conformidad con el numeral 7.2 de los Lineamientos Generales para la aplicación de Instructivos de las Modalidades de Titulación de las Facultades de la Universidad Técnica de Ambato.

Ambato, Mayo de 2016

TUTOR

Ing. Freddy Robalino, Mg.

AUTORÍA

El presente proyecto de investigación titulado: “**Sistema inteligente de reconocimiento de patrones con visión artificial para la alerta automática de intrusos en las áreas de almacenamiento de las PYMES**”, es absolutamente original, auténtico y personal, en tal virtud, el contenido, efectos legales y académicos que se desprenden del mismo son de exclusiva responsabilidad del autor.

Ambato, Mayo de 2016

AUTOR

Gabriel Andrés López Márquez

C.C.: 180392048-5

DERECHOS DE AUTOR

Autorizo a la Universidad Técnica de Ambato, para que haga uso de este Trabajo de Titulación como un documento disponible para la lectura, consulta y procesos de investigación.

Cedo los derechos de mi trabajo de Titulación, con fines de difusión pública, además autorizo su reproducción dentro de las regulaciones de la universidad.

Ambato, Mayo de 2016

AUTOR

Gabriel Andrés López Márquez
C.C.: 180392048-5

APROBACIÓN DE LA COMISIÓN CALIFICADORA

La Comisión Calificadora del presente trabajo conformada por los señores docentes, Ing. Mg. Santiago Manzano e Ing. Mg. Vicente Morales revisó y aprobó el Informe Final del trabajo de graduación titulado: **“Sistema inteligente de reconocimiento de patrones con visión artificial para la alerta automática de intrusos en las áreas de almacenamiento de las PYMES”**, presentado por el señor Gabriel Andrés López Márquez de acuerdo al numeral 9.1 de los Lineamientos Generales para la aplicación de Instructivos de las Modalidades de Titulación de las Facultades de la Universidad Técnica de Ambato.

Ambato, Mayo de 2016

Ing. Vicente Morales L., Mg.
PRESIDENTE DEL TRIBUNAL

Ing. Mg. Santiago Manzano
DOCENTE CALIFICADOR

Ing. Mg. Vicente Morales
DOCENTE CALIFICADOR

DEDICATORIA

El esfuerzo invertido en este trabajo que representa un éxito dentro de mi vida, lo dedico a mi familia.

A mi tía Lola Márquez por ayudarme a vencer los obstáculos protegiéndome en cada paso que doy, siempre estará presente en mis pensamientos. A mi madre por ser la fuente de amor, protección y apoyo durante toda mi vida.

A mi padre por ser mi ejemplo, por alentarme a superarme y apoyar cada decisión que he tomado, forjando de la mejor manera mi futuro. A mi hermana por su confianza y paciencia

.

Gabriel Andrés López Márquez

AGRADECIMIENTO

La gratitud es un don de la humanidad, por tal motivo agradezco a mi familia por su constante apoyo y preocupación.

A mi madre por ser mi guía, el motor primordial de mi vida, a mi padre por la confianza depositada en mí y en mis capacidades, a mi hermana por formar parte de mis éxitos y tropiezos.

A los docentes de la facultad de Ingeniería en Sistema, Electrónica e Industrial por brindarme sus conocimientos durante todos mis años de estudio, un especial agradecimiento a mi tutor Ing.Mg. Edgar Freddy Robalino Peña por su aporte en el desarrollo y culminación de mi proyecto de investigación.

Gabriel Andrés López Márquez

ÍNDICE GENERAL

CONTENIDOS	PÁGINAS
PORTADA	i
APROBACIÓN DEL TUTOR	ii
AUTORÍA	iii
DERECHOS DE AUTOR.....	iv
APROBACIÓN DE LA COMISION CALIFICADORA	v
DEDICATORIA	vi
AGRADECIMIENTO	vii
ÍNDICE GENERAL.....	viii
ÍNDICE DE FIGURAS	xi
ÍNDICE DE TABLAS	xiii
RESUMEN	xv
SUMMARY	xvi
ACRÓNIMOS	xvii
GLOSARIO.....	xviii
INTRODUCCIÓN	xx

CAPÍTULO I

EL PROBLEMA

1.1. Tema	2
1.2. Planteamiento del Problema	2
1.3. Delimitación	3
1.3.1 Delimitación de Contenido	3
1.3.2 Delimitación Espacial	4
1.3.3 Delimitación Temporal	4
1.4. Justificación	4
1.5. Objetivos	5
1.5.1 Objetivo General	5
1.5.2 Objetivos Específicos	6

CAPÍTULO II

MARCO TEÓRICO

2.1. Antecedentes Investigativos	7
--	---

2.2. Fundamentación Teórica	9
2.2.1 Sistema Inteligente	9
2.2.2 Introducción al Procesado Digital de Imágenes (PDI)	12
2.2.3 Clases principales de PDI	19
2.2.4 Detección de bordes	23
2.2.5 Microcontrolador	25
2.3. Propuesta de Solución	27

**CAPÍTULO III
METODOLOGÍA**

3.1. Modalidad de la Investigación	28
3.2. Población y Muestra	28
3.3. Recolección de Información	28
3.4. Procesamiento y Análisis de datos	29
3.5. Desarrollo del proyecto	29

**CAPÍTULO IV
DESARROLLO DE LA PROPUESTA**

4.1. Definición del sistema inteligente y reconocimiento de patrones	31
4.1.1. Sistema Inteligente	31
4.1.2. Reconocimiento de Patrones	35
4.2. Establecimiento de características y elementos del sistema	37
4.2.1. Elementos del Sistema Inteligente	37
4.2.2. Características del sistema inteligente	38
4.3. Análisis de requerimientos, estudio de herramientas y componentes	46
4.3.1. Software	46
4.3.2. Hardware	50
4.4. Diseño de los diagramas y esquemas del Sistema Inteligente	61
4.5. Especificación de variables y algoritmos de procesamiento	64
4.6. Arquitectura del Software de seguridad	71
4.7. Programación del software de seguridad	73
4.8. Pruebas de funcionamiento	74
4.8.1. Interface de usuario y visualización	83
4.9. Estudio Económico	86

4.9.1 Descripción General	86
4.9.2 Precios Unitarios	88
4.9.3 Sumas Parciales	89
4.9.4 Presupuesto General	91

CAPÍTULO V

CONCLUSIONES Y RECOMENDACIONES

5.1. Conclusiones	93
5.2. Recomendaciones	94
BIBLIOGRAFÍA	96
ANEXOS	100

ÍNDICE DE FIGURAS

FIGURA	PÁGINAS
Figura 2.1: Elementos de un sistema inteligente	10
Figura 2.2: Tipos de imagenes	14
Figura 2.3: Espectro electromagnético	16
Figura 2.4: Escáner PET del cerebro humano	16
Figura 2.5: Cerebro de un ratón visto en un microscopio de fluorescencia	17
Figura 2.6: Detección de vegetación con imagen infrarroja	17
Figura 2.7: Montañas en el Tíbet vistas por un radar topográfico	18
Figura 2.8: Detección de emisiones de gases golfo de México con sonar	18
Figura 2.9: Circuito integrado ampliado 2000 veces con SEM	19
Figura 2.10: Resultado del operador Sobel	23
Figura 2.11: Resultado del opetador Canny	24
Figura 4.1: Sistema de monitoreo Continuo	33
Figura 4.2: Sistema de monitoreo Periódico	34
Figura 4.3: Elementos del sistema de reconocimiento de patrones	37
Figura 4.4: Elementos de un sistema inteligente	37
Figura 4.5: Tipos de Iluminacion	39
Figura 4.6: Tipos de retroiluminacion	40
Figura 4.7: Diagrama del sistema inteligente.	62
Figura 4.8: Diagrama de flujo del sistema inteligente.....	64
Figura 4.9: Arquitectura del sistema inteligente propuesto	72
Figura 4.10: Arquitectura del sistema de reconocimiento de patrones	72
Figura 4.11: Prueba durante el dia aplicada en exteriores	78
Figura 4.12: Prueba durante el dia aplicada en exteriores	78

Figura 4.13: Prueba durante el dia aplicada en exteriores	79
Figura 4.14: Prueba durante el dia aplicada en interiores.....	79
Figura 4.15: Prueba durante el dia aplicada en interiores.....	80
Figura 4.16: Prueba durante la noche aplicada en áreas de almacenamiento.....	81
Figura 4.17: Prueba durante la noche aplicada en áreas de almacenamiento.....	81
Figura 4.18: Prueba durante la noche aplicada en áreas de almacenamiento.....	82
Figura 4.19: Prueba durante la noche aplicada en áreas de almacenamiento.....	82
Figura 4.20: Prueba durante la noche aplicada en áreas de almacenamiento.....	83
Figura 4.21: Entorno grafico del sistema SISREP	83
Figura 4.22: Sub-ítem opción reconocer video del disco duro.....	84
Figura 4.23: Sub-ítem opción reconocer cámara existente.....	84
Figura 4.24: Captura del video	85
Figura 4.25: Reconocimiento del patrón y contorno corporal del individuo.....	85

ÍNDICE DE TABLAS

TABLA	PÁGINAS
Tabla 2.1: Tipos de imagenes	14
Tabla 4.1: Características del sistema inteligente propuesto	32
Tabla 4.2: Arquitectura de sistemas inteligentes de vision artificial	35
Tabla 4.3: Reconocimiento de patrones	36
Tabla 4.4: Tipos de iluminacion.....	40
Tabla 4.5: Ventajas y desventajas de los sistemas inteligentes con iluminacion	41
Tabla 4.6: Tipos de tecnologias de camaras	43
Tabla 4.7: Tipos de PID.....	45
Tabla 4.8: Software para vision artificial	47
Tabla 4.9: Comparativa entre Matlab y OpenCV	48
Tabla 4.10: Características de Matlab	50
Tabla 4.11: Tipo de camaras web	53
Tabla 4.12: Carcterísticas de la camara.	54
Tabla 4.13: Características del computador	55
Tabla 4.14: Co,parativa entre los microcontroladores	56
Tabla 4.15: Comparativa de Alertas Automaticas.....	59
Tabla 4.16: Características del modulo GSM	60
Tabla 4.17: Tabla de variables en MATLAB	67
Tabla 4.18: Prueba módulo único	75
Tabla 4.19: Prueba grupo de módulos	75
Tabla 4.20: Prueba sistema completo	76
Tabla 4.21: Verificación basada en la operatividad	77
Tabla 4.22: Hardware	86

Tabla 4.23: Software	87
Tabla 4.24: Herramientas	87
Tabla 4.25: Mano de obra directa.....	87
Tabla 4.26: Hardware - Precio Unitario	88
Tabla 4.27: Software - Precio Unitario.....	88
Tabla 4.28: Herramientas - Precio Unitario	89
Tabla 4.29: Mano de obra directa - Precio Unitario	89
Tabla 4.30: Hardware - Suma Parcial	90
Tabla 4.31: Software - Suma Parcial	90
Tabla 4.32: Herramientas - Suma Parcial	91
Tabla 4.33: Mano de obra directa - Suma Parcial	91
Tabla 4.34: Presupuesto General	92

RESUMEN

En la actualidad, los sistemas de seguridad mediante video vigilancia poseen al operador como parte considerable en la seguridad, teniendo como principal inconveniente el índice de atención visual que poseen las personas siendo este no superior a 45 minutos, perdiendo eficiencia en el monitoreo a largo plazo. Para ello se brindó como solución un sistema inteligente de video vigilancia, facilitando de esta manera la obtención de escenas de relevancia mediante el tratamiento de imágenes, teniendo como objetivo principal la detección de intrusos.

El presente proyecto tiene como enfoque el reconocimiento de patrones de una persona en las áreas de almacenamiento de las PYMES, para conseguirlo se hizo hincapié en la detección de bordes y desarrollo de algoritmos de procesamiento en tiempo real, consiguiendo así un sistema robusto con eficiencia en la detección de individuos. Dentro de los sistemas inteligentes se localiza la rama de la visión artificial la cual es una disciplina que se encuentra en pleno auge debido a lo cual está alcanzando un gran crecimiento, gracias a esto encontramos varias plataformas en las que se puede desarrollar dichos sistemas, la plataforma empleada en el presente proyecto de investigación fue Matlab, donde se creó una interfaz gráfica de fácil uso y comprensión para el usuario sobre el funcionamiento del sistema inteligente de video vigilancia, teniendo en cuenta que es necesario proporcionar al sistema la capacidad computacional suficiente para el procesamiento de imágenes en tiempo real.

SUMMARY

Currently, security systems using video surveillance have the operator and considerable part in security, with the main drawback rate visual attention that people have and this no more than 45 minutes, losing efficiency in the long-term monitoring. To this solution was provided as an intelligent video surveillance system, thereby facilitating obtaining relevant scenes through image processing, with the main objective intrusion detection.

This project is focusing pattern recognition of a person in storage areas of PYMES, to get stressed edge detection and development of processing algorithms in real time, thus achieving a robust system efficiency detecting individuals. Within intelligent systems branch of artificial vision which is a discipline that is booming because of which is achieving great growth, thanks to this are several platforms that can develop such systems, is located the platform used in this research project was Matlab, where a graphical interface easy to use and understand for the user on the operation of intelligent video surveillance system was established, taking into account the need to provide the system with sufficient computing capacity for image processing in real time..

ACRÓNIMOS

A/D	Análogo/Digital
CCTV	Closed TV Circuit /Circuito Cerrado de Televisión
CMOS	Complementary Metal Oxide Semiconductor /Semiconductor de Oxido de Metal Complementario
CCD	Charge Coupled Device /Dispositivo de Cargas Acopladas
D/A	Digital/Análogo
GSM	Global System for Mobile communications / Sistema global para comunicaciones móviles
GUI	Graphical User Interface /Interfaz Gráfica de Usuario
GUIDE	Interfase gráfica de usuario del entorno de desarrollo
HD	High Definition / Alta Definición
IA	Artificial Intelligence /Inteligencia Artificial
IPT	Image Processing Toolbox / Herramientas para el procesamiento de imágenes
JPEG	Join Photograph Expert Group / Grupo Experto de fotografía unido
MATLAB	MATrix LABoratory / Laboratorio de Matrices
PDI	Procesamiento digital de imágenes
PDU	Protocol Data Unit / Protocolo de información
PYMES	Pequeñas y Medianas Empresas
SDK	Software Development Kit / Herramientas de desarrollo de software
SISREP	Sistema Inteligente de Reconocimiento de Patrones

GLOSARIO DE TERMINOS

ALGORITMO: Es un conjunto prescrito de instrucciones o reglas bien definidas, ordenadas y finitas que permite realizar una actividad mediante pasos sucesivos que no generen dudas a quien deba realizar dicha actividad.

ARDUINO: Plataforma de hardware libre, basada en una placa con un microcontrolador y un entorno de desarrollo, diseñada para facilitar el uso de la electrónica en proyectos multidisciplinarios.

CONVOLUCIÓN: Operador matemático que transforma dos funciones f y g en una tercera función que en cierto sentido representa la magnitud en la que se superponen f y una versión trasladada e inversa de g .

INTERFAZ: Dispositivo capaz de transformar las señales generadas por un aparato en señales comprensibles por otro.

MICROCONTROLADOR: Circuito integrado programable, capaz de ejecutar las órdenes grabadas en su memoria. Está compuesto de varios bloques funcionales, los cuales cumplen una tarea específica.

MONOCROMÁTICA: Tiene un solo color que está encendido o apagado o también que tiene tonos intermedios, por lo que tiene la misma ambigüedad que blanco y negro. Para una imagen el término monocromo es esencialmente blanco y negro, pero a veces se prefiere el término monocromo para indicar que puede ser en realidad (blanco y verde), (verde y negro), etc.

PIXEL: Un píxel es la menor unidad en color que forma parte de una imagen digital, ya sea una fotografía, un fotograma de vídeo o un gráfico. Las imágenes se forman como una sucesión de píxeles. La sucesión marca la coherencia de la información presentada, siendo su conjunto una matriz coherente de información para el uso digital.

SEGMENTACIÓN: La segmentación en el campo de la visión artificial es el proceso de dividir una imagen digital en varias partes (grupos de píxeles) u objetos. El objetivo de la segmentación es simplificar y/o cambiar la representación de una imagen en otra

más significativa y más fácil de analizar. La segmentación se usa tanto para localizar objetos como para encontrar los límites de estos dentro de una imagen.

INTRODUCCIÓN

En el presente trabajo de investigación se realizó el desarrollo de un sistema inteligente de video vigilancia el cual es capaz mediante la detección de patrones de identificar la presencia de individuos en las áreas de almacenamiento, en caso de tratarse con la presencia de intrusos en el área se alertara al operador de manera inmediata de su aparición, para un mejor entendimiento, se lo ha dividido en los siguientes capítulos:

En el CAPÍTULO I se argumenta el problema, identifica el problema en las áreas de almacenamiento, la delimitación, la justificación de la implementación de un prototipo y los objetivos del estudio.

En el CAPÍTULO II se presentan antecedentes investigativos sobre trabajos relacionados con el tema propuesto, también se detalla la fundamentación teórica y la propuesta solución.

En el CAPÍTULO III se presenta la modalidad de investigación y los pasos utilizados para el desarrollo del proyecto.

En el CAPÍTULO IV se presenta el desarrollo de la propuesta, el análisis de requisitos, el diseño del software, la codificación, el diseño del sistema inteligente, arquitectura del sistema y características del sistema; fehacientes a la implementación del Sistema inteligente de reconocimiento de patrones con visión artificial para la alerta automática de intrusos.

Finalmente en el CAPÍTULO V se redactan las conclusiones y recomendaciones; al término del proyecto de investigación se incorpora la bibliografía y los anexos.

CAPÍTULO I

EL PROBLEMA

1.1. Tema

“SISTEMA INTELIGENTE DE RECONOCIMIENTO DE PATRONES CON VISIÓN ARTIFICIAL PARA LA ALERTA AUTOMÁTICA DE INTRUSOS EN LAS ÁREAS DE ALMACENAMIENTO DE LAS PYMES”.

1.2. Planteamiento del Problema

En la actualidad existe dependencia entre el usuario y los dispositivos electrónicos, en donde para su manejo siempre ha sido necesario que el usuario realice un contacto físico con el dispositivo para que este funcione, hoy en día existen tecnologías que abren una puerta hacia la innovación y la creatividad, las cuales nos permiten una gama de herramientas para desarrollar diversos sistemas siendo la visión artificial un campo de investigación muy activo en donde todavía se considera una tarea muy difícil al reconocimiento de patrones a causa de la alta variabilidad en la apariencia del contorno humano[1].

La seguridad es un estamento muy importante para las PYMES (pequeñas y medianas empresas), así como también para las grandes empresas y las multinacionales. En la actualidad muchos de estos tipos de empresas poseen CCTV (Circuitos Cerrados de Televisión) para el control y vigilancia de sus instalaciones y pertenencias. Pero dichos tipos de sistemas no cumplen con el requisito de tener Visión Artificial o Procesamiento de Imágenes [3].

A nivel mundial, en especial en los países desarrollados en el aspecto tecnológico; los sistemas de seguridad a través de cámaras han trascendido al uso de cámaras infrarrojas

y a la Visión Artificial, es decir; una habitación es vigilada continuamente, incluso en la noche. Cuando sucede un robo en la oscuridad, las cámaras infrarrojas pueden detectar bordes o patrones y extraer rasgos, por medio de un software de computadora; con el cual se reconoce la silueta de una persona; se graba su rostro y se alerta en forma silenciosa al propietario del local, la presencia de un intruso.

En América Latina, la venta de sistemas de visión artificial creció en un 6% en el segundo semestre de 2013, los cuales son empleados en su mayoría en aplicaciones de control de calidad y control de procesos, siendo orientados pocos sistemas a la vigilancia, existiendo una gran concentración de uso en países como Chile y Brasil [2].

Ecuador avanza en el desarrollo tecnológico, debido al emprendimiento de profesionales en el área de la informática, la electrónica, la mecánica y las telecomunicaciones; sobre todo por la influencia de trabajos de graduación, proyectos de investigación o tesis de grado; que impulsan las Universidades, Escuelas Politécnicas y Centros de Educación Superior. En el país, un sistema de seguridad con Visión Artificial; es un tema, que poco o casi nada se lo ha puesto en práctica, se lo ha diseñado o implementado.

El Ecuador aún opera su seguridad con los CCTV, y es por esto que muchos locales y empresas aún siguen siendo robadas; domicilios, oficinas y hasta bodegas de almacenamiento de productos, son blanco fácil para la delincuencia [4].

1.3. Delimitación

En el presente numeral se describe la delimitación de contenido, espacial y temporal.

1.3.1 Delimitación de Contenido

Área académica:	Comunicaciones
Línea de investigación:	Tecnologías de Comunicación.
Sublínea de investigación:	Procesamiento digital de señales e imágenes.

1.3.2 Delimitación Espacial

El proyecto de investigación se realizó como trabajo estructurado de manera independiente; para la Facultad de Ingeniería en Sistemas, Electrónica e Industrial, de la Universidad Técnica de Ambato, en la obtención del título de Ingeniero en Electrónica y Comunicaciones.

1.3.3 Delimitación Temporal

La presente investigación se ejecutó en seis meses, a partir de su aprobación por el Honorable Consejo Directivo de la Facultad de Ingeniería en Sistemas, Electrónica e Industrial.

1.4. Justificación

La Visión Artificial es una herramienta útil para evitar las pérdidas económicas y materiales al prevenir de la presencia de individuos en lugares donde se pueden efectuar robos, al tener cámaras web y otras cámaras de seguridad controladas por un software que pueda procesar imágenes y extraer rasgos o patrones característicos de una imagen; se puede prevenir el cometimiento de robos al llevar la vigilancia a un entorno inteligente; propio de los algoritmos de Visión Artificial.

Existen programas de computadora como MATLAB, LABView, OpenCV, NeuroCHECK, MVIAT, APHELION que poseen algoritmos de procesamiento de imágenes; además existen módulos como Arduino que permiten establecer comunicación entre los sistemas inteligentes y el móvil; al combinar la tecnología de Visión Artificial que proporciona MATLAB con las comunicaciones, la seguridad que otorga una cámara web puede trascender a un mejor nivel.

Los beneficiarios directos del trabajo de investigación son los empresarios de las PYMES y las personas encargadas de la vigilancia, ya que controlan de una manera más eficiente las distintas áreas de almacenamiento. Adicionalmente, si el software de seguridad se decidiera implementar en un domicilio, los beneficiarios serían mayores y la sociedad ambateña tendría una herramienta de seguridad para la vigilancia de sus hogares.

Los beneficiarios indirectos fueron los estudiantes de la carrera de Ingeniería en Electrónica y Comunicaciones; de la Facultad de Ingeniería en Sistemas, Electrónica e Industrial, de la Universidad Técnica de Ambato; que dispusieron de este proyecto como guía para trabajos futuros, investigaciones o en agrado a su conocimiento.

El desarrollo de este trabajo de investigación se fundamentó en la disposición de suficiente fuente bibliográfica. Procedió con la ayuda de un tutor y fue apoyado por docentes de la FISEI. Además, contó con la existencia de una plataforma informática eficiente en algoritmos de procesamiento de imágenes como es MATLAB; y su herramienta GUIDE para la creación de interfaces gráficas de usuario (GUI) del tipo técnicos.

La implementación de un sistema inteligente de reconocimiento de patrones con visión artificial para la alerta automática de intrusos hacia el móvil; fue factible implementarlo, en virtud al avance tecnológico en el software MATLAB, las telecomunicaciones, el Ancho de Banda y otras categorías técnicas que facilitaron la consecución de este trabajo de investigación.

Vincular la tecnología de Visión Artificial con los sistemas de seguridad y vigilancia, para buscar el bienestar de la sociedad ambateña, justifico el desarrollo de este trabajo de investigación; ya que promovió el uso de la técnica y los conocimientos para buscar la solución a un problema real, que causan pérdidas económicas desde hace mucho tiempo a las empresas que han sido víctimas de robo.

1.5. Objetivos

Este apartado del Trabajo de Graduación redacta el objetivo general y los objetivos específicos, parámetros sistemáticos que matizan el trabajo de investigación; los objetivos se aclaran con el desarrollo de la propuesta.

1.5.1 Objetivo General

- Implementar un sistema inteligente de reconocimiento de patrones con visión artificial para la alerta automática de intrusos en las áreas de almacenamiento de las PYMES.

1.5.2 Objetivos Específicos

- Determinar las características del sistema inteligente de reconocimiento de patrones con visión artificial.
- Establecer el método de gestión de alerta automática de intrusos.
- Desarrollar un prototipo de un sistema inteligente de reconocimiento de patrones con visión artificial que permita la alerta automática de intrusos en las áreas de almacenamiento de las PYMES

CAPÍTULO II

MARCO TEÓRICO

2.1. Antecedentes Investigativos

Al realizar una investigación bibliográfica en repositorios digitales se encontraron los siguientes temas afines:

En la Universidad Autónoma de Madrid, Facultad de Ingeniería de Telecomunicaciones en el año 2015, el Ingeniero Jaime Mateo Herrero con el tema: “Preservación de privacidad de personas en video seguridad” empleando la plataforma Matlab ;determina la existencia de numerosas técnicas que permiten ocultar los rasgos identificativos de las personas en secuencias de video, así como la característica fundamental que requiere una técnica de preservación de privacidad es la reversibilidad; dado que resulta poco práctico ocultar la identidad de una persona si luego es imposible recuperar esa imagen original. Como en un trabajo futuro se propuso mejorar el algoritmo scrambling[5].

En la Escuela Politécnica Nacional del Ecuador, Facultad de Ingeniería Eléctrica y Electrónica ,en el año 2014 los Ingenieros María Teresa Calderón y Diego Javier Maldonado con el tema:“ Control por visión de un cuadrúpedo utilizando ROS” empleando OpenCV concluyeron; que mediante el uso de recursos y herramienta contenidos en el framework ROS, es posible desarrollar diferentes aplicaciones para el seguimiento de objetos , ya sea por su color o forma básica, pues el framework ROS se caracteriza por apoyar a la reutilización de código para la investigación y desarrollo de la robótica; también recomienda implementar un algoritmo de procesado muy robusto y sensible que permita el desarrollo de aplicaciones en entornos poco controlados[6].

En la Escuela Politécnica del Ejercito seda Latacunga, Facultad de Ingeniería en Electrónica e Instrumentación en el año 2015 los Ingenieros María Gabriela Chávez y Iván Marcelo Tatayo con el tema: “ Desarrollo e implementación de un sistema para el control de un robot móvil para acceso a lugares remotos utilizando tecnología Kinect” empleando la tecnología Kinect y la plataforma Visual Studio 2010 mencionan que por medio de los estudios de los códigos fuente SDK de Kinect, se logró desarrollar una aplicación demostrativa capaz de capturar los movimientos de los brazos y que la estatura optima del usuario para una mejor detección ,es de 1.6m; recomendando una distancia establecida como optima dentro del proyecto de 1.8m a 2m[1].

En la Universidad Autónoma del Estado de Hidalgo, Facultad de Ingeniería en Electrónica y Telecomunicaciones en el año 2014 el Ingeniero Gilberto Sánchez Rodríguez con el tema: “Generador de código CNC para proceso de frenado, empleando visión computarizada” utilizando la plataforma Matlab refiere que para la comunicación entre micro-controladores es significativo indagar en la compatibilidad de velocidad de transmisión y recepción de datos, como en el algoritmo utilizado en el procesamiento de bordes puesto que es exclusivo para piezas rectangulares; como trabajo futuro se propuso mejorar dicho algoritmo para piezas circulares[7].

En la Universidad de Veracruzana, Facultad de Ingeniería en Electrónica y Comunicaciones en el año 2014 los Ingenieros Ricardo Vaquier y José Aguilar con el tema: “Sistema de visión artificial para la detección de cuerpos sólidos en botellas” utilizando la plataforma LABView, resumen que la técnica para la detección de objetos solidos se basa en tres algoritmos que son la escala de grises, conversión a binario y los métodos proporcionados por NI Vision; dicho proyecto fue pensado para brindar un producto simple y flexible[8].

En la Universidad Carlos III de Madrid, Facultad de Ingeniería en Sistemas y Automática en el año 2013 el Ingeniero Álvaro Loras con el tema: “Percepción de la actividad de un conductor mediante cámaras” empleando Lenguaje C++ concluye que para diferenciar entre un brazo y otro se segmento sus centroides por profundidad y filtrado, que el proyecto funciona mejor en ambientes de interior que de exterior ya que esto compromete el cálculo computacional para que el resto de la aplicación funcione correctamente en tiempo real[9].

En la Universidad Autónoma de Madrid, Facultad de Electrónica y Comunicaciones en el año 2014 el Ingeniero Gonzalo Ballesteros Villareal con el tema: “ Verificación de vehículos mediante técnicas de visión artificial” empleando la plataforma Matlab menciona que los sistemas basados en simetrías obtienen los rendimientos más bajos y su tiempo de ejecución es mayor al ofrecido por los bordes, en la base de datos GTI se ofrece un conjunto de datos públicos que permiten analizar comparativamente estrategias de verificación; como trabajo futuro se propuso estudiar a más profundidad los vectores de características de bordes a mayor profundidad[10].

2.2. Fundamentación Teórica

Este numeral del Trabajo de Graduación fundamenta de forma teórica el contenido científico para el desarrollo de la propuesta.

2.2.1. Sistema Inteligente

Se puede definir un sistema inteligente como un programa de computación que cuenta con características y comportamientos similares a los de la inteligencia humana o animal, es decir, que cuenta con la capacidad de decidir por sí mismo qué acciones realizará para alcanzar sus objetivos basándose en sus percepciones, conocimientos y experiencias acumuladas [11].

Un sistema inteligente puede ser considerado completo si cumple con varias características como:

- a) Ser capaz de aprender. Esto implica que el sistema aprenda conceptos a partir de información recibida de los sentidos teniendo en cuenta un recuerdo el cual se mantenga para el aprendizaje, es decir tiene que ser capaz de aprender de sus propias experiencias. Estas se encuentran en la memoria que es el lugar físico de almacenamiento de los conceptos y reglas de actuación, las cuales relacionan la situación y las consecuencias de la acción [11].
- b) Ser autónomo. Es decir, que deberá ser capaz de hacer las cosas por sí mismo (sin embargo puede optar por aceptar ayuda) [11].

- c) Ser capaz de razonar. Es decir, se debe utilizar algún tipo de razonamiento, basado en hechos conocidos y capaces de producir las ideas que más tarde se hacen hechos conocidos [11].
- d) Ser capaz de desarrollar su autoconciencia. Esto no sólo se relaciona con la autonomía, el razonamiento y el aprendizaje, sino también con la consideración de la necesidad de los sentidos. [11].

La figura 2.1 muestra los elementos de un sistema inteligente.

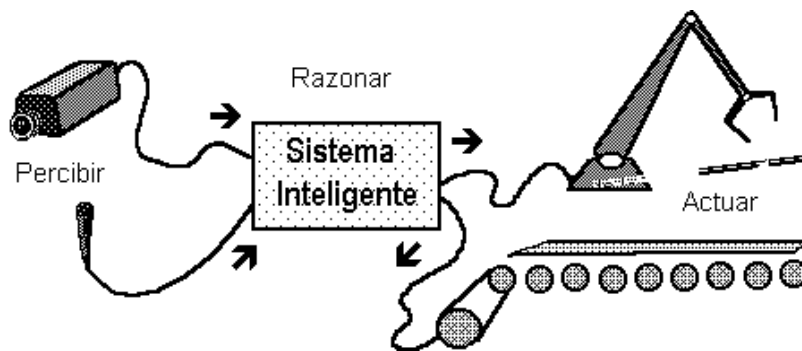


Figura 2.1 Elementos de un sistema inteligente.
Fuente: http://www.depi.itch.edu.mx/apacheco/ai/s_intel.htm

Inteligencia Artificial

La inteligencia artificial, también conocida por sus siglas IA es un área multidisciplinaria que a través de ciencias computacionales estudia la creación y diseño de entidades capaces de resolver cuestiones por sí mismas utilizando como paradigma la inteligencia humana.

Reúne amplios campos, los cuales tienen en común la creación de máquinas capaces de pensar. Adquiriendo datos mediante sensores físicos y sensores mecánicos en máquinas, pulsos eléctricos u ópticos en computadoras, entre otros. Entonces, se puede decir que la IA incluye características humanas tales como el aprendizaje, la adaptación, el razonamiento, la autocorrección y el mejoramiento implícito. Varios ejemplos se encuentran en el área de control de sistemas, planificación automática, la habilidad de responder a diagnósticos y a consultas de los consumidores, reconocimiento de escritura, reconocimiento del habla y reconocimiento de patrones. [12].

Visión Artificial

Es el conjunto de técnicas para obtener informaciones útiles de una o varias imágenes por medio de un dispositivo electrónico programable, para el control de un sistema automático.

Los sistemas de visión artificial completan tareas de inspección con un alto nivel de flexibilidad y repetitividad; nunca se cansan, aburren o distraen y pueden trabajar en ambientes donde los inspectores humanos no podrían realizar la inspección visual.

En un ser humano, los ojos proporcionan al cerebro información del ambiente que lo rodea, basado en experiencias previas con objetos similares, interpreta la asociación entre las imágenes recibidas y es capaz de tomar decisiones. De forma similar, los sistemas de visión artificial ven al objeto por medio de una cámara, interpretan y procesan la imagen a través de una aplicación en un sistema computarizado [13].

Componentes de un Sistema de Visión Artificial:

Los componentes fundamentales de un sistema de visión artificial son: los componentes de adquisición de imágenes, tarjetas capturadoras de video, el software necesario para desarrollar la aplicación y actuadores externos.

- ***Componentes de Adquisición de Imágenes***

Son los componentes encargados de la captura y adquisición de las imágenes. En el mercado se pueden encontrar diferentes componentes que permiten la adquisición de imágenes como por ejemplo: las cámaras de video y las webcam.

La cámara es la encargada de capturar la imagen proyectada en un sensor integrado, para poder transferirla a un sistema electrónico y así poder ser interpretada, almacenada y/o visualizada. Este sistema electrónico puede ser un monitor para visualizar la imagen, un grabador de vídeo para almacenarla o un ordenador para visualizar, almacenar y procesar la imagen [13].

- ***Tarjetas Capturadoras de Video:***

Las tarjetas capturadoras han sido utilizadas desde hace mucho tiempo para poder capturar las imágenes a alta velocidad, el desarrollo de este tipo de placas

de captura ha caminado en paralelo al rápido avance de la velocidad de los ordenadores.

- ***Software de aplicaciones de visión por computadora***

El software es un elemento muy importante en una aplicación de visión por computadora, pues es el encargado de analizar, procesar e identificar características de las imágenes.

- ***Actuadores externos***

Aquí se ubican todos los equipos o dispositivos que reciben la información del computador y cumplen la función de actuadores dentro del proceso o aplicación.

Por lo general los sistemas de visión artificial emplean microcontroladores en sus interfaces siendo estos un circuito integrado programable, capaz de ejecutar las órdenes grabadas en su memoria. Está compuesto de varios bloques funcionales, los cuales cumplen una tarea específica. Un microcontrolador incluye en su interior las tres principales unidades funcionales de una computadora: unidad central de procesamiento, memoria y periféricos de entrada/salida [13].

2.2.2 Introducción al Procesado Digital de Imágenes (PDI)

El procesado digital de imágenes (PDI) es el procesado, entendiendo éste como el almacenamiento, transmisión y representación de información, de imágenes digitales por medio de una computadora digital.

El término imagen se refiere a una función bidimensional de intensidad de luz $f(x,y)$ donde (x) e (y) denotan las coordenadas espaciales, y el valor de (f) en cualquier punto (x,y) es proporcional a la intensidad de la imagen en ese punto. Una imagen digital puede escribirse como una matriz cuyos índices de fila y columna identifican un punto en la imagen y cuyo valor coincide con el nivel de intensidad de luz en ese punto. Cada elemento del array se corresponde con un elemento en la imagen y se le denomina pixel[14].

Imagen Digital

Una imagen digital es una celda compuesta por elementos llamados píxeles, cada píxel es un espacio en la memoria de la computadora donde se almacena un número y este número representa la definición de color y brillo de una parte de la imagen. La imagen digital se puede representar como una función $F(x,y)$ de valores ordenados en una matriz de dos dimensiones, donde el valor de la intensidad se obtiene por el indexado de las coordenadas x e y .

Una imagen puede definirse como una función bidimensional que cuantifica la intensidad de luz siendo el espectro visible es el más común. Es decir, que a cada píxel se le asigna un valor que representa la intensidad de luz incidente en una determinada coordenada del plano XY . Un píxel se designa con el nombre de la matriz y los subíndices que indican la fila y la columna al cual pertenece como por ejemplo: $A(i,j)=127$ indica que el píxel que ocupa la fila i y la columna j en la imagen A tiene un valor de intensidad igual a 127[15].

La tabla 2.1 muestra las diferentes imágenes utilizadas para el PDI detalladas a continuación con sus características que las distinguen unas de otras:

Tabla 2.1 Tipos de imágenes

TIPOS DE IMÁGENES DIGITALES			
RGB(RED- GREEN- BLUE)	INDEXADAS	ESCALA DE GRISES	BINARIAS
Utilizan tres canales para reproducir los colores en la pantalla. Utilizan 8 bits por canal (8 bits x 3), es decir, 24 bits de color para cada píxel. Reproducen hasta 16,7 millones de colores. Soporta algunos formatos como: JPG, BMP, etc.	Reduce los colores de la imagen a un máximo de 256. Admiten los formatos GIF y PNG-8 y muchas aplicaciones multimedia. Reduce el tamaño de archivo porque elimina la información del color.	Utilizan distintos tonos de gris. En imágenes de 8 bits, puede tener hasta 256 tonos de gris. Cada píxel tiene un valor de brillo comprendido entre 0 (negro) y 255 (blanco).	Tienen una profundidad de color de 1 bit. Utiliza uno de los dos valores de color (blanco o negro) para representar los píxeles de una imagen. Contiene solo valores de 1 y 0.

Fuente: Constante P, “Diseño e implementación de un prototipo de brazo robótico para clasificación de formas”

Elaborado por: Gabriel López

A continuación en la figura 2.2 se puede apreciar un ejemplo de los diferentes tipos de imágenes:



Figura 2.2 (a) RGB (b) Indexada (c) Escala de Grises (d) Binaria

Fuente: <http://www.grimaldos.es/cursos/imgdig/tipos.html>

El interés en el procesado digital de imágenes se basa esencialmente en dos aspectos: la mejora de la información contenida en una imagen para la interpretación humana y el tratamiento de los datos de una escena para favorecer la percepción autónoma por parte de una máquina.

Debido al amplio rango de tipos de imágenes empleadas en el PDI, no existe un límite claro respecto dónde se encuentra la línea divisoria entre el PDI y otras áreas afines, como el análisis de imágenes o la visión por computador, entre otras. El análisis de imágenes se refiere al procesado por cual se extrae información cuantitativa de la imagen y en donde el resultado del análisis es siempre una tabla de datos, una gráfica o cualquier representación de los datos numéricos. El procesado de imágenes, sin embargo, siempre produce otra imagen como resultado de la operación, por lo que por lo general se pretende mejorar la calidad de una imagen para poder apreciar mejor determinados detalles. La visión por computador o visión artificial es un subcampo de la inteligencia artificial cuyo propósito es programar un computador para que “entienda” una escena o las características de una imagen, emulando la visión humana.

Atendiendo a los tipos de procesos implicados en estas disciplinas, se suele hacer una clasificación de tres niveles: nivel bajo (procesado), nivel medio (análisis) y nivel alto (interpretación) [14].

Áreas de aplicación

Además de su aplicación en los programas espaciales, las técnicas de procesado digital de imagen comenzaron a utilizarse en una gran variedad de ámbitos y problemas que compartían la necesidad de mejora de las imágenes para su interpretación y análisis.

Actualmente existe una inmensa gama de áreas donde el PDI se utiliza de manera habitual. Un criterio de clasificación para diferenciar cada una de las áreas implicadas es el de la fuente de las imágenes. La principal fuente de energía de las imágenes es el espectro electromagnético (EM), especialmente la banda de rayos X y la del espectro visible. Otras fuentes son la acústica, la ultrasónica y la electrónica. [16]. La figura 2.3 muestra el espectro electromagnético con sus respectivas longitudes de onda.

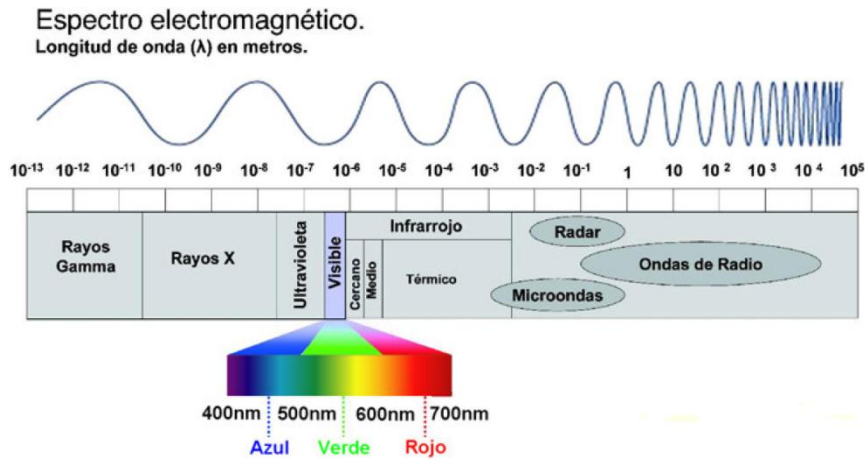


Figura 2.3: Espectro electromagnético

Fuente: Procesamiento Digital de Imágenes: Fundamentos y prácticas con MATLAB.

Empezando por los rayos Gamma, existen aplicaciones que procesan esas fuentes de imagen en medicina nuclear, como en la Tomografía por Emisión de Positrones (PET) donde se le inyecta al paciente un isótopo radioactivo que emite rayos gamma que posteriormente son capturados, y en las observaciones astronómicas para detectar la radiación natural de rayos gamma de los astros. La siguiente banda, la de los rayos X, tiene sus principales aplicaciones en medicina (radiografía, angiografía, Tomografía Axial Computarizada (TAC, etc.), pero también en astronomía e industria, como en el control de calidad de placas de circuito impreso o de alimentos. [16]. La figura 2.4 muestra la tomografía del cerebro humano.

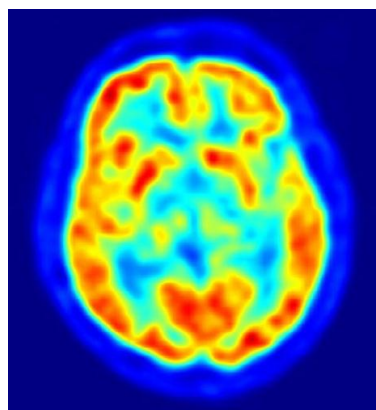


Figura 2.4 Escáner PET del cerebro humano

Fuente: Image processing: principles and applications

También es posible obtener imágenes empleando la banda ultravioleta del espectro electromagnético, teniendo como principales aplicaciones la litografía, la inspección industrial, la microscopía, los láseres, las imágenes biológicas y la astronomía. La figura 2.5 muestra la microscopía del cerebro de un ratón.

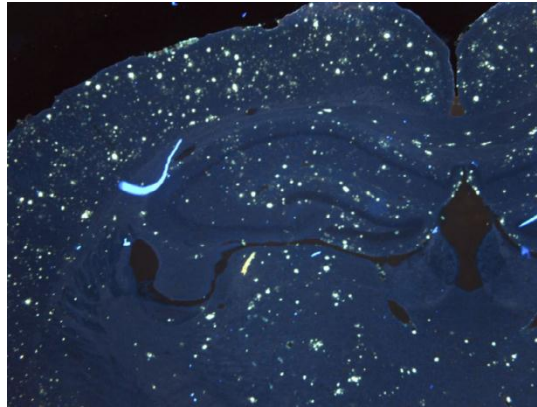


Figura 2.5 Cerebro de un ratón visto en un microscopio de fluorescencia
Fuente: Image processing: principles and applications.

Respecto a las imágenes captadas en la banda visible e infrarroja, estas son por mucho las aplicaciones más numerosas. La banda infrarroja se utiliza habitualmente en conjunto con la banda de espectro visible, por ejemplo en microscopía, astronomía, detección remota (con varias bandas que detectan diferentes detalles del terreno cada una), controles de calidad en la industria y en las aplicaciones de identificación de la policía [17]. La figura 2.6 muestra la imagen infrarroja de la vegetación.



Figura 2.6: Detección de vegetación con imagen infrarroja
Fuente: Image processing: principles and applications.

Las imágenes de la banda de microondas tienen al radar como aplicación principal ya que este puede obtener datos de virtualmente cualquier región y en cualquier momento, sin importar condiciones de clima o de luz ambiental (puede penetrar nubes, vegetación, hielo, arena) utilizando su propia “iluminación” (pulsos de microondas) para obtener una imagen a partir de la energía reflejada. Las imágenes de la banda de radio tienen su mayor aplicación en astronomía y medicina. En medicina las ondas de radio se utilizan principalmente en Imágenes de Resonancia Magnética (MRI) [17]. La figura 2.7 muestra la imagen de las montañas del Tíbet mediante un radar topográfico.

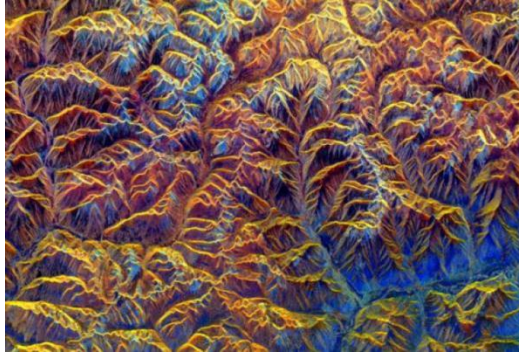


Figura 2.7: Montañas en el Tíbet vistas por un radar topográfico
Fuente: Image processing: principles and applications.

Otras formas de obtener imágenes, en este caso imágenes acústicas, es a través del sonido. Se aplica en exploración geológica (petróleo y minerales), en la industria y en medicina. El ejemplo más notorio es el ultrasonido, con el que se puede realizar un seguimiento al embarazo, o captar imágenes de partes del cuerpo como la tiroides o las capas musculares [17]. La figura 2.8 muestra la detección de gases mediante sonar.

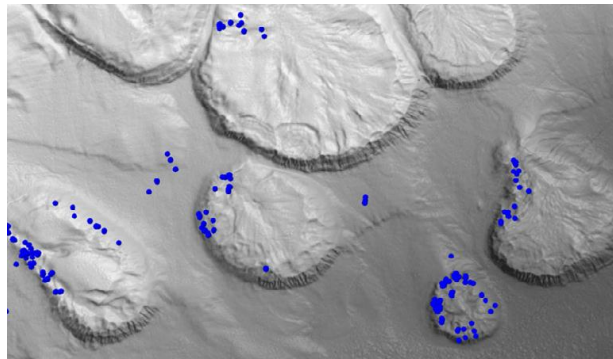


Figura 2.8: Detección de emisiones de gases golfo de México con sonar
Fuente: Image processing: principles and applications.

En las imágenes electrónicas, como las de la microscopía electrónica, la microscopía de escaneo de electrones (SEM) y la microscopía de transmisión de electrones (TEM), se utilizan rayos de electrones enfocados en vez de rayos de luz como los microscopios ópticos. A través de esta técnica es posible captar un filamento de tungsteno o detectar un circuito integrado dañado [17]. La figura 2.9 muestra un circuito integrado.

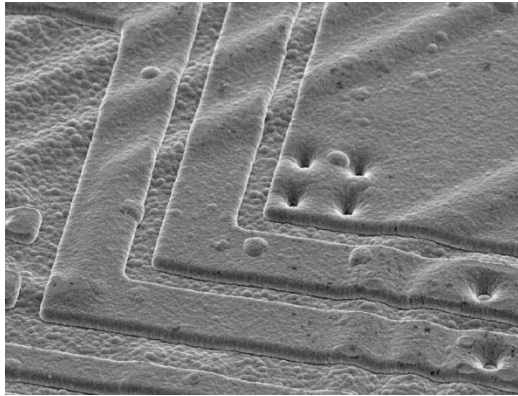


Figura 2.9: Circuito integrado ampliado 2000 veces con SEM

Fuente: Image processing: principles and applications

2.2.3 Clases principales de PDI

Conviene clasificar los diferentes procesos involucrados en el procesado digital de imágenes para tener una visión general más estructurada. Podemos englobar la mayor parte de tareas en tres categorías, cada una de ellas con diferentes algoritmos involucrados:

- **Mejora o realce de la imagen:** Procesado de la imagen para que el resultado sea más apropiado para una aplicación en particular. Tareas habituales: mejora de la nitidez o aclarado de imágenes desenfocadas, eliminación del ruido, mejora del contraste, mejora del brillo, detección de bordes, etc.
- **Restauración de la imagen:** Se puede considerar como revertir el daño ocasionado a la imagen por una causa conocida. Tareas habituales: eliminar el desenfoco por moimiento, eliminar distorsiones ópticas, eliminar interferencia periódica, etc.
- **Segmentación de la imagen:** Subdivide la imagen en partes o aísla ciertos objetos de una imagen. Entre las cuales: búsqueda y selección de formas determinadas en la imagen, máscaras de la imagen, etc [18].

Estas clases suelen aparecer en algún momento del procesado de la imagen aunque no siempre deben hacerlo y dependerá del objetivo y del problema específico en cada caso.

Pasos fundamentales en el PDI

La complejidad de la tarea determinará el número de procesos necesarios para resolver el problema, pero los pasos fundamentales y cotidianos en el PDI son los siguientes:

- a) **Adquisición de la imagen:** Los sensores digitalizan las señales. Generalmente incluye pre-procesado de la imagen (ejemplo: escalar la imagen).
- b) **Mejora de la imagen:** Procesos de manipulación de la imagen para lograr un resultado más adecuado que el original para una aplicación específica (obtener detalles que se no se veían, o simplemente destacar ciertas características de interés).
- c) **Restauración de la imagen:** también mejora la apariencia de la imagen pero a diferencia de la mejora de la imagen que es subjetiva, la restauración es objetiva en el sentido en que las técnicas de restauración tienden a ser modelos probabilísticos o matemáticos de degradación de la imagen (¿Cómo era la imagen antes de estropearse?).
- d) **Procesado del color:** Determinado color puede dar o resaltar más información
- e) **Ondículas (wavelets):** Utilizadas fundamentalmente para representar imágenes en varios grados de resolución. Se utiliza principalmente en compresión.
- f) **Compresión:** Reduce el almacenamiento requerido para guardar una imagen, o el ancho de banda para transmitirla.
- g) **Operaciones morfológicas:** Herramientas para extraer componentes de la imagen útiles para la representación y descripción de formas.
- h) **Segmentación:** Divide una imagen en sus partes constituyentes. Los objetos se extraen o aíslan del resto de la imagen para su posterior análisis. Es una de las tareas más difíciles del PDI.
- i) **Representación y descripción:** casi siempre recibe una imagen segmentada que consta solamente de fronteras o de regiones. Se toman decisiones tales como si la forma obtenida debe ser tratada como una frontera o una región, y se extraen atributos que resultan en información cuantitativa de interés o básicos para diferenciar clases de objetos
- j) **Reconocimiento:** El proceso que asigna una etiqueta (ejemplo: “vehículo”) a un objeto basándose en sus descriptores o bien le da un significado a un grupo de objetos ya reconocidos [18].

El margen de aplicaciones del reconocimiento de patrones es muy amplio, permiten identificar objetos, señales (de voz, imagen, video, comunicaciones, etc.), sin embargo las más importantes están relacionadas con la visión y audición por parte de una máquina. Existen dos formas en las que se puede enfocar el reconocimiento de patrones los cuales son el estadístico(o teoría de la decisión) y el sintáctico (o estructural).

En el enfoque estadístico como el basado en redes neuronales utiliza patrones de los que se extraen propiedades de naturaleza cuantitativa, mientras que el enfoque sintáctico se fundamenta en las relaciones geométricas asociadas a la forma de los objetos y el enfoque basado en la apariencia considera distintas formas de vista de los mismos[19].

Características del Reconocimiento de Patrones:

- Tienen que ser robustos: la extracción debe ser insensible al ruido de captura e iluminación
- Discriminantes (clasificación): las características deben servir para distinguir objetos de clases distintas
- Tienen que poseer determinadas invariancias dependientes de la aplicación:
- Traslación: los valores de las características son independientes de la posición
- Rotación y escalado: identificador de la orientación del objeto y de su tamaño.
- Códigos de cadena: consiste en representar mediante un código incremental la frontera (1-pixel, cerrada) del objeto
- Se escoge un punto inicial y se recorre la frontera en el sentido de las agujas del reloj indicando la dirección que sigue la frontera.
- Sensibilidad al ruido: para aumentarla se sub muestrea la frontera con una malla más basta.
- Invariancia a traslación: es inmediata.
- Invariancia a rotación: es preciso poder determinar un punto inicial que pueda ser detectado robustamente en el objeto rotado.
- Invariancia ha escalado: depende del sub muestreo de la frontera.
- Invariancia a deformaciones: no existe en general.

- Capacidad discriminante: reducida, exige métodos eficientes de clasificación de secuencia [20].

Selección de Variables

Esta etapa se encarga de determinar las características o cualidades de las señales recibidas para así poder preparar los datos para su clasificación. Por lo general esta etapa viene a formar parte de la etapa de clasificación como tal. Entre las selecciones tenemos:

- **Para la Clasificación:** La selección de características relevantes, a partir del conjunto total de características que describen a los objetos, se hace con dos motivos fundamentales: mejorar la clasificación y/o aumentar la velocidad de procesamiento.
- **Para la Representación:** Decidir cuáles características representan mejor a cierto tipo de objetos

El clasificador es la etapa de toma de decisiones en el sistema, su rol es asignar los patrones de clase desconocida a la categoría apropiada. Tenemos tres tipos de clasificaciones las cuales son:

- **Clasificación supervisada:** también es conocida como clasificación con aprendizaje. Se basa en la disponibilidad de áreas de entrenamiento.
- **Clasificación parcialmente supervisada:** también conocida como de aprendizaje parcial. Se basa en que existe una muestra de objetos sólo en algunas áreas de entrenamiento.
- **Clasificación no supervisada:** también conocida como clasificación sin aprendizaje. Se utilizan algoritmos de clasificación automática en los que los individuos más próximos se van agrupando formando clases [21].

2.2.4 Detección de bordes

La detección de bordes es una herramienta fundamental en el procesamiento de imágenes y visión artificial, particularmente en las áreas de detección y extracción de características, los bordes de una imagen digital se pueden definir como transiciones entre dos regiones de niveles de gris significativamente distintos suministrando así una valiosa información sobre las fronteras de los objetos [22].

La detección de bordes es la técnica más frecuentemente utilizada, ya que en la mayoría de las aplicaciones no es útil encontrar puntos y línea.

- **Operador SOBEL**

El operador Sobel calcula el gradiente de la intensidad de una imagen en cada punto (píxel). Para cada punto, este operador da la magnitud del mayor cambio posible, la dirección de éste y el sentido desde oscuro a claro. El resultado muestra cómo de abruptamente o suavemente cambia una imagen en cada punto analizado y, en consecuencia, cuán probable es que éste represente un borde en la imagen y, también, la orientación a la que tiende ese borde.

Matemáticamente, el gradiente de una función de dos variables (en este caso, la función de intensidad de la imagen) para cada punto es un vector bidimensional cuyos componentes están dados por las primeras derivadas de las direcciones verticales y horizontales. Para cada punto de la imagen, el vector gradiente apunta en dirección del incremento máximo posible de la intensidad, y la magnitud del vector gradiente corresponde a la cantidad de cambio de la intensidad en esa dirección.

Lo dicho en los párrafos anteriores implica que el resultado de aplicar el operador Sobel sobre una región con intensidad de imagen constante es un vector cero, y el resultado de aplicarlo en un punto sobre un borde es un vector que cruza el borde (perpendicular) cuyo sentido es de los puntos más oscuros a los más claros.

Los operadores de Sobel son rápidos y efectivos. Sin embargo, no proporcionan el valor real de gradiente, sino una imagen sobre la que se pueden realizar cálculos referentes al contorno, este método da relieve al contraste (marca bordes) entre regiones homogéneas, al mismo tiempo que produce un adelgazamiento [22].

A continuación en la figura 2.10 se puede apreciar un ejemplo del operador Sobel:



Figura 2.10 Resultado del operador Sobel
Fuente: CARDENAS F, M, Rosendo O. “Automatización de un sistema de centrado”

- **Operador CANNY**

Es considerado por muchos como el mejor método existente para la detección de bordes, este algoritmo fue diseñado con el objetivo de:

- Minimizar la probabilidad de múltiples respuestas ante un borde único
- Minimizar la probabilidad de dejar bordes sin detectar
- Minimizar la distancia entre el borde encontrado y el borde verdadero

El proceso de detección de bordes usando el operador Canny sigue los siguientes procedimientos:

- Suavizar la imagen por medio de un filtro gaussiano, con el objetivo de minimizar el ruido de la imagen
- Aplicar una derivada de primer orden a toda la imagen para poder determinar los bordes y dirección de estos
- Eliminar bordes que no tengan la suficiente fuerza

Para satisfacer estos requisitos Canny utiliza el cálculo de variaciones - una técnica que encuentra la función que optimiza un funcional indicado. La función óptima en el algoritmo de Canny es descrito por la suma de cuatro términos exponenciales, pero se puede aproximar por la primera derivada de una gaussiana [22].

A continuación en la figura 2.11 se puede apreciar un ejemplo del operador Canny:

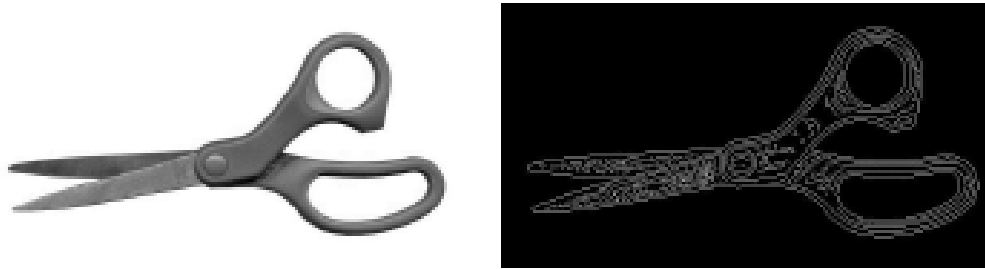


Figura 2.11 Resultado del operador Canny

Fuente: CARDENAS F, M, Rosendo O. “Automatización de un sistema de centrado”

2.2.5 Microcontrolador

Un microcontrolador (abreviado μC , UC o MCU) es un circuito integrado programable, capaz de ejecutar las órdenes grabadas en su memoria. Está compuesto de varios bloques funcionales, los cuales cumplen una tarea específica. Un microcontrolador incluye en su interior las tres principales unidades funcionales de una computadora: unidad central de procesamiento, memoria y periféricos de entrada/salida [23].

Opciones para microcontroladores

Launchpad MSP430

El Texas Instruments MSP430 es muy similar al chip microcontrolador ATmega de Atmel. Las diferencias notables incluyen un precio muy bajo, así como algunas mejoras interesantes para un bajo consumo de energía. La forma más fácil de familiarizarse es con una tarjeta de desarrollo TI LaunchPad.

La principal diferencia entre LaunchPad y Arduino es el costo. Mientras que un Arduino Uno te costará unos \$30 dólares, y un Leonardo \$25, el LaunchPad MSP430 sólo \$10 directamente de TI o de un distribuidor importante (e incluye un cable USB), aunque el chip MSP430G2553, mismo que utiliza la LaunchPad, sólo tiene 14 pines I/O y 16 K de memoria de programa, en comparación con el ATmega328 del Uno (con 32 K y 20 pines I/O), quizás sea todo lo que necesitas para un proyecto en particular. Para una generación de makers acostumbrados al fácil uso del IDE de Arduino, el viejo entorno de desarrollo basado en Eclipse del MSP430 parecía demasiado complicado y difícil de

usar. Sin embargo, la nueva plataforma de prototipado de código abierto Energia ha cambiado todo eso [23].

Picaxe

La casi omnipresencia de la plataforma Arduino ha llevado a muchos sistemas, que no son compatibles con el software, a ser por lo menos compatibles físicamente. El Shield Base PICAXE-28×2 replica la forma de Arduino, estableciendo compatibilidad con los cientos de shields Arduino existentes en el mercado.

Wiring

El gran éxito de Arduino le ha robado un poco de amor y atención a la tarjeta Wiring y su entorno de programación. Probablemente se merece más. La tarjeta está basada en el mismo entorno de desarrollo derivado de Processing del que proviene Arduino — aunque por ahora es una rama un tanto diferente del árbol de la familia — y debe sentirse familiar para cualquiera que esté acostumbrado al IDE de Arduino.

El entorno de programación de Wiring es compatible con cualquier hardware basado en la serie de procesadores AVR de Atmel, no sólo las propias tarjetas Wiring. Y esto incluye la línea de Arduino.

La última placa, la Wiring S, es similar al viejo Arduino Diecimila, pero con un procesador más grande. Al igual que el Shield Base Picaxe, emparejando la tarjeta con un Shield Play Wiring S hace compatible a sus pines con la forma de Arduino, para que puedas volver a utilizar tus shields Arduino con tus tarjetas Wiring[23].

Netduino

La Netduino también adopta la forma de Arduino puedes conectar la mayoría de los shields de Arduino existentes. Pero ahí es donde termina la similitud. Hay muchas tarjetas Netduino disponibles, y a diferencia de las tarjetas que hemos conocido hasta ahora, que han sido en torno a microcontroladores de 8 o 16 bits de un tipo u otro, la Netduino es una placa basada en un ARM Cortex construida alrededor del microcontrolador STMicro STM32Fx de 32 bits.

El sistema operativo de la tarjeta es el NET Micro Framework. Estas tarjetas son programables en C#, directamente desde Microsoft Visual C# Express 2010, y son extremadamente potentes y flexibles. Los desarrolladores de C# en plataformas que no

sean Windows no están del todo abandonados, ya que existe cierto apoyo para OS X y Linux.

Arduino

Arduino es una plataforma de hardware de código abierto, basada en una sencilla placa de circuito impreso que contiene un microcontrolador de la marca “ATMEL” que cuenta con entradas y salidas, analógicas y digitales, en un entorno de desarrollo que está basado en el lenguaje de programación processing. El dispositivo conecta el mundo físico con el mundo virtual, o el mundo analógico con el digital controlando, sensores, alarmas, sistemas de luces, motores, sistemas comunicaciones y actuadores físicos. [23].

2.3. Propuesta de Solución

La implementación de un software de seguridad con reconocimiento de patrones y alerta automática de intrusos hacia el móvil, facilitara la detección de individuos previniendo de esta manera las pérdidas económicas de los robos en las áreas de almacenamiento de las PYMES. Mejorando la seguridad en las empresas e impulsando el uso de Visión Artificial en los sistemas de control y vigilancia con cámaras.

CAPÍTULO III

METODOLOGÍA

3.1. Modalidad de la Investigación

Se realizó una investigación de campo, ya que fue necesario realizar un estudio sistemático de los hechos donde se genera el problema, con el fin de obtener información que fue útil en el desarrollo de la propuesta y apoyo a la fundamentación de los objetivos planteados.

De igual manera se realizó una investigación bibliográfica-documental mediante libros, revistas científicas y publicaciones electrónicas para poder profundizar diferentes enfoques con respecto al tema de investigación, de esta manera se recopiló información valiosa que sirvió como sustento científico del proyecto, ampliando teorías, conceptualizaciones y criterios de diversos autores, acorde a los requerimientos del proyecto..

3.2. Población y Muestra

La presente investigación por su característica no requirió población y muestra.

3.3. Recolección de Información

Para la recolección de información se empleó fuentes bibliográficas, repositorios de publicaciones, además se acudió a diversos lugares de información como archivos, bibliotecas, librerías, videotecas, institutos de investigación, Internet y guía del tutor para el desarrollo de la parte técnica para registrar una descripción concreta y concisa de los avances que se obtuvieron durante el desarrollo del proyecto de investigación. Para ello fue importante tener presente las diversas fuentes que fueron útiles en la tarea de conseguir información para la investigación.

3.4. Procesamiento y Análisis de datos

Para la realización del procesamiento y análisis de datos se consideró las siguientes actividades:

- Recolección de información mediante libros, artículos, tesis, paper, archivos, bibliotecas, librerías, videotecas, institutos de investigación, Internet.
- Análisis de la información, que ayudaron a plantear estrategias para la solución del problema.
- Lectura de artículos relacionados con la investigación propuesta.
- Interpretación de los resultados, es decir la relación entre las variables más importantes que denotaron la investigación del proyecto y generaron una solución a partir del mismo.

3.5. Desarrollo del proyecto

El desarrollo de la implementación de un sistema inteligente de reconocimiento de patrones con visión artificial para la alerta automática de intrusos, tuvo el siguiente lineamiento:

- Definición del sistema inteligente y el reconocimiento de patrones.
- Establecimiento de las características y elementos del sistema.
- Análisis de requerimientos.
- Estudio de las herramientas y componentes.
- Diseño de los diagramas y esquemas de programación.
- Especificación de variables y algoritmos de procesamiento de imágenes.
- Descripción de los métodos alertas automáticas de intrusos.
- Diseño y arquitectura del software de seguridad.
- Programación del software de seguridad con Visión Artificial.
 - Algoritmo de adquisición de imagen y video de una cámara web.
 - Procesamiento de la imagen y video.
 - Algoritmo de extracción de patrones, rasgos o bordes.
 - Algoritmo para el guardado de imágenes.
- Programación de la aplicación con software libre.
 - Módulo SMS.
 - Módulo Llamada.

- Pruebas de funcionamiento del sistema inteligente.
- Depuración de errores del software de seguridad
- Integración del software y componentes físicos en un solo prototipo.
- Documentación del Sistema.
- Desarrollo de un manual de usuario

CAPÍTULO IV

DESARROLLO DE LA PROPUESTA

4.1. Definición del sistema inteligente y el reconocimiento de patrones

El sistema inteligente desarrollado es capaz de determinar los patrones de una persona presente en las áreas de almacenamiento por lo que fue necesario dotar al sistema de la potencia computacional suficiente para procesar las imágenes a la velocidad deseada teniendo como resultado una respuesta instantánea. Al tratarse de zonas de almacenamiento las imágenes no siempre se presentaran en un formato adecuado para su análisis, debido a que no son ambientes controlados por lo que se optó en realizar el preprocesamiento de las imágenes, permitiendo de esta manera una mejora en la calidad de las imágenes.

4.1.1. Sistema Inteligente

El sistema inteligente fue desarrollado para la vigilancia remota en un ambiente no controlado, haciendo uso de imágenes proporcionadas por la cámara permitiendo de esta manera recibir la información del entorno en el que se encuentre. El sistema actúa continuamente teniendo como objetivo la detección de intrusos en las áreas de almacenamiento y para alcanzarlo, debe realizar un análisis de las imágenes proporcionadas por el sistema.

El sistema inteligente al ser orientado a la vigilancia cumple con varias características como son mostradas en la tabla 4.1

Tabla 4.1. Características del sistema inteligente

SISTEMA INTELIGENTE	
CARACTERISTICAS	DESEMPEÑO
Inteligencia	Emplea el reconocimiento de patrones de las personas
Capacidad Sensorial	Recibe comunicaciones del entorno mediante el empleo de la cámara
Objetivo	Detección de intrusos en el área
Reglas de actuación	Interpreta las imágenes adquiridas dentro del sistema
Sistematización	Correlación entre las partes del sistema permitiendo un desempeño optimo
Robustez	Extracción insensible al ruido de captura e iluminación

Fuente: Herrero L., “Visión Artificial”
Elaborado por: Gabriel López

Con las características expuestas anteriormente se pueden emplear tres tipos de arquitecturas que definen los componentes de hardware y software a utilizar para el desarrollo del sistema inteligente. Además con esto se definirá las limitantes y alcances del sistema.

Las tres arquitecturas a identificarse fueron:

a) *Sistema de Monitoreo Continuo*

Son aquellos sistemas de visión que necesitan recibir una respuesta de la inspección visual en tiempo real. Esto implica que la retroalimentación de las imágenes que se analizan es continua.

Los sistemas de monitoreo continuo se utilizan en aplicaciones en las que es crítico que exista una secuencia totalmente continua de imágenes, que únicamente se logra analizando una secuencia de video [13].

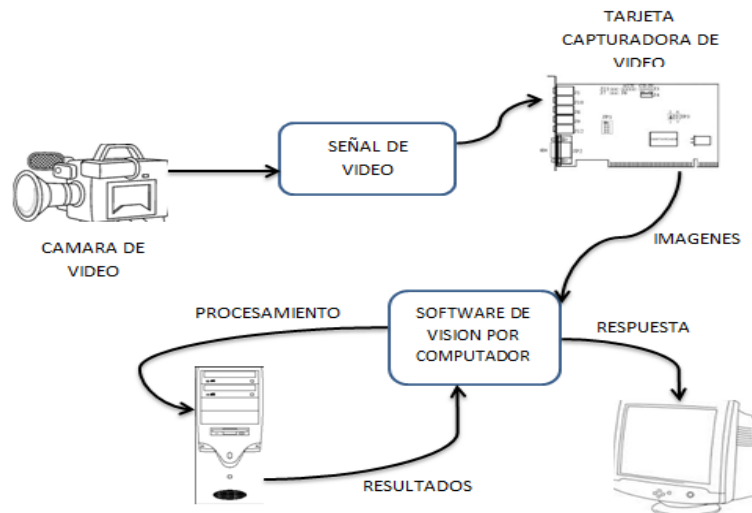


Figura 4.1 Sistema de monitoreo Continuo

Fuente: Ramírez J., “Desarrollo de una aplicación de inspección visual utilizando visión por computadora”

b) Sistema de Monitoreo Periódico

Los sistemas que realizan una inspección utilizando secuencia de imágenes que no son totalmente continuas, reciben el nombre de sistemas de inspección periódica.

Se debe tener presente las limitantes que posee, por no contar con una secuencia completa de imágenes y no tener la capacidad de procesamiento necesaria para una respuesta más rápida [13].

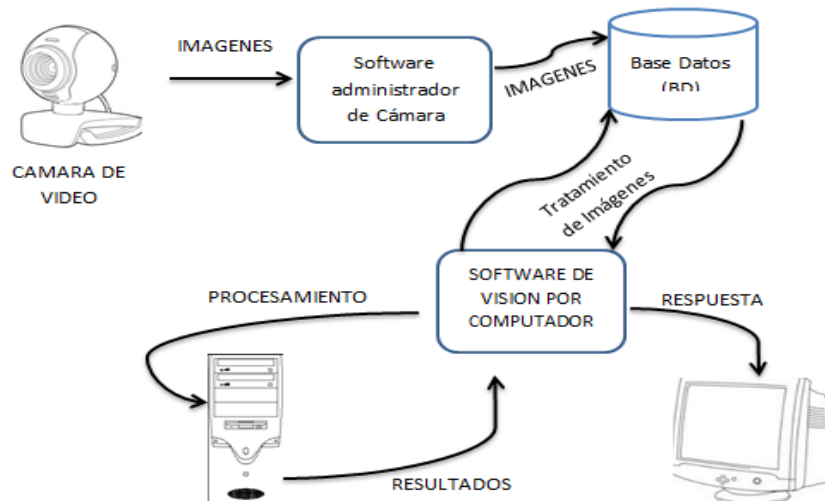


Figura 4.2 Sistema de monitoreo Periódico

Fuente: Ramírez J., “Desarrollo de una aplicación de inspección visual utilizando visión por computadora”

c) *Sistemas Integrales de Visión*

Se denomina sistemas de visión integrales aquellos sistemas que incorporan el hardware y el software para desarrollar aplicaciones de visión artificial en un mismo sistema.

Este tipo de componentes de reducidas dimensiones, están preparados para poder ser instalados en cualquier aplicación de visión con los mínimos requisitos de programación. Un reducido tamaño es una característica fundamental para la implementación en un equipo que necesita la implementación de una cámara de visión artificial.

El desempeño óptimo de este sistema depende de cámaras inteligentes las cuales son sistemas de visión integrados que tienen el aspecto de una cámara, pero que además de incorporar el sensor integran a su vez un procesador, memoria, y puertos de comunicación con el exterior como por ejemplo: Ethernet, I/O, RS-232 [13].

La opción de comunicación Ethernet en este tipo de sistemas permite visualizar y controlar lo que están haciendo todas las cámaras inteligentes que existen en una planta desde cualquier ordenador de las instalaciones.

La tabla 4.2 representa las características en la arquitectura de los sistemas inteligentes de visión artificial.

Tabla 4.2: Arquitectura de los sistemas inteligentes de visión artificial.

CARACTERISTICAS DE ARQUITECTURA DE LOS SISTEMAS INTELIGENTES DE VISION ARTIFICIAL		
SISTEMA DE MONITOREO CONTINUO	SISTEMA DE MONITOREO PERIODICO	SISTEMA INTEGRAL DE VISION
Inspección visual en tiempo real	Inspección visual no es en tiempo real	Inspección visual en tiempo real
Procesamiento de imágenes en tiempo real	Procesamiento de imágenes no es en tiempo real	Procesamiento de imágenes en tiempo real
Retroalimentación de Imágenes continua	Retroalimentación de Imágenes no continua	Retroalimentación de Imágenes continua
No necesita el uso de cámaras inteligentes	No necesita el uso de cámaras inteligentes	Necesita el uso de cámaras inteligentes

Fuente: Ramírez J., “Desarrollo de una aplicación de inspección visual utilizando visión por computadora”

Elaborado por: Gabriel López

Se seleccionó para el sistema inteligente la arquitectura de monitoreo continuo, ya que esta se adapta a las áreas de almacenamiento para la detección de intrusos en tiempo real, teniendo en cuenta que este tipo de arquitectura al no depender de cámaras inteligentes y software evolucionados lo vuelve más accesible y de uso más sencillo.

4.1.2. Reconocimiento de patrones

El reconocimiento de patrones en el sistema propuesto es capaz de determinar la presencia de individuos en las áreas de almacenamiento con el propósito de extraer información, que permita alertar la presencia de estos en dichas áreas. Gracias al reconocimiento de patrones el sistema puede interpretar el mundo.

La aplicación del reconocimiento de patrones dentro del sistema tiene las siguientes características las cuales están detalladas en la tabla 4.3

Tabla 4.3 Reconocimiento de patrones

Reconocimiento de patrones	
Característica	Descripción
Ser robustos	Extracción insensible al ruido
Discriminantes (clasificación)	Distinguen objetos de clases distintas
Códigos de cadena	Representa mediante un código incremental la frontera
Sensibilidad al ruido	Se sub muestrea la frontera
Invariancia a rotación	Determina un punto inicial que pueda ser detectado en el objeto rotado
Invariancia a traslación	Es inmediata
Invariancia a escalado	Depende del sub muestreo de la frontera
Invariancia a deformaciones	No existe en general
Capacidad discriminante	Exige métodos eficientes de clasificación de secuencia

Fuente: Vélez L., “Sistemas de Visión Artificial”

Elaborado por: Gabriel López

Etapas del Reconocimiento de Patrones

En base a las características mencionadas anteriormente se empleó para el reconocimiento de patrones, una cámara que pueda obtener percepciones fieles de la realidad. La información obtenida por parte de la cámara es enviada a un extractor de características cuya función radica en eliminar datos que resulten irrelevantes o redundantes. Terminado el proceso de filtración los datos son enviados al Clasificador en donde los patrones son agrupados por sus propiedades distintivas.

La figura 4.3 muestra los elementos empleados en el sistema de reconocimiento de patrones.

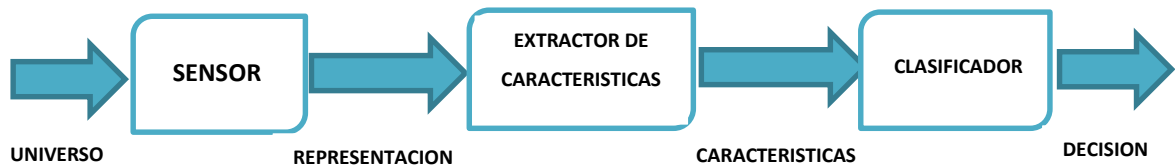


Figura 4.3: Elementos del sistema reconocimiento de patrones
Fuente: Mendoza D, Romero O. “sistema de control para monitorear el acceso vehicular”

La función que desempeña cada una de estas etapas fue primordial ya que gracias a estas se realizaron ajustes en el sistema para la clasificación de las señales de entrada en una de las clases predefinidas. En base al análisis de ciertas características con el empleo de algoritmos para la detección de bordes de las personas se pudo clasificar satisfactoriamente señales de entrada.

Para la extracción de bordes en los individuos se tomó en cuenta los factores como fuentes de ruido, deformaciones, giros bruscos en la captura de imágenes y deformaciones en ellas mediante lo cual se pudo establecer los elementos del sistema inteligente.

4.2. Establecimiento de las características y elementos del sistema

Mediante las características del sistema inteligente expuestas anteriormente se establecieron los siguientes elementos necesarios para un correcto desempeño del sistema propuesto.

4.2.1. Elementos del sistema inteligente

A continuación como se muestra en la figura 4.4 se puede observar los elementos del sistema de visión artificial:

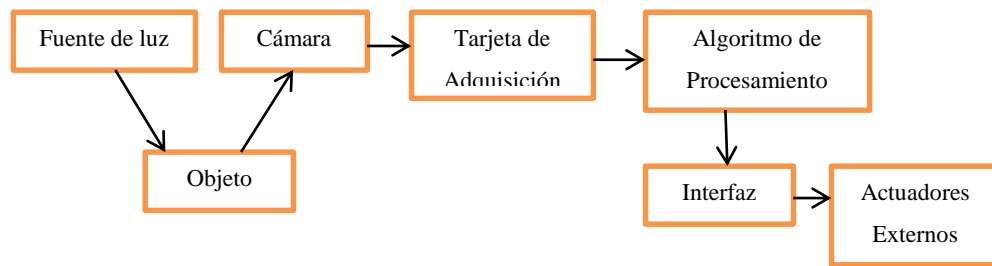


Figura 4.4: Elementos del Sistema Inteligente
Fuente: CARDENAS F, M, Rosendo O. “Automatización de un sistema de centrado”

4.2.2. Características del sistema inteligente

A continuación se detallaran los elementos que se tomaron en cuenta al desarrollar el sistema inteligente.

➤ Fuente de luz

Uno de los aspectos más importantes es la iluminación correcta la cual debe ser uniforme e independiente del entorno, con el objetivo de facilitar la extracción de los rasgos de interés para la aplicación.

Fundamentalmente existen dos tipos de iluminación:

- ***Iluminación frontal***

La luz incide directamente sobre el objeto, en forma vertical, horizontal, oblicua o difusa, este tipo de iluminación permite distinguir claramente los detalles del objeto, así como su forma, color, detalles internos, etc.; que permiten una mejor segmentación, pero también produce fenómenos contraproducentes como son la creación de sombras y reflejos.

El inconveniente de la iluminación oblicua es que crea sombras, lo que interfiere en la detección correcta de la forma de los objetos, en mayor proporción cuando estos son oscuros, debido a esto, siempre se trata de evitar las sombras. Al iluminar objetos brillantes se producen brillos que también son adversos perjudicando al momento de realizar la detección de los contornos y detalle de los objetos [22].

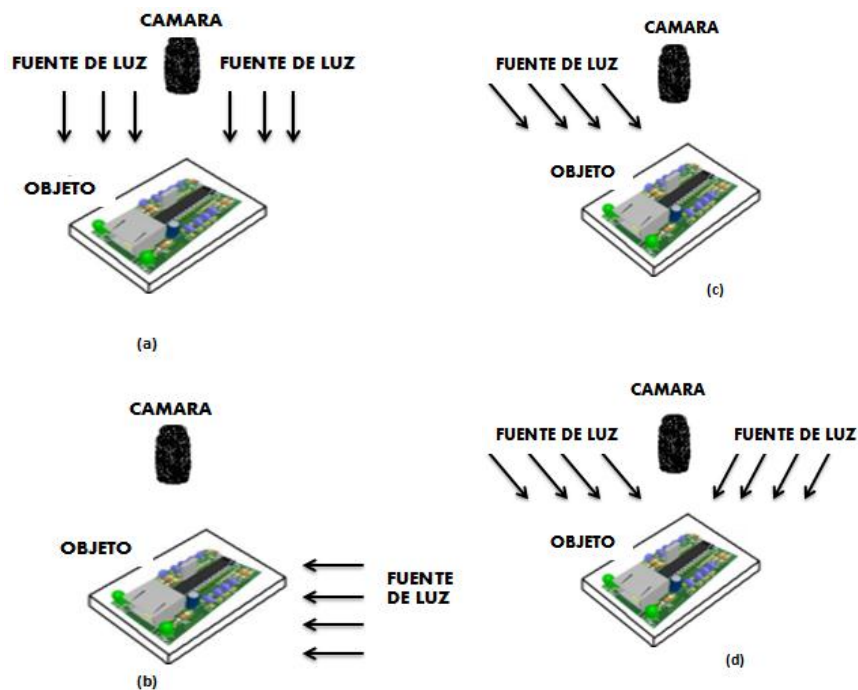


Figura 4.5 (a) Iluminación Vertical (b) Iluminación Horizontal (c) Iluminación Oblicua (d) Iluminación Difusa

Fuente: CARDENAS F, M, Rosendo O. “Automatización de un sistema de centrado”

- **Retroiluminación**

En este caso se ilumina una pantalla. Este tipo de iluminación solo sirve para la detección de contornos.

La retroiluminación se puede realizar de dos formas: colocando el objeto entre la pantalla y la fuente de luz, de forma que lo que la cámara captura es la sombra proyectada sobre la pantalla; o situando el objeto entre la cámara y la pantalla de forma que lo que se ilumina es el fondo de la escena. En este caso la iluminación debe estar perpendicular a la pantalla, con el fin de no deformar la sombra del objeto [22].

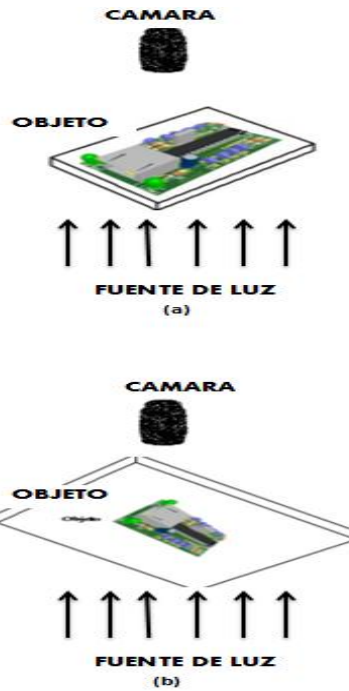


Figura 4.6 (a) Proyección sobre la pantalla (b) Proyección sobre la pantalla
Fuente: CARDENAS F, M, Rosendo O. “Automatización de un sistema de centrado”

La tabla 4.4 muestra las características de los tipos de iluminación existentes en un sistema inteligente

Tabla 4.4 Tipos de iluminación en un sistema inteligente.

TIPOS DE ILUMINACION		
	ILUMINACION FRONTAL	RETROILUMINACION
Incidencia de la luz en el objeto	Vertical. Horizontal, Oblicua o Difusa	El objeto se ubica entre la fuente de luz y la pantalla
Distinción del objeto	Forma, Color, Detalles Internos	Contornos
Sombras en la iluminación ambiental	Si	No

Fuente: Fuente: CARDENAS F, M, Rosendo O. “Automatización de un sistema de centrado”
Elaborado por: Gabriel López

Se optó por la iluminación frontal como fuente de luz para el sistema inteligente propuesto debido a que las áreas de almacenamiento cuentan con este tipo de iluminación. Hay que tener en cuenta que al tratarse de un ambiente no controlado se tuvo que hacer énfasis en el preprocesamiento de las imágenes.

A continuación en la tabla 4.5 se detallara las ventajas y desventajas de los sistemas con y sin iluminación.

Tabla 4.5 Ventajas y desventajas de los sistemas inteligentes con iluminación.

	VENTAJAS	DESVENTAJAS
CON ILUMINACIÓN	<p>Mayor distinción del objeto como forma y contornos</p> <p>Mejor procesamiento de imágenes al tener mayor distinción del medio</p> <p>Presencia mínima de ruido en las imágenes</p> <p>Uso de elementos menos costosos</p>	<p>Al existir demasiada iluminación puede causar la reflexión de la luz en los objetos brillantes</p> <p>Se necesita más enfoque en la resolución de imágenes</p>
SIN ILUMINACIÓN	<p>No se produce sombra en los objetos</p> <p>Programas más robustos en el filtrado de imágenes</p> <p>Entrega de imágenes en grises al emplear tecnología infrarroja</p>	<p>Menor eficiencia en la detección de objetos</p> <p>Mayor énfasis en el preprocesamiento de imágenes</p> <p>Uso de elementos más costosos</p>

Fuente: CARDENAS F, M, Rosendo O. "Automatización de un sistema de centrado"

Elaborado por: Gabriel López

➤ **Objeto**

Al orientar el sistema de visión artificial a la vigilancia en las áreas de almacenamiento los objetos a detectar serán individuos.

➤ **Sensor de Imagen (Cámara)**

El sensor de imagen más común y conocido es la cámara, en la actualidad se puede presenciar en los mercados gran variedad de cámaras, las más regulares poseen características como: costos menores, ahorro en el consumo de energía, resolución adecuada y facilidad para su uso.

Para el procesamiento de imágenes es necesario el empleo de una cámara con buena calidad de imagen, en cuyo interior podemos apreciar dos etapas de construcción básica como son el sensor de imagen y la interfaz digital.

El sensor de imagen es una matriz de cientos y miles de células sensibles a la luz, llamadas pixeles, estos pixeles convierten la intensidad de la luz en un voltaje de salida. La interfaz digital convierte el voltaje de salida del pixel en niveles grises digitales, pone todos los niveles grises juntos para formar una imagen y transferir esta imagen a la computadora.

Actualmente existen dos tecnologías para la fabricación de sensores destinados a las cámaras digitales: Los CCD (Charge Coupled Device o Dispositivo de Cargas Acopladas), que fueron los primeros en aparecer en el mercado, y los más recientes CMOS (Complementary Metal Oxide Semiconductor o Semiconductor de Oxido de Metal Complementario)[24].

La tabla 4.6 muestra las diferencias notables entre ambas tecnologías

Tabla 4.6 Tipos de tecnologías en Cámaras.

DIFERENCIAS	CCD	CMOS
Electrónica de Control (Convertir los datos de cada pixel en datos digitales binarios)	Fuera del dispositivo de captura	Integrada en el dispositivo de captura
Rango Dinámico (Cociente entre el nivel de saturación de los pixeles y el umbral por debajo del cual no captan señal)	Supera a CMOS en un factor de dos	Bajo rango
Sensibilidad (Capacidad que tiene el sensor de generar carga eléctrica por unidad de luz que incide sobre el)	Inferior	Superior(integra elementos amplificadores en cada celda)
Blooming (Cuando un fotodiodo se satura puede afectar a otros fotodiodos próximos a él)	Si se da este efecto	No se da este efecto
Velocidad de obturación	Mayor velocidad	Menor velocidad
Ruido	Menos ruido	Mayor ruido
Consumo eléctrico	Mayor consumo	Menor consumo
Costo	Más costoso	Más económico

Fuente: <http://www.neoteo.com/ccd-vs-cmos>

Se eligió para el sistema inteligente una cámara web con sensor CMOS ya que con poca luz esta se comporta mejor, y a que la velocidad de su sensor es mayor debido a que el procesado se realiza dentro del mismo sensor características suficientes para el desempeño adecuado del sistema inteligente.

➤ **Sistema de adquisición(Tarjeta Gráfica)**

En esta etapa del sistema inteligente se toma la información capturada por el sensor de imagen y la convierte a una señal digital que pueda ser procesada en un computador, pues se aplican los filtros para la extracción de información de las imágenes adquiridas.

Para la elección de la tarjeta de adquisición se tomó en cuenta parámetros de velocidad en la transmisión, velocidad de transferencia en la información, profundidad del pixel y capacidad de captura por disparo.

Se trabajó con la tarjeta gráfica interna de un computador portátil ya que la tarjeta que este posee fue suficiente para cumplir con los requerimientos de velocidad de transmisión, formato de los datos, profundidad del pixel y capacidad de captura por disparo.

➤ **Procesamiento digital de Imágenes (PDI)**

En esta etapa se da el tratamiento de imágenes mediante medios electrónicos, teniendo en cuenta que las imágenes capturadas no provienen únicamente del censado de luz visible para el ojo humano sino que también se puede hacer uso de otros tipos de censados de luz dentro del área del espectro electromagnético[24].

Existen tres tipos de procesamiento:

- ***Procesos de bajo nivel:***

Involucra operaciones primitivas tales como: el procesamiento de imágenes para reducir el ruido, realce de contraste y filtros de enfoque. Un proceso de bajo nivel está caracterizado por el factor de que ambos (estas son entradas y salidas) son imágenes.

- ***Procesos de nivel medio***

Involucra tareas como: segmentación (dividir una imagen en regiones u objetos), descripción de los objetos reduciéndolos a una forma adecuada para el procesamiento en la computadora y clasificación de objetos individuales. Un procesamiento de nivel medio se caracteriza por el factor de que las entradas generalmente son imágenes, pero las salidas son atributos extraídos de estas imágenes (Por ejemplo: bordes, contornos y la identidad de objetos individuales).

- **Proceso de alto nivel**

Implica obtener algún significado de un conjunto de objetos reconocidos en el análisis de imágenes y finalmente realizar las funciones cognitivas asociadas con la vista [24].

La tabla 4.7 nos muestra una comparativa entre los tipos de PDI

Tabla 4.7 Tabla comparativa de los tipos de PDI.

Tipos de Procesamiento			
Características	Nivel Bajo:	Nivel Medio	Nivel Alto
Entrada	Imagen	Imagen	Imagen
Salida	Imagen	Atributos de objetos: bordes, contornos, identidades de objetos individuales	Análisis de la imagen(información, sentido a los objetos)
Proceso	Reducir el ruido, Realce de Contraste, Realce de características	Segmentación , Clasificación de Objetos, Descripción de objetos	Interpretación, Análisis de la imagen, Funciones cognitivas

Fuente: Mendoza D, Romero O. “Sistema de control para monitorear el acceso vehicular”
Elaborado por: Gabriel López

Según las características expuestas en la tabla se optó por trabajar con un procesamiento de nivel medio ya que el sistema inteligente tiene como función reconocer los patrones atributos que se realizaron con la detección de bordes y contornos de los individuos mediante el empleo de los distintos filtros.

➤ **Interfaz**

En esta parte del sistema se comunica a los actuadores externos la decisión tomada por el algoritmo de procesamiento, para la comunicación se empleó una placa Arduino.

➤ **Actuadores Externos**

Envía la orden al sistema operativo para realizar su función correspondiente la cual será la alerta automática.

4.3. Análisis de requerimientos, estudio de las herramientas y componentes

Al ser un proyecto orientado a la vigilancia de eventos que se producen en las áreas de almacenamiento se tomó en cuenta varios parámetros como las características del entorno y las condiciones meteorológicas ya que estas afectan a los resultados del sistema inteligente desarrollado. También los cambios en la iluminación debido al instante del día y a las condiciones meteorológicas provocan la presencia de sombras o viento, dando lugar a una incertidumbre en la percepción de los objetos que debe ser tratada de forma inteligente por el sistema.

Esto llevó al desarrollo de un sistema robusto, que aporta información redundante, de modo que se posibilite la eliminación de la incertidumbre, por lo que se realizó un análisis y estudio de requerimientos en las herramientas para obtener componentes que permitieron un correcto desempeño del sistema.

4.3.1. Software

Dentro del área de procesamiento de imágenes existen algunos programas que permiten el tratamiento de una imagen, entre éstos se tiene: Neuro Check, MVIAT, LabView, Aphelion, Matlab, OpenCV. A continuación en la tabla 4.8 se detallaran las características principales de cada programa para de esta manera poder elegir el que se adapte mejor al sistema inteligente desarrollado:

Tabla 4.8 Software de Visión Artificial

	NEURO CHECK	MVIAT	LabView	Aphelion	Matlab	OpenCV
PROPO SITO	Configuración y operación de sistemas industriales de control de calidad visual	Independiente de propósito general en la adquisición y procesamiento de imágenes	Adquisición de datos, comunicación y control, automatización industrial, visión artificial, robótica y redes	Reconocimiento de formas, la clasificación e interpretación de imágenes	Software matemático incluyendo señales y procesamiento de imágenes	Procesamiento de imágenes
TIPO DE PROGR AMACI ON	No requiere ningún conocimiento de programación de software.	Fácil de usar no requiere ningún conocimiento de programación	Programación gráfica (Lenguaje G)	Lenguaje de macro-comandos compatible con Visual Basic	Lenguaje de programación propio (lenguaje M).	Lenguaje de programación en códigos C y C++

Fuente: Mendoza D, Romero O. “Sistema de control para monitorear el acceso vehicular”[22]

De los programas descritos con anterioridad los más destacados son Matlab y OpenCV, a continuación en la tabla 4.9 se detalla una comparativa entre ambos programas.

Tabla 4.9 Comparativa entre Matlab Y OpenCV

	CARACTERISTICAS	VENTAJAS	DESVENTAJAS
OpenCV	Creado por Intel para el procesamiento de imágenes	<ul style="list-style-type: none"> ✓ Utilizado con herramientas de desarrollo para crear interfaces graficas de usuario ✓ Capacidad ligera de instalación en el disco duro ✓ Entrenamiento de cascada de haarcascade para detección de objetos ✓ Uso de memoria Ram ligera 	<ul style="list-style-type: none"> ✓ Librería de procesamiento de imágenes y visión artificial ✓ Desarrollo de aplicaciones en win32 ✓ Uso de envoltorios para creación de aplicaciones de interfaz; ejemplo: emguCV, AForge.net, OpencvSharp. ✓ Falencias en sistemas operativos de 64 bits ✓ Direccionamiento del path desde el entorno de variable en el sistema operativo
Matlab	Creado por Matworks para visión artificial	<ul style="list-style-type: none"> ✓ Herramienta informática para el procesamiento de imágenes y visión artificial ✓ Desarrollo de aplicaciones de tipo Windows form ✓ Uso de algoritmo haarcascade directamente ✓ Uso de toolbox para el desarrollo de visión artificial 	<ul style="list-style-type: none"> ✓ Capacidad de disco duro nominal en la instalación ✓ Uso de memoria Ram prioritaria

Fuente: <http://docs.opencv.org/3.0-alpha/modules/core/doc/core.html>

Elaborado por: Gabriel López

Para llevar a cabo el proyecto se decidió emplear Matlab ya que en relación con OpenCV es una herramienta robusta en visión artificial y procesamiento de imágenes digitales; al contrario de OpenCV que es una librería que crea aplicaciones win32 y que necesita de un envoltorio para crear aplicaciones tipo Windows form, es decir en Matlab es necesario programación orientada a objetos y programación C++; en OpenCV es necesario programación orientada a objetos, programación orientada a aspectos y C++.

Se utilizó la versión Matlab 2013 debido a su potencia en el desarrollo de este tipo de aplicaciones, integración de funciones, facilidad de desarrollo, conocimientos previos, gran documentación y a la librería dedicada al procesamiento digital de imágenes denominada Image Processing Toolbox.

El Toolbox de adquisición de imágenes se compone de una colección de funciones que permiten, entre otras cosas, adquirir imágenes de diversos dispositivos (desde frame grabbers profesionales hasta USB Webcams) tanto para visualizar videos en vivo como para la adquisición de imágenes mediante triggers o para importar datos hacia el entorno de trabajo de MATLAB. En otras palabras, es la herramienta seleccionada para diseñar la aplicación en tiempo real. El Toolbox de Procesamiento de Imágenes proporciona a MATLAB un conjunto de funciones que amplía las capacidades del producto para realizar desarrollo de aplicaciones y de nuevos algoritmos en el campo del proceso y análisis de imágenes [25].

El entorno matemático y de creación de MATLAB fue ideal para el procesado de imágenes, ya que las imágenes son, al fin y al cabo, matrices. Este toolbox incorpora funciones para:

- Diseño de filtros.
- Mejora y retocado de imágenes.
- Análisis y estadística de imágenes.
- Operaciones morfológicas, geométricas y de color.
- Transformaciones 2D.

En la tabla 4.9 se muestran las diversas características del software Matlab 2013.

Tabla 4.10 Características Matlab 2013.

 <p>The image shows the MATLAB logo in blue serif font at the bottom left. To its right is a 3D surface plot with a color gradient from blue to red, set against a light gray grid background.</p>	<p>Características:</p> <ul style="list-style-type: none">▪ Cálculos intensivos desde un punto de vista numérico.▪ Gráficos y visualización avanzada. Lenguaje de alto nivel basado en vectores, arreglos y matrices.▪ Colección muy útil de funciones de aplicación.
	<p>Ventajas:</p> <ul style="list-style-type: none">▪ Facilidad de programación▪ Gran potencia de cálculo en operaciones con matrices▪ Gran facilidad para obtener resultados gráficos▪ Código abierto disponible para Windows y Linux▪ Ideal para desarrollo teórico de algoritmos y tratamiento de imágenes

Fuente: <http://www.mathworks.com/products/matlab/>

Elaborado por: Gabriel López

4.3.2. Hardware

Para el correcto funcionamiento del sistema inteligente se requirió de los siguientes elementos:

➤ **Iluminación**

Al emplear una iluminación frontal como fuente de luz para el sistema inteligente propuesto en áreas de almacenamiento se pueden utilizar tres fuentes que nos pueden prestar la iluminación deseada para el correcto uso de la aplicación las cuales son:

- **Lámparas incandescentes**

Es la fuente de iluminación más común y consiste en un filamento de tungsteno o halógeno-tungsteno. Su ventaja radica en que existe una gran variedad de potencias y como desventaja, es que se reduce su luminosidad con el tiempo, lo que puede provocar problemas en algunos sistemas de visión.

- **Tubos Fluorescentes**

Más eficaces que las lámparas y suministran una luz más difusa, que es bueno para piezas reflectoras. Existe una gran variedad, tanto en forma como en el tamaño con lo que son ampliamente utilizados.

- **Láseres**

Empleados para una iluminación con luz estructurada, mediante un control adecuado del láser. Tiene el inconveniente de presentar un mal comportamiento frente a superficies que absorben luz [26].

El presente proyecto al ser orientado a la vigilancia de eventos que se producidos en el exterior se empleó la arquitectura de los sistemas integrales de visión artificial. Por lo cual dependerá tanto de la luz del sol o la luna, o la proveniente de las lámparas incandescentes ya que al tratarse de áreas de almacenamiento este tipo de fuentes de iluminación es la más común. Al recurrir a este tipo de iluminación estamos refiriéndonos a una iluminación frontal cuyas características fueron descritas anteriormente.

Si el sistema carece de iluminación el prototipo deberá en su versión 2.0 aplicar algoritmos de conversión de colores a modo de imagen negativa, al ser un software prototipo tiende a recibir mejoras debido al desarrollo de los programas de los sistemas operativos, de los compiladores, de los lenguajes de programación además de la memoria RAM y ROM, tanto en versiones de Matlab, OpenCV, LabView,etc

➤ **Cámara**

Al depender tanto de la luz del exterior como la proporcionada por la fuente de luz en el área se decidió trabajar con una Cámara de categoría CMOS ya que esta cumple con las

características necesarias para un desempeño óptimo en el sistema. Dentro de esta categoría se pudo encontrar cámaras web las que dependerán de la iluminación proporcionada.

También son muy utilizadas en mensajería instantánea y chat como en Skype, Line, Hangouts etc. Por lo general puede transmitir imágenes en vivo, pero también puede capturar imágenes o pequeños videos (dependiendo del programa de la cámara web) que pueden ser grabados y transmitidos por Internet. Dentro de la distinción tradicional de los accesorios de PC, la cámara web es un dispositivo de entrada, ya que por medio de él podemos transmitir imágenes hacia la computadora, y tomarse fotos y editarlas.

Además del uso cotidiano, es frecuente la aplicación de cámaras web de cierta calidad en diferentes áreas de la investigación científica. Por ejemplo, en la astronomía de aficionado, las cámaras web de cierta calidad pueden ser utilizadas para registrar tomas de satélites lejanos y estrellas. Ciertas modificaciones pueden lograr exposiciones prolongadas que permiten obtener imágenes de objetos tenues de cielo profundo como galaxias, nebulosas, etc. En Psicología cognitiva se emplean cámaras web modificadas para realizar pruebas de seguimiento ocular [25].

La tabla 4.11 muestra las cámaras a considerar para nuestro sistema

Tabla 4.11 Tipos de cámaras web para el sistema inteligente propuesto

CAMARA	RESOLUCION	CALIDAD DE FOTOGRAFIAS	CALIDAD DE VIDEO	TIPO DE ENFOQUE	CORRECCION DE ILUMINACION AUTOMATICA
GENIUS FaceCam 321	640 p	2 MP	Regular	Constante	Estándar
GENIUS FaceCam 1020	720 p	4 MP	Buena	Ajustable	Estándar
ETRON 6 LEDCam	640	3 MP	Buena	Ajustable	Estándar
LOGITECH c270	720	3 MP	Buena	Constante Siempre focalizado	Estándar
LOGITECH C310	720	5 MP	Muy Buena	Constante Siempre focalizado	De gama alta

Fuente: <http://www.neoteo.com>
Elaborado por: Gabriel López

Teniendo en cuenta que la cámara debe estar posicionada en un lugar fijo, para de forma poder obtener una imagen de manera clara y concisa la cual se empleara en el reconocimiento de patrones y la cobertura de la cámara debe ser amplia por eso es recomendable ubicarla en la zona más alta de las áreas de almacenamiento para poder tener una cobertura ideal de toda el área.

Se ha procedido a elegir la cámara web Logitech c270 como se muestra en la tabla 4.12 ya que esta debido a sus características tendrá un desempeño eficiente dentro del sistema inteligente propuesto:

Tabla 4.12 Características de la cámara.

CAMARA	CARACTERISTICAS
 A black Logitech HD Webcam C310 is shown against a white and grey checkerboard background. The camera has a lens in the center, a small microphone grille to the right, and the Logitech logo on the right side. The text 'HD 720P' is visible near the lens.	<p>Captura de video HD de alta calidad 1280 x 720 y 3 MP</p> <p>Tecnología Logitech Fluid CrystaL: Es un algoritmo de Logitech que ajusta automáticamente la frecuencia de cuadro, el color y la nitidez para obtener las mejores imágenes estáticas y en movimiento. No importa cuales sean las condiciones de iluminación.</p> <p>Componentes ópticos Carl Zeiss con enfoque automático.</p> <p>Enfoque automático.</p> <p>Entrega los siguientes formatos de video: H.264 / MPEG4 / MJPEG</p> <p>Corrección inteligente de luz automático para una imagen nítida</p> <p>Dimensiones 3,8 x 1,2 x 1,3 pulgadas</p>


Fuente: <http://www.logitech.com/es-es/product/hd-webcam-c310>

Elaborado por: Gabriel López

➤ **Computador**

Para realizar el presente proyecto se empleó una computadora portátil marca Toshiba Satellite como se muestra en la tabla 4.13 la cual cumple con los requisitos necesarios para poder instalar y ejecutar en ella Matlab 2013 a.

Tabla 4.13 Características del computador.

COMPUTADOR	CRACTERISTICAS
 <p style="text-align: center;">Toshiba Satellite</p>	<ul style="list-style-type: none"> ▪ SISTEMA OPERATIVO: Windows 8.1 ▪ PROCESADOR: Intel® Core™ i7-4720HQ Processor ▪ MEMORIA: 16GB DDR3L 1600MHz memory ▪ MOTOR DE GRÁFICOS* AMD Radeon™ R9 M265X, Mobile Intel® HM86 Express Chipset ▪ MEMORIA DE GRÁFICOS* 2GB GDDR5 discrete graphics memory ▪ PUERTOS USB 1 puerto USB v3.0, 1 puerto USB 3.0 con función de USB Sleep and Charge, 2 puertos USB v2.0, Puerto LAN RJ-45 ▪ BATERÍA Batería de iones de litios de 6 celdas/48 Wh

Fuente: <http://www.toshiba.es/laptops/product-filter/?sFamily=satellite>

Elaborado por: Gabriel López

➤ **Interfaz**

Para la interfaz del sistema inteligente se cuenta con una amplia gama de microcontroladores por lo cual en la tabla 4.13 se muestra la comparativa entre ellos.

Tabla 4.14 Comparativa microcontroladores.

	LAUNCHPAD MSP430	PICAXE	WIRING	NETDUINO	ARDUINO
Voltaje operativo	5 V	5 V	5 V	5 V	5 V
Voltaje de entrada recomendado	7-10 V	7-9 V	7-10 V	7-12 V	7-12 V
Voltaje de entrada límite	6-20 V	6-20 V	6-20 V	6-20 V	6-20 V
Contactos de entrada y salida digital	SI	SI	SI	SI	SI
Contactos de entrada analógica	SI	SI	SI	SI	SI
Intensidad de corriente	30 mA	40 mA	20 mA	30 Ma	40 Ma
<u>Memoria Flash</u>	16KB	32KB	128KB	32KB (4KB reservados para el bootloader)	128KB (4KB reservados para el bootloader)
<u>SRAM</u>	1 KB	2 KB	8 KB	2 KB	8 KB
<u>EEPROM</u>	512 bytes	1 KB	4 KB	1 KB	4 KB
Frecuencia de reloj	16 MHz	16 MHz	16 MHz	16 MHz	16 MHz

Fuente: Guía de aprendizaje para microcontroladores

Para el desarrollo del sistema inteligente se optó por la placa arduino uno debido a su reducido tamaño, bajo precio y su microcontrolador ATmega16U2 el cual es suficiente para la ejecución y desempeño del programa que se cargó en él.

➤ **Actuadores Externos**

Existen varios métodos para la alerta automática que se pueden aplicar entre los que tenemos:

- ***Teléfonos de emergencia:***

Los teléfonos de emergencia permiten que un usuario, víctima de un accidente, se ponga en contacto con el centro de control. Además de establecer un enlace de voz, el uso de un teléfono de emergencia por parte de un usuario también permite conocer su posición exacta. También son números fáciles de aprender, cortos, que comunican con los servicios de asistencia inmediata que

generalmente trabajan mediante entidades estatales o de las comunidades autónomas. Por ello, sirven para notificar de una emergencia sanitaria, un accidente, catástrofe o cualquier otro incidente en que el haya algún peligro potencial.

Normalmente son números gratuitos y prefijados. Algunos números, como el 112 están disponibles siempre que haya cobertura GSM de algún operador. Aunque debemos tener en cuenta que en zonas rurales, montañosas o muy apartadas, sin no hay ninguna cobertura de red, no se podrá efectuar la llamada. Por este motivo, no se debe confiar únicamente en el teléfono móvil para avisar de una emergencia [27].

- ***Pulsadores de alarma.***

Los pulsadores de alarma permiten a un usuario enviar una alarma al centro de control en caso de accidente. Puesto que no se trata de equipos muy caros, se pueden instalar a intervalos frecuentes. Cuando el pulsador manual de alarma o estación manual es activado, transmite una señal al centro de control.

Esta activa un buzzer (alarma de bajo volumen ubicada en el mismo centro) y después de unos segundos o minutos, según las normativas, activará las sirenas, bien generales, bien para un sector determinado (sirenas programables), dependiendo de la edificación y programación; y efectúa las maniobras de seguridad, tal como abrir o cerrar puertas. Las sirenas y maniobras son programadas siguiendo el plan de seguridad del lugar. Existen 2 tipos tales como:

- **Pulsadores de alarma convencional**

Un pulsador manual o estación manual convencional está conectada al panel a través de una zona que informa al panel de seguridad que zona ha sido activada. La desventaja de este sistema es que no especifica exactamente cual elemento fue activado. En una instalación con varios elementos en una zona hay que buscar punto a punto en la ZONA para saber dónde se ha producido la emergencia.[27].

- **Pulsadores de alarma direccionable**

Un pulsador manual o estación manual direccionable se conecta al panel de seguridad a través de un bucle o loop. Este sistema se comunica a través de un sistema binario que permite identificar en el panel cada elemento del sistema. Cuando un pulsador manual o estación manual direccionable es activada el panel informará del lugar exacto del elemento.

- **Alarma automática.**

Cuando se utilizan sistemas de emergencia orientados a los usuarios. Muy a menudo suele haber sensores instalados para detectar un cambio de estado y comunicar esta información al centro de control y mando utilizando un sistema. Entonces se informa al operador de que un usuario solicita ayuda. Existen dos tipos:

- **Alertas al móvil**

Al detectar en el sitio especificado algún intruso este avisara al usuario vía sms la presencia de este. También existen aplicaciones para los móviles los cuales podrán ser alertados dependiendo de la función de dicha aplicación con el sistema operativo instalado.

- **Alertas por correo electrónico**

Al detectar algún intruso en el lugar especificado será enviado un correo electrónico inmediatamente en donde se especificara la presencia de dicho intruso [27].

- **Detección automática de incidentes.**

Cuando un sitio está equipado de un sistema de video vigilancia las imágenes procedentes del sitio y sus inmediaciones se muestran en pantallas instaladas en el centro de control y mando. Es difícil que un operador pueda supervisar más de unas cuantas pantallas simultáneamente con una vigilancia constante durante varias horas.

- **Megafonía.**

Altavoces instalados a intervalos fijos en lugares para proporcionar información e instrucciones a los usuarios. Altavoces (o sirenas) instalados a intervalos fijos en lugares que emiten una señal acústica que indica un peligro. Altavoces (o balizas sonoras) instalados cerca de las salidas de emergencia para proporcionar información a los usuarios sobre la necesidad de usar una salida y dónde está ubicada [27].

La tabla 4.15 nos muestra las características de las diferentes alertas automáticas.

Tabla 4.15 Tabla comparativa alertas automáticas.


Teléfonos de emergencia:	Pulsadores de alarma.	Alarma automática.	Detección automática de incidentes.	Megafonía.
<p>Permiten que un usuario, víctima de un accidente, se ponga en contacto con el centro de control</p> <p>Normalmente son números gratuitos y prefijados</p>	<p>Permiten a un usuario enviar una alarma al centro de control</p> <p>Activa un buzzer y después de unos segundos o minutos, según las normativas, activará las sirenas, bien generales o bien para un sector determinado</p>	<p>Tienen sensores instalados para detectar un cambio de estado y comunicar esta información al centro de control.</p> <p>Existen dos tipos: la alerta móvil la alerta por correo electrónico.</p>	<p>Está equipado de un sistema de video vigilancia las imágenes procedentes del sitio se muestran en pantallas instaladas en el centro de control.</p> <p>Dependen que un operador este al mando de la vigilancia</p>	<p>Altavoces instalados a intervalos fijos en lugares para proporcionar información e instrucciones a los usuarios</p>

Fuente: <http://www.xatakafoto.com>
Elaborado por: Gabriel López

Se seleccionó el método de alerta al móvil ya que este es el más efectivo al tratarse de un sistema de vigilancia remota alertando de manera inmediata al dueño o encargado de las zonas de almacenamiento provocando una acción inmediata por parte de estos.

Para poder emplear dicho método fue necesario el empleamiento de un módulo GSM el cual tiene la función de realizar la llamada y envío de mensaje de texto de encontrarse algún intruso en el área de almacenamiento ya que la aplicación debe poder desempeñarse de manera óptima e inmediata al tratarse de un sistema de seguridad en tiempo real no pueden existir retrasos en su conexión. Para el desarrollo del sistema inteligente propuesto se empleara el módulo GSM SIM 908. La tabla 4.16 muestra las características del módulo GSM.

Tabla 4.16 Características del módulo GSM SIM 908

Sistema Operativo Móvil	CARACTERISTICAS
	<ul style="list-style-type: none"> ▪ Alimentación: 6-12v ▪ Consumo en reposo: (100mA/7v - en modo GSM) ▪ Quad-Band 850/900/1800/1900MHz ▪ GPRS multi-slot class 10 ▪ GPRS mobile station class B ▪ Soporta GSM phase 2/2+ ▪ Clase 4 (2 W a 850/900 MHz) ▪ Clase 1 (1 W a 1800/1900MHz) ▪ Receptor GPS incorporado ▪ Antena SMD integrada para GPS y GSM ▪ Conexión para teclado 4x4 ▪ Interruptor USB/Arduino ▪ Indicadores LED para alimentación, estado de la red y estado del módulo ▪ Jack para salida de audio ▪ Jack para entrada de audio ▪ Dimensiones: 81x70mm

	<ul style="list-style-type: none">▪ Especificaciones para SMS por GSM / GPRS: Punto a punto MO y MT SMS cell broadcast Texto u modo PDU
--	---

Fuente: <http://es.aliexpress.com/item/GPS-GSM-module-SIM908/>
Elaborado por: Gabriel López

4.4. Diseño de los diagramas y representación del sistema inteligente

A continuación en la figura 4.7 se puede observar el diagrama de funcionamiento del sistema inteligente propuesto.

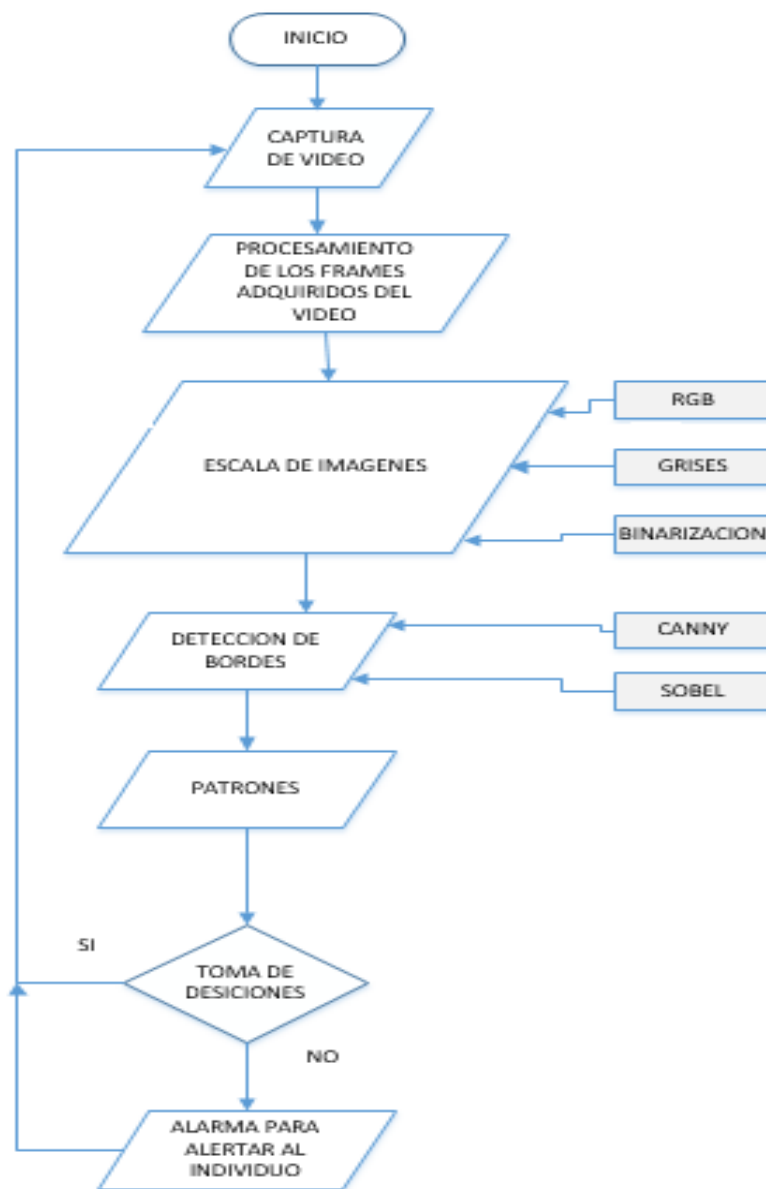
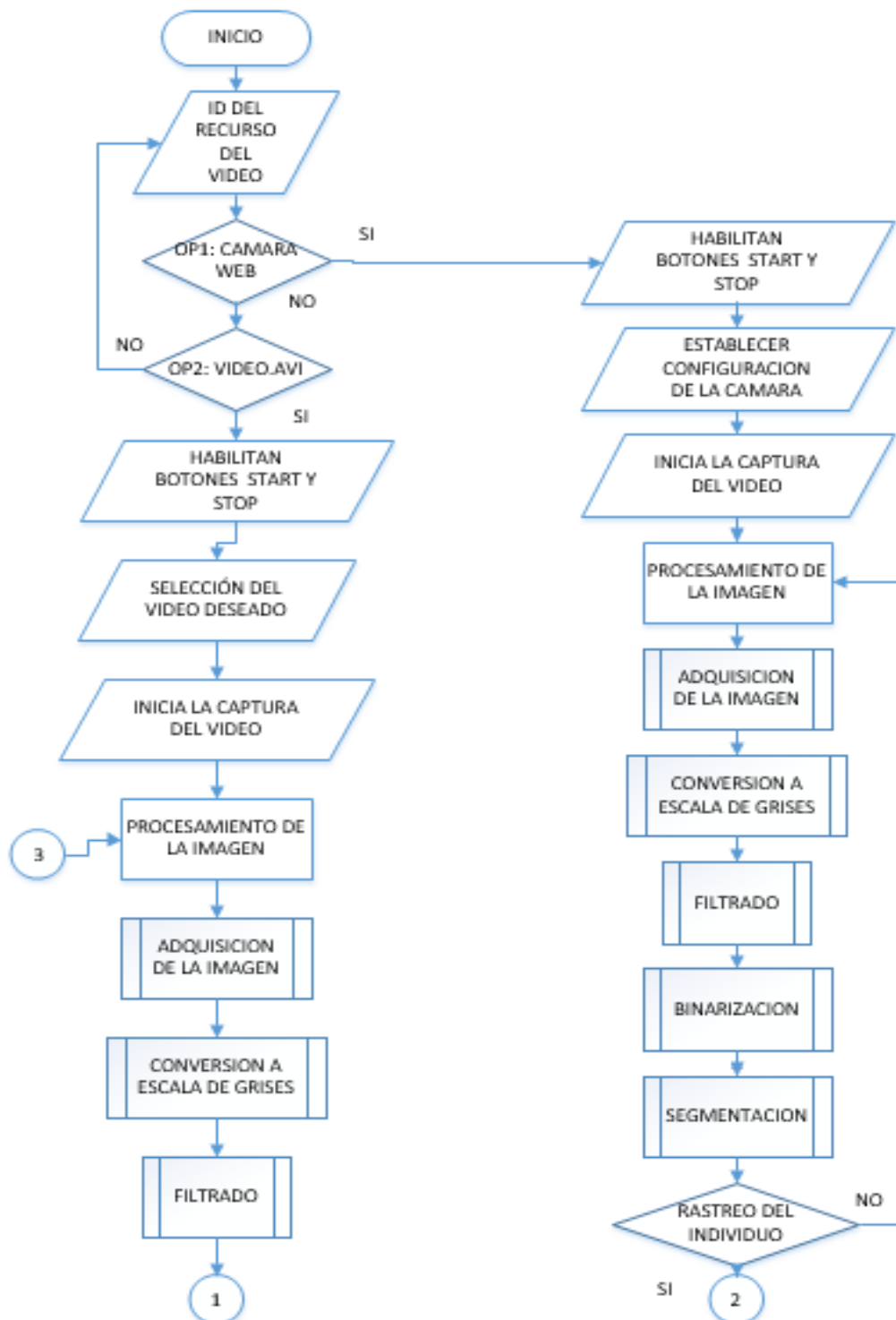


Figura 4.7 Diagrama de bloques del sistema inteligente
Elaborado por: Gabriel López

El reconocimiento de patrones es una etapa esencial en el sistema inteligente de visión artificial por lo que se debe contar con imágenes con el menor ruido posible para evitar errores en la detección de los individuos, lo que se da gracias a un procesado eficiente. El uso de filtros nos ayudó de gran manera a disminuir el ruido en las imágenes pero si las imágenes adquiridas son de muy mala calidad el sistema siempre tendrá errores. El procesamiento digital de las imágenes en esta etapa es muy riguroso para que no existan

errores en la identificación de los elementos. A continuación se muestra un diagrama en la figura 4.8 de los principales pasos en el reconocimiento de patrones.



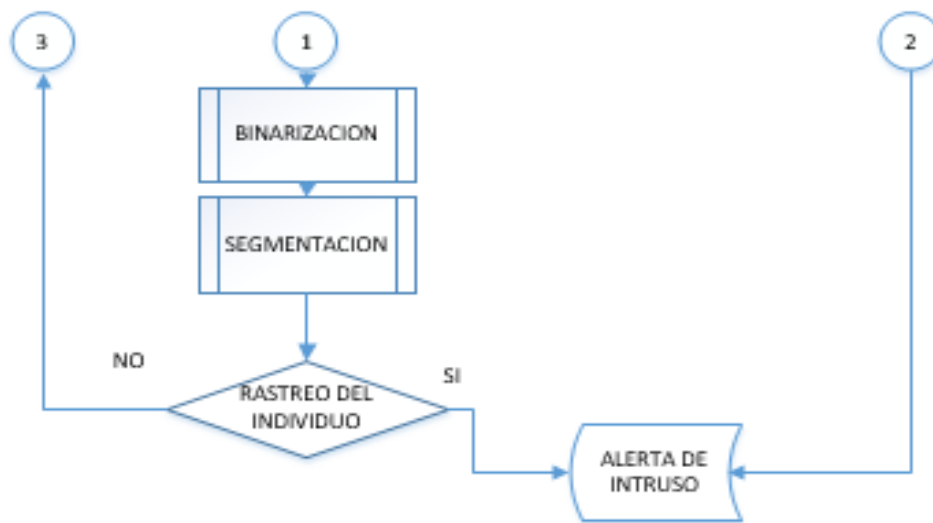


Figura 4.8 Diagrama de flujo de la programación del sistema inteligente.
Elaborado por: Gabriel López

4.5. Especificación de variables y algoritmos de procesamiento de imágenes

La programación de los algoritmos está basada en MATLAB, ya que este software es apropiado para el procesamiento digital de imágenes del sistema inteligente propuesto, dado que su utilidad proviene de la capacidad de manipular matrices que contienen información sobre la imagen adquirida.

Adquisición de imágenes

Esta herramienta usa adaptadores de dispositivos de hardware para conectar los dispositivos a través de los drivers.

Para conectar un dispositivo de adquisición de imágenes desde MATLAB se debe crear un objeto de entrada de video. Este objeto representa la conexión entre el dispositivo y MATLAB. Se puede manipular las propiedades de este objeto para controlar distintos aspectos de la adquisición. Para crear este objeto se necesita información como:

Nombre del Adaptador

Un adaptador es el software que utiliza MATLAB para comunicarse con el dispositivo. Para determinar de cuales adaptadores dispone el sistema se debe llamar a la función

imaqhwinfo la cual entrega información acerca de los adaptadores instalados y disponibles [28].

En el presente proyecto se utilizó MATLAB 2013a, y la función imaqwinfo en la ventana de comandos da como resultado la siguiente información:

```
>> imaqhwinfo
ans =
    InstalledAdaptors: {'matrox' 'winvideo'}
    MATLABVersion: '7.14 (R2013a)'
    ToolboxName: 'Image Acquisition Toolbox'
    ToolboxVersion: '4.3 (R2013a)'
```

Los adaptadores cargados y disponibles son matrox, y winvideo.

ID del dispositivo

El adaptador asigna un número único para cada dispositivo con el que se puede comunicar. El primer dispositivo detectado toma el ID de 1, el segundo dispositivo el ID de 2, y así sucesivamente.

Con la función imaqhwinfo se puede encontrar el ID de un determinado dispositivo, especificando el nombre del adaptador. El comando debe ser escrito de la siguiente manera:

```
>> info=imaqhwinfo('winvideo')
info=
AdaptorDllName: 'C:\ProgramFiles\MATLAB\R2013a\toolbox\imaq\imaqadaptor
s\win64\mwwinvideoimaq.dll'
    AdaptorDllVersion: '4.3 (R2013a)'
    AdaptorName: 'winvideo'
    DeviceIDs: {[1] [2]}
    DeviceInfo: [1x1 struct]
```

El proyecto consta con un dispositivo de entrada que es asignado con ID=2 ya que el ID=1 es el de la cámara web propia de la PC. En el caso de que sea una PC sin cámara web el dispositivo toma el ID=1[28].

Formato del Video

Especifica las características del video (formato, nombre del dispositivo, resolución de la imagen, etc.). Los dispositivos de adquisición de imágenes soportan múltiples formatos de video que se especifican cuando se crea el objeto de entrada de video.

Para obtener información sobre el nombre del dispositivo, los formatos que soporta y formato por defecto se aplican los siguientes comandos:

```
>> info=imaqhwinfo('winvideo');
>> device1 = info.DeviceInfo(1)
device1 =
    DefaultFormat: 'YUY2_160x120'
    DeviceFileSupported: 0
    DeviceName: 'Video WebCam'
    DeviceID: 1
    VideoInputConstructor: 'videoinput('winvideo', 1)'
    VideoDeviceConstructor: 'imaq.VideoDevice('winvideo', 1)'
    SupportedFormats: {'YUY2_160x120' 'YUY2_176x144'
'YUY2_320x240' 'YUY2_352x288' 'YUY2_640x480'}
```

Por tanto el comando para la adquisición de la imagen de la cámara será:

```
vid = videoinput('winvideo', 2, 'UYVY_720x576');
```

Captura de imagen

Las variaciones en intensidad debidas al ruido, por deficiencias en la iluminación o la obtención de imágenes de bajo contraste hacen necesario un pre-procesamiento con el objetivo de corregir estos problemas de manera que acentúen las características que se deseen extraer de las mismas, para que facilite las operaciones de las etapas posteriores

La captura de la imagen será realizada mediante la sentencia getsnapshot la cual hace tomas continuas del video, con el fin de que los procesos posteriores puedan actuar con más facilidad, las cuales son extraídas directamente desde la variable handles.vidobj que fue usada en la barra de herramientas en el icono de capturar imagen y las almacena en una variable [28].

Por lo tanto el comando para la captura de la imagen deseada será:

```
I = getsnapshot(handles.vidobj);
```

Representación de imágenes

Una vez adquirida la imagen de la cámara, esta ingresa a un arreglo de elementos llamados píxeles. MATLAB almacena las imágenes como un arreglo de dos dimensiones, de esta forma se obtienen una matriz de números de tamaño $M \times N$, en la cual cada elemento de la matriz corresponde a un píxel de la imagen adquirida.

La intensidad de cada píxel es cuantizada en 256 niveles de grises que son representados por valores, asignando el valor de 255 como el nivel más claro y 0 como el nivel más oscuro [29].

Para pasar de un tipo de imagen a otro la Toolbox de procesamiento digital de imágenes de MATLAB (Image Processing Toolbox "IPT") proporciona varias funciones que permiten la conversión. En la tabla 4.17 se muestran las más habituales:

Tabla 4.17 Tabla de variables en MATLAB.

Función	Imagen de Salida	Imagen de Entrada
Dither	imagen indexada (mediante tramado)	Imagen RGB
Grayscale	imagen indexada (mediante umbralización multinivel)	Imagen de intensidades en escala de grises
gray2ind	imagen indexada	Imagen de intensidades en escala de grises
ind2gray	Imagen indexada en escala de grises	Imagen indexada
rgb2ind	Imagen indexada	Imagen RGB
Ind2rgb	Imagen RGB	Imagen indexada
rgb2gray	Imagen en escala de grises	Imagen RGB

Fuente: <http://www.mathworks.com/products/matlab/>

Elaborado por: Gabriel López

Escala de grises

Representan a cada pixel de la imagen como un elemento de una matriz en donde cada uno tiene un valor entre 0 y 255, correspondiendo el valor 0 al negro absoluto y 255 al blanco absoluto.

Esta configuración es muy importante porque es más fácil trabajar con dos escalas de colores. El comando para hacer este proceso es `rgb2gray`, el cual hace el cambio de escala RGB a escala de grises automáticamente; la misma que se aplica directamente a la imagen tomada y almacenada en la variable `I` para luego almacenarla en otra llamada `Igray`. [29]

Por lo tanto el comando para obtener una imagen en escala de grises será:

```
Igray=rgb2gray(I);
```

Binarización

Permite la conversión de una imagen en escala de grises a una imagen en blanco y negro. Una imagen binaria es una imagen que contiene regiones de partículas con pixeles de valor 1 y una región de fondo con pixeles de valor 0, es decir el color blanco tendrá el valor de 1 y el color negro el valor de 0.

La binarización de imágenes es el método más común de segmentar imágenes en regiones de partículas y regiones de fondo. Así, se resaltan las características de la imagen que desee. Debido a que el proceso es subjetivo, la imagen binaria resultante puede contener información no deseada, como partículas de ruido, partículas tocando el borde de las imágenes, partículas tocándose mutuamente y partículas con bordes irregulares e inclusive pérdida de información. Es por esto que es recomendable después de la binarización la aplicación de filtros. [29]

Por lo tanto el comando para obtener una imagen binarizada será:

```
lavel=graythresh(I1);  
intensidad=lavel*255  
BW=im2bw(I1,lavel);
```

Filtros

El filtro Máximo consiste en asignar al píxel de referencia el valor más alto de los encontrados en la máscara, por lo que su efecto es ensanchar las zonas claras y adelgazar las oscuras. El filtro Mínimo hace justo lo contrario, seleccionando el valor más bajo. En este caso el nuevo valor del píxel de referencia será el valor más bajo de los encontrados en la máscara, teniendo como efecto ensanchar y adelgazar las zonas claras y oscuras respectivamente.

En el Filtro Mediana se escoge el píxel de en medio del vector (ordenado previamente de menor a mayor). Se basa en sustituir el valor del píxel por el de la mediana del conjunto formado por el mismo y sus otros vecinos, logrando con esto homogenizar los píxeles de la imagen, se lo utiliza cuando existe la presencia de ruido aleatorio [29].

Extracción de bordes

Una de las técnicas para poder identificar la discontinuidad de la intensidad en los píxeles de una imagen es la detección de bordes, un borde es conocido como un cambio significativo en el valor en la escala de grises entre los píxeles vecinos de la imagen. La intensidad del borde es afectada por la iluminación, si esta es baja la intensidad del borde es pequeño lo que hace difícil de detectar, otro de los factores que afectan es la presencia de objetos brillantes ya que gracias a estos los otros objetos aparecen en la imagen con poca intensidad.

La detección de bordes es la forma más utilizada en segmentación para encontrar discontinuidades en valores de intensidad. La IPT cuenta con una función específica para ello, la función `edge`, desde la que se puede detectar los bordes de una imagen utilizando diferentes métodos .

`BW = edge(I)` recibe como entrada una imagen binaria o en escala de grises `I` y devuelve una imagen binaria `BW` del mismo tamaño que `I`, con unos donde la función haya encontrado bordes y ceros en el resto. `I` es un array disperso tipo `numeric`. `BW` es de tipo `logical`.

`edge` utiliza por defecto el método Sobel para detectar los bordes, pero se pueden utilizar otros métodos que se listan a continuación:

- El método Sobel detecta los bordes usando la aproximación Sobel. Suele obtener buenos resultados con las opciones por defecto. (Por defecto).
- El método Prewitt detecta los bordes usando la aproximación Prewitt. Suele obtener resultados parecidos a los obtenidos con el método Sobel.
- El método Roberts detecta los bordes usando la aproximación Roberts. Uno de los métodos más simples y antiguos de detección de bordes. No suele usarse mucho salvo que se necesite simpleza y rapidez.
- El método del Laplaciano del Gaussiano detecta los bordes buscando cortes con cero después de filtrar I con un filtro Laplaciano o Gaussiano. Obtiene normalmente mejores resultados que el método Sobel pero a costa de tener que modificar los parámetros por defecto.
- El método de cruce por cero detecta los bordes buscando cruces con cero después de filtrar I con el filtro que se especifique. Está basado en el mismo concepto que el método del Laplaciano del Gaussiano pero se utiliza un filtro especificado para realizar la convolución.
- El método Canny detecta los bordes buscando máximos locales del gradiente de I. Este método es el más potente de todos y por ello es el más utilizado. Es menos propenso a confundirse por el ruido y más propenso a detectar bordes débiles. Para obtener el mejor resultado normalmente se necesitan modificar sus parámetros por defecto [29].

A continuación se detalla la sintaxis para el empleo de cada método:

Método Sobel:

`BW = edge(I, 'sobel')` especifica que se usará el método Sobel

Método Prewitt:

`BW = edge(I, 'prewitt')` especifica que se usará el método de Prewitt.

Método Roberts:

`BW = edge(I, 'roberts')` especifica que se usará el método de Roberts.

Método Laplaciano del Gaussiano:

`BW = edge(I, 'log')` especifica que se usará el método Laplaciano del Gaussiano

Método de Cruce por cero:

`BW = edge(I, 'zerocross', thresh, h)` especifica que se usará el método de Cruce por cero

Método Canny:

`BW = edge(I, 'canny')` especifica que se usará el método Canny

4.6. Arquitectura del software de seguridad

El sistema inteligente se ha diseñado y desarrollado en base a las características de un sistema de visión artificial para la vigilancia en las áreas de almacenamiento. Siendo el sistema capaz de determinar los patrones de una persona presente en dicha área.

Por ello es necesario dotar al sistema de la potencia computacional suficiente para procesar las imágenes a la velocidad deseada para dar una respuesta a tiempo. Este hecho sugiere la necesidad de diseñar arquitecturas de alto rendimiento especializadas para los problemas de visión.

Al analizar todos los elementos y características de los sistemas inteligentes de visión artificial se ha decidido la arquitectura mostrada en la figura 4.10.

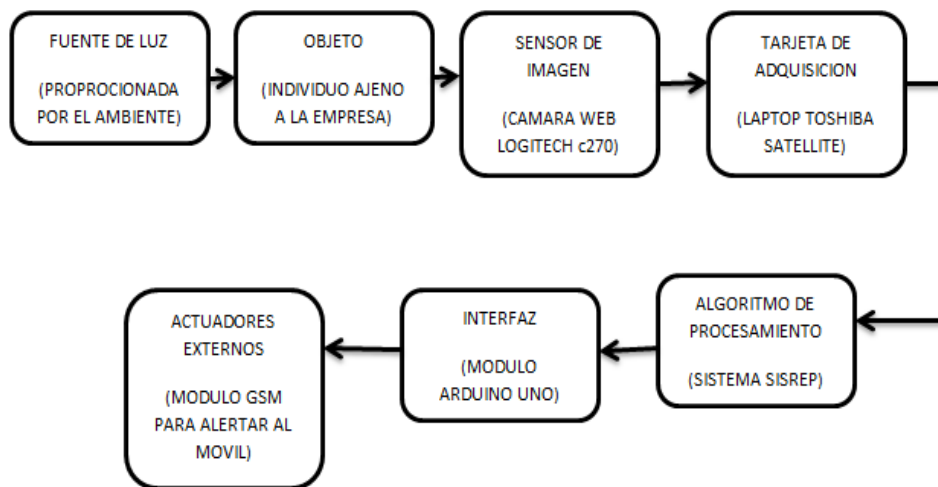


Figura 4.10 Arquitectura del sistema inteligente propuesto.
Elaborado por Gabriel López

Se tiene interés, asimismo, en el desarrollo de algoritmos de procesamiento en tiempo real que aporten capacidad de refinamiento en la respuesta, en función del tiempo disponible. El análisis completo debería realizarse en menos de 5 segundos por cada imagen para la detección de intrusos en el área. En la figura 4.11 se muestra la arquitectura del reconocimiento de patrones para el sistema SISREP.

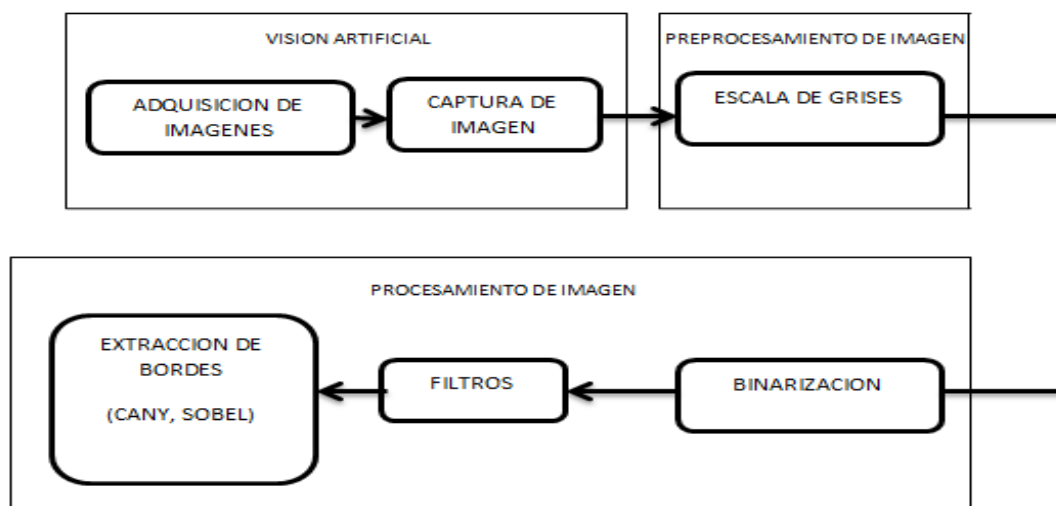


Figura 4.11 Arquitectura del sistema de reconocimiento de patrones.
Elaborado por: Gabriel López

4.7. Programación del software de seguridad con visión artificial

Visión Artificial:

Código para adquirir video	Descripción
<pre>handles.vidobj = videoinput('winvideo',1); start(handles.vidobj); guidata(hObject, handles); src_obj = getselectedsource(handles.vidobj); get(src_obj); vidRes = get(handles.vidobj, 'VideoResolution'); preview(handles.vidobj,hImage); axis off catch</pre>	Evento que se ejecuta al oprimir el botón cámara
Código para capturar imagen	Descripción
<pre>[nombre,ruta]=uigetfile('*.avi','SELECCIONE EL VIDEO EN FORMATO AVI'); if nombre == 0 set(hObject,'Enable','on'); set(handles.inicio,'Enable','on'); set(handles.parar,'Enable','on'); return</pre>	Evento que se ejecuta al oprimir el botón video

Preprocesamiento de imagen:

Código para escala de grises	Descripción
<pre>grisesb=rgb2gray(imagengray); background3=grisesb; axes(handles.background3); axis off; imshow(background3); handles.imagen_M=grisesb; guidata(hObject, handles);</pre>	Evento que se ejecuta automáticamente para el preprocesamiento de la imagen

Procesamiento de la imagen:

Código para los filtros	Descripción
<pre>var=fix(handles.slider_ruido); se = strel('disk',var); filtroRGB = imsubtract(imadd(imagenruido,imtophat(imagenruido,se)), imbothat(imagenruido,se)); else var=fix(handles.slider_ruido); se = strel('disk',var); filtroRGB = imsubtract(imadd(imagenruido,imtophat(imagenruido,se)), imbothat(imagenruido,se));</pre>	Evento en el que se ejecutan diversos filtros para evitar el ruido en la imagen

<code>end</code>	
Código para binarización	
<code>binaria1=im2bw(imagenbina,0.5); background3=binaria1; axes(handles.background3); axis off; imshow(background3); handles.imagen_M=binaria1; guidata(hObject, handles);</code>	Evento en el que se transforma la imagen adquirida
Código para extracción de bordes	
<code>I3=edge(segmento, 'canny'); I3=~I3; for i=1:800, waitbar(i/800) end</code>	Evento en el que se extraen los rasgos a

Alerta al móvil:

Código para los filtros	Descripción
<code>set(PS, 'Baudrate', 9600); set(PS, 'StopBits', 1); set(PS, 'DataBits', 8); set(PS, 'Parity', 'none'); set(PS, 'Terminator', 'CR/LF'); set(PS, 'OutputBufferSize', 2); set(PS, 'InputBufferSize', 2); set(PS, 'Timeout', 5);</code>	Evento en el que se ejecuta la alerta al móvil sobre intrusos

4.8. Pruebas de funcionamiento

Al término de la implementación del sistema desarrollado, objeto de esta tesis se han obtenido los siguientes resultados de acuerdo con los objetivos planteados en un inicio. La valoración del software está afirmada en su arquitectura. A continuación la Tabla 4.18 muestra la prueba de módulo único

Tabla 4.18: Prueba módulo único

PRUEBAS DEL SISTEMA INTELIGENTE DE RECONOCIMIENTO DE PATRONES			
Nombre Prueba		Módulo único	
Realizada por		Programador	
Descripción		Evaluar y depurar cada algoritmo de programación del sistema	
Nro.	Categoría	Respuesta esperada del sistema	Resp.
1	Visión Artificial	Captura y visualización de video	OK
2	Visión Artificial	Extracción de bordes y contornos	OK
3	Visión Artificial	Segmentación	OK
4	Visión Artificial	Detección y reconocimiento de patrones	OK
5	Comunicación inalámbrica	Sistema de alerta automática	OK

Elaborado por: Gabriel López

Fuente: Sistema inteligente de reconocimiento de patrones con visión artificial. SISREP

A continuación, la Tabla 4.19 representa la prueba grupo de módulos.

Tabla 4.19: Prueba grupo de módulos

PRUEBAS DEL SISTEMA INTELIGENTE DE RECONOCIMIENTO DE PATRONES				
Nombre Prueba		Grupo de módulos		
Realizada por		Programador		
Descripción		Evaluar y depurar cada interfaz de usuario del sistema		
Nro.	Categoría	Respuesta esperada de la interfaz de usuario	Incremental ascendente	Incremental descendente
1	Visión Artificial	Captura y visualización de video	OK	OK
2	Visión Artificial	Extracción de bordes y contornos	OK	---
3	Visión Artificial	Segmentación	OK	---
4	Visión Artificial	Detección y reconocimiento de patrones	OK	---
5	Comunicación inalámbrica	Sistema de alerta automática	OK	---

Elaborado por: Gabriel López

Fuente: Sistema inteligente de reconocimiento de patrones con visión artificial. SISREP

A continuación, la Tabla 4.20 describe la prueba de sistema completo.

Tabla 4.20: Prueba sistema completo

PRUEBAS DEL SISTEMA INTELIGENTE DE RECONOCIMIENTO DE PATRONES					
Nombre Prueba		Sistema completo			
Realizada por		Programador			
Descripción		Evaluar y depurar el sistema SISREP completo			
Nro.	Categoría	Respuesta esperada del sistema PIVRA	Velocidad	Exactitud	Fiabilidad
1	Visión Artificial	Captura y visualización de video	100%	100%	100%
2	Visión Artificial	Extracción de bordes y contornos	90%	100%	95%
3	Visión Artificial	Segmentación	100%	100%	95%
4	Visión Artificial	Detección y reconocimiento de patrones	95%	100%	90%
5	Comunicación inalámbrica	Sistema de alerta automática	100%	100%	100%

Elaborado por: Gabriel López

Fuente: Sistema inteligente de reconocimiento de patrones con visión artificial. SISREP

La evaluación es favorable, ya que el sistema SISREP cumple con los objetivos establecidos en el desarrollo de la propuesta y ha superado la prueba de módulo único y la prueba de grupo de módulos, con éxito. A su vez, la prueba de sistema completo obtuvo excelentes resultados en velocidad y exactitud; en la fiabilidad existe una reducción porcentual de entre 5% a 10%, aproximadamente, debido al entorno operativo e iluminación del ambiente de trabajo del sistema.

Complementariamente a las pruebas del sistema SISREP se realizó una verificación basada en los diagramas de flujo del sistema. La Tabla 4.21 muestra la verificación basada en la operatividad.

Tabla 4.21: Verificación basada en la operatividad

PRUEBAS DEL SISTEMA INTELIGENTE DE RECONOCIMIENTO DE PATRONES			
Nombre		Basada en el uso	
Realizada por		Programador	
Descripción		Verificar el uso y el entorno de trabajo del sistema SISREP	
Nro.	Categoría	Respuesta esperada del sistema PIVRA	Perfil operacional (Bodegas PYMES)
1	Visión Artificial	Captura y visualización de video	90%
2	Visión Artificial	Extracción de bordes y contornos	90%
3	Visión Artificial	Segmentación	95%
4	Visión Artificial	Detección y reconocimiento de patrones	98%
5	Comunicación inalámbrica	Sistema de alerta automática	100%

Elaborado por: Gabriel López

Fuente: Sistema inteligente de reconocimiento de patrones con visión artificial. SISREP

La verificación del sistema SISREP basada en la operatividad, estipula buenos resultados en virtud al entorno de trabajo; un valor porcentual promedio de 94,6% reflejan aquel valor cualitativo. Además, el valor porcentual beneficia la ergonomía y fiabilidad del sistema SISREP.

El diseño ha sido el óptimo puesto que la cámara permitió tener la visión correcta del producto haciendo que el sistema funcione adecuadamente, mediante la utilización de módulos Arduinos y la utilización de cable USB para la conexión del computador, evitando así una interrupción o un corte en la transferencia de datos durante el proceso.

Para facilitar y darle una mayor rapidez al desarrollo del sistema se utilizó Matlab, el cual permitió diseñar una interface más simple y una programación más detallada mediante códigos de acuerdo a la necesidad de este.

Pruebas realizadas durante el día

Las pruebas realizadas para el reconocimiento del patrón corporal del ser humano, durante el día se han capturado en el transcurso de la mañana y tarde dentro del horario de 8 a.m hasta las 5 p.m

Para establecer la confiabilidad del sistema se han realizado 5 capturas del procesamiento las cuales se muestran en las figuras 4.10 - 4.14, en las que se puede observar la imagen de salida con la detección de las personas sometidas a las pruebas

(fotografía izquierda) y la detección de los contornos de estas en el procesamiento de los frames del video (fotografía derecha).



Figura 4.10 Prueba durante el día aplicada en el mal Materia Plaza.
Elaborado por Gabriel López



Figura 4.11 Prueba durante el día aplicada en el mal Materia Plaza.
Elaborado por Gabriel López



Figura 4.12 Prueba durante el día aplicada en el parque Montalvo.
Elaborado por Gabriel López



Figura 4.13 Prueba durante el día aplicada en el mal Materia Plaza.
Elaborado por Gabriel López



Figura 4.14 Prueba durante el día aplicada en el mal Materia Plaza.
Elaborado por Gabriel López

Pruebas realizadas durante la noche

Las pruebas realizadas para el reconocimiento del patrón corporal del ser humano, durante la noche se han capturado en el transcurso del horario de 9 p.m hasta las 11 p.m

Al igual que se ha realizado en el día, para determinar la confiabilidad del sistema se han realizado 5 capturas del procesamiento las cuales se muestran en las figuras 4.15 - 4.19.

Se puede observar igual que anteriormente la imagen de salida con la detección de las personas sometidas a las pruebas (fotografía izquierda) y la detección de los contornos de estas en el procesamiento de los frames del video (fotografía derecha).



Figura 4.15 Prueba durante la noche aplicada en un área de almacenamiento.
Elaborado por Gabriel López



Figura 4.16 Prueba durante la noche aplicada en un área de almacenamiento.
Elaborado por Gabriel López



Figura 4.17 Prueba durante la noche aplicada en un área de almacenamiento.
Elaborado por Gabriel López



Figura 4.18 Prueba durante la noche aplicada en un área de almacenamiento.
Elaborado por Gabriel López



Figura 4.19 Prueba durante la noche aplicada en un área de almacenamiento.
Elaborado por Gabriel López

4.8.1. Interface de usuario y visualización

Entorno Gráfico

Para un entorno amigable al usuario se diseñó una interface la cual posee botones, etiquetas y video utilizando la opción GUI Builder de Matlab. La pantalla principal consta de ítems los cuales se encuentran agrupados en la parte de recursos, sistema y controles

La figura 4.20 muestra el entorno gráfico del sistema SISREP.

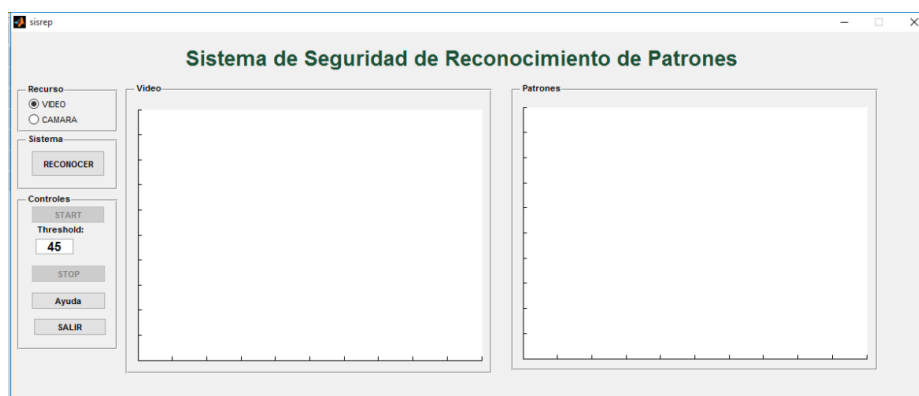


Figura 4.20 Entorno gráfico del sistema SISREP.
Elaborado por: Gabriel López

En la parte de recursos se puede observar los ítems de video y cámara. La parte de video permite la captura de videos en formato .AVI del disco duro del ordenador.

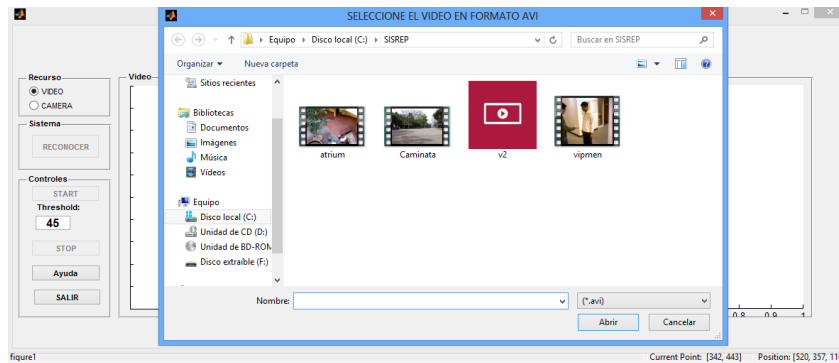


Figura 4.21 Sub-Ítem Sistema opción reconocer video del disco duro.

Elaborado por: Gabriel López

La parte de cámara permite la adquisición del video en tiempo real en donde se podrá seleccionar la cámara que se desee emplear para el reconocimiento de patrones.

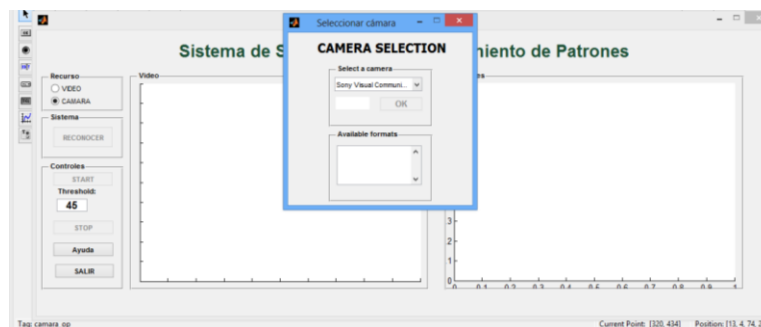


Figura 4.22 Sub-Ítem Sistema opción reconocer cámara existente

Elaborado por: Gabriel López

Visualización

Una visualización sin errores es fundamental para el reconocimiento del objeto; en la interface de usuario el check botón Cámara es el encargado de comenzar el video. La figura 4.13 muestra la captura de video



Figura 4.23 Captura de video.
Elaborado por: Gabriel López

La imagen de salida con la detección del parámetro corporal de las personas en la parte izquierda y la detección de los contornos de estas se encuentra en la parte derecha.

Reconocimiento

Iniciada la visualización se procesó el video para comenzar con el reconocimiento. La figura 4.24 muestra el reconocimiento del contorno y patrón corporal del ser humano.



Figura 4.24 Reconocimiento del contorno y patrón corporal de la persona.
Elaborado por: Gabriel López

4.9. Estudio Económico

Estudio Económico

El presente Trabajo de Graduación no está dirigido a generar beneficios tangibles o económicos; pero es imperante realizar un análisis económico del capital invertido en el desarrollo e implementación del sistema SISREP

4.9.1. Descripción General

En este apartado se detallan los valores financieros necesarios para la implementación del sistema inteligente; los costos pueden variar al cambiar los modelos y/o marcas del hardware, así como el tipo de licencias de software utilizados en el desarrollo.

Hardware:

La Tabla 4.22 muestra el hardware utilizado en el desarrollo e implementación del sistema inteligente:

Tabla 4.22: Hardware

Hardware	Cantidad	Horas de proyecto	Horas de uso al año
Ordenador Toshiba core i7, 2GHz, 8G RAM.	1	960	1440
Cámara Web Logitech c270, 720MP.	1	800	1136
Impresora HP LaserJet CP1515n	1	56	56
Arduino Uno	1	800	1136

Fuente. Trabajo de Graduación. Sistema SISREP.
Elaborado por. Gabriel López

Software:

La Tabla 4.23 muestra en detalle el software utilizado en el diseño y desarrollo del sistema inteligente:

Tabla 4.23: Software

Software	Cantidad	Horas de proyecto	Horas de uso al año
Sistema Operativo Microsoft Windows 8.1	1	960	1440
Matlab 2013 ^a	1	800	1136
Arduino 1.6.6	1	800	1136
Microsoft Office 2010	1	960	1440

Fuente. Trabajo de Graduación. Sistema SISREP.
Elaborado por. Gabriel López

Herramientas:

La Tabla 4.24 muestra las herramientas utilizadas para la implementación y funcionamiento del sistema inteligente:

Tabla 4.24: Herramientas

Elemento	Cantidad	Horas de proyecto	Horas de uso al año
Soporte para Cámara	1	800	1136
Carcasa para el modulo	1	640	851

Fuente. Trabajo de Graduación. Sistema SISREP
Elaborado por. Gabriel López

Mano de obra directa:

La Tabla 4.25 muestra las horas de mano de obra directa requerida en el proceso de implementación del sistema inteligente:

Tabla 4.25: Mano de obra directa

Actividad	Horas
Ingeniería de software	40
Programación del software SISREP	640
Ingeniería de usabilidad	24
Implementación, pruebas y solución de problemas	40
Documentación del proyecto	360
Horas totales	1104

Fuente. Trabajo de Graduación. Sistema SISREP.
Elaborado por. Gabriel López

4.9.2. Precios Unitarios

En este apartado se detallan los precios de cada uno de los elementos necesarios para llevar a cabo la implementación del sistema SISREP, los cuales se han citado anteriormente.

Hardware:

La Tabla 4.26 muestra el hardware del sistema inteligente y su precio unitario:

Tabla 4.26: Hardware - Precio Unitario

Hardware	Cantidad	Precio (\$/unid)
Ordenador Toshiba core i7, 2GHz, 8G RAM.	1	780,00
Cámara Web Logitech c270, 720MP.	1	36,00
Impresora HP LaserJet CP1515n	1	280,00
Arduino Uno	1	20,00

Fuente. Trabajo de Graduación. Sistema SISREP.
Elaborado por. Gabriel López

Software:

La Tabla 4.27 detalla el software utilizado en el sistema inteligente con sus respectivos precios unitarios:

Tabla 4.27: Software - Precio Unitario

Software	Cantidad	Precio (\$/unid)
Sistema Operativo Microsoft Windows 8.1	1	150,00
Matlab 2013 ^a	1	4,50
Arduino 1.6.6	1	0,00
Microsoft Office 2010	1	4,50

Fuente. Trabajo de Graduación. Sistema SISREP
Elaborado por. Gabriel López

Herramientas:

La Tabla 4.28 describe las herramientas necesarias para el funcionamiento del sistema SISREP con sus precios unitarios:

Tabla 4.28: Herramientas - Precio Unitario

Elemento	Cantidad	Precio (\$/unid)
Soporte para Cámara	1	30,00
Carcasa para Modulo	1	10,00

Fuente. Trabajo de Graduación. Sistema SISREP.
Elaborado por. Gabriel López

Mano de obra directa:

La Tabla 4.29 muestra la mano de obra directa útil para la implementación del sistema inteligente, con sus respectivos precios unitarios:

Tabla 4.29: Mano de obra directa - Precio Unitario

Actividad	Horas	Precio (\$/hora)
Ingeniería de software	40	5,00
Programación del software SISREP	640	8,00
Ingeniería de usabilidad	24	3,00
Implementación, pruebas y solución de problemas	40	3,00
Documentación del proyecto	360	3,00

Fuente. Trabajo de Graduación. Sistema SISREP
Elaborado por. Gabriel López

4.9.3. Sumas Parciales

En este apartado se detallan los importes parciales de cada uno de los elementos que componen las distintas categorías, que han sido calculados a partir de la descripción general y los precios unitarios.

Hardware:

La Tabla 4.30 muestra el hardware del sistema inteligente con su respectiva suma parcial:

Tabla 4.30: Hardware - Suma Parcial

Hardware	Cantidad	Horas de proyecto	Horas de uso al año	Precio (\$/unid)	Amortización anual	Coste (\$)
Ordenador Toshiba core i7, 2GHz, 8G RAM.	1	960	1440	780,00	26%	135,20
Cámara Web Logitech c270, 720MP.	1	800	1136	36,00	26%	6,59
Impresora HP LaserJet CP1515n	1	56	56	280,00	26%	72,80
Arduino Uno	1	800	1136	20,00	26%	3,66
SUBTOTAL						218,25

Fuente. Trabajo de Graduación. Sistema SISREP
Elaborado por. Gabriel López

Software:

La Tabla 4.31 muestra el software utilizado para el diseño y desarrollo del sistema inteligente con la respectiva suma parcial:

Tabla 4.31: Software - Suma Parcial

Software	Cantidad	Horas de proyecto	Horas de uso al año	Precio (\$/unid)	Amortización anual	Coste (\$)
Sistema Operativo Microsoft Windows 8.1	1	960	1440	150,00	26%	26,00
Matlab 2013 ^a	1	800	1136	4,50	26%	0,82
Arduino 1.6.6	1	800	1136	0,00	26%	0,00
Microsoft Office 2010	1	960	1440	4,50	26%	0,78
SUBTOTAL						27,6

Fuente. Trabajo de Graduación. Sistema SISREP
Elaborado por. Gabriel López

Herramientas:

La Tabla 4.32 muestra las herramientas utilizadas en la implementación y operatividad del sistema SISREP con la suma parcial:

Tabla 4.32: Herramientas - Suma Parcial

Elemento	Cantidad	Horas de proyecto	Horas de uso al año	Precio (\$/unid)	Amortización anual	Coste (\$)
Soporte para Cámara	1	800	1136	30,00	30%	2,75
Carcasa para el modulo	1	640	851	10,00	30%	1,80
SUBTOTAL						3,83

Fuente. Trabajo de Graduación. Sistema SISREP
Elaborado por. Gabriel López

Mano de obra directa:

La Tabla 4.32 muestra el coste de las horas de mano de obra directa, requerida en el proceso de implementación del sistema SISREP

Tabla 4.32: Mano de obra directa - Suma Parcial

Actividad	Horas	Precio (\$/hora)	Coste (\$)
Ingeniería de software	40	5,00	200,00
Programación del software SISREP	640	8,00	5.120,00
Ingeniería de usabilidad	24	3,00	72,00
Implementación, pruebas y solución de problemas	40	3,00	120,00
Documentación del proyecto	360	3,00	1.080,00
SUBTOTAL			1.477,12

Fuente. Trabajo de Graduación. Sistema SISREP
Elaborado por. Gabriel López

4.9.4 Presupuesto General

Sumando la contribución de todas las categorías anteriores, se obtiene que el coste total del proyecto; impuestos incluidos, asciende al valor estipulado en la Tabla 4.33:

Tabla 4.33: Presupuesto General

Concepto	Coste Subtotal (\$)
Hardware	218,25
Software	27,6
Herramientas	30,21
Mano de obra directa	1.477,12
Imprevistos (5%)	432,54
TOTAL	2185.72

Fuente. Trabajo de Graduación. Sistema SISREP
Elaborado por. Gabriel López

CAPÍTULO V

CONCLUSIONES Y RECOMENDACIONES

5.1. Conclusiones

El desarrollo del Trabajo de Graduación presentó aspectos relevantes en lo concerniente al Sistema Inteligente de Reconocimiento de Patrones con visión artificial (SISREP); detallados en las siguientes conclusiones:

- Se diseñó e implemento un sistema de reconocimiento de patrones para la alerta automática de intrusos con una interfaz gráfica de usuario en un 98% útil para la adaptación al dominio de la propuesta y de fácil manipulación para el usuario.
- Arduino en conjunto con Matlab permite crear un algoritmo vinculado que envía en forma inalámbrica un uno lógico para alertar la presencia de intrusos y cero lógico para la ausencia de intrusos gestionando la alerta hacia el teléfono móvil.
- El prototipo SISREP con visión artificial, permite la alerta automática de intrusos en áreas de almacenamiento de las PYMES; es capaz de capturar, detectar y reconocer patrones en forma de la silueta humana con un alto nivel de acierto y desvinculado en un 95 % a perturbaciones en la imagen digital en movimiento; para una óptima aplicación de filtrado de bordes Canny y Sobel.
- La detección y el reconocimiento de patrones del prototipo SISREP, en los casos experimentados, ha sido correcto; obteniendo un acierto del 95%, en aproximadamente de 2 a 3 segundos de tiempo de respuesta; condición eficaz para la menor sensación posible de que se trata de una secuencia de fotogramas procesados con técnicas de Visión Artificial a una distancia máxima aproximada de entre 5,5 m a 10,5 m.
- Si se careciera de iluminación el software SISREP en la versión 2.2 podría utilizar el algoritmo de conversión de colores a modo negativo.

- Los algoritmos de procesamiento de fotogramas en movimiento, en especial filtros de imágenes y corrección de píxeles; en conjunto a la extracción de bordes Sobel y Canny, cooperan para la consecución de un algoritmo de extracción de rasgos óptimo en el 99 % a la visualización del patrón humano en forma binaria.
- El prototipo SISREP utiliza imágenes dinámicas en color, trabaja con las imágenes en el espacio de color RGB, con su transformación en escala de grises y binaria. Consigue de esta manera un pre-procesado y procesado de información rápido, logrando facilitar el reconocimiento de patrones en ordenadores de nominales prestaciones.

5.2. Recomendaciones

A continuación se presentan las principales recomendaciones efectuadas en el desarrollo y la implementación del Sistema Inteligente de Reconocimiento de Patrones con visión artificial (SISREP):

- Establecer variables de denominación coherente a las sentencias de programación en MATLAB, para tener mayor organización y control en el caso de la depuración y ejecución de las funciones de dicho lenguaje en procesamiento digital de imágenes y visión artificial.
- Es importante tener en cuenta la ubicación, posición y calidad de imagen de la cámara, además de otros parámetros importantes de calibración que incurren en el funcionamiento eficiente del prototipo SISREP.
- Al utilizar un entorno con iluminación propia, el GUIDE de MATLAB enfocado a Visión Artificial, algoritmos robustos al procesamiento de imágenes y extracción de rasgos; se logró mayor precisión y desempeño del prototipo SISREP en la forma del patrón de la imagen segmentada.
- El tipo de imágenes en movimiento que se utiliza un algoritmo de reconocimiento de patrones, influyen de manera determinante en los resultados que se obtienen, y en la elección de determinados parámetros inherentes al prototipo SISREP.
- Además de los algoritmos de extracción de bordes Sobel y Canny, existen muchos otros con los que queda pendiente su experimentación; como son los árboles de decisión, las máquinas de soporte vectorial o las redes neuronales.
- En el diseño, implementación y perfeccionamiento de los diferentes métodos para la caracterización de figuras humanas han jugado un papel muy importante los

conocimientos adquiridos durante la carrera en el ámbito del diseño de proyectos, la programación, DSP (Procesamiento Digital de Señales), etc.

- La detección automática de humanos para su futura aplicación en el mundo real, por ejemplo en la industria del automóvil que estudia la realización de vehículos “respetuosos con los peatones”; es decir, que sean capaces de minimizar las lesiones en caso de atropello e incluso evitarlo. Una vez contrastados los mejores módulos (en cuanto a exactitud y tiempo de proceso), queda el camino abierto para la creación de aplicaciones en virtud al mejoramiento de las condiciones de vida del ser humano con la Visión Artificial.

BIBLIOGRAFÍA

- [1] M.G. Chavez, I.M. Tatayo, “Desarrollo e implementación de un sistema para el control de un robot móvil para acceso a lugares remotos utilizando tecnología Kinect”, Thesis, Ing. Electrónica e Instrumentación, Escuela Politécnica del Ejercito, Latacunga, Ecuador, 2015.
- [2] A.H. Poveda “Sistema de seguridad integral para Generica S.A”, Thesis, Ing. Administración y Negocios, Universidad Nacional Abierta a Distancia, Bogota, Colombia, 20143.
- [3] J. Herrero, “Preservación de privacidad de personas en video seguridad” Thesis, Ing. en Telecomunicaciones, Universidad Autónoma de Madrid, Madrid, España, 2015.
- [4] J.E. Perez, “Reconocimiento de placas vehiculares mediante procesamiento de imágenes para optimizar el acceso a los parqueaderos de la UTA campus Huachi”, thesis, Universidad Técnica de Ambato, Ambato, Ecuador, 2014.
- [5] J. Herrero, “Preservación de privacidad de personas en video seguridad” Thesis, Ing. en Telecomunicaciones, Universidad Autónoma de Madrid, Madrid, España, 2015.
- [6] M.T. Calderón, D.J.Maldonado “Control por visión de un cuadricoptero utilizando ROS”, Thesis, Ing. Electrónica y Eléctrica, Escuela Politécnica Nacional, Quito, Ecuador, 2014.
- [7] G. Rodriguez, “Generador de código CNC para proceso de frenado, empleando visión computarizada”, Thesis, Ing. Electrónica y Telecomunicaciones, Universidad Autónoma del Estado de Hidalgo, Hidalgo, México, 2014.
- [8] R. Vaquier, J.Aguilar, “Sistema de visión artificial para la detección de cuerpos sólidos, en botellas a través del procesamiento de imágenes con LabView”, Thesis, Ing. Electrónica y Telecomunicaciones, Universidad Autónoma de Estado de Veracruzana, Veracruz, México, 2014.
- [9] A. Loras, “Percepción de la actividad de un conductor mediante cámaras 3D”, Thesis, Ing. Electrónica Industrial y Automática, Universidad Carlos III de Madrid, Madrid, España, 2013.

- [10] G. Ballesteros, “Verificación de vehículos mediante técnicas de visión artificial”, Thesis, Ing. en Electrónica y Comunicaciones, Universidad Autónoma de Madrid, Madrid, España, 2015.
- [11] “Definición de Sistema Inteligente, requisitos e Inteligencia Artificial”, Julio 2015, [Online] Disponible en: <http://b.se-todo.com/pravo/4699/index.html>
- [12] “Acerca de la Inteligencia de los Sistemas Inteligentes”, Julio 2015 [Online] Disponible en: <http://www.acadning.org.ar/anales/2005/I%20%20Incorporaciones/Titulares/1.%20DAquila%20-%202005/3.%20Conferencia%20Ing.%20DAquila.pdf>
- [13] J.D.Cruz, “Clasificador automático de tapas rosca de plástico para procesos de producción continua, basados en la inspección de defectos superficiales”, thesis, Universidad Técnica del Norte, Ibarra, Ecuador, 2010.
- [14] R.C. Gonzalez, P. Wintz, “Procesamiento digital de imágenes”, Primera edición. USA. Addison-Weslwy, 2010, pp. 21-25
- [15] Constante P., “Diseño e implementación de un prototipo de brazo robótico para aplicaciones de clasificación y reconocimiento de formas en el proceso de politizado empleando procesamiento digital de imágenes”, thesis, Escuela Politécnica del Ejercito, Latacunga, Ecuador, 2012
- [16] L. Sánchez, R. Fernández, “Procesamiento Digital de Imágenes: Fundamentos y prácticas con MATLAB” Ph.D. dissertation, Universidad de León., España, 2011, pp. 10-23.
- [17] T. Acharya, A. Ray, “Image processing: principles and applications”. Second Edition, New York, John Wiley & Sons, 2012, pp. 15-20
- [18] R. Woods, L. Eddins, “Digital image processing using MATLAB”, First Edition, USA, Woods & Eddins, 2010, pp. 22-25.
- [19] J.L.Calle, B.S.Gamboa, “Diseño e implementación del sistema de seguridad del centro de investigaciones y control ambiental (CICAM)”, Thesis Ing. Electrónica y Control, Escuela Politécnica Nacional, Quito, Ecuador, 2011.

- [20] Llumiquinga F., “Procesamiento de imágenes mediante software libre Python para el análisis metalográfico en aceros de bajo contenido de carbono”, Thesis Ing. Mecánica, Escuela Politécnica Nacional, Quito, Ecuador, 2014
- [21] E.M. Acurio, “Diseño y construcción de un módulo didáctico de visión artificial orientado al control de calidad de llenado de botellas de vidrio con diferentes tipo de líquido“, Thesis, Ing. Electrónica y Control, Escuela Politécnica Nacional, Quito, Ecuador, 2011
- [22] Cárdenas F., Rosendo O., “Automatización de un sistema de centrado de componentes utilizando visión artificial”, thesis, Universidad Politécnica Salesiana, Cuenca, Ecuador, 2012
- [23] Rodriguez A., “Control y monitoreo de la red de datos, internet y climatización de la central de datos del hospital municipal Nestra Señora de la Merced”, thesis, Universidad Tecnica de Ambato, Ambato, Ecuador, 2015.
- [24] Mendoza D. Romero W., “Diseño y construcción de un prototipo de sistema de control para monitorear e incrementar la seguridad en el acceso vehicular al parqueadero de la ESPE-L utilizando procesamiento digital de imágenes”, thesis, Escuela Politécnica del Ejercito, Latacunga, Ecuador, 2012
- [25] Constante P., “Diseño e implementación de un prototipo de brazo robótico para aplicaciones de clasificación y reconocimiento de formas en el proceso de politizado empleando procesamiento digital de imágenes”, thesis, Escuela Politécnica del Ejercito, Latacunga, Ecuador, 2012
- [26] R.Llobregat, “Procesado digital de imágenes de video para la detección de humo”, M.S. thesis, Escuela Politécnica de Valencia, Valencia, España, 2011
- [27] “Manual AIRCR de túneles de carretera”, agosto 2015, [Online] Disponible en: <http://tunnels.piarc.org/tunnels/ressources/4/515,Chap-8.3-Equipment-and-systems-ES-v-.pdf>

[28] Rodríguez E., Topapanta A. “Diseño y construcción del sistema de enfoque automático para una cámara térmica, usando procesamiento de imágenes en matlab”, thesis, Escuela Politécnica Nacional, Quito, Ecuador, 2014

[29] E. Laorden, “Descripción, comparación y ejemplos de uso de las funciones de la toolbox de procesamiento digital de imágenes”, M.S. thesis, Universidad Politécnica de Madrid, Madrid, España, 2012

ANEXOS

ANEXOS

ÍNDICE DE ANEXOS

ANEXO A: Manual de Usuario	101
ANEXO B: Diagramas de Componentes	108
ANEXO C: Programación	114

ANEXO A: Manual de Usuario

ÍNDICE MANUAL DE USUARIO

1. INTRODUCCIÓN	103
2. OBJETIVOS	103
3. REQUISITOS	105
4. CONOCIMIENTOS PREVIOS	105
5. SISTEMA SISREP	105

1. INTRODUCCIÓN

El Sistema Inteligente de Reconocimiento de Patrones con Visión Artificial (SISREP) en su entorno de herramienta de seguridad emplea, por un lado; técnicas de Visión Computacional como tecnología base para el reconocimiento de patrones de carácter facial y corporal; con el propósito de alertar en forma inalámbrica a través de mensaje o llamada la presencia de intrusos en las áreas de almacenamiento de las PYMES; proceso realizado con tecnología Arduino y módulo GSM. En este entorno se da la posibilidad de detección y reconocimiento del propietario de la PYMES o caso contrario la alerta de un posible delincuente.

SISREP focaliza su operatividad en el análisis de una imagen mediante un sistema de visión artificial, para identificar patrones de interés. El sistema SISREP comprueba la presencia de uno o varios individuos que transiten en un rango frontal a la cámara web; si existiera la presencia del propietario de la PYMES, éste no es alertado automáticamente a su teléfono inteligente, ya que es conocido para el sistema; lo que no ocurre al tratarse con otras personas desvinculadas a la empresa. La característica distintiva del sistema SISREP, está en alertar con una llamada o un mensaje al teléfono inteligente o smartphone de manera automática gracias a la detección y reconocimiento de patrones.

2. OBJETIVOS DEL MANUAL

- El objetivo principal es facilitar la utilización del sistema SISREP obteniendo información de cada uno de los formularios que posee la aplicación, a manera de guía de usuario.
- Conocer cómo manipular el sistema SISREP mediante la descripción detallada e ilustrada de todas las opciones.

3. REQUISITOS DEL SISTEMA

El sistema funcionará sin problemas con las siguientes características mínimas:

Sistemas operativos admitidos:

- Microsoft Windows 8.1. SP1.
- Microsoft Windows 8.

Procesador:

- Arquitecturas de 32 y 64 bits.

Memoria RA:

- Al menos 2GB para SO: Windows 8 y Windows 8.1 SP1.

Dispositivo gráfico:

- Dispositivo gráfico compatible con DirectX 9.
- Memoria gráfica de al menos 2 GB compartida o dedicada.

Dispositivos:

- Cámara web.
- Arduino Uno.
- Módulo GSM.

Herramientas y suministros:

- Soporte de madera.

Cárcasa del sistema SISREP.

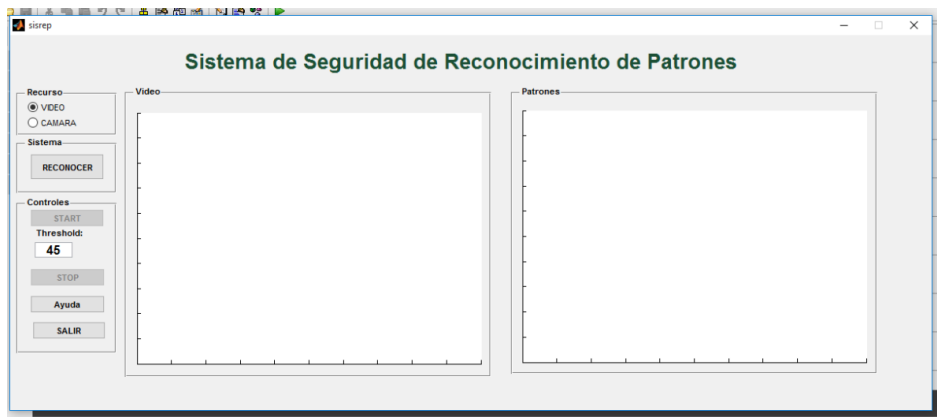
4. CONOCIMIENTOS PREVIOS

Los conocimientos mínimos que deben tener las personas que van a operar el sistema SISREP, y que deberán utilizar este manual son:

- Conocimientos sobre Matlab 2013a.
- Conocimientos sobre GUI Builder de Matlab.
- Conocimientos sobre Toolbox de procesamiento de imágenes.
- Conocimientos sobre plataforma Arduino y módulo GSM.

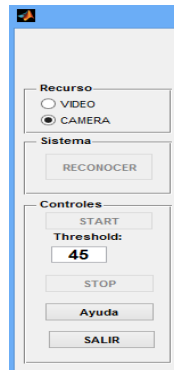
5. SISTEMA SISREP

Para ingresar al sistema, el usuario necesita ejecutar la aplicación, accediendo a la siguiente interfaz:

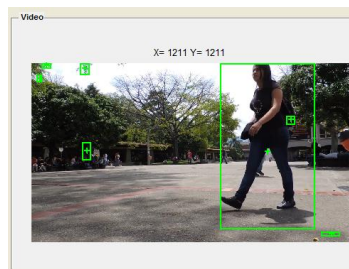


Interfaz. 1: Ventana inicio sistema SISREP

A continuación se describirá cada ítem del formulario principal. Mismo que se compone de tres partes importantes como son: Recurso, sistema y controles.



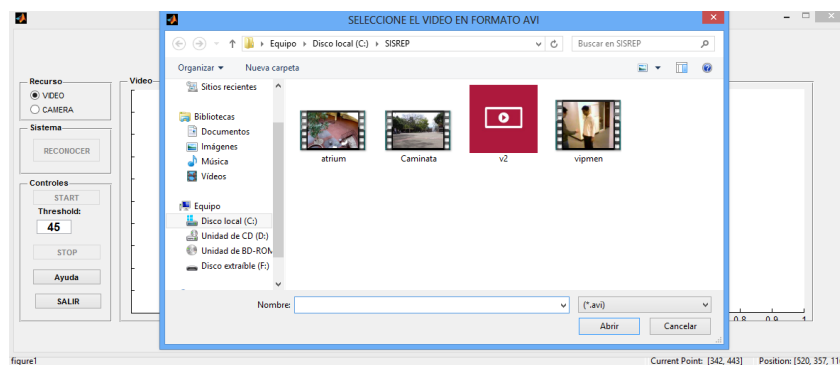
Interfaz. 2: Ítems del sistema SISREP



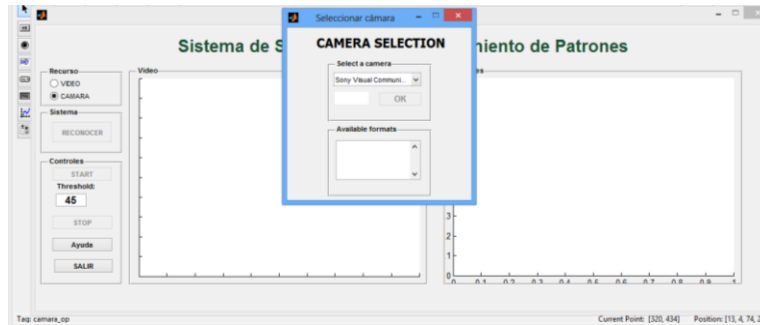
Interfaz. 3: Sub-Ítem Recurso opción video



Interfaz. 4: Sub-Ítem Recurso opción cámara



Interfaz. 5: Sub-Ítem Sistema opción reconocer video del disco duro



Interfaz. 6: Sub-Ítem Sistema opción reconocer cámara existente

Recurso: Permite la captura de video ya sea a través de un video en formato .AVI del disco duro del ordenador; a su vez, permite la adquisición del video en tiempo real.

Sistema: Una vez seleccionado el recurso, la categoría sistema; permite habilitar la ventana para abrir un video en formato .AVI con el checkbox seleccionado en VIDEO o también habilita una ventana para la selección de la cámara web conectada al ordenador y su respectivo formato que soporta dicho dispositivo.

Controles: Posee los controles del sistema SISREP a través de botones, con los cuales se puede iniciar (Start), detener (Stop), ayuda y salir del sistema.

- **Start:** Permite dar inicio a todos los procesos de reconocimiento de patrones de SISREP.
- **Stop:** Permite detener el proceso del sistema SISREP.
- **Ayuda:** Permite abrir un formulario con la autoría del sistemas, además de un resumen funcional de SISREP.

Salir: Cierra el sistema y todas las ventanas.

ANEXO B: Diagramas de Componentes

ÍNDICE DIAGRAMAS DE COMPONENTES

1. INTRODUCCIÓN	109
2. ESTRUCTURA DEL MODULO ARDUINO UNO	109
3. CARACTERISTICAS DE LA CAMARA WEB.....	110
4. CARACTERISTICAS LAPTOP TOSHIBA SATELLITE	112

1. INTRODUCCIÓN

Los presentes diagramas electrónicos y características de los equipos forma parte del sistema SISREP, son diseños de ingeniería útiles para cumplir con el propósito del reconocimiento de patrones con visión artificial para la alerta automática de intrusos en las áreas de almacenamiento de las PYMES. La parte técnica de los bosquejos están consolidados para el cumplimiento efectivo del propósito de cada circuito. La fuente de voltaje 5V/40mA, la cámara web y el circuito receptor de alerta.

Para efectuar el montaje de los circuitos electrónicos y para evaluarlos, es necesario disponer de equipos de laboratorios que debe incluir multímetro digital. Para el desarrollo del circuito de recepción se utilizó un Arduino UNO el mismo que consta con su software en versión 1.6.7, mismo que es la versión actualizada a la elaboración del presente proyecto de investigación.

2. ESTRUCTURA DEL MODULO ARDUINO UNO

El modulo mostrado a continuación, representa la estructura y forma de empleamiento del Arduino, misma que funciona con una fuente de voltaje de 5V/40mA. El Diagrama Electrónico B.1 muestra el diseño del módulo Arduino UNO.

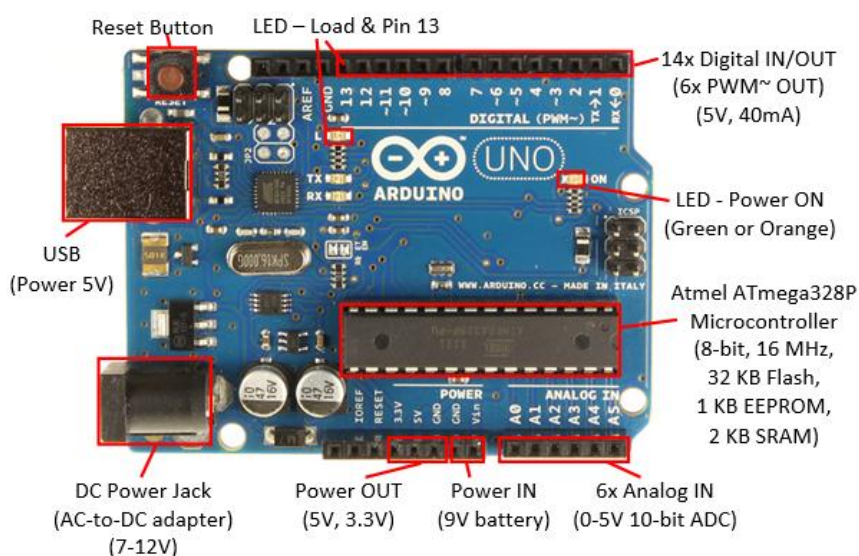


Diagrama Electrónico B.1: Modulo Arduino Uno
Fuente. Proyecto de Investigación. Sistema SISREP

3. CARACTERISTICAS DE LA CAMARA WEB

El Diagrama Electrónico B.2, representa características de la cámara web a emplear, mismo que está constituida por tecnología CMOS, produciendo una sensibilidad superior con los elementos amplificadores de cada celda siendo una desventaja el ruido producido por esta.



Figura B.1: Cámara Web Logitech c270

Fuente. <http://www.logitech.com/es-roam/product/hd-webcam-c270>

Logitech C270 especificaciones:

- Videoconferencias HD 720p en pantalla panorámica, fáciles de realizar en la mayoría de aplicaciones de mensajería instantánea, incluida Logitech Vid™ HD.
- Videoconferencias en alta definición :
Con vídeo HD 720p en pantalla panorámica en la mayoría de aplicaciones de mensajería instantánea y Logitech Vid™ HD.
- Instantáneas de 3 megapíxeles
- Incluye Logitech Vid™ HD
- Micrófono integrado con RightSound™
- Le ofrece conversaciones con un sonido nítido y sin molestos ruidos de fondo.

- Tecnología RightLight™
- Ajusta automáticamente la imagen durante la videoconferencia para proporcionar la mejor calidad incluso en condiciones de iluminación escasa.
- Compatibilidad con MI
- Funciona con Skype™, Windows Live™ Messenger, Yahoo!® Messenger, AOL® Instant Messenger (AIM®) y otras conocidas aplicaciones de mensajería instantánea.
- Software de edición de vídeo gratuito
- Logitech colabora con MAGIX para ofrecer MAGIX™ Photo Manager 9 y MAGIX™ Video Easy integrados en la nueva cámara Web Logitech. Y de forma gratuita.*
- MAGIX es gratuito pero requiere registro por correo electrónico.
- Especificaciones:
- Requisitos del sistema
- Lo que necesita:
- Windows® XP (SP2 o superior), Windows Vista® o Windows® 7 y Windows® 8 (32 bits o 64 bits)
- Requisitos básicos:
- 1 GHz
- 512 MB de RAM o más
- 200 MB de espacio en disco duro
- Conexión a Internet
- Puerto USB 1.1 (se recomienda 2.0)
- Para videoconferencias y grabaciones HD fluidas de 720p:
- Intel® Core™2 Duo a 2,4 GHz
- 2 GB de RAM
- 200 MB de espacio en disco duro
- Puerto USB 2.0
- Velocidad de carga de 1 Mbps o superior
- Resolución de pantalla de 1280 x 720

- Contenido de la caja
- Cámara Web con cable de 1,5 m
- Documentación del usuario
- Especificaciones:
 - Videoconferencias HD (1280 x 720 píxeles) con el sistema recomendado
 - Captura de vídeo: hasta 1280 x 720 píxeles
 - Fotos: hasta 3 megapíxeles (mejora por software)
 - Micrófono integrado con tecnología Logitech RightSound™
 - Certificación USB 2.0 de alta velocidad (se recomienda)
 - Clip universal para monitores LCD, CRT o portátiles
 - Software de cámara Web Logitech:
 - Logitech Vid™ HD
 - Tecnología Logitech RightLight™
 - Captura de vídeo y fotos

3. CARACTERISTICAS LAPTOP TOSHIBA SATELLITE

El Diagrama Electrónico B.3, representa las características de la laptop Toshiba Satellite en la cual se desarrolló el software SISREP.



Figura B.2: Laptop Toshiba Satellite
Fuente. Proyecto de Investigación. Sistema SISREP

Especificaciones principales:

Toshiba Satellite L455-SP2903R	
Especificaciones Técnicas:	
Sistema Operativo	<ul style="list-style-type: none">• Windows 7 Home Basic (SP1)
Procesador	<ul style="list-style-type: none">• Procesador Intel Pentium Dual-Core T4300- 2.10GHz, 1MB L2 Cache, 800MHz FSB
Memoria	<ul style="list-style-type: none">• 2GB DDR2 (Máxima memoria soportada: 4GB)
Disco Duro	<ul style="list-style-type: none">• 250GB (5400 RPM) Serial ATA (SATA)
Unidad Óptica	<ul style="list-style-type: none">• Unidad Óptica DVD SuperMulti soporta 11 formatos:- Velocidad Máxima y compatibilidad: CD-ROM (24x), CD-R (24x), CD-RW (24x), DVD-ROM (8x), DVD-R (Una Capa, (8x)), DVD-R (Doble Capa, (6x)), DVD-RW (6x), DVD+R (Una Capa, (8x)), DVD+R (Doble Capa, (6x)), DVD+RW (8x), DVD-RAM (5x)
Pantalla	<ul style="list-style-type: none">• Pantalla de 15.6" diagonal widescreen TruBrite TFT LCD 1366 x 768 resolución nativa (HD)
Gráficos	<ul style="list-style-type: none">• Intel Graphics Media Accelerator 4500M
Sonido	<ul style="list-style-type: none">• Bocinas estéreo internas
Dispositivos de Entrada	<ul style="list-style-type: none">• Teclado de 86 teclas en Español (negro)• Dispositivo Apuntador TouchPad
Comunicaciones	<ul style="list-style-type: none">• Webcam y micrófono• 10/100 Ethernet• Realtek 802.11b/g wireless LAN
Expansiones	<ul style="list-style-type: none">• 2 ranuras de memoria principal. Ambas ranuras pueden estar ocupadas.
Lector de tarjetas	<ul style="list-style-type: none">• Secure Digital, Memory Stick, Multi Media Card [ranura compartida]
Puertos	<ul style="list-style-type: none">• Video:<ul style="list-style-type: none">- Puerto de salida de monitor RGB (VGA)• Audio:<ul style="list-style-type: none">- Puerto de entrada de micrófono externo- Puerto de salida de audífonos• Datos:<ul style="list-style-type: none">- 3 Puertos Universal Serial Bus USB v2.0:- Puerto RJ-45 LAN
Seguridad	<ul style="list-style-type: none">• Ranura de seguridad de candado
Batería	<ul style="list-style-type: none">• Batería de Ion Litio de 44 Wh (6 celdas)

ANEXO C: Programación

```
% Autor: Gabriel López
% Tutor: Ing. Freddy Robalino
% Universidad Técnica de Ambato
% Facultad de Ingeniería en Sistemas, Electrónica e Industrial
%


---



gui_Singleton = 1;
gui_State = struct('gui_Name',       mfilename, ...
                  'gui_Singleton',  gui_Singleton, ...
                  'gui_OpeningFcn', @sisrep_OpeningFcn, ...
                  'gui_OutputFcn',  @sisrep_OutputFcn, ...
                  'gui_LayoutFcn',  [] , ...
                  'gui_Callback',   []);
if nargin && ischar(varargin{1})
    gui_State.gui_Callback = str2func(varargin{1});
end

if nargout
    [varargout{1:nargout}] = gui_mainfcn(gui_State, varargin{:});
else
    gui_mainfcn(gui_State, varargin{:});
end

% --- Executes just before sisrep is made visible.
function sisrep_OpeningFcn(hObject, eventdata, handles, varargin)

movegui(hObject, 'center')
imaqreset
% ID del recurso del video
handles.fuente=2;
%Desabilita los botones "Start" y "Stop"
set(handles.inicio, 'Enable', 'off');
set(handles.parar, 'Enable', 'off');
set(hObject, 'UserData', 0)
set(handles.axes1, 'XTickLabel', [], 'YTickLabel', [])
set(handles.axes2, 'XTickLabel', [], 'YTickLabel', [])
% Choose default command line output for sisrep
handles.output = hObject;

% Update handles structure
guidata(hObject, handles);

% --- Outputs from this function are returned to the command line.
function varargout = sisrep_OutputFcn(hObject, eventdata, handles)

% Get default command line output from handles structure
varargout{1} = handles.output;

% Habilita los botones "Start" y "Stop"
set(handles.inicio, 'Enable', 'off');
set(handles.parar, 'Enable', 'off');
% Desabilita el boton concurrente
set(hObject, 'Enable', 'off');
```

```

% adquiere el recurso por defecto
sel_fuente=handles.fuente;
switch sel_fuente

case 1 %CAMARA WEB
    % Abre el GUI para seleccionar la camara a usar
    sel_camera
    %
    uiwait
    % Establece la configuracion de la camara
    % La variable es_web_ext= indica si es camara externa o de la
    % laptop
    global id es_web_ext
    % Determina el formato dependiendo del tipo de camara a usar
    if es_web_ext==0
        formato='YUY2_640x480';
    else
        formato='RGB24_640x480';
    end
    end
    try
        % Crea el objeto de video
        vid = videoinput('winvideo',id,formato);
        handles.vidobj=(vid);
        % Update handles
        guidata(hObject, handles);
    catch
        % Mensaje de error
        msgbox('Verifique la conexion de la camara','Camara')
        % Remueve los labels de los axis
        set(handles.axes1,'XTick',[ ],'YTick',[ ])
        set(handles.axes2,'XTick',[ ],'YTick',[ ])
    end
    %Especifica como obtiene o adquiere un frame desde el video
    vid.FrameGrabInterval = 1;
    set(vid,'TriggerRepeat',Inf);
    % Inicia la captura de video
    % _____ Otorga el fondo _____
    vid.FramesPerTrigger=50;
    start(vid);
    data = getdata(vid,50);
    if es_web_ext==0
        fondo=double(ycbcr2rgb(data(:,:, :,50)));
    else
        fondo=double(data(:,:, :,50));
    end
    end

    %Toma la ultima imagen como fondo
    %Muestra el fondo
    imshow(uint8(fondo));
    axes(handles.axes2);
    imshow(uint8(fondo));
    axes(handles.axes1);
    hold on

    % Reseteo el objeto de video
    stop(vid);
    clear vid
    imaqreset

```

```

    case 2%VIDEO AVI
        [nombre, ruta]=uigetfile('*.avi','SELECCIONE EL VIDEO EN
FORMATO AVI');
        if nombre == 0 %si presiona el boton cancel, regresa
            set(hObject,'Enable','on');
            set(handles.inicio,'Enable','on');
            set(handles.parar,'Enable','on');
            return
        end
        handles.xyloObj = VideoReader(fullfile(ruta,nombre));
        axes(handles.axes2)
        % MOSTRAR LA IMAGEN AXES1
        fondo=double(read(handles.xyloObj,10));
        imagesc(uint8(fondo))
        axis image on
        handles.ruta=ruta;
        handles.nombre=nombre;
        %-----
        handles.xyloObj = VideoReader(fullfile(ruta,nombre));
        axes(handles.axes1)
        % MOSTRAR LA IMAGEN AXES2
        fondo=double(read(handles.xyloObj,10));
        imagesc(uint8(fondo))
        axis image on
        handles.ruta=ruta;
        handles.nombre=nombre;

end

% Save background
handles.backg=fondo;

guidata(hObject,handles)
% Enable buttons
set(hObject,'Enable','on');
set(handles.inicio,'Enable','on');
set(handles.parar,'Enable','on');

% FUNCION QUE EMPIEZA EL RASTREO Y RECONOCIMIENTO DEL INDIVIDUO
function inicio_Callback(hObject, eventdata, handles)

% Desabilita el boton Start
set(handles.inicio,'Enable','off');
% Adquiere el umbral (threshold)
umbral=str2double(get(handles.umbral,'String'));
% Verifica el estado del valor numerico del unbral (threshold)
if umbral<0 || isnan(umbral) || isempty(umbral)
    errordlg('Ingreso del valor numérico','ERROR')
    return
end
% Usa dato 0 (1 detiene la captura)
set(handles.parar,'UserData',0)
% TOMA EL RECURSO DE VIDEO
sel_fuente=handles.fuente;
%TOMA EL CONTRAFONDO
fondo=handles.backg;
switch sel_fuente

```

```

case 1 % WEB CAM
    global id es_web_ext

    if es_web_ext==0
        formato='YUY2_640x480';
    else
        formato='RGB24_640x480';
    end
    try
        vid = videoinput('winvideo',id,formato);

        %////////GUARDO EL VIDEO EN UNA VARIABLE GENERAL////////
        handles.vidobj = vid;
        guidata(hObject, handles);
    catch
        msgbox('Verifique la conexion de la camara','Camara')
        set(handles.axes1,'XTick',[ ],'YTick',[ ])
        set(handles.axes2,'XTick',[ ],'YTick',[ ])
    end
    % Especifica cuan amenudo adquiere un frame del video stream
    vid.FrameGrabInterval = 2;
    set(vid,'TriggerRepeat',Inf);
    %Inicia la captura
    % Capturas
    start(vid);
    while 1
        if get(handles.parar,'UserData') % Data from "Stop" button
            break
        end
        % Adquiere imagen
        if es_web_ext==0
            imagen = ycbcr2rgb(getdata(vid,1));

        else
            imagen = getdata(vid,1);

        end
        % Muestra imagen
        image(imagen)
        % Convierte la imagen a double
        im_ent = double(imagen);
        axis image on

        METODO DE CANNY EN LA WEBCAM
        im1=rgb2gray(imagen);
        imcopy=im1;
        imcopy=double(imcopy)/256;
        %Prewitt Mascara que se aplica
        mP1=[-1 -1 -1;0 0 0;1 1 1];
        mP2=[-1 0 1;-1 0 1;-1 0 1];
        %Prewitt se realiza la convolucion por codigo
        cp1=convn(imcopy,mP1);
        cp2=convn(imcopy,mP2);
        cp3=max(cp1,cp2);
        cp3=uint8(cp3);
        imcopy=uint8(imcopy);
        I3=edge(imcopy,'canny');
        I3=(~I3);
    end
end

```



```

axes(handles.axes2);
imshow(I3);
hold off
axes(handles.axes1);
imshow(imagen);
hold off

% Llama la funcion "compare"
compare(im_ent,fondo,umbral);
drawnow
end
stop(vid);
delete(vid)
clear vid
imaqreset

case 2 % VIDEO AVI
nFrames = handles.xyloObj.NumberOfFrames;
for cnt = 1:2:nFrames
    if get(handles.parar,'UserData')% Detiene con el boton
"Stop" el video precedido
        break
    end
    % Muestra la imagen
    the_image=read(handles.xyloObj,cnt);
    image(the_image);
    axis image off
    im_ent=double(the_image);

METODO DE SOBEL EN EL VIDEO
    I = rgb2gray(the_image);
    gra = graythresh(I);
    [junk threshold] = edge(I, 'sobel');
    fudgeFactor = .5;
    BWs = edge(I,'sobel', threshold * fudgeFactor);
    se90 = strel('line', 3, 90);
    se0 = strel('line', 3, 0);
    BWsdil = imdilate(BWs, [se90 se0]);
    BWdfill = imfill(BWsdil, 'holes');
    BWnobord = imclearborder(BWdfill,4);
    seD = strel('diamond',1);
    BWfinal = imerode(BWnobord,seD);
    BWfinal1 = imerode(BWfinal,seD);
    BWfinal2 = imerode(BWfinal1,seD);
    BWfinal = imerode(BWfinal2,seD);
    BWoutline = bwperim(BWfinal);
    Segout = I;
    Segout(BWoutline) = 255;
    axes(handles.axes2);
    imshow(BWfinal2);
    hold off
    axes(handles.axes1);
    imshow(the_image);
    hold off

% Llama la funcion "compare"
compare(im_ent,fondo,umbral);%
drawnow;
end

```

```

% Habilita el boton "Start"
set(handles.inicio,'Enable','on');
% Usa data 0 (1 detiene captura)
set(handles.parar,'UserData',0)
guidata(hObject, handles);

% --- BOTON STOP ---.
function parar_Callback(hObject, eventdata, handles)
set(hObject,'userdata',1)
guidata(hObject, handles)

% --- SELECCION DEL RECURSO DE VIDEO
function uipanel1_SelectionChangeFcn(hObject, eventdata, handles)
if hObject==handles.video_op %VIDEO AVI
    handles.fuente=2;
else
    handles.fuente=1;    % WEBCAM
end
guidata(hObject,handles)

% --- Executes on button press in salir.
function salir_Callback(hObject, eventdata, handles)

menusalir=questdlg('¿Está seguro de salir de la
aplicación?', 'SISREP_v1.0', 'Si', 'No', 'Cancelar');
switch menusalir
    case 'Si',
        close(gcf);
    case 'No'
        gcbf
    case 'Cancelar'
        gcbf
end

```