

UNIVERSIDAD TÉCNICA DEL NORTE

FACULTAD DE INGENIERÍA EN CIENCIAS APLICADAS
ESCUELA DE INGENIERÍA MECATRÓNICA



TEMA:

**MÓDULO DIDÁCTICO PARA RECONOCIMIENTO Y
CLASIFICACIÓN DE PIEZAS MECANIZADAS MEDIANTE VISIÓN
ARTIFICIAL.**

**TRABAJO DE GRADO PARA OPTAR POR EL TÍTULO DE INGENIERO
MECATRÓNICO**

AUTOR:

Fausto Ramiro Navarrete Olmedo

DIRECTOR:

Ing. Fernando Valencia.

Ibarra – Ecuador

2014

UNIVERSIDAD TÉCNICA DEL NORTE

FACULTAD DE INGENIERÍAS EN CIENCIAS APLICADAS

ESCUELA DE INGENIERÍA MECATRÓNICA

TEMA:

**MÓDULO DIDÁCTICO PARA RECONOCIMIENTO Y CLASIFICACIÓN
DE PIEZAS MECANIZADAS MEDIANTE VISIÓN ARTIFICIAL**

APROBACIÓN DEL DIRECTOR

En calidad de Director del Trabajo de grado presentado por el señor FAUSTO RAMIRO NAVARRETE OLMEDO, como requisito previo para optar por el Título de Ingeniero, luego de haber revisado minuciosamente, doy fe de que dicho trabajo reúne los requisitos y méritos suficientes para ser sometido a presentación pública y evaluado por parte del Tribunal Calificador, siendo responsable de la dirección del trabajo de investigación contenido en el presente documento.

En la ciudad de Ibarra, a 15 de Febrero del 2014



Ing. Fernando Valencia

DIRECTOR

DECLARACIÓN DE AUTORÍA

Manifiesto que la presente obra es original y se la desarrolló, sin violar derechos de autor de terceros, por lo tanto es original y que soy el titular de los derechos patrimoniales; por lo que asumo la responsabilidad sobre el contenido de la misma y saldré en defensa de la Universidad Técnica del Norte en caso de reclamación por parte de terceros.

Ibarra, a los 15 días del mes de Febrero de 2014.



FAUSTO RAMIRO NAVARRETE OLMEDO

CI. N°. 1002998779



ING. FORTALEZA VALERIO

DIRECTOR DE TESIS

CERTIFICACIÓN DE AUTORÍA

Certifico que el presente trabajo fue desarrollado por el señor Fausto Ramiro Navarrete Olmedo, bajo mi supervisión.

Ing. Fernando Valencia
DIRECTOR DE TESIS

Fecha: 12 de febrero de 2014

CESIÓN DE DERECHOS DE AUTOR DEL TRABAJO DE GRADO A FAVOR DE LA UNIVERSIDAD TÉCNICA DEL NORTE

Yo, Fausto Ramiro Navarrete Olmedo, con cédula de identidad Nro. 1002998779, manifiesto mi voluntad de ceder a la Universidad Técnica del Norte los derechos patrimoniales consagrados en la Ley de Propiedad Intelectual del Ecuador, artículos 4, 5 y 6, en calidad de autor de la obra o trabajo de grado denominado: MODULO DIDACTICO DE RECONOCIMIENTO Y CLASIFICACION DE PIEZAS MECANIZADAS MEDIANTE VISION ARTIFICIAL, que ha sido desarrollado para optar por el título de: Ingeniero Mecatrónico en la Universidad Técnica del Norte, quedando la Universidad facultada para ejercer plenamente los derechos cedidos anteriormente. En mi condición de autor me reservo los derechos morales de la obra antes citada. En concordancia suscribo este documento en el momento que hago entrega del trabajo final en formato impreso y digital a la Biblioteca de la Universidad Técnica del Norte.

Ibarra, a los 15 días del mes de febrero de 2014.



Fausto Ramiro Navarrete Olmedo

CI. N°. 1002998779

DEDICATORIA

Con todo el amor del mundo dedico este trabajo a Dios, mis padres, hermanos, profesores y demás personas importantes que estuvieron presentes en cada paso de la investigación y especialmente en mi vida, por ese estímulo permanente que supieron brindar para poder superar los obstáculos que se me presentaban, que por su afán y sacrificio fue posible culminar con éxito esta etapa de estudios, que siempre pondré al servicio del bien, la verdad y la justicia.

Ramiro

AGRADECIMIENTO

En el presente trabajo, dejo constancia de mi agradecimiento a la Universidad Técnica Del Norte, por haberme dado la oportunidad de superarme y servir a la Patria, los profesores de manera especial a quienes me brindaron sus valiosos conocimientos que fueron un aporte importante para la dirección y orientación en el desarrollo de este trabajo. A mis padres por ser el soporte moral y económico. A mis amigos por haberme motivado para la culminación de mi carrera profesional.

Ramiro

ÍNDICE DE CONTENIDOS

ÍNDICE GENERAL

CONTENIDO	PAG.
APROVACIÓN DEL DIRECTOR	ii
DECLARACIÓN DE AUTORIA	iii
CERTIFICACIÓN DE AUTORIA	iv
CESIÓN DE DERECHOS	v
DEDICATORIA	vi
AGRADECIMIENTO	vii
ÍNDICE DE CONTENIDOS	viii
Índice de Tablas	xii
Índice de Figuras	xiii
Índice de Ecuaciones	xiv
Índice de Anexos	xv
RESUMEN	xvi
ABSTRACT	xvii
CAPÍTULO I	
1 VISIÓN ARTIFICIAL	1
1.1 INTRODUCCIÓN	1
1.2 CONCEPTO DE VISIÓN ARTIFICIAL	1
1.3 APLICACIÓN DE LA VISIÓN ARTIFICIAL	2
1.4 ETAPAS DE LA VISIÓN ARTIFICIAL	3
1.5 ADQUISICIÓN DE IMÁGENES	4
1.5.1 CÁMARA	4
1.5.2 DIGITALIZADOR	5
1.5.2.1 Muestreo	5
1.5.2.2 Cuantización	6
1.5.3 SISTEMAS DE ILUMINACIÓN	6
1.5.3.1 Iluminación Posterior	6
1.5.3.2 Iluminación Frontal	7
1.5.3.3 Luz Direccional	7
1.5.3.4 Luz Estructurada	8
1.6 FUENTE DE ILUMINACIÓN	9
1.7 PROCESAMIENTO DE LA IMAGEN	9
1.7.1 CONVERSIÓN DE UNA IMAGEN EN FORMATO DE COLOR RGB, A NIVELES DE GRIS	10
1.7.2 CONVERSIÓN DE UNA IMAGEN EN NIVELES DE GRIS A BLANCO Y NEGRO	11
1.7.3 HISTOGRAMA	12
1.8 SEGMENTACIÓN	13
1.8.1 SEGMENTACIÓN BASADA EN PÍXELES	13
1.9 REPRESENTACIÓN Y DESCRIPCIÓN	15

1.9.1	DESCRIPTOR DE COLOR	16
1.9.1.1	Modelo RGB (Red, Green, Blue)	17
1.9.1.2	MODELO YCBCR	17
1.9.1.3	MODELO XYZ	18
1.10	RASGOS GEOMÉTRICOS DE UN OBJETO	19
1.10.1	ÁREA	19
1.10.2	PERÍMETRO	19
1.10.3	FACTOR DE COMPASIDAD	19
1.11	TÉCNICAS DE RECONOCIMIENTO DE PATRONES Y MODELOS DE CLASIFICACIÓN	20
1.12	MODELO DE CLASIFICADORES	21
1.12.1	APRENDIZAJE SUPERVISADO	21
1.12.2	MODELO PARAMÉTRICO	22
1.14.3	MODELO NO PARAMÉTRICO	22

CAPÍTULO II

2	DESARROLLO DEL SISTEMA DE VISIÓN ARTIFICIAL	23
2.1	INTRODUCCIÓN	23
2.2	IMPLEMENTACIÓN DEL HARDWARE	23
2.2.1	CÁMARA	23
2.2.2	SISTEMA DE ILUMINACIÓN	25
2.3	IMPLEMENTACIÓN DEL SOFTWARE	26
2.3.1	CAPTURA DE LA IMAGEN	26
2.3.2	PREPROCESAMIENTO Y SEGMENTACIÓN DE LA IMAGEN	28
2.3.2.1	Conversión a Niveles de Grises	29
2.3.2.2	Filtrado	30
2.3.2.3	Umbralización	31
2.3.2.4	Filtro para la Extracción de Bordes	31
2.3.2.4.1	Descripción de la detección de Bordes	32
2.3.2.5	Descripción del Método Sobel	33
2.4	DESCRIPTORES DE REGIONES	35
2.4.1	ÁREA	35
2.4.2	PERÍMETRO	35
2.5	PROPIEDADES DE OBJETO	36
2.5.1	ALGORITMOS DE RECONOCIMIENTO	36
2.5.2	SELECCIÓN DE OBJETOS	37
2.5.3	TOMA DE DECISIONES	38
2.6	SELECCIÓN DE TÉCNICA PARA CLASIFICACIÓN POR COLOR	39
2.6.1	MODELO HSV	40
2.6.2	ECUACIÓN DE CONVERSIÓN DE RGB A HSV	41
2.7	HMI DE CAPTURA Y CLASIFICACIÓN DE IMÁGENES	43
2.7.1	DESCRIPCION DE PARTES CONSTITUTIVAS DEL HMI	44

CAPÍTULO III

3	IMPLEMENTACIÓN DEL MÓDULO DE CLASIFICACIÓN DE PIEZAS MECANIZADAS	45
3.1	INTRODUCCIÓN	45
3.2	DESARROLLO DEL PROCESO	45
3.3	COMPONENTES NEUMÁTICOS DEL MÓDULO	46
3.4	DIMENSIONAMIENTO DEL CILINDRO NEUMÁTICO DISPENSADOR DE PIEZAS	46
3.4.1	LONGITUD DE CARRERA	47
3.4.2	AIRE DE ALIMENTACIÓN AL SISTEMA NEUMÁTICO	49
3.4.3	SELECCIÓN DE ELECTROVALVULAS	50
3.5	ALIMENTADOR DE PIEZAS	52
3.6	CRITERIO DE SELECCIÓN DE COMPONENTES DE LA BANDA TRANSPORTADORA	54
3.6.1	MOTOR	55
3.6.1.1	CÁLCULO DE LA POTENCIA DEL MOTOR	56
3.6.2	RODILLOS	61
3.7	BANDA TRANSPORTADORA	62
3.7.1	CARACTERÍSTICAS DEL PROTOTIPO DE LA BANDA TRANSPORTADORA	63
3.8	DEPOSITO CIRCULAR CLASIFICADOR DE PIEZAS	65
3.8.1	ESPECIFICACIONES TÉCNICAS DEL MOTOR	66
3.9	DISEÑO ELECTRÓNICO DE CONTROL	69
3.9.1	CONTROL DEL SISTEMA	69
3.9.2	DISEÑO DEL PROGRAMA PRINCIPAL	70
3.9.3	SENSO DE POSICIONAMIENTO DE PIEZAS	70
3.9.4	SENSOR DETECTOR DE PIEZAS METÁLICAS	71
3.9.5	PROTOCOLO DE INTERCAMBIO DE DATOS	72
3.10	DIAGRAMAS DE CIRCUITO DE POTENCIA	72
3.11	CONTENIDO DE ELEMENTOS EN EL PANEL DE MANDO Y CONTROL	74
3.12	FUENTE DE PODER	75
3.13	DESCRIPCIÓN DE PUERTOS UTILIZADOS	76
3.14	IMPLEMENTACIÓN COMPLETA DEL MÓDULO RECONOCIMIENTO Y CLASIFICACIÓN	77
 CAPÍTULO IV		
4	PRUEBAS DE FUNCIONAMIENTO DEL SISTEMA	78
4.1	INTRODUCCIÓN	78
4.2	PRUEBA DE FUNCIONAMIENTO DE TARJETA CONTROLADORA	78
4.3	PRUEBA DE FUNCIONAMIENTO DE SENSORES	79
4.4	PRUEBAS DEL SISTEMA COMPLETO	79
4.4.1	RESULTADO EN LA ADQUISICIÓN DE LA IMAGEN	80
4.4.2	MEDICIONES DE DESEMPEÑO REAL DEL SISTEMA	82
4.4.3	DETECCIÓN DE COLORES	82
4.4.4	PRUEBA POR FORMA GEOMÉTRICA	83

4.4.5	PRUEBAS POR TIPO DE MATERIAL	85
4.4.6	PRUEBAS DEL DEPÓSITO CLASIFICADOR	85
4.5	ANÁLISIS DE RESULTADOS	86
4.6	ANÁLISIS DE RENDIMIENTO	87
CAPÍTULO V		
5	MANUAL DE USO	88
5.1	INTRODUCCIÓN	88
5.2	CONDICIONES PARA LA INSTALACIÓN DEL SOFTWARE	88
5.3	PERFIL DEL USUARIO	89
5.4	INTERFACE USUARIO DEL SISTEMA DE VISIÓN	89
5.4.1	ESTADO MANUAL DEL SISTEMA	90
5.5	INFORMACIÓN DE SEGURIDAD	91
5.6	MANTENIMIENTO DEL MÓDULO DIDÁCTICO	91
5.6.1	MANTENIMIENTO AL SISTEMA NEUMÁTICO	92
5.6.2	MANTENIMIENTO DE LA BANDA TRANSPORTADORA	92
5.7	ESPECIFICACIONES DE PINES UTILIZADOS EN LA TARJETA ARDUINO	93
5.8	DESARROLLO DE GUÍA DE PRÁCTICAS	94
5.8.1	ESQUEMATIZACIÓN DE LAS PRÁCTICAS	94
5.9	PRÁCTICA Nº 1	95
5.9.1	OBJETIVO	95
5.9.2	EQUIPO REQUERIDO	95
5.9.3	PROCEDIMIENTO	95
5.9.4	VARIABLES A UTILIZAR	96
5.9.5	CONCLUSIONES Y RECOMENDACIONES	96
5.10	PRÁCTICA Nº 2	96
5.10.1	OBJETIVO	96
5.10.2	EQUIPO REQUERIDO	97
5.10.3	PROCEDIMIENTO	97
5.10.4	VARIABLES A UTILIZAR	97
5.10.5	FUNCIONAMIENTO Y CONCLUSIONES	97
5.11	PRÁCTICA Nº 3	98
5.11.1	OBJETIVO	98
5.11.2	EQUIPO REQUERIDO	98
5.11.3	PROCEDIMIENTO	98
5.11.4	VARIABLES A UTILIZAR	99
5.11.5	FUNCIONAMIENTO Y CONCLUSIONES	99
5.12	PRÁCTICA Nº 4	99
5.12.1	OBJETIVO	99
5.12.2	EQUIPO REQUERIDO	100
5.12.3	PROCEDIMIENTO	100
5.12.4	VARIABLES A UTILIZAR	100
5.12.5	FUNCIONAMIENTO Y CONCLUSIONES	101
5.13	PRÁCTICA Nº 5	101
5.13.1	OBJETIVO	101

5.13.2	EQUIPO REQUERIDO	101
5.13.3	PROCEDIMIENTO	102
5.13.4	VARIABLES A UTILIZAR	102
5.13.5	FUNCIONAMIENTO Y CONCLUSIONES	102
	CONCLUSIONES Y RECOMENACIONES	103
	BIBLIOGRAFIA	105
	ANEXOS	106

ÍNDICE DE TABLAS

Tabla 1	Cálculo de Factores de Compacidad de figuras	21
Tabla 2	Características del Cilindro Alimentador de Piezas	53
Tabla 3	Calidad del Aire Recomendado	54
Tabla 4	Características de Electroválvulas Seleccionadas	56
Tabla 5	Datos del Motor Eléctrico	59
Tabla 6	Características de la placa ARDUINO	73
Tabla 7	Elementos del Panel de Control	79
Tabla 8	Características de la Fuente de Voltaje	80
Tabla 9	Identificación de Entradas y Salidas	80
Tabla 10	Resultado de Prueba en Tarjeta Controladora	83
Tabla 11	Prueba de Sensores	84
Tabla 12	Resultado de la Pruebas de Colores	87
Tabla 13	Condiciones de Matlab	88
Tabla 14	Resultados de Command Windows de Matlab	88
Tabla 15	Resultados de las Pruebas de Formas Geométricas	89
Tabla 16	Resultados de Pruebas de Tipo de Material	89
Tabla 17	Resultados de las Pruebas por Deposito Clasificador	90
Tabla 18	Análisis de Resultados	91

ÍNDICE DE FIGURAS

Figura 1	Diagrama de Bloques de las Etapas en un Sistema de Visión Artificial	4
Figura 2	Iluminación Posterior	7
Figura 3	Iluminación Frontal	7
Figura 4	Iluminación Direccional	8
Figura 5	Iluminación Estructurada	8
Figura 6	Representación de una imagen a color	10
Figura 7	Representación de una imagen a formato de grises	11
Figura 8	Representación de una imagen en binaria	12
Figura 9	Diferentes tipos de histogramas	13
Figura 10	Segmentación de la imagen	15
Figura 11	Distribución de colores en el cubo RGB	17
Figura 12	Distribución de color YCbCr	18
Figura 14	Cámara Utilizada	24

Figura 15	Manguera LED Blanco Frio	25
Figura 16	Descripción de Procesos que Realiza el Programa	26
Figura 17	Imágenes Capturadas por la Web Cam	28
Figura 18	Representación de una Imagen en Color con sus tres matrices de intensidades	29
Figura 19	Fotografía a Escala de Grises	29
Figura 20	Imagen Binarizada	31
Figura 21	Imágenes Resultados de la Aplicación del Logaritmo de Sobel.	33
Figura 22	Mascara del detector sobel	34
Figura 23	Representación matricial de la imagen	34
Figura 24	Detección de bordes con el método sobel	34
Figura 25	Diagrama de Flujo para la Clasificación por Forma	40
Figura 26	Diagrama de Flujo para la Clasificación por Color	41
Figura 27	Representación Gráfica del Modelo de Color RGB	41
Figura 28	Modelo de Imagen HSV	42
Figura 29	Captura de Imagen de HMI	43
Figura 30	Diagrama del Módulo Didáctico	46
Figura 31	Cilindro Neumático	48
Figura 32	Dispositivo que Permite Purificar el Aire	51
Figura 33	Electroválvula 3/2 para el Accionamiento del Cilindro de Ingreso de Piezas	51
Figura 34	Dispensador de Piezas	53
Figura 35	Motor Reductor	55
Figura 36	Rodamientos de la Banda Transportadora	61
Figura 37	Banda transportadora	62
Figura 38	Fuerzas que Intervienen en el Eje Motriz de la Banda Transportadora	63
Figura 39	Servomotor POWER HN-6001MG con Engranaje de Acero	66
Figura 40	Depósito Clasificador de Piezas	66
Figura 41	Placa ARDUINO UNO	67
Figura 42	Descripción del Proceso del Módulo	70
Figura 43	Diagrama esquemático del sensor qrd1114 Diagrama de conexión del sensor hacia la tarjeta controladora	71
Figura 44		71
Figura 45	Sensor Sn04-n física e internamente	72
Figura 46	Diagrama Esquemático del Circuito de Potencia	73
Figura 47	Panel de control del Módulo	77
Figura 48	Ensamblaje completo del módulo	79
Figura 49	Adquisición de la Imagen	80
Figura 50	Resultado de las Imágenes Captadas	81
Figura 51	Imágenes mal Captadas	82
Figura 52	Descripción de Regiones del HMI	86

Figura 53	Especificaciones de Conexiones de Borneras	94
-----------	--	----

ÍNDICE DE ECUACIONES

Ecuación 1.1	La Obtención de Este Espacio de Color a Partir del RGB	18
Ecuación 1.2	Componentes RGB	19
Ecuación 1.3	Cálculo del Área	19
Ecuación 1.4	Cálculo del Perímetro	19
Ecuación 1.5	Cálculo del Factor de Compacidad	19
Ecuación 2.1	Intensidad de una Imagen	30
Ecuación 2.2	Derivada de primer orden para una imagen	33
Ecuación 2.3	La Magnitud del Vector	33
Ecuación 2.4	Simplificando la Ecuación 2.3	33
Ecuación 2.5	Obtención de su Valor Absoluto	33
Ecuación 2.6	Método Sobel	33
Ecuación 2.7	Coeficiente de Matrices	33
Ecuación 2.8	Cálculo del Coeficiente	38
Ecuación 2.9	Ecuación de Conversión de RGB a HSV	41
Ecuación 2.10	Ecuación de Conversión de HSV	41
Ecuación 3.1	Fuerza Teórica del Embolo	48
Ecuación 3.2	Área del Embolo	48
Ecuación 3.3	Cálculo de la Tensión del Motor	56
Ecuación 3.4	Cálculo del Coeficiente de Fricción	56
Ecuación 3.5	Tensión necesaria para mover una carga horizontal	57
Ecuación 3.6	Coeficiente de fricción	57
Ecuación 3.7	Razón de carga	57
Ecuación 3.8	Velocidad de banda transportadora	58
Ecuación 3.9	Velocidad angular	58
Ecuación 3.10	Tensión Efectiva	59
Ecuación 3.11	Potencia del motor	59
Ecuación 3.12	Potencia del motor en vatios	60
Ecuación 3.13	Relación torque velocidad	60
Ecuación 3.14	Fuerza que se transfiere del engranaje al motor	60
Ecuación 3.15	Cálculo para el Diseño de Rodillos	61
Ecuación 3.16	Cargas radiales	62
Ecuación 3.17	Cálculo del Esfuerzo Cortante Torsional	64
Ecuación 3.18	Cálculo del Momento Polar de Inercia	64

INDICE DE GRÁFICOS

Gráfico 1	Histograma de una Imagen 2	12
Gráfico 2	Histograma de una Botella	14

ÍNDICE DE ANEXOS

ANEXO1. RGB de la tabla de colores HSV	116
ANEXO 2. Catalogo FESTO de cilindros neumáticos	117
ANEXO 3. Norma <i>DIN ISO 8573-1</i>	118
ANEXO 4. Diagramas de cortes y momentos en el eje XY	119
ANEXO 5. Programa realizado en arduino	120
ANEXO 6. Circuito impreso de la placa de control	121
ANEXO 7. Características del sensor inductivo	122
ANEXO 8. Planos de construcción del módulo didáctico	123
ANEXO 9. Diagrama eléctrico del módulo didáctico	124
ANEXO10. Programa principal realizado en matlab	125

RESUMEN

En la formación profesional, se ve la necesidad de aumentar experiencia a los estudiantes, por medio de laboratorios que posean módulos para realizar prácticas de simulación de procesos industriales y automatización; pero debido a la falta de tecnológicas se ve necesario implementar material didáctico con el cual se tenga la oportunidad de desarrollar las destrezas que complementan los conocimientos teóricos. Con la implementación de este módulo didáctico de reconocimiento y clasificación de piezas se contribuirá a la Facultad a fortalecer el aprendizaje de sus estudiantes. El modulo consta de un cilindro neumático que transfiere 3 tipos de piezas de diferente forma geométrica y distinto color (cilíndricas (rojo), cubicas (azul), piramidales (negro)) de 50mm, a una banda transportadora, sobre la que estará la cámara web inspeccionando las piezas mediante un sistema de visión artificial con procesamiento digital de imágenes, determinando el estado las características determinadas en el programa de Matlab. Una vez llevados a cabo los procesos de alimentación de piezas desde el depósito hacia la banda transportadora, y su respectiva inspección, se procede a la clasificación de las mismas en un depósito colector circular con seis divisiones adaptado al eje de un servomotor que estará al final, el cual se lo posicionó en el ángulo deseado para la ubicación de las piezas según las especificaciones de la práctica. También están ubicados de manera estratégica en el módulo sensores de tipo inductivo para que identifiquen el tipo de material de cada pieza (madera, metal). El motor de la banda transportadora es de velocidad variable el cual sirve para realizar prácticas que simulen el montaje de sistemas mecatrónicos complejos con fines de control de flujo del material en procesamiento. Incluye un tablero de control o de mando el cual contiene los indicadores y pulsadores necesarios para el funcionamiento como: pulsadores marcha/paro, lámparas indicadoras. El control de las operaciones necesarias para el modulo se lo realizara mediante una DAC (Arduino), se dispondrá de una guía de usuario para facilitar su funcionamiento para lograr el mayor provecho posible. En las industrias ecuatorianas se está implementando el uso de detectores para sistemas de automatización de procesos de fabricación, los sistemas de procesamiento de imágenes representan el sector más innovador. Lo que se ve reflejado en el funcionamiento y manual de este módulo didáctico implementado para esos fines.

ABSTRACT

In vocational training, is the need to increase experience to students, through laboratories possessing practical modules for simulation and automation of industrial processes, but due to lack of technology is necessary to implement training materials with which will have the opportunity to develop skills that complement theoretical knowledge. With the implementation of this training module recognition and classification of pieces will contribute to the School to strengthen student learning. The module consists of a pneumatic cylinder that transfers 3 types of pieces of different geometrical shape and different color (cylindrical (red), cubic (blue) pyramidal (black)) 50mm , to a conveyor belt on which the camera is web inspecting parts using a machine vision system with digital image processing , determining the status of certain features in the program Matlab. Once the processes carried out parts feeding from the reservoir to the conveyor, and the respective inspection , proceed to sorting them on a circular manifold tank six divisions adapted to the spindle of a servomotor which is ultimately , which is positioned in the desired location of the parts to the specifications of the practice angle. They are also strategically located in the inductive sensor module to identify the type of material of each piece (wood, metal). The conveyor motor is variable speed which serves for convenient mounting simulating complex mechatronic systems for control purposes in material flow processing. Includes a control panel or command which contains buttons and indicators necessary for the operation as push buttons for start / stop, indicator lamps. The control of the operations necessary for the module would be via a DAC (Arduino) will provide a user guide for easy operation to achieve the best of it. In the Ecuadorian industries are implementing the use of sensors for process automation systems for manufacturing, image processing systems represent the most innovative sector. What is reflected in this operation manual and training module implemented for this purpose.

CAPÍTULO 1

VISIÓN ARTIFICIAL

1.1 INTRODUCCIÓN

En el presente capítulo se describe los fundamentos teóricos que permitan dar una mayor comprensión y selección de los elementos necesarios para la automatización del proyecto.

1.2 CONCEPTO DE VISIÓN ARTIFICIAL

La Visión Artificial VA, trata de simular el proceso del sentido de la visión de los seres vivos, según el cual un objeto es captado mediante los receptores de la retina y es transformado en impulsos nerviosos que son procesados por el cerebro; consistiendo en la captación de imágenes en línea mediante cámaras basadas en matrices de sensores sensibles a la luz (CCD o CMOS), y el posterior tratamiento de las mismas mediante técnicas de análisis de imagen y la actuación sobre el proceso. Las principales ventajas son la reducción de los costos del control de calidad, el aumento de la producción, la inspección del 100% de la producción y el funcionamiento 24 horas al día sin pares en la producción donde se ha implementado el sistema.(Rojas. 2006).

La captura de la imagen es el conjunto de operaciones que se efectúan para transformar la iluminación de una escena en una señal digital.

Las imágenes no siempre se presentan en un formato adecuado para su análisis, por lo que los siguientes pasos son el pre-procesamiento de una imagen. Donde se utilizan técnicas encaminadas a realizar la mejora de la imagen, como son el nivel

de gris, contraste, eliminación del ruido, el realce de algunas características de interés, etc. Una vez que ésta imagen está en condiciones de ser procesada, se tienen que hallar los objetos dentro de la imagen de forma independiente, y esto se hace a través de la segmentación, que es un proceso que divide la escena en objetos. Cada uno de los objetos puede ser clasificado, por lo que la siguiente tarea es la clasificación o extracción de características para el reconocimiento. El reconocimiento es la identificación de cada objeto en la escena mediante una etiqueta. (Sobrado, 2003).

1.3 APLICACIONES DE LA VISIÓN ARTIFICIAL

El amplio espectro de aplicaciones cubierto por la VA, se debe a que permite extraer y analizar información espectral, espacial, temporal de los distintos objetos. La información espectral incluye frecuencia (color) e intensidad (tonos de gris). La información espacial se refiere a aspectos como forma y posición (una, dos y tres dimensiones). La información temporal comprende aspectos estacionarios (presencia y/o ausencia) dependientes del tiempo (eventos, movimientos, procesos). La mayoría de las aplicaciones de la visión artificial se clasifica por el tipo de tarea, entre las que se menciona a continuación: La medición o calibración se refiere a la correlación cuantitativa con los datos del diseño, asegurando que las mediciones cumplan con las especificaciones del diseño. Por ejemplo, el comprobar que un cable tenga el espesor recomendado, y la detección de fallas es un análisis cualitativo que involucra la detección de defectos o artefactos no deseados, con forma desconocida en una posición desconocida. Por ejemplo, encontrar defectos en la pintura de un auto nuevo, o agujeros en hojas de papel. (Gonzales, 2002). La verificación es el chequeo cualitativo de que una operación de ensamblaje ha sido llevada a cabo correctamente. Por ejemplo, que no falte ninguna tecla en un teclado, o que no falten componentes en un circuito impreso. (Sandoval, 2010). El reconocimiento involucra la identificación de un objeto con base en descriptores asociados con el objeto. Por ejemplo, la clasificación de cítricos (limones, naranjas, mandarinas, etc.) por color y tamaño. Las inspecciones realizadas por los seres humanos, a menudo no pueden cumplir con los requisitos de la industria moderna

respecto a la velocidad de producción, calidad de producto y costos de producción. Los humanos se cansan, cometen errores y los criterios que se aplican durante las inspecciones son inevitablemente subjetivos. En algunos casos, no es humanamente posible llevar a cabo las tareas de inspección debido a las condiciones ambientales. Las cámaras y los sistemas que componen un sistema de Visión Artificial, por el contrario, llevan a cabo las mediciones con una precisión constante y a un ritmo que es establecido por el propio proceso de producción. Estas ventajas han llevado a una creciente aceptación de la visión por industrias de todo el mundo. (Sobrado, 2003).

1.4 ETAPAS DE LA VISIÓN ARTIFICIAL

Se ha visto que el ser humano captura la luz a través de los ojos, y que esta información circula a través del nervio óptico hasta el cerebro donde se procesa. Existen razones para creer que la primera etapa de este proceso consiste en encontrar elementos más simples en los que se descompone la imagen en segmentos y arcos. Después el cerebro interpreta la escena y por último actúa en consecuencia.

La visión artificial, en un intento de reproducir este comportamiento, define tradicionalmente cuatro fases principales: 1) La primera fase, que es puramente sensorial, consiste en la *captura* o adquisición de las imágenes digitales mediante algún tipo de sensor. 2) La segunda etapa consiste en el tratamiento digital de las imágenes, con objeto de facilitar las etapas posteriores. En esta etapa de *procesamiento previo* es donde, mediante filtros y transformaciones geométricas, se eliminan partes indeseables de la imagen o se realzan partes interesantes de la misma. 3) La siguiente fase se conoce como *segmentación*, y consiste en aislar los elementos que interesan de una escena para comprenderla. 4) Por último se llega a la etapa de *reconocimiento* o *clasificación*. Figura 1. En ella se pretende distinguir los objetos segmentados, gracias al análisis de ciertas características que se establecen previamente para diferenciarlos. Estas cuatro

fases no se siguen siempre de manera secuencial, sino que en ocasiones deben realimentarse hacia atrás. (Reynoso, 2002)



Figura 1. Diagrama de bloques de las etapas en un sistema de visión artificial.

Fuente: Autor

1.5 ADQUISICIÓN DE IMÁGENES

Al tratar los aspectos más relevantes del proceso de *captura* y *digitalización* de una imagen, esto es, la adquisición de la imagen del mundo físico y su paso al dominio discreto y virtual informático. Una vez digitalizada una imagen bidimensional digital está constituida por un conjunto de elementos llamados *píxeles*. Cada píxel ofrece cierta información sobre una región elemental de la imagen. En imágenes en niveles de gris esta información es el brillo. En imágenes en color, la información corresponde a la intensidad de cada una de las componentes de una base de color. (García, 2009).

1.5.1 CÁMARA.

Es el dispositivo encargado de transformar las señales luminosas que aparecen en la escena, en señales analógicas capaces de ser transmitidas por un cable coaxial. Se divide en dos partes, el sensor, que captura las propiedades del objeto en forma de señales luminosas y lo transforma en señales analógicas, y la óptica que se encarga de proyectar los elementos adecuados de la escena ajustando una

distancia focal adecuada. Las cámaras de visión artificial son más sofisticadas que las convencionales, pues brindan un completo control de los tiempos, señales de video, velocidad de captura y sensibilidad. Estos son factores fundamentales tanto en aplicaciones científicas como industriales.

1.5.2 DIGITALIZACIÓN

Vélez y Moreno (2003) señalan que es el proceso de paso del mundo continuo (o analógico) al mundo discreto (o digital). En la digitalización normalmente se distinguen dos procesos: el muestreo y la cuantización.

1.5.2.1 Muestreo

El muestreo de una señal continua consiste en la medición a intervalos (discretización) respecto de alguna variable (generalmente el tiempo o el espacio), siendo su parámetro fundamental la frecuencia de muestreo, que representa el número de veces que se mide un valor analógico por unidad de cambio.

Mediante el muestreo se convierte una imagen I_C , que es algo continuo, en una matriz discreta I_D de $N \times M$ píxeles. El número de muestras por unidad de espacio sobre el objeto original conduce al concepto de resolución espacial de la imagen.

Ésta se define como la distancia, sobre el objeto original, entre dos píxeles adyacentes. Sin embargo la unidad de medida de resolución espacial más habitual suele ser los píxeles por pulgada, siempre medidos sobre el objeto original.

1.5.2.2 Cuantización

La segunda operación es la Cuantización de la señal, que consiste en la discretización de los posibles valores de cada píxel. Los niveles de Cuantización

suelen ser potencias de 2 para facilitar el almacenamiento en el computador de las imágenes, ya que éstos utilizan el byte10 como unidad mínima de memoria directamente. Vélez y Moreno (2003)

1.5.3 SISTEMAS DE ILUMINACIÓN

La iluminación es el aspecto más decisivo de cualquier aplicación de visión artificial. Eligiendo la técnica adecuada de iluminación se puede lograr un aumento en la exactitud, en la fiabilidad del sistema y en su tiempo de respuesta. Es un error muy serio y costoso asumir que se puede compensar una iluminación inadecuada con un algoritmo. Existen reglas bien establecidas para la elección de un tipo de lente. Por el contrario, son mucho menos las reglas establecidas para la elección de una buena iluminación, aun cuando esta última es tan importante como la elección adecuada de la lente para obtener imágenes utilizables. Para que una característica aparezca en una imagen, la luz debe venir de la fuente de iluminación, reflejarse en el objeto y ser recolectada por la lente. (Sánchez, 2011).

1.5.3.1 Iluminación Posterior.

La iluminación posterior permite delinear el contorno de las piezas, minimiza la visibilidad de gabinetes traslúcidos y permite visualizar perforaciones pasantes. Esta técnica brinda un contraste muy alto y es fácil de implementar. Dentro de las técnicas de iluminación posterior se pueden encontrar la difusa y la direccional. En la iluminación posterior difusa (figura 2 a) los rayos de luz se transmiten en diversos ángulos. En la iluminación posterior direccional (figura 2 b), un colimador o sistema de luz divergente hace que todos los rayos de luz se propaguen en direcciones paralelas.

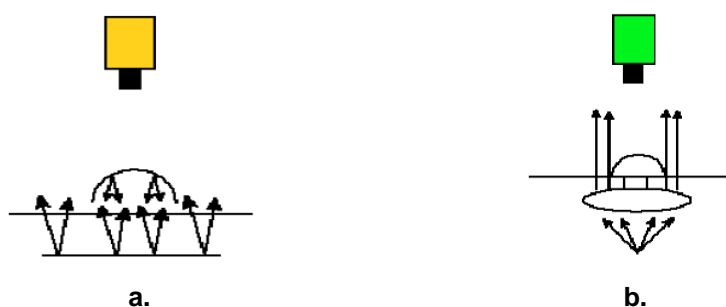


Figura 2. Iluminación Posterior:

a) Iluminación posterior difusa. b.) Iluminación posterior direccional.

Fuente: Sobrado, (2006).

1.5.3.2 Iluminación Frontal

Es la más usada, y consiste en iluminar frontalmente la pieza. Presenta más problemas para obtener un buen contraste entre la pieza y el fondo, debido a la aparición de brillos y sombras que alteran las propiedades de las piezas a estudio como se observa en la figura 3, se emplea en piezas poco reflectoras para evitar los brillos que son bastante molestos, usándose una iluminación difusa muy estudiada para piezas muy reflectoras.

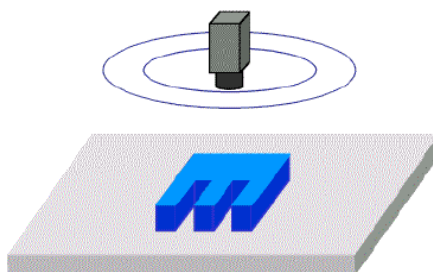


Figura 3. Iluminación Frontal.

Fuente: Sobrado, (2006).

1.5.3.3 Iluminación Direccional

Consiste en una iluminación direccionada en algún sentido en el espacio para destacar una característica concreta del objeto así como se muestra en la figura 4. La principal virtud es la creación de sombras sobre el objeto, lo que puede ayudar a aumentar el contraste de partes tridimensionales y obtener la consiguiente información 3D.

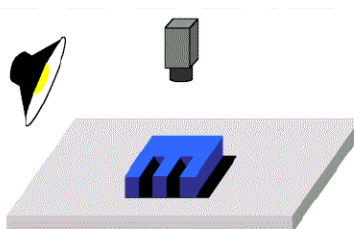


Figura 4. Iluminación Direccional

Fuente: Sobrado, (2006).

1.5.3.4 Iluminación Estructurada

Consiste en proyectar sobre la pieza unos patrones de luz conocidos modulados (proyección de puntos, franjas o rejillas sobre la superficie de trabajo) y observando la luz reflejada, que también viene modulada, obtener información sobre la estructura de la superficie del objeto, la cual puede ser reconstruida mediante triangulación, como se mira en la figura 5.

Las fuentes de luz empleadas deben de ser especiales pues deben ser capaces de emitir luz estructurada y suelen ser láseres. Se usa para reconstrucciones 3D de objetos y para conocer su forma.

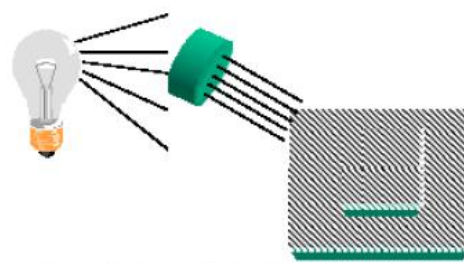


Figura 5. Iluminación Estructurada.

Fuente: Sobrado, (2006).

1.6 FUENTES DE ILUMINACIÓN.

Es de gran importancia prestar atención a las diferentes fuentes de iluminación.

- a. Lámparas incandescentes.- Es la fuente de iluminación más común y consiste en un filamento de tungsteno o halógeno-tungsteno. Como ventaja tiene que existe gran variedad de potencias y como desventaja, que reduce su luminosidad con el tiempo, lo que puede provocar problemas en algunos sistemas de visión.

- b. Tubos fluorescentes.- Más eficaces que las lámparas y suministran una luz más difusa, que es bueno para piezas muy reflectoras. Existe una gran variedad, tanto en forma (circulares, lineales), como en tamaño con lo que son ampliamente utilizados.

- c. Fibra óptica.- Para iluminar zonas de difícil acceso o extremadamente pequeñas. Proporciona iluminación constante.

- d. Láser.- Empleados para una iluminación con luz estructurada, ya que el láser es capaz de emitir luz estructurada con un control adecuado. Tiene el inconveniente de presentar un mal comportamiento frente a superficies que absorben luz. (Cartagena, 2005).

1.7 PROCESAMIENTO DE LA IMAGEN

Al obtener una imagen a través de cualquier sistema de captura, muchas de las veces ésta no se la utiliza directamente por el sistema de visión, debido a que existen varios factores los cuales dificultan la buena resolución de la imagen tales como: variaciones en intensidad debidas al ruido, por deficiencias en la iluminación o la obtención de imágenes de bajo contraste. Por estos motivos es necesario un pre-procesamiento de la imagen con el objetivo de corregir estos problemas, además de aplicar aquellas transformaciones a la imagen que acentúen las

características que se deseen extraer de las mismas, de manera que se facilite las operaciones de las etapas posteriores. (García, 2008).

Pudiendo ser mencionadas las diferentes técnicas empleadas para el mejoramiento de la imagen y las transformaciones necesarias para la aplicación de estas técnicas.

1.7.1 CONVERSIÓN DE UNA IMAGEN EN FORMATO DE COLOR RGB (RED-GREEN-BLUE) A NIVELES DE GRIS

En el proceso de captación de imagen, se obtuvo una imagen digital en el espacio de color RGB con sus tres matrices de intensidades para los colores rojo, verde y azul respectivamente, ver figura 6, muestra las tres matrices correspondientes únicamente al recuadro de la imagen con intensidades para cada color que van del 0 al 255.



Figura 6. Representación de una imagen a color.

Fuente: (Autor)

El proceso de conversión a niveles de gris consiste en calcular el promedio de cada intensidad para las matrices de los colores rojo, verde y azul. Cada valor de la conversión se redondea para formar una matriz de intensidades donde el valor correspondiente a cada punto indica el valor de intensidad que puede ir de [0 a

255]; el cero representa el negro absoluto y el 255 el blanco absoluto, como se mira en la figura 7.



Figura 7. Representación de una imagen a formato de grises.

Fuente: (Autor)

1.7.2 CONVERSIÓN DE UNA IMAGEN EN NIVELES DE GRIS A BLANCO Y NEGRO (BINARIZACIÓN)

Con esta técnica, los valores de píxel en la imagen de entrada menores a un cierto umbral (entre 0 y 255) son convertidos a negro mientras que los píxeles con valores mayores o iguales al umbral son convertidos a blanco. En la figura 8 se muestra la transformación de la imagen en escala de gris a blanco y negro. El valor de la intensidad mayor o igual a un umbral (127) se les asigna el máximo valor de gris (255=blanco). Por su parte, los píxeles comprendidos por debajo de este valor toman el valor mínimo (0=negro).



Figura 8. Representación de una Imagen en Binaria.
Fuente: (Autor).

1.7.3 HISTOGRAMA

El histograma de una imagen es una función que representa el número de píxeles existentes para cada valor de nivel de gris. Para una imagen con 256 niveles de gris posibles, la abscisa del histograma varía entre 0 y 255 (Gráfico 1). (Córdova y Romero, 2005).

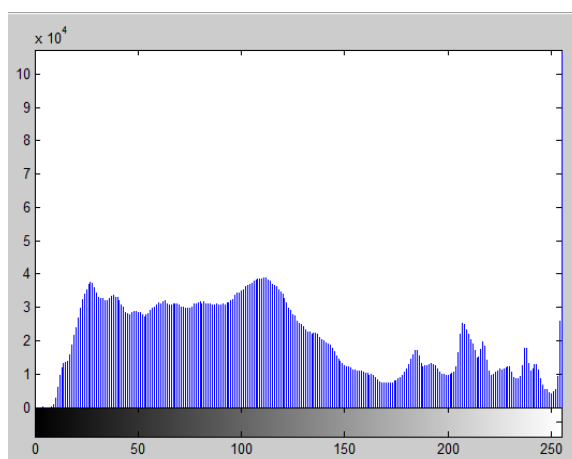


Gráfico 1. Histograma de la figura 6
Fuente: Autor

Un histograma con la gráfica hacia la izquierda resulta muy oscuro, si está hacia la derecha contiene mucho brillo y si se encuentra al centro tiene poco contraste. La representación de un histograma ideal es una recta horizontal, porque nos indica

que todos los posibles valores de grises están distribuidos de manera uniforme en la imagen (figura 9).

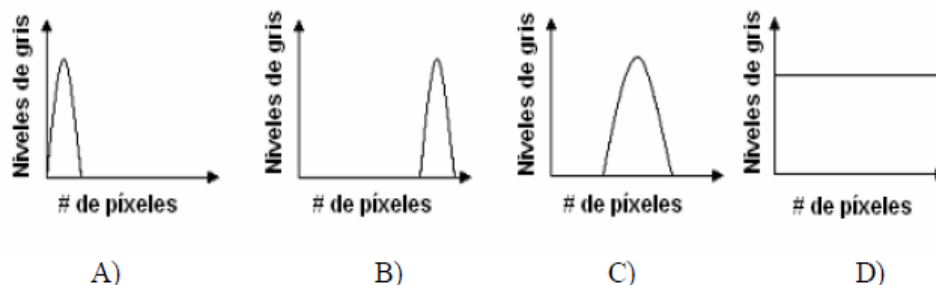


Figura 9. Diferentes tipos de histogramas: A) imagen muy oscura, B) imagen muy clara, C) imagen con poco contraste, D) histograma ideal.

Fuente: Córdova y Romero, (2005).

1.8 SEGMENTACIÓN

La segmentación es un proceso que consiste en dividir una imagen digital en regiones homogéneas con respecto a una o más características (como por ejemplo el brillo o el color) con el fin de facilitar un posterior análisis o reconocimiento automático. Localizar la cara de una persona dentro de la imagen de una fotografía o encontrar los límites de una palabra dentro de una imagen de un texto, constituyen ejemplos de problemas de segmentación. (Martínez, 2010).

1.8.1 SEGMENTACIÓN BASADA EN PÍXELES

Este método de segmentación toma en cuenta sólo el valor de gris de un pixel, para decidir si el mismo pertenece o no al objeto de interés. Para ello, se debe encontrar el rango de valores de gris que caracterizan dicho objeto, lo que requiere entonces la búsqueda y el análisis del histograma de la imagen.

El objetivo de este método, es el de encontrar de una manera óptima los valores característicos de la imagen que establecen la separación del objeto de interés, con respecto a las regiones que no pertenecen al mismo, debido a esta característica y si los valores de gris del objeto y del resto de la imagen difieren claramente, entonces el histograma mostrará una distribución bimodal, con dos máximos

distintos, lo que debiera generar, la existencia de una zona del histograma ubicada entre los dos máximos, que no presenten los valores característicos, y que idealmente fuera igual a cero, con lo cual se logrará una separación perfecta entre el objeto y la región de la imagen que lo circunda, al establecer un valor umbral ubicado en esta región del histograma. (Martínez, 2010). Por lo tanto cada pixel de la imagen, es asignado a una de dos categorías, dependiendo si el valor umbral es excedido o no.

Si el valor del histograma ubicado entre los dos máximos, es distinto de cero, las funciones de probabilidad de los valores de gris del objeto y de la región restante se remplazan, de tal manera que algunos pixeles del objeto se toman como pertenecientes a la región circundante y viceversa. Conocida la distribución de la función de probabilidad de los pixeles del objeto y de la región circundante, es posible aplicar análisis estadístico en el proceso de buscar un umbral óptimo, con el número mínimo de correspondencias erróneas. Estas distribuciones pueden ser estimadas por histogramas locales, ver gráfico 1.2, los cuales solamente incluyen las regiones correspondientes de la imagen.

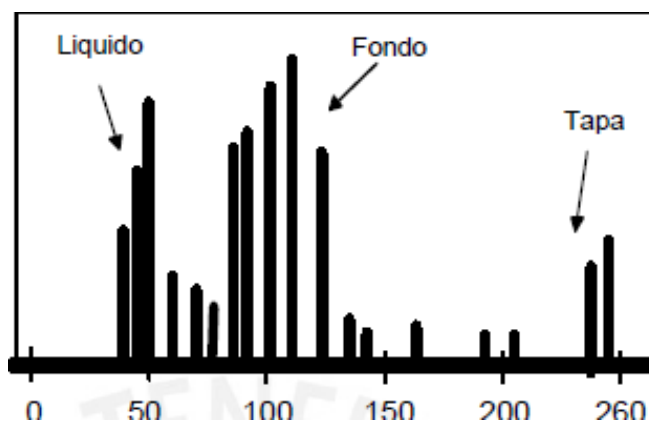


Gráfico 2. Histograma de una botella
Fuente: Gonzales y Woods, (2001).

En imágenes industriales se tiene un alto contraste lo cual permite aplicar segmentaciones muy simples como el mencionado, para distinguir dos clases de regiones en la imagen: región objeto y región fondo como se muestra en la figura 10.



Figura 10. Segmentación de la imagen
Fuente: Gonzales y Woods, (2001)

El indica que el fondo (gris claro) es lo que ocupa más espacio en la imagen, porque es el que contiene mayor número de píxeles en la escala de grises 125 (aprox.). Luego se tiene el líquido (refresco, color negro), la tapa y partes de la etiqueta (color blanco). Si aplicamos un umbral a esta imagen, a partir del límite que establece el líquido (aproximadamente nivel 50 de la escala de grises), todo lo demás se convierte en color negro (0), y el líquido se convierte en color blanco (1). De esta forma, la imagen pasa a ser binaria, y el líquido queda claramente separado de todo lo demás que aparece en la imagen. (Martínez, 2010).

1.9 REPRESENTACIÓN Y DESCRIPCIÓN

Una vez que los objetos de interés han sido aislados, el siguiente paso para el Reconocimiento del objeto es: La representación y descripción. En esta etapa, se extrae a la imagen un conjunto de características que describen las propiedades físicas de los objetos como son: el color, textura, área, perímetro, ancho, alto, promedio de intensidad, centroide, orientación, número de hoyos, entre otros. Las propiedades que deben tener estos descriptores de características son:

- Ser capaces de hacer una adecuada discriminación: Deben proporcionar valores numéricos diferentes para objetos de clases distintas.

- Ser suficientemente fiables: debe de haber cambios numéricos pequeños para objetos de la misma clase.
- Rapidez: El número de descriptores debe ser pequeño y calcularse en un tiempo aceptable.

1.9.1 DESCRIPTOR DE COLOR

El color es una característica importante para la clasificación y descripción de objetos en la visión por computadora. Existen diferentes modelos de color los cuales se describen a continuación:

1.9.1.1 Modelo RGB (Red, Green, Blue)

El modelo RGB (*Red, Green, Blue*), maneja 3 canales, uno para el rojo, uno para el verde y uno para el color azul, la imagen resultante tiene tonalidades que surgen de las combinaciones entre los colores primarios. En esta codificación, por cada píxel se necesitan 24 bits, es decir 3 bytes, codificando cada color primario con un byte (*8 bits*) que pueden tomar 256 valores con un rango de 0 a 255. Los píxeles RGB son tridimensionales, $\text{pixel RGB} = f(R, G, B)$, donde el valor de f es la intensidad del color rojo, verde y azul en las coordenadas (x,y) del píxel RGB. De esta forma los colores se representan en coordenadas cartesianas dentro de un cubo unitario figura 11. (Barrera, 2010).

Cada color se representa como un vector del origen y la diagonal principal corresponde a la escala de grises. En este modelo se basan las cámaras y receptores de televisión.

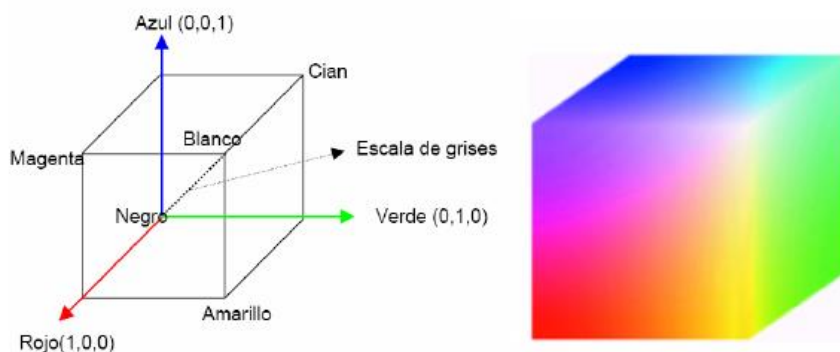


Figura 11. Distribución de colores en el cubo RGB
Fuente: http://www2.elo.utfsm.cl/~elo328/PDI13_Color.pdf

1.9.1.2 Modelo YCBCR

En el modelo YCbCr el color es representado por la luminancia (Y) y por dos valores diferentes de color (Cb y Cr) que son características colorimétricas del color. La luminancia es la cantidad lineal de luz que puede ser calculada como la suma ponderada de los componentes lineales del espacio de color RGB.

La obtención de este espacio de color a partir del RGB es la siguiente de la ecuación (1.1):

$$\begin{aligned}
 Y &= 0.2989 * R + 0.5866 * G + 0.1145 * B \\
 C_b &= -0.1688 * R - 0.3312 * G + 0.5 * B \\
 C_r &= 0.5 * R - 0.4184 * G - 0.08116 * B
 \end{aligned}
 \tag{Ecuación 1.1}$$

Siendo R, G y B son los valores del canal rojo, verde y azul respectivamente como se puede observar en la figura 12.

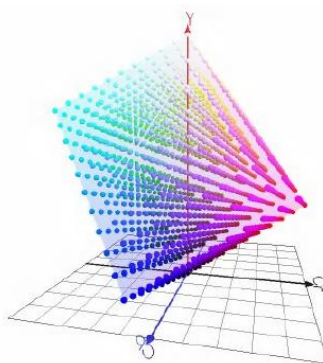


Figura 12. Distribución de color YCbCr

Fuente:

http://www2.elo.utfsm.cl/~elo328/PDI13_Color.pdf

1.9.1.3 MODELO XYZ

Es el primer espacio de color estandarizado en 1931. Estas componentes no son reales, sino imaginarias, pero cualquier color se puede definir como combinación de ellas. De esta forma, toda la información de intensidad está comprendida únicamente en la componente Y. Estos valores se pueden obtener de las componentes RGB como se ve en la ecuación (1.2):

$$\begin{pmatrix} X \\ Y \\ Z \end{pmatrix} = \begin{pmatrix} 0.607 & 0.174 & 0.200 \\ 0.299 & 0.587 & 0.114 \\ 0.000 & 0.066 & 1.116 \end{pmatrix} \begin{pmatrix} R \\ G \\ B \end{pmatrix}$$

(Ecuación 1.2)

En la figura 13 se muestra cómo los colores se distribuyen en la curva en función de su longitud de onda. Se puede observar que con este espacio de color no es fácil la separación de colores.

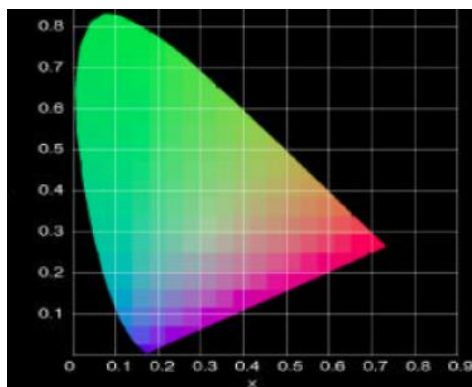


Figura 13. Distribución del color en el modelo XYZ

Fuente:

http://www2.elo.utfsm.cl/~elo328/PDI13_Color.pdf

1.10 RASGOS GEOMÉTRICOS DE UN OBJETO

Son mediciones que pueden ser usados para describir un objeto, se obtienen a partir de los píxeles que componen la región del objeto o los píxeles obtenidos del contorno del objeto. (Sánchez, 2011).

1.10.1 ÁREA

En una imagen digital, el área de un objeto está dada por el número de píxeles que representan al mismo, por lo tanto el cálculo del área se realiza contando el número de píxeles, la cual en la ecuación (1.3), se realiza un recorrido por todas las columnas C y los renglones R de la imagen y se van sumando los píxeles correspondientes al objeto:

$$A = \sum_{x=0}^C \sum_{y=0}^R f(x, y) \quad (\text{Ecuación 1.3})$$

1.10.2 PERÍMETRO

Es un parámetro geométrico que al igual que el área, puede ser calculado a partir del código de cadena. Para realizar este cálculo, es necesario contar la longitud del código, y tomar en consideración que los pasos en direcciones diagonales deben ser multiplicados por un factor igual a raíz cuadrada de dos. El perímetro p de un contorno codificado en código de cadena se expresa en la ecuación (1.4):

$$P = n_e + n_o\sqrt{2} \quad (\text{Ecuación 1.4})$$

Donde:

n_e = Número de pasos pares del código

n_o = Número pasos impares de código

1.10.3 FACTOR DE COMPACIDAD (FC)

Este rasgo es de gran importancia y muy empleado para la clasificación. Se define como el perímetro al cuadrado del objeto dividido por (4π) veces su área (ecuación 1.5) de esta forma se compara la forma del objeto con un círculo.

$$FC = \frac{P^2}{4\pi(A)} \quad (\text{Ecuación 1.5})$$

Donde:




FC = Factor de Compacidad el cual su valor depende de su figura geométrica (ver Tabla 1.1).

P = Perímetro

A = Área

La importancia de este rasgo característico radica en que es invariante a traslaciones, rotaciones y cambios de escala. (Sánchez, 2011).

Tabla 1. Cálculo del Factor de Compacidad para diferentes figura geométricas.

Figura	Medidas	Factor de Compacidad
Circulo  $r = 1$	Circunferencia = $2\pi r = 6.28$ Área = πr^2	$FC = \frac{(6.28)^2}{4(3.1416)(3.1416)} = \frac{39.44}{39.45} \approx 1$
Cuadrado  5 5	Perímetro = 20 Área = 25	$FC = \frac{(20)^2}{4(3.1416)(25)} = \frac{400}{314.16} = 1.27$
Rectángulo  34 2	Perímetro = 72 Área = 68	$FC = \frac{72^2}{4(3.1416)(68)} = \frac{5184}{854.48} = 6.06$

Fuente: Sánchez, (2011).

1.11 TECNICAS DE RECONOCIMIENTO DE PATRONES Y MODELOS DE CLASIFICACIÓN

Reconocimiento de Patrones "es la categorización de datos de entrada en clases identificadas, por medio de la extracción de características significativas o atributos de los datos extraídos de un medio ambiente que contiene detalles irrelevantes". Matemáticamente hablando, la clasificación consiste en la partición del espacio n-dimensional definido por las características de un objeto, en varias regiones, donde cada región corresponde a una clase. (Seijas, 2011).

- Un sistema de reconocimiento de patrones tiene uno de los siguientes objetivos: Identificar el patrón como miembro de una clase ya definida (clasificación supervisada).
- Asignar el patrón a una clase todavía no definida (clasificación no supervisada, agrupamiento o clustering).

El reconocimiento automático de formas gira entorno a varios conceptos claves que se definen algunos a continuación:

- **Patrón:** un patrón es una descripción cuantitativa o estructural de un objeto o alguna entidad de interés.
- **Clase:** una clase de patrones es un conjunto de patrones que comparten algunas propiedades.
- **Vector de características:** la composición de varias características en un vector se denomina vector de características, esta contiene la medida de las características de un patrón; puede estar formado de números binarios o valores reales. Un vector característica define puntos en un espacio n-dimensional.

1.12 MODELO DE CLASIFICADORES

Se suele utilizar indistintamente los términos aprendizaje y entrenamiento para referirse al proceso de construcción del clasificador. El aprendizaje en los clasificadores puede realizarse de dos maneras muy diferentes: aprendizaje Supervisado y aprendizaje no Supervisado. (Sobrado, 2003)

1.12.1 CLASIFICADORES DE APRENDIAJE SUPERVISADO Y NO SUPERVISADO

En los supervisados, la muestra se divide en las diferentes clases ya conocidas en las que se desea clasificar. A grandes rasgos las etapas 0en la construcción de un clasificador supervisado son: determinación de las clases, elección y test de las características discriminantes, selección de la muestra, cálculo de funciones discriminantes y test del clasificador. (Vélez y Moreno, 2003)

En los no supervisados este proceso se realiza de manera automática, sin la necesidad de ningún supervisor externo. Para ello se emplean técnicas de agrupamiento, gracias a las cuales el sistema selecciona y aprende los patrones que poseen características similares, determinándose automáticamente las clases. (Vélez y Moreno, 2003)

1.12.2 MODELO PARAMÉTRICO

Se puede modelar las clases mediante funciones de densidad de probabilidad conocidas: esta forma de abordar el problema se conoce como aproximación paramétrica. El aprendizaje se reduce a la estimación de los parámetros que determinan las funciones de densidad de probabilidad de las clases. La herramienta fundamental que es la estadística y particularmente la teoría de la decisión de Bayes o clasificador bayesiano el cual se especifica más adelante.

1.12.3 MODELO NO PARAMÉTRICO

Si no se supone un determinado modelo estadístico, bien por desconocimiento o por la imposibilidad de asumir un modelo paramétrico adecuado, el problema resulta más complejo y se puede abordar desde diferentes perspectivas. Los clasificadores construidos sin esta suposición se conocen como *clasificadores no paramétricos*. Cuando no puede suponerse un modelo paramétrico para las funciones de densidad de probabilidad asociadas a las clases se utilizan modelos no paramétricos para estimar las funciones de densidad de probabilidad. En estos casos la única información disponible para la clasificación es el conjunto de prototipos. El método más simple entre los no paramétricos es el del *vecino más cercano*, que consiste en etiquetar un patrón con la etiqueta del prototipo más cercano.

CAPÍTULO 2

DESAROLLO DEL SISTEMA DE VISION ARTIFICIAL

2.1 INTRODUCCIÓN

En este capítulo se describe el diseño e implementación del sistema de Visión Artificial el cual permite seleccionar piezas mecanizadas mediante una DAQ y un motor de posicionamiento. En el desarrollo de este capítulo se describió el hardware empleado, además las partes más importantes y sobresalientes del código del programas de cada etapa del sistema, de manera que se pueda entender claramente el desarrollo del software.

2.2 IMPLEMENTACIÓN DEL HARDWARE

Para la implementación del hardware se describe los elementos utilizados para el desarrollo del sistema de visión artificial y los procesos necesarios para la vinculación de cada uno de los elementos que son parte del proyecto, los que detallados a continuación

2.2.1 CÁMARA

Para la selección de la cámara web utilizada en este proyecto se buscó una de características adecuadas enfocándonos en la resolución, su compatibilidad con los diferentes sistemas operativos y un pixelaje acorde a la aplicación, resultándonos idónea una Cámara USB Apex, la cual permite capturar imágenes de alta calidad, ayudando a que el procesamiento digital de imágenes sea lo más óptimo posible.

La colocación de la cámara es de gran importancia para poder obtener una mejor imagen, por lo tanto se la colocó en un lugar que no esté propenso a vibraciones provocadas por el sistema, obteniendo como resultado imágenes de alta calidad ocasionando que nuestro procesamiento sea correcto. En la figura 14 se muestra la Web Cam utilizada.



Figura 14. Cámara USB Apex

Fuente: (Autor)

La cámara utilizada para la toma de imágenes dentro del desarrollo del proceso cuenta con las siguientes especificaciones y características descritas a continuación.

- Cámara CMOS USB
- 1,3Mpx pixeles de alta definición
- Resolución 8.0Mpx
- USB 2.0 plug and play
- 3 leds para visión nocturna
- Enfoque manual

2.2.2 SISTEMA DE ILUMINACIÓN

La iluminación es un factor muy importante en el proceso de visión, para la obtención de resultados óptimos. Una deficiente iluminación implica obtener imágenes de baja calidad, lo que a su vez con lleva a resultados erróneos o un procesamiento excesivo. En nuestro caso se estableció un sistema de iluminación formado por una Manguera Diseñada con leds de alta potencia, lo que generó un mejor desempeño, mayor flujo luminoso y una mejor vida útil, esta manguera puede adaptarse a cualquier área donde requiera una iluminación pareja y sin sombras. (Figura 15), las cuales se las ubicó en la sección cerrada adaptada al módulo para evitar el ingreso de la luz externa mejorando así el enfoque de las piezas a clasificar y para que no haya problemas en el reconocimiento de sus características necesarias.



Figura 15. Manguera LED Blanco Frío

Fuente: <http://siecled.com/productos/frames/fr-manguerasLed.html>

La manguera LED utilizada para la iluminación dentro del desarrollo del proceso cuenta con las siguientes especificaciones y características descritas a continuación.

- Voltaje de Operación 12VDC
- Consumo Máximo 20W
- Índice de Protección IP68
- Luxes 150Lx

- Ángulo de Apertura 120°
- Temperatura 5000°K
- Tiempo de Vida 50.000Hrs
- Dimensiones 500L, 12W, 4H mt

2.3 IMPLEMENTACIÓN DEL SOFTWARE

En esta parte se detalla las partes principales del código fuente, el cual realiza los procesos que se indica en el diagrama de flujo en la figura 16.



Figura 16. Descripción de procesos que realiza el programa
Fuente: Autor

2.3.1 CAPTURA DE LA IMAGEN

Se optó por desarrollar el código en MATLAB versión 7.13.0, debido a su versatilidad de programación e implementación de algoritmos útiles para nuestra aplicación. Las imágenes se capturaron en formato JPG por ser un formato que no tiene compresión y por lo tanto no distorsiona los niveles de color.

La captura de las imágenes se lo realiza con el código puesto a continuación:

```

*****EMPIEZA CAPTURA*****
try
handles.rgb = getsnapshot(handles.videobj);
  
```

```

catch
handles.rgb = imread('perfil.jpg');
end

```

```

% Update handles structure

imshow(handles.rgb)
guidata(hObject, handles);
image(handles.rgb, 'Parent', handles.fotografia);
axes(handles.fotografia)
axis off

```

A continuación se describe el código que guarda la imagen para luego poder procesarla. En el anexo 10 se describe ampliamente todo el código fuente realizado en Matlab utilizado en este proyecto.

```

%*****GUARDAR LA IMAGEN*****

% --- Executes on button press in guardar.
function guardar_Callback(hObject, eventdata, handles)
rgb = getimage(handles.fotografia);
if isempty(rgb), return, end

```

```

%guardar como archivo

fileTypes = supportedImageTypes; % Función auxiliar.
[f,p] = uiputfile(fileTypes);
if f==0, return, end

```

```
fName = fullfile(p,f);
imwrite(rgb,fName);
msgbox(['Imagen guardada en ' fName]);
```

```
%Cambio al directorio donde se ha guardado la imagen (prescindible)
if ~strcmp(p,pwd)
cd(p);
end
function fileTypes = supportedImageTypes
% Función auxiliar: formatos de imágenes.
fileTypes = {'*.jpg','JPEG (*.jpg)';*.tif','TIFF (*.tif)';...
'*.bmp','Bitmap (*.bmp)';*.*','All files (*.*)'};
```

Cada elemento de la matriz corresponde a un pixel de la imagen, y en nuestro caso se tienen tres matrices, una para cada combinación de color, de rojo, azul y verde de la imagen (RGB) en la figura 17 vemos los tipos de figuras geométricas que vamos a reconocer y clasificar con la implementación del software utilizado.



Figura 17. Imágenes capturadas con la web-cam
Fuente: (Autor)

2.3.2 PREPROCESAMIENTO Y SEGMENTACIÓN DE LA IMAGEN

En esta etapa se eliminó distorsiones y se mejoró las características de la imagen original para su posterior tratamiento. Ya que las imágenes originales capturadas con la web-cam no estaban en las mejores condiciones para nuestros objetivos,

debido a varios motivos aleatorios en los procesos de captación, transmisión y digitalización de la imagen, así como de iluminación.

En el desarrollo de este proyecto se analizó una serie de imágenes digitales, a las cuales se aplicó las técnicas de procesamiento de imagen y así determinar las que utilizaremos. El objetivo final de esta etapa es separar al objeto que se quiere inspeccionar del fondo y de los alrededores para posteriormente extraer características del objeto dentro de la imagen.

A continuación se muestran los pasos realizados en la etapa del pre procesamiento y segmentación de la imagen. La secuencia que se siguió en esta etapa se muestra a continuación:

- Conversión a niveles de gris.
- Filtrado
- Umbralización.
- Filtro para la extracción de bordes

2.3.2.1 Conversión a Niveles de Grises

Las imágenes capturadas son a color debido a la cámara utilizada, entonces el primer paso consiste en convertir la imagen a niveles de gris para tener píxeles con niveles de intensidad entre 0 y 255. La figura 18 muestra tres matrices correspondiente al cuadro de la imagen con intensidad.

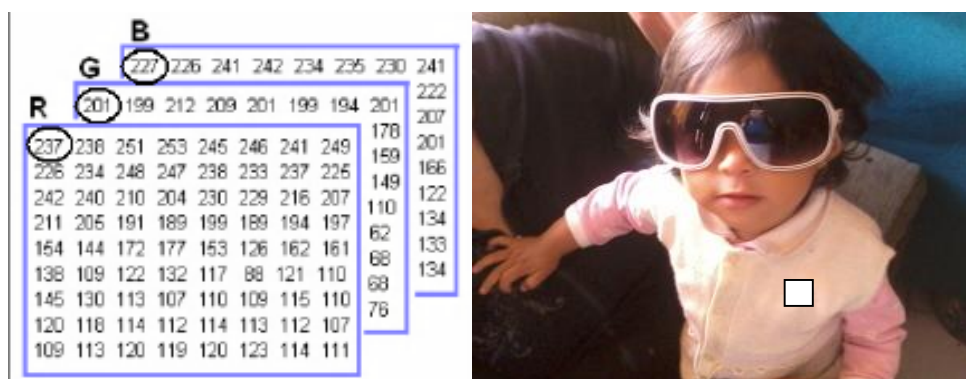


Figura 18. Representación de una imagen en color con sus tres matrices de intensidades Fuente: Autor

Para la conversión a niveles de gris se calcula el promedio de cada intensidad para las matrices de los colores rojo, verde y azul. Cada valor de la conversión se redondea para formar una matriz de intensidades donde el valor correspondiente a cada punto indica el valor de intensidad que puede ir de [0 a 255]; el cero representa el negro absoluto y el 255 el blanco absoluto. La figura (19) se obtuvo como resultado de aplicar a la figura 18 la siguiente Ecuación:

$$I = \text{Round} \left\{ \frac{1}{3} (R + G + B) \right\} \quad \text{Ecuación 2.1}$$

Cálculo de los dos primeros elementos de la matriz en niveles de gris:

$$I = \text{Round} \left\{ \frac{1}{3} (237 + 201 + 207) \right\} = \text{Round}\{221.6\} = 222$$

$$I = \text{Round} \left\{ \frac{1}{3} (238 + 199 + 226) \right\} = \text{Round}\{221.0\} = 221$$

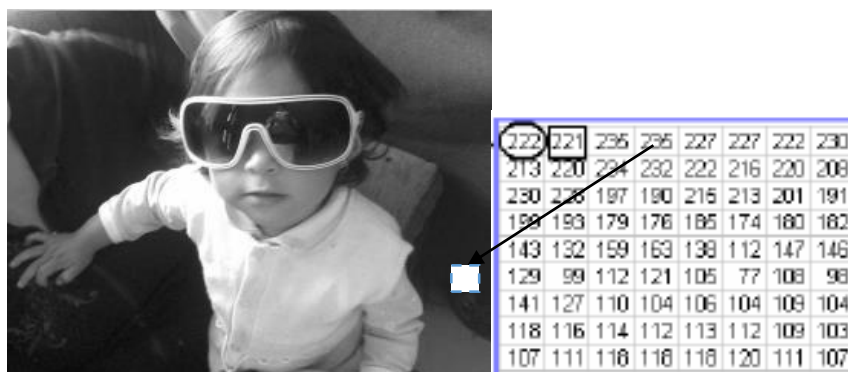


Figura 19. Fotografía a Escala de grises
Fuente: Autor

2.3.2.2 Filtrado

Inicialmente se aplicó filtros para eliminar ruidos y mejorar la imagen, sin embargo en la mayoría de los casos cuando la imagen es poco ruidosa, el no emplear un filtro no afecta al momento de segmentar, es por ello que para evitar mayor tiempo

de procesamiento se puede prescindir de un filtro, y se evitó su uso en la implementación final.

2.3.2.3 Umbralización

En esta etapa se convierte la imagen de diferentes escalas de grises, en una imagen binaria (blanco y negro) ver figura 20, en la cual la imagen pasa a tener todos los pixeles que son menores al límite especificado en el threshold a un valor de cero (0), y todos los demás a un valor de uno (1).



*Figura 20. Imagen Binarizada
Fuente: Autor*

Es importante mencionar que la imagen binaria ocupa menos memoria que una imagen de escala de grises o de color, pero contiene la información de relevancia.

2.3.2.4 Filtro para la Extracción de Bordes

En visión computacional es de utilidad para hacer reconocimiento de objetos o bien para segmentar regiones, extraer los bordes de objetos (que en teoría delimitan sus tamaños y regiones) ver figura 21. La función *EDGE* dio la posibilidad de obtener los bordes de la imagen. La función permitió encontrar los bordes a partir de dos diferentes algoritmos que pueden ser elegidos, *CANNY* y *sobel*.

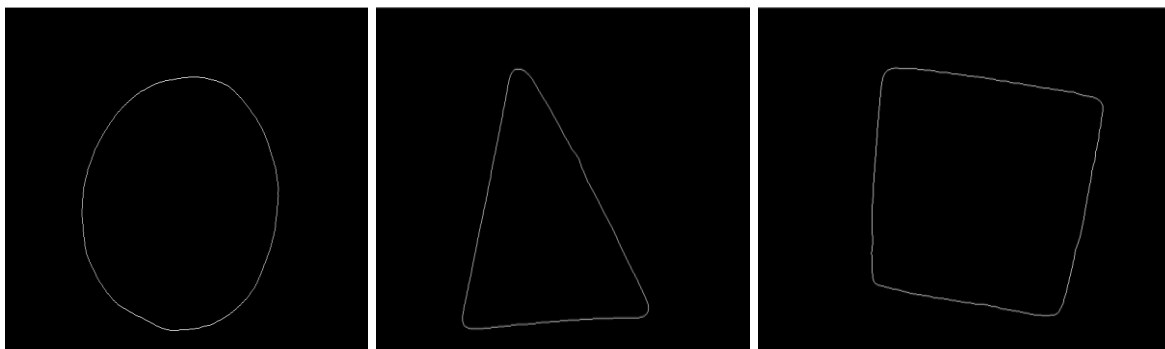


Figura 21. Imágenes resultado de la aplicación del algoritmo de sobel
Fuente: Autor

Los códigos que se utilizan para el procesamiento y segmentación de las imágenes se muestran a continuación:

```

im=handles.rgb;
bn=im2bw(im);           %%convertir a blanco y negro la imagen
axes(handles.proc1)
imshow(bn);
bn=bwareaopen(bn,100); %% elimina areas menores a 1000
pixeles
axes(handles.proc2)
imshow(bn);
area = bwarea(bn);      %%area total del objeto
[g,tg] = edge(bn, 'sobel'); %%asignando bordes a la imagen
axes(handles.proc3)
imshow(g);

```

2.3.2.4.1 Descripción de la detección de bordes

Un borde, dentro de una imagen, es considerado un tipo de discontinuidad, por lo que este puede ser detectado usando derivadas de primer y segundo orden. La derivada de primer orden para una imagen es el gradiente, el cual, para una función en dos dimensiones, $f(x,y)$ viene dado por el cálculo de la secuencia de ecuaciones de la 2.2 hasta la 2.5:

Ecuación 2.2 Derivada de primer orden para una imagen

$$\nabla f = [G_x \ G_y] \left[\frac{df}{dx} \quad \frac{df}{dy} \right]$$

Fuente: (Vélez y Moreno, 2003)

Ecuación 2.3 La magnitud del vector

$$\nabla f = \text{mag}(Vf) = [G_x^2 + G_y^2]^{1/2}$$

$$= \left[\left(\frac{df}{dx} \right)^2 + \left(\frac{df}{dy} \right)^2 \right]^{1/2}$$

Fuente: (Vélez y Moreno, 2003)

Ecuación 2.4 Simplificando la ecuación 2.3

$$\nabla f = G_x^2 + G_y^2$$

Fuente: (Vélez y Moreno, 2003)

Ecuación 2.5 Obtención de su valor absoluto

$$\nabla f = [G_x] + [G_y]$$

Fuente: (Vélez y Moreno, 2003)

2.3.2.5 Descripción del Método SOBEL

El método Sobel se utilizó en el procesamiento de las imágenes, especialmente en algoritmos de detección de bordes, se puede decir que es un método que calcula una aproximación al gradiente de la función de intensidad de una imagen.

El método de extracción de borde Sobel usa la máscara que se muestra en la figura 2.9 para aproximar digitalmente las primeras derivadas G_x y G_y , (ecuación 2.6 y 2.7):

Ecuación 2.6

$$G_x = (Z_7 + 2Z_8 + Z_9) - (Z_1 + 2Z_2 + Z_3)$$

Ecuación 2.7

$$G_y = (Z_3 + 2Z_6 + Z_9) - (Z_1 + 2Z_4 + Z_7)$$

Fuente: Vélez y Moreno, (2003).

$$\begin{array}{ccc} -1 & -2 & -1 \\ 0 & 0 & 0 \\ 1 & 2 & 1 \end{array} \quad \begin{array}{ccc} -1 & 0 & 1 \\ -2 & 0 & 2 \\ -1 & 0 & 1 \end{array}$$

$$(Z_7 + 2Z_8 + Z_9) - (Z_1 + 2Z_2 + Z_3) \quad (Z_3 + 2Z_6 + Z_9) - (Z_1 + 2Z_4 + Z_7)$$

Figura 22. Mascaras de detector sobel

Fuente: Vélez y Moreno, (2003).

La matriz de la figura 23 representa la imagen de la figura 22 la cual se calcula el gradiente en el píxel marcado con el punto negro. La región de la máscara en donde se desea calcular el gradiente está marcada con los puntos en blanco.

$$\begin{bmatrix} \circ 0 & \circ 8 & \circ 8 & 8 & 8 & 8 \\ \circ 1 & \bullet 8 & \circ 9 & 7 & 6 & 8 \\ \circ 2 & \circ 3 & \circ 4 & 8 & 8 & 8 \\ 2 & 2 & 2 & 8 & 8 & 8 \\ 2 & 2 & 2 & 2 & 8 & 8 \end{bmatrix}$$

Figura 23. Representación matricial de la imagen

Fuente : Autor

Para calcular el gradiente se calcula las ecuaciones 2.6 y 2.7 obteniendo $G_x = 30 - 4 = 26$ $G_y = 12 - 24 = -12$. Entonces $G = 48$, si fijamos un valor de umbral $T=30$, el píxel marcado con el punto negro sería un punto de borde y si por el contrario, el umbral es mayor que 48, dicho punto no sería punto de borde.

La figura 24 se muestra la detección de bordes utilizando los operadores de Sobel.



Figura 24. Utilizando el Método sobel
Fuente: Autor

2.4 DESCRIPTORES DE REGIONES

Se encargan de caracterizar una región resultante de una segmentación, cumpliendo los mismos requisitos que los descriptores de contornos en cuanto a que deben ser invariantes.

En el código anterior se observó el proceso para obtener el contorno o el borde de una imagen de la cual se necesita determinar el área de dichos borde.

2.4.1 ÁREA

El área de una región determinada que está limitada y vendrá dada por la ecuación (1.6) que se encuentra descrita en el capítulo 1:

$$A = \sum_{x=0}^{M-1} \sum_{y=0}^{N-1} f(x, y)$$

Donde $f(x,y)$ es la imagen binaria.

El código fuente realizado en Matlab el cual cumple con este proceso se lo presenta más adelante.

2.4.2 PERÍMETRO

El perímetro P está constituido por el número de píxeles exteriores de la región, es decir los del contorno de la figura que se esté procesando su información durante el desarrollo del análisis del proyecto. Más adelante se describe el código fuente realizado en Matlab el cual desarrolla este proceso

2.5 PROPIEDADES DE OBJETOS

Con las propiedades básicas (área, volumen, perímetro, centroide, etc.) de los objetos de diferentes figuras como círculos, cuadrados, triángulos, se desea plantear una solución para lograr el reconocimiento del objeto reflector y esta forma se lo puede lograr al aplicar las diferentes funciones que ofrece MATLAB como se muestra a continuación:

```
perimetro = bwarea(g);      %% calculando el perímetro
k=perimetro^2;
cociente=k/area;          %%cociente
disp('cociente=')         %%visualiza cocientes
disp (cociente)
```

2.5.1 ALGORITMO DE RECONOCIMIENTO DE OBJETO REFLECTOR.

Una vez establecidas las funciones por medio de MATLAB se obtuvieron también las propiedades del objeto, donde se procedió a plantear un algoritmo que ayudo a

definir al objeto como principal, con el análisis de estas propiedades se trató de analizar un posible algoritmo de comparación de áreas y perímetros.

La solución que se deseaba plantear es la que por medio de un algoritmo se obtenga el objeto redondeado y se compare el valor del cociente con el fin de determinar si el objeto identificado fue realmente el objeto deseado y este algoritmo se procede de la siguiente manera.

```
%% determinando que figura es

if ((cociente>5) & (cociente<7))
    a=strcat('EL OBJETO ES UN TRIANGULO'),...
    set(handles.figura,'String',a)
else
if ((cociente>3) & (cociente<5))
    a1=strcat ('EL OBJETO ES UN CIRCULO'),...
    set(handles.figura,'String',a1)
else
if (cociente>7)
    a2=strcat ('EL OBJETO ES UN CUADRADO'),...
    set(handles.figura,'String',a2)
```

En el anexo 10 se detalla todo el código fuente desarrollado en este proyecto.

2.5.2 SELECCIÓN DE OBJETOS

Visto que las funciones de MATLAB cumplen correctamente nuestro fin que es aplicar las transformaciones necesarias y extracción de información de las

imágenes capturadas resulta de especial utilidad de poder aislar objetos de una imagen binaria con un método rápido e interactivo

2.5.3 TOMA DE DECISIONES

Nuestra cámara toma alrededor de 12 cuadros por segundo, y como se tiene un sensor óptico que se activa cuando el objeto está en la posición adecuada, bastará con tomar como imagen solamente un cuadro para que se procese dentro del computador. Se diseñó el programa, para detectar colores y para las formas de los objetos. En la Figura 25 se muestra el diagrama de flujo para el proceso de selección y clasificación del objeto según sea su forma. Para la selección de formas se utilizó la ecuación 2.8 y con los resultados que se los denominan coeficientes, se realiza una comparación matemática entre ellos para poder clasificarlos.

$$\frac{(\text{Perimetro})^2}{\text{Area}} \quad (\text{Ecuación 2.8})$$

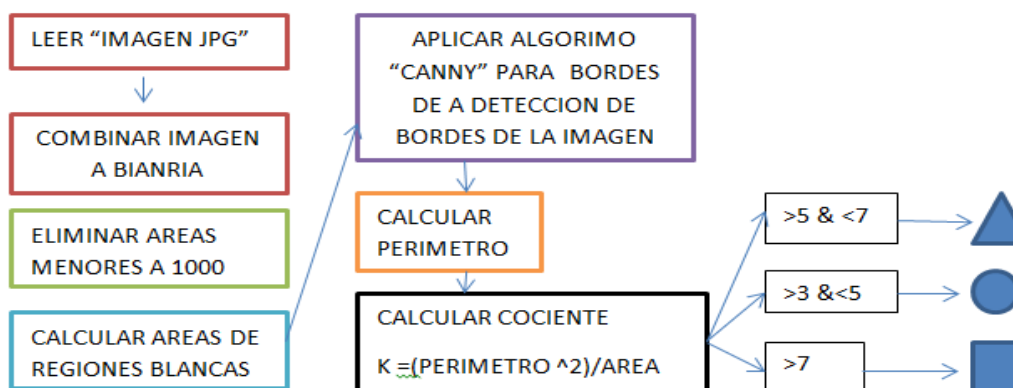


Figura 25. Diagrama de flujo para la clasificación por forma
Fuente: Autor

2.6 SELECCIÓN DE TÉCNICA PARA LA CLASIFICACIÓN POR COLOR

La selección de la técnica que se eligió cumple dos características muy importantes: un tiempo de procesamiento menor a un segundo y adecuada definición de las regiones segmentadas, para una mejor comprensión se describe mediante un diagrama de flujo el proceso que realiza el programa en matlab para la clasificación de piezas en base a su color como se observa en la figura 26.

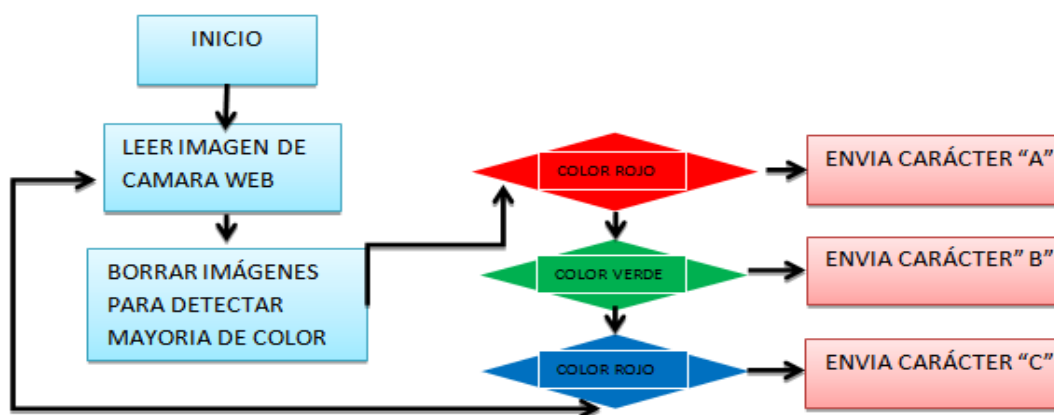


Figura 26. Diagrama de flujo para la clasificación por color
Fuente: Autor

Se optó por transformar la imagen original de modelo RGB (figura 27) a un modelo HSV (figura 28), porque el modelo RGB no es necesariamente la representación más eficiente del color, respecto de las capacidades del ojo humano, por tal motivo para codificar el color se utiliza una transformación conocida.

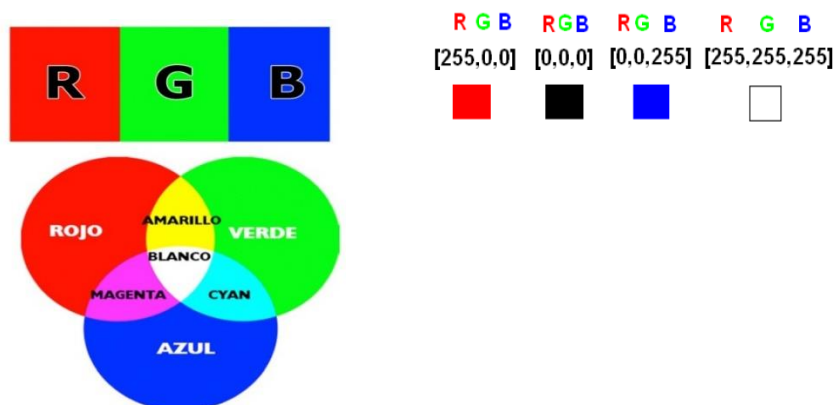


Figura 27. Representación gráfica del modelo de color RGB
Fuente: http://www2.elo.utfsm.cl/~elo328/PDI13_Color.pdf

Las imágenes están formadas por miles de puntos (píxeles), a cada píxel le corresponde un color, este color está determinado por un vector de 3 componentes, R (Rojo), G (Verde), B (Azul).

2.6.1 MODELO HSV

Es un espacio cilíndrico, pero normalmente asociado a un cono o cono hexagonal, debido a que es un subconjunto visible del espacio original con valores válidos de RGB. Las siglas H, S y V corresponden a Tono (hue), Saturación (saturation) y valor (value) respectivamente. El sistema coordenado es cilíndrico, y el subconjunto de este espacio donde se define el color es una pirámide de base hexagonal.

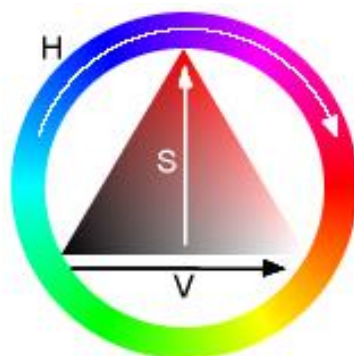


Figura 28. Modelo de imagen HSV

Fuente:

<http://sobrecolors.blogspot.com/2010/12/espacio-de-color-hsv.html>

Tonalidad (Hue): Se refiere a la frecuencia dominante del color dentro del espectro visible.

Saturación (Saturation): Se refiere a la cantidad del color o a la "pureza" de éste. Va de un color "claro" a un color más vivo (azul cielo – azul oscuro). También se puede considerar como la mezcla de un color con blanco o gris.

Valor (Value): Es la intensidad de luz de un color. Dicho de otra manera, es la cantidad de blanco o de negro que posee un color. (Gonzales y Woods, 2001)

2.6.2 ECUACIÓN DE CONVERSIÓN DE RGB A HSV

La transformada básica para la tonalidad (H) supone valores RGB [0-1], como se observa en la ecuación 2.8.

$$H = \cos^{-1} \left[\frac{\frac{1}{2}[(R - G) + (R - G)]}{\sqrt{(R - G)^2 + (R - B)(G - B)}} \right]$$

(Ecuación 2.9)

Si $B > G$, entonces $H = 360 - H$. El valor del matiz es un número entero entre $[0-360^\circ]$, como se puede observar en las ecuaciones 2.9.

$$S = 1 - 3 \frac{\min(R, G, B)}{R + G + B}$$

(Ecuación 2.10)

$$I = \frac{R + G + B}{3}$$

Las transformaciones para S e I, son de menor complejidad computacional. (Gonzales y Woods, 2001).

A continuación se describe el código fuente para la clasificación de colores.

```
% % PRUEBA CON VFM
```

```
K=handles.rgb;
```

```
Imagehsv=rgb2hsv(K);
```

```
axes(handles.proceso4)
```

```
imshow(Imagehsv);
```

```
%Extraer componentes individuales de color
```

```
size(K);
```

```
image(K);
```

```

R = K(:,:,1);

G = K(:,:,2);

B = K(:,:,3);

mR = sum (sum(R));

mG = sum (sum(G));

mB = sum (sum(B));

if((mR > mG)& (mR > mB))

    x=strcat('COLOR ROJO'),...

    set(handles.ramiro,'String',x)

    fprintf (s,'A');

elseif( (mB > mR) & (mB > mG))

    y=strcat('COLOR AZUL'),...

    set(handles.ramiro,'String',y)

    fprintf (s,'B');

elseif ((mG>mR) & (mG> mB))

    z=strcat('COLOR VERDE'),...

    set(handles.ramiro,'String',z)

    fprintf (s,'C');

```

La función `rgb2hsv` de matlab tiene como finalidad Convertir el mapa de color RGB a mapa de color HSV

Sintaxis

`CMAP = rgb2hsv (M)`

`= hsv_image rgb2hsv (rgb_image)`

Descripción

CMAP = rgb2hsv (M) convierte un mapa de colores RGB M a un mapa de colores HSV CMAP. Ambos mapas de colores son mx3 matrices. Los elementos de los dos mapas de colores están en el rango de 0 a 1. Las columnas de la matriz de entrada M representan intensidades de rojo, verde, y azul, respectivamente. Las columnas de la matriz de salida cmap representan matiz, saturación y valor, respectivamente como se puede observar en el Anexo 1.

2.7 HMI DE CAPTURA Y CLASIFICACION DE IMÁGENES

Con las prestaciones multimedia de que se tiene en la actualidad se creó un HMI (Interfaz Humano Maquina) interactivo lo más amigables posibles (Figura 29), porque no basta con diseñar e implementar un sistema que cumpla con los objetivos iniciales de cualquier proyecto, si no que sea de fácil comprensión en su manejo incluso para personas que no tengan muchos conocimientos sobre el software en el cual se desarrolló el programa. Matlab cuenta con dos pantallas una llamada Editor que guarda sus archivos en una extensión (.m) que es donde se desarrolla el código fuente y otra Guide la misma que tiene una extensión (.fig) que en la que desarrolla la programación gráfica que es la que se presenta al usuario final, es en ésta última donde se tiene que tratar de ser los más específicos posibles a más de evitar ambigüedades que puedan causar confusiones para el operador o usuario final.

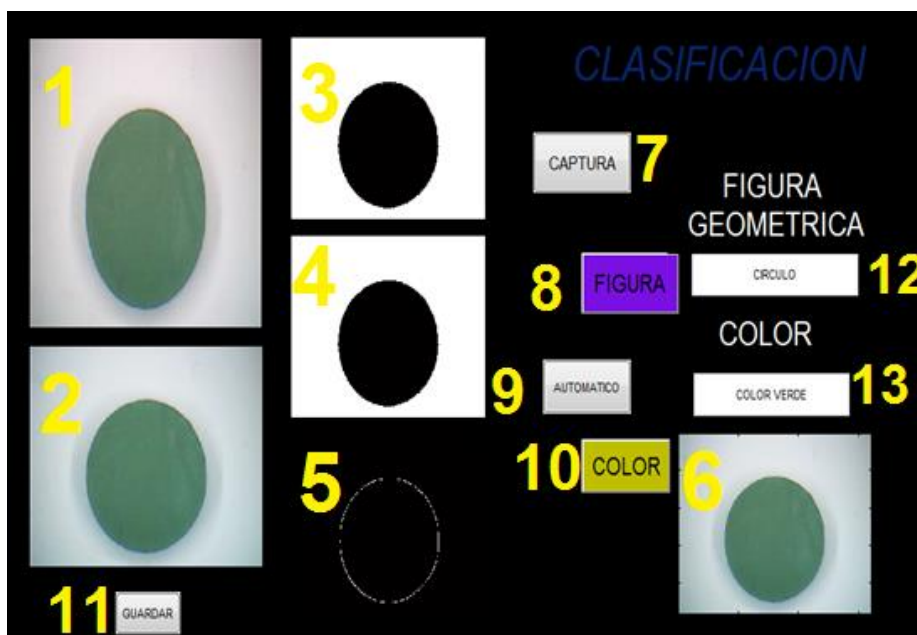


Figura 29. Captura de Imagen de HMI
Fuente: Autor

2.7.1 DESCRIPCION DE PARTES CONSTITUTIVAS DEL HMI

Para tener una mayor descripción de los procesos que realiza el programa hecho en matlab del interfaz humano máquina (HMI), se realiza una descripción de cada uno de los componentes utilitarios los cuales son:

VISUALIZACIÓN PROCESOS

- 1.- Muestra imagen en tiempo real.
- 2.- Fotografía capturada para luego para luego realizar el proceso de filtrado.
- 3.- Primer filtro de imagen, en la cual mediante el programa convierte la imagen real en imagen binaria (negro y blanco).
- 4.- Segundo filtro, eliminación de ruido o de áreas no deseadas en la imagen.
- 5.- Extracción de borde o contorno de la imagen capturada.
- 6.- Determinación del color del objeto

BOTONES QUE FUNCIONAN EN ESTADO MANUAL

- 7.- Captura la imagen en tiempo real.

- 8.- Realiza el proceso para la extracción de la figura geométrica.
- 9.- Cambio de estado MANUAL /AUTOMATICO.
- 10.- Extracción de color de la imagen.
- 11.- Guarda la imagen en la carpeta antes seleccionada.
- 12.- visualizador de resultados para la forma geométrica de la pieza.
- 13.- Visualizador de resultados para el tipo de color de la pieza.

CAPÍTULO 3

IMPLEMENTACIÓN DEL MÓDULO DE CLASIFICACIÓN DE PIEZAS MECANIZADAS

3.1 INTRODUCCIÓN

En este capítulo se dará a conocer la implementación de los componentes del módulo describiendo las características básicas que los alumnos requieren para iniciar una adecuada formación técnica, así como la familiarización de elementos industriales se combinó conocimientos de neumática y control. Ciencias que interactúan entre sí, dependiendo de las necesidades y requerimientos que se precisen en la industria.

3.2 DESARROLLO DEL PROCESO

El desarrollo del proceso de reconocimiento y clasificación de piezas inicia al montar un sistema de visión artificial en un entorno de producción continuo. Se tiene tres tipos de figuras geométricas (triángulo, cuadrado, círculo) de distinto color (rojo, verde, azul) las cuales se quiere clasificar en un depósito circular con divisiones. Estas figuras viajarán por una banda transportadora. El sistema tiene que ser capaz de, captar una imagen del objeto y procesarla para verificar el estado en base a su forma color o tipo de material, dependiendo del resultado de la inspección, la figura será depositada al caer de la banda transportadora.

A continuación, se muestra en la Figura 30 un diagrama general del módulo didáctico.

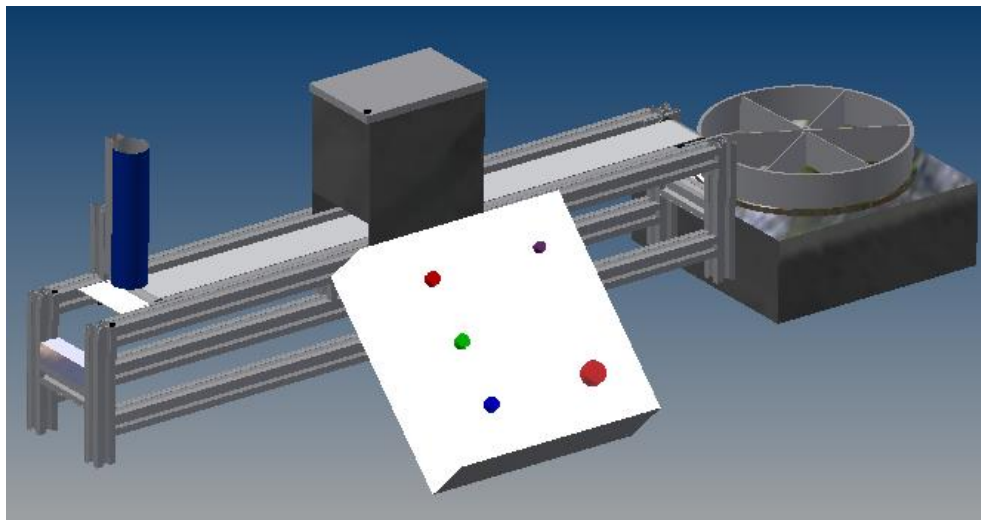


Figura 30. Diagrama del Módulo Didáctico
Fuente: Autor

3.3 COMPONENTES NEUMATICOS DEL MÓDULO

Para la realización de este proyecto se analizó aspectos constructivos como se representa en la Figura 30 donde se muestra varios de los implementos necesarios para su montaje. El modulo se dimensiona considerando los elementos de trabajo que intervendrán en el proyecto, para ello analizamos el sitio designado para el cilindro y la electroválvula, que son los dispensadores de las piezas a clasificar.

3.4 DIMENSIONAMIENTO DEL CILINDRO NEUMATICO DISPENSADOR DE PIEZAS

Este cilindro neumático es el encargado de ingresar las piezas una a una hacia la banda transportadora mediante un rango de tiempo, ubicando a las piezas en posición para su traslado al siguiente proceso que es la toma de imágenes.

3.4.1 LONGITUD DE CARRERA

Para el dimensionamiento del cilindro neumático se debe considerar los parámetros como: fuerza ejercida por un elemento de trabajo, que depende de la presión del aire y el diámetro del cilindro. La fuerza teórica del embolo se calculó de la siguiente manera:

$$F_{teórica} = 10 * P * A * n \quad (\text{Ecuación 3.1})$$

$$A = \frac{\pi * D^2}{4} \quad (\text{Ecuación 3.2})$$

Donde:

$F_{teórica}$ = Fuerza teórica del émbolo (N)

A = Superficie útil del embolo (cm)

P = Presión de trabajo (kPa, 10^5 N/m², Bar)

D = Diámetro del embolo (mm)

n= factor de fricción

El cual se establece debido a las condiciones en las cuales el cilindro va a trabajar en este caso en 0.8. Una vez definido el tipo de material y obteniendo un peso promedio de las piezas de madera y de aluminio se tiene un peso de 0.07 kg, se determina la fuerza mínima de 2N que debería tener el cilindro para poder desplazar la pieza hacia la banda, de acuerdo a los datos técnicos presentados en el catálogo de cilindros neumáticos normalizados, el cual se muestra en el Anexo 2, con un diámetro del embolo de 20mm y 50mm de carrera alcanza una fuerza de avance de 121 N, con una presión de trabajo de 5 bar. (Figura 31) Esta fuerza de avance es totalmente suficiente para desplazar las piezas.

A continuación se muestra la forma de calcular el área del cilindro para de esta manera obtener el volumen total que contiene éste mediante la ecuación 3.2 citada anteriormente.

$$A = \frac{\pi * 2^2}{4} = 3.141mm^2$$

Establecida una presión de trabajo de 5 bares y en base a la formula general de la ecuación (3.1).

$$F = 10 * P * \pi \left(\frac{(2)^2}{4} \right) * n$$

Debido a las condiciones en las cuales el cilindro va a trabajar se establece el factor de fricción en 0.8, entonces:

$$F = 10 * P * \pi \left(\frac{(2)^2}{4} \right) * 0.8$$

$$F = 125 N$$

Suficiente como para empujar el peso de las piezas y superar las fuerzas de fricción que estén presentes.



Figura 31 Cilindro neumático simple efecto (metal work)
Fuente : Autor

En la tabla 2 que se muestra a continuación se encuentran presentes las diferentes características del cilindro alimentador de piezas del módulo didáctico.

Tabla 2. Características Del Cilindro Alimentador De Piezas

DATOS TÉCNICOS	
PRESIÓN DE TRABAJO	Máx 10bar (máx 1Mpa)
RANGO DE TEMPERATURA °C	-10 ; 80
FLUIDO	Aire sin lubricación, Si se utiliza lubricación, esta debe ser continuada
DIÁMETRO DEL EMBOLO O CAMISA	Ø20mm
DIÁMETRO VASTAGO	Ø 8mm
CARRERA	50 mm
CONSTRUCCION	Cabezales achaflanados
TIPO	Simple efecto
MARCA	METALWORK

Fuente: Catálogo de accesorios neumaticos de metalwork

3.4.2 AIRE DE ALIMENTACIÓN AL SISTEMA NEUMÁTICO

Para garantizar la fiabilidad del mando neumático en el proceso es fundamental que el aire alimentado al sistema tenga un nivel de calidad suficiente. Esto implica que para que el aire atmosférico se transforme en una forma de energía “aire comprimido”, es necesario conseguir que el aire reduzca su volumen considerable, la compresión del aire implica determinados problemas, ya que al comprimirse el

aire también se comprimen todas sus impurezas que contiene tales como el polvo, hollín, gérmenes, suciedades, vapor de agua. A estas impurezas se suman las que provienen del propio compresor.

Tabla 3. Calidad del aire recomendado

Aplicaciones	Cuerpos Sólidos (μm)	Punto de condensación del agua (0°C)	Contenido max. de aceite (mg/m^3)	Clase de filtración recomendada
Minería	40	-	25	40 μm
Lavandería	40	+10	5	40 μm
Máquinas soldadoras	40	+10	25	40 μm
Máquinas herramienta	40	+3	25	40 μm
Cilindros neumáticos	40	+3	25	40 μm
Válvulas neumáticas	40 ó 50	+3	25	40 ó 50 μm
Máquinas de embalaje	40	+3	1	5 μm -1 μm

Fuente: DIN ISO 8573-1. Norma que detalla sobre la calidad del aire y la clase de filtración recomendada

En tabla 3 se observa la calidad de aire recomendadas indica la calidad de aire comprimido según los tipos de impurezas, además se puede observarlos parámetros indispensables para generar un aire de calidad en el área de la neumática, la misma que involucra cilindros y válvulas neumáticas. Las clases de calidad se definen en concordancia con la norma *DIN ISO 8573-1*, que la veremos en el anexo 3.

3.4.3 SELECCIÓN DE ELECTROVÁLVULA Y UNIDAD DE MANTENIMIENTO

La función principal de la unidad de mantenimiento es permitir, orientar o detener el flujo de aire para distribuirlo hacia el actuador neumático. La unidad de mantenimiento se designó de acuerdo al número de vías, posiciones, y de acuerdo con la función cumplen, en la figura 32 vemos la unidad de mantenimiento utilizada.



Figura 32. Dispositivo que permite la purificación de aire (Unidad de mantenimiento)
Fuente: Autor

Debido a que el cilindro que se utiliza para el ingreso de las piezas hacia la banda es de simple efecto, es necesario que la electroválvula sea de 3 vías y de 2 posiciones que se muestra en la figura 33.



Figura 33. Electroválvula 3/2 para el accionamiento del cilindro de ingreso de piezas
Fuente: Autor

Especificaciones que debe poseer la electroválvula para el correcto funcionamiento del cilindro neumático, las mismas que se muestran en la tabla en la tabla 4.

Tabla 4. Características De Electroválvulas Seleccionadas

SERIE	E304
TIPO	Válvula 3/2 distribuidor monoestable (Bobina-Muelle)
CONEXIÓN	G1/8
TEMPERATURA	5-60 GRADOS CENTÍGRADOS
FLUIDO	AIRE COMPRIMIDO FILTRADO
PRESIÓN DE TRABAJO	MAX 135 PSI
SOLENOIDE	24VDC 2.4 W

Fuente: Catálogo de accesorios neumaticos de metalwork

3.5 ALIMENTADOR DE PIEZAS

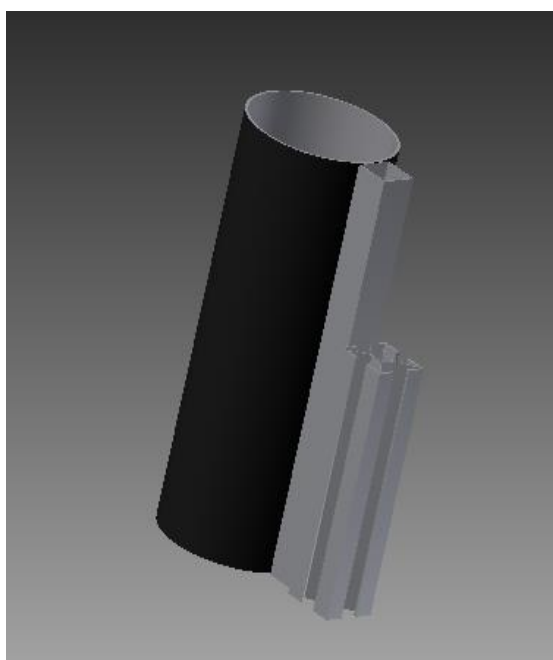
El alimentador de piezas fue construido de acuerdo con las especificaciones establecidas que se enlistan a continuación:

- **Capacidad Media de Producción.** El diseño de la maquina está establecida para alimentar 10 tapas por minuto.
- **Material de Construcción.** El material del cual fue construido la estructura el alimentador de piezas es un cilindro de PVC de 2 pulgadas de diámetro el cual tiene una alta resistencias al desgaste.
- **Tipo de Uniones.** La unión que se utiliza en el diseño de este tipo de dispensador será de tipo empernada debido a que este tipo de unión da facilidades de ser desmontado, y además genera facilidades de transporte.

- **Capacidad de dispensador.** El cilindro debe ser capaz de contener 10 piezas.

La figura 33 muestra el dispensador de las piezas implantado en la estructura del módulo didáctico, mismo que para una mayor durabilidad se escogió que fuese de material PVC, el que puede acoplarse a los diferentes diámetros de las piezas, según la necesidad lo amerite.

Diseño en Autodesk Inventor Profesional



Pieza real implementada en el modulo



Figura 34. *Dispensador de piezas de material PVC*
Fuente: Autor

3.6 CRITERIO DE SELECCIÓN DE COMPONENTES DE LA BANDA TRANSPORTADORA

Para la selección del tipo de transportador a construir se investigó la teoría de transportadores. Se escogió el transportador de bandas planas por las siguientes razones:

- Construcción sencilla, la estructura básica requiere de una banda con dos rodillos, uno acoplado a un motor y otro libre.
- Es apropiado para el desplazamiento de objetos con base plana.
- Para el sistema de transmisión de movimiento es apropiado utilizar un motor reductor DC de 12 V con acople adaptada al eje un rodillo por motivos enunciados a continuación:
- Facilidad de controlar
- Es una máquina relativamente fácil de conseguir.

3.6.1 MOTOR

Es el actuador, encargado de que se produzca el movimiento y tracción de la banda transportadora mediante la interacción entre el campo magnético de su rotor y el imán permanente de su estator. En la figura 35 se muestra el motor utilizado en el módulo didáctico el cual lo se selecciona en base a sus características de funcionamiento que son los requerimientos según nuestros cálculos realizados.



Figura 35. Motor Reductor empleado
Fuente: Autor

El motor que se implementó cumple con varios componentes que son indispensables para el funcionamiento y debe cumplir con características de fábrica que se muestran en la tabla 5.

Tabla 5. Datos del motor eléctrico de la banda transportadora

DATOS DE FABRICANTE	
Marca	POPOLU
Alimentación	12 V
Revoluciones	150 RPM
Consumo sin carga	300 mA
Consumo con carga	5A
Torque máximo	(14 kg-cm)
Dimensiones	37D X 54L mm

Fuente: Catálogo de fabricación

3.6.1.1 CÁLCULO DE LA POTENCIA DEL MOTOR

La tensión o fuerza total efectiva se define como la fuerza total transmitida por el motor cuando la banda se mueve en torno a la polea de accionamiento, y está representada mediante la ecuación 3.3:

$$T_E = T_X + T_Y \pm T_Z \quad \text{Ecuación (3.3)}$$

Donde:

T_E = tensión efectiva.

T_X = tensión necesaria para mover la banda vacía.

T_Y = tensión necesaria para mover la carga en sentido horizontal.

T_Z = tensión necesaria para levantar o bajar la carga.

Así tenemos que la tensión necesaria para mover la banda vacía (T_X) es:

Basamos en el manual de bandas para transmisión y transporte de la empresa ICOBANDAS S.A. del Anexo 4, tenemos dos valores de F_x , para equipo nuevo y equipo usado, donde se recomienda utilizar el valor para equipo usado utilizando en ella la ecuación (3.4).

$$F_x = L_c + G \quad \text{Ecuación (3.4)}$$

$$F_x = 0.035$$

Donde:

F_x = coeficiente de fricción de enrollamiento entre la banda y los rodillos deslizantes.

L_c = Valor ajustado de la distancia entre centros.

G = Peso de las piezas en movimiento.

El valor de LC, se ha determinado con la siguiente condición.

$$LC = L \text{ para } L \leq 250 \text{ ft.}$$

$$LC = 4.43 \text{ ft.}$$

Para el valor de G, se consideran los pesos de las piezas móviles (rodillos deslizantes, poleas etc.)

$$G = 2.5 \text{ lb/ft.}$$

Reemplazando valores tenemos que:

$$T_X = 0.035 * 4.43 \text{ ft} * 2.5 \frac{\text{lb}}{\text{ft}}$$

$$T_X = 0.38 \text{ lb.}$$

La tensión necesaria para mover la carga en sentido horizontal (T_Y) es:

$$T_Y = F_Y \cdot L_C \cdot Q \quad \text{Ecuación (3.5)}$$

Donde:

F_Y = Coeficiente de fricción entre la banda y los rodillos deslizantes.

Q = Razón de carga, en libras de carga por pie de distancia transportado

Para el coeficiente de fricción entre la banda y los rodillos se usa normalmente un valor:

$$F_Y = 0.04 \quad \text{Ecuación (3.6)}$$

La razón de carga está determinada por la siguiente fórmula:

$$Q = \frac{33.3 \cdot C}{S} \quad \text{Ecuación (3.7)}$$

Donde:

C = Carga en t/hr.

S = Velocidad de la banda en ft/min

Para el valor de la carga suponemos que los seis piezas de 0.07kg. Son transportados en un minuto.

$$C = 0.07 \frac{\text{kg}}{\text{min}} = 0.00420 \frac{\text{t}}{\text{hr}}$$

Para determinar la velocidad de la banda transportadora utilizamos la siguiente fórmula:

$$S = w \cdot r \quad \text{Ecuación 3.8}$$

$$w = 2 \cdot \pi \cdot N \quad \text{Ecuación 3.9}$$

Donde:

w = velocidad angular

r = radio del rodillo de la banda 0.035 m

N = revoluciones por minuto motor 150 RPM

$$S = 2 \pi (150) \cdot (0.035)$$

$$S = 2.20 \text{ m/min}$$

$$S = 7,21 \text{ ft/min}$$

Reemplazando los valores en la fórmula tenemos:

$$Q = \frac{33.3 * 0.00420 \text{ t/hr}}{7,21 \text{ ft/min}}$$

$$\mathbf{Q = 0.010 lb/ft}$$

Reemplazando los valores en la fórmula de (T_Y):

$$T = 0.04 * 4.43 \text{ ft} * \frac{0.01 \text{ lb}}{\text{ft}}$$

$$T = 0.0017 \text{ lb}$$

El valor de la tensión necesaria para levantar o bajar carga (T_Z), en este caso es cero ya que no hay diferencia de nivel.

Así la tensión efectiva T_E es:

$$T_E = T_X + T_Y + T_Z \quad \text{Ecuación 3.10}$$

$$T_E = 0.38 \text{ lb}$$

Para el cálculo de la potencia del motor se utiliza la siguiente fórmula:

$$P = \frac{T_E \cdot S}{33000} \quad \text{Ecuación 3.11}$$

$$P = \frac{0.38 \text{ lb} * 7.21 \text{ ft/min}}{33000}$$

$$P = 8.33 \times 10^{-5} \text{ hp}$$

Tomando en cuenta la eficiencia aproximada en la reducción de velocidad de un 90% se tiene:

$$P_n = \frac{P}{0.90}$$

$$P_n = 8.33 \times 10^{-3} \text{ Hp}$$

$$P_n = \frac{P}{0.90}$$

$$P_n = 9.2 \text{ Watt}$$

De tal forma que se dispone de un motor eléctrico de 12 [VDC] que consume una corriente 300mA, con revoluciones de 150 [rpm]; Entonces la potencia del motor será calculada por la ecuación 3.12 que la se detalla a continuación:

Donde:

- V Voltaje de motor en Voltios
- I Corriente de motor con carga en Amperios
- P Potencia en Vatios

$$P = V \times I \quad \text{Ecuación 3.12}$$

$$P = 12 \times 0,3$$

$$P = 40[W]$$

Donde:

- P potencia en vatios [W]
- τ Torque [N.m]
- n velocidad de giro [rad/s]

La relación torque velocidad de giro la exponemos en la ecuación 3.13

$$P = \tau \times n \quad \text{Ecuación 3.13}$$

$$\tau = 40/7.21$$

$$\tau = 5.547[N.m]$$

Al ser necesaria la fuerza que se transfiere del engrane del motor al eje en la entrada se tiene que:

- τ Torque [$N.m$]
- F Fuerza [N]
- r Distancia [m]

$$\tau = F \times r \quad \text{Ecuación 3.14}$$

$$F = \frac{5.54}{0.03}$$

$$F = 184[N]$$

3.6.2 RODILLOS

El motor transmite el movimiento del eje y permite un desplazamiento uniforme. Se utilizaron rodillos torneados, a los cuales se adaptaron 2 rodamientos acoplados a los extremos de cada los cilindro. Son estructuras cilíndricas adaptados para nuestra necesidad el cual tiene de diámetro dos pulgadas. En la figura 36 se presenta el diseño realizado en Inventor.

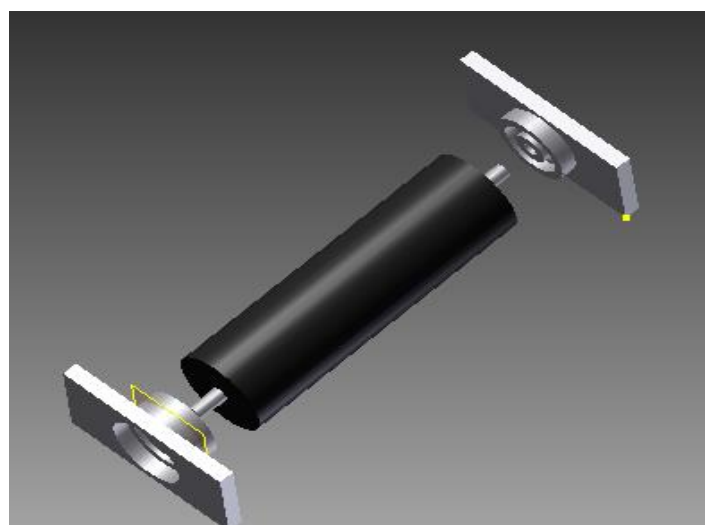


Figura 36. Rodamientos de la banda transportadora
Fuente: Autor

A continuación se describe los cálculos para el diseño de los rodillos en la ecuación 3.15.

$$Peso_{rodillo} = 0.450 \text{ kg} \cdot 9.81 \frac{\text{m}}{\text{s}^2} = 4.41 \text{ N} \quad \text{Ecuación 3.15}$$

Para soportar 4.41 N (Peso rodillo) de carga radial, de un eje que gira a 150 rpm. La duración del diseño debe ser de 1000h. El rodamiento será montado en un eje cuyo diámetro mínimo aceptable es de 13mm. Como se muestra en la ecuación 3.16.

$K=3$ Para cargas radiales

$$L_d = (h) * (rpm) * \left(60 \frac{\text{min}}{\text{h}}\right) \quad \text{Ecuación 3.16}$$

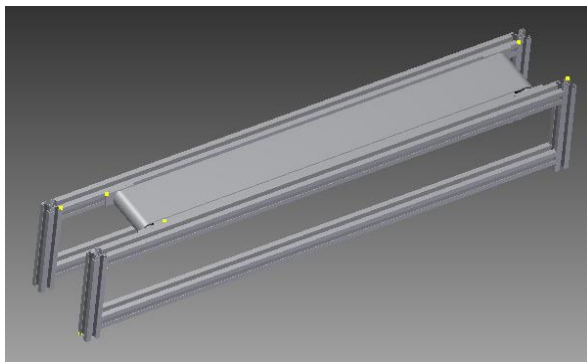
$$L_d = (1000h) * (150rpm) * \left(60 \frac{\text{min}}{\text{h}}\right) = 9 \times 10^5 \text{ rev}$$

3.7 BANDA TRANSPORTADORA

Es la encargada del transporte de las piezas para que posteriormente sean clasificadas según sea la necesidad.

Una vez ya seleccionados los componentes que realizan la transmisión de movimiento de la banda tales como el motor DC y los rodillos, se realiza el ensamblaje de banda transportadora de 70cm x 10cm con la capacidad de transportar un mínimo de 10 piezas para su inspección. En la figura 37 se puede apreciar el diseño en inventor y la estructura armada.

Diseño en Autodesk Inventor Profesional



Banda implementada en el modulo



Figura 37. Banda transportadora de caucho blanco
Fuente: Autor

3.7.1 CARACTERÍSTICAS DEL PROTOTIPO DE BANDA TRANSPORTADORA

Al momento de realizar una implementación de la banda transportadora, la mayoría de fabricantes recomienda tener en cuenta los siguientes puntos:

- Contar con un sistema de ajuste apropiado para evitar problemas de tensión.
- Minimizar costos en tiempo y distancia.
- Minimizar curvas y terrenos irregulares.
- Seleccionar la longitud y el ancho del transportador apropiados.
- Cuando el diseño de la cinta indica restricciones de carga, éstos deben ser respetados y chequeados, mediante sistemas que eviten la sobrecarga de material.

En la figura 38 identifica las fuerzas que intervienen en el eje motriz para poder desarrollar los respectivos diagramas de cortes y momento de nuestra banda transportadora.

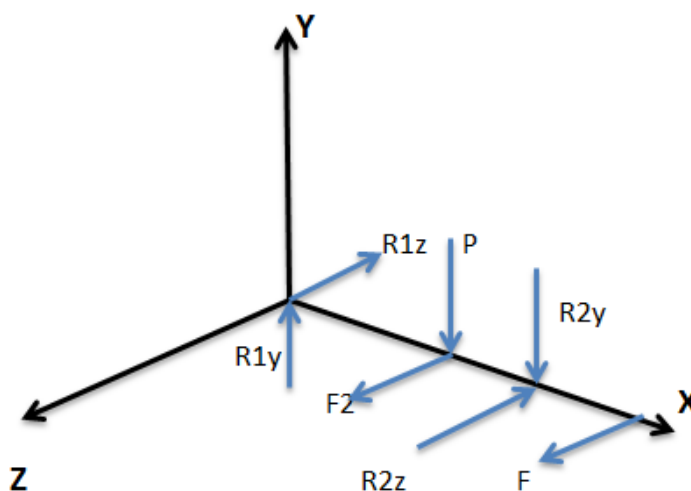


Figura 38. Fuerzas que intervienen en el eje motriz de la banda transportadora
Fuente: Autor

Donde:

F= Fuerza que ejerce el motor en el eje

F_2 = Fuerza correspondiente a la tensión que se transmite a través del rodillo

P = Peso del rodillo que actúa directamente sobre este

R_{1z} , R_{2z} , R_{1y} , R_{2y} = Reacciones correspondientes a rodamientos

Con los datos que obtuvimos anteriormente determinamos que el peso del rodillo es $0.450 \text{ kg} \cdot 9.81 \frac{\text{m}}{\text{s}^2} = 4.41 \text{ N}$ entonces procedemos graficar los diagramas de cortes y momentos en el eje XY y en el eje XZ como observamos en Anexo 4.

Ahora calculamos el esfuerzo cortante torsional con la ecuación 3.17

$$\tau_{max} = \frac{Tc}{J} \quad \text{Ecuación 3.17}$$

Donde

Tc = Radio de la superficie extrema del eje

J = Momento polar de inercia

Entonces cómo se calculó anteriormente los datos $T=5.54 \text{ [N.m]}$ y se tiene un eje de $c = 13 \text{ [mm]}$

Se calcula el momento polar de inercia utilizando la ecuación 3.18:

$$J = \pi D^4 / 32 \quad \text{Ecuación 3.18}$$

$$J = \pi 13^4 / 32$$

$$J = 2803.97$$

Remplazando los valores de en la ecuación 3.16 obtenemos:

$$\tau_{\max} = ((5547)(13))/2804$$

$$\tau_{\max} = 25.7[\text{MPa}]$$

Luego:

S_y = Esfuerzo permisible del material (AISI 304 =310[MPa])

n = Factor de seguridad

Determinamos el Factor de seguridad

$$\sigma = \frac{S_y}{n}$$

$$n = 11$$

Demostrando de tal forma que el diseño está sobredimensionado para nuestro modulo con fines didácticos por que las fuerzas a las que van a estar sometidos los elementos son muy pequeñas en comparación con las fuerzas que se involucradas en la industria.

3.8 DEPOSITO CIRCULAR CLASIFICADOR DE PIEZAS

Se ha diseñado un depósito de aluminio dividido en 6 partes iguales el cual está adaptado al eje de dos servomotores con engranes de acero de 6kg ver figura 37, de tal forma que proporcione los ángulos exactos en sus movimientos y el torque necesario para poder mover la estructura del depósito de las piezas.

Se ha optado por el aluminio para el diseño del depósito circular debido a sus características técnicas y su ductilidad al momento de darle la forma requerida para construirlo como lo podemos apreciar en la figura 38.



Figura 39. Servomotor POWER HD-6001MG con engranajes de acero
Fuente: Autor

3.8.1 ESPECIFICACIONES TÉCNICAS DEL MOTOR

Los servomotores analógicos POWER HD-6001MG necesarios para el funcionamiento del deposito módulo deben cumplir con las especificaciones que se detallan a continuación:

Peso: 56g (menos conector)

Dimensión: 40.7x20.5x39.5mm

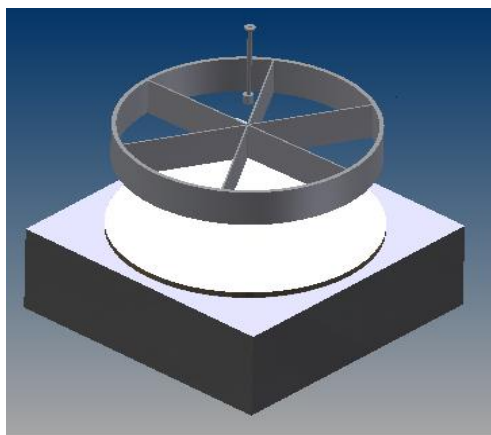
Suministro de voltaje: 4.5~6.0v

Torque plaza: 7kg. Cm

Velocidad de funcionamiento: sec 0.14/60 & deg;(ninguna carga en marcha)

Temperatura de almacenamiento:- 20c~55c

Diseño en Autodesk Inventor Profesional



Banda implementada en el modulo



Figura 40. Deposito clasificador de piezas
Fuente: Autor

3.9 DISEÑO ELECTRÓNICO DE CONTROL

A continuación se describe todos los elementos electrónicos que intervienen en el control automático de clasificación y reconocimiento de piezas.

3.9.1 CONTROL DEL SISTEMA

El control del sistema se basa en la placa ARDUINO UNO la cual su diseño es de libre distribución y utilización. En la figura 41 se observa el aspecto de la placa y sus características, y de igual manera sus características se las enlistan en la tabla 6.



Figura 41. Placa ARDUINO UNO
Fuente: Guía de usuario de Arduino

Descripción de puertos:

14 entrada/salida digitales, de los cuales 6 pueden ser usados como salidas PWM

6 entradas analógicas

Los pin 0 y 1 pueden funcionar como RX y TX serial.

Un oscilador de crystal de 16 MHz

Conector USB

Un jack de poder

Un conector ICSP

Botón de Reset

Tabla 6. Características de la placa ARDUINO UNO

Microcontrolador	ATmega328
Voltaje de operación	5V
Voltaje de entrada (recomendado)	7-12 V
Voltaje de entrada (limites)	6-20 V
Digital I/O Pins	14 (de los cuales 6 proporcionan PWM)
Pines de entrada analógica	6
Corriente continua para los pines E / S	40 Ma
Corriente de la CC para Pin 3.3V	50 Ma
Flash Memory	32 KB (of which 0.5 KB used by bootloader)
SRAM	2 KB
EEPROM	1 KB
Clock Speed	16 MHz

Fuente: Guía de usuario de Arduino

El programa se implementó haciendo uso del entorno propio de arduino y se transfiere empleando un cable USB, el mismo que proporciona alimentación a la placa y evitar utilizar una fuente de alimentación externa. En caso de necesitar una fuente tiene que ser una que proporcione voltajes de entre 6 y 25 Voltios. Para programar en la placa es necesario tener el entorno de desarrollo (IDE). En el (Anexo 5 a) se muestra el aspecto. Lo primero que se hace para comenzar a trabajar con el entorno de desarrollo de Arduino es configurar las comunicaciones entre la placa arduino y la PC. Para ello se debe abrir en el menú “Tools” la opción “Serial Port”. En esta opción se selecciona el puerto serial al que está conectado nuestra placa, la que se encuentra representado en el (Anexo 5 b).

3.9.2 DISEÑO DEL PROGRAMA PRINCIPAL

Para tratar de enfrentar el problema planteado al inicio de esta tesis, se realizó un sistema basado en prueba y error; es decir, se inició con la captura de una imagen mediante una cámara USB cuyo tratamiento se lo hizo con la ayuda de Matlab. Luego se familiarizó con el programa y se procedió a realizar varios cambios en el mismo, implementando filtros para el procesamiento de la imagen obteniendo la mejor para realizar las respectivas operaciones con ellas, llegando así a los resultados requeridos en las prácticas posteriores.

Se desarrolló los dos programas tanto en Matlab como en la plataforma de arduino en segmentos debido a que con esta metodología es más fácil corregir errores o realizar cambios acordes con las exigencias del momento. Para una mejor persuasión se pone a disposición el código fuente completo de los dos programas en el anexo 5 de arduino y en el 10 el de Matlab. En la figura 40 se describe mediante un diagrama de flujo la secuencia básica que realiza el módulo completamente ensamblado.

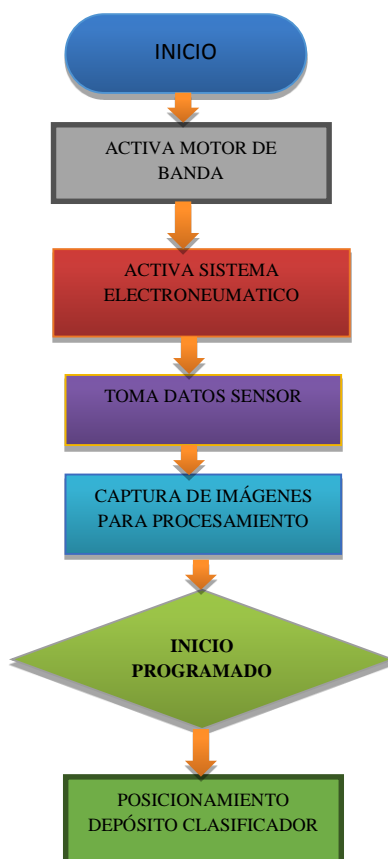


Figura 42. Descripción del proceso que realiza el módulo
Fuente: Autor

3.9.3 SENSOR DE POSICIONAMIENTO DE PIEZAS

En este caso se utilizó un Sensor óptico qrd1114, el cual está compuesto por un diodo y un fototransistor para detectar el reflejo de luz infrarroja y así poder detectar la presencia de objetos en un rango desde 0.5 a 1 centímetro. Cuando se encuentra un objeto, éste refleja la luz emitida por el emisor y es captada por el transistor, el cual se pone en saturación. Cuando no hay luz reflejada el transistor está en corte.

En la figura 41 se muestra el circuito para la conexión del sensor. El resistor R2 le suministra corriente al emisor, cuando se recibe una señal infrarroja reflejada, esta es convertida a corriente por el foto-transistor. La polarización de este último se hace con el resistor R1. Cuando no se recibe nada de señal el voltaje en P1.1 está en 3.3V, a medida que empieza a recibir señal, el voltaje va disminuyendo de este valor hasta 0V (cuando se recibe una señal muy intensa).

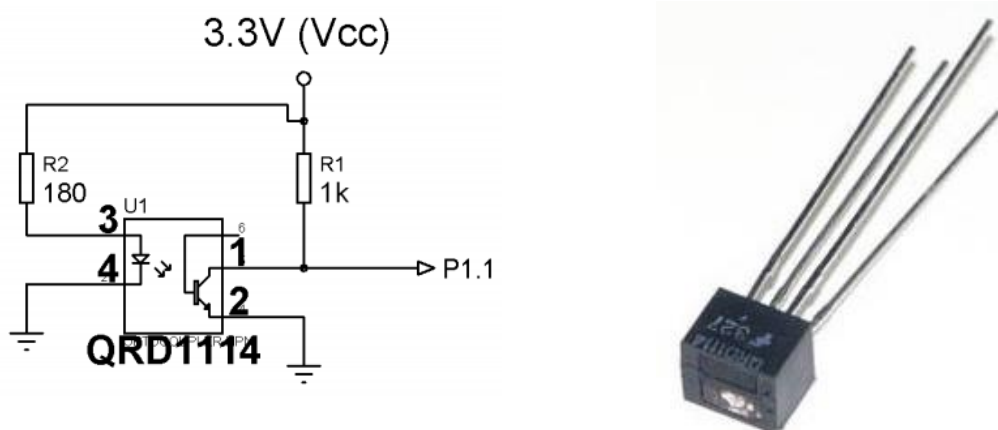


Figura 43. Diagrama esquemático del circuito de conexión del sensor y fotografía del sensor qrd1114.

Fuente: Datasheet del sensor qrd114

En la figura 44 se muestra el diagrama de conexión realizado en el simulador Fritzing de Arduino

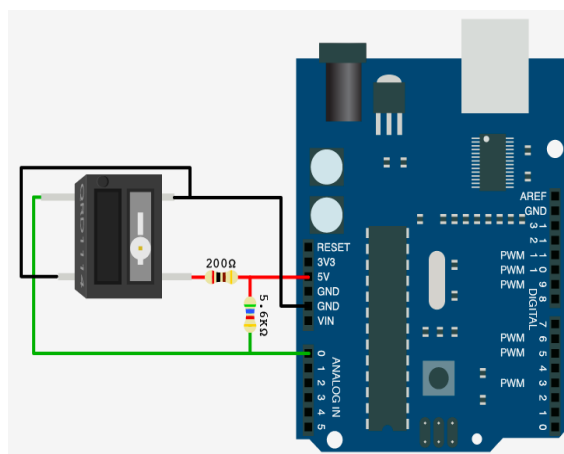


Figura 44. Diagrama de conexión del sensor hacia la tarjeta controladora

Fuente: Autor

3.9.4 SENSOR DETECTOR DE PIEZAS METALICAS

Una de las principales características que debió cumplir el módulo didáctico es la clasificación de piezas en base a su tipo de material para lo cual se adquirió un sensor de proximidad inductivo Sn04-n, el cual cumple con nuestras necesidades debido a que es un interruptor electrónico que trabajan sin contacto el cual se lo observa en la figura 45.



Figura 45. Sensor Sn04-n física e internamente
Fuente: Datasheet del sensor

Este sensor genera un campo magnético y detecta las pérdidas de corriente de dicho campo generadas al introducirse en él las piezas de aluminio para clasificarlas.

El sensor consiste en una bobina con núcleo de ferrita, un oscilador, un sensor de nivel de disparo de la señal y un circuito de salida. Al aproximarse un objeto “metálico” se inducen corrientes de histéresis en el objeto. Debido a ello hay una pérdida de energía y una menor amplitud de oscilación.

El circuito sensor reconoce entonces un cambio específico de amplitud y genera una señal que conmuta la salida de estado sólido o la posición “ON” y “OFF”.

Las características técnicas y de conexiones del sensor inductivo se las detalla en el anexo 7.

3.9.5 PROTOCOLO DE INTERCAMBIO DE DATOS

La comunicación permite a los demás componentes cruzar información con una computadora cediendo el control a un software especializado que realice otras tareas. La tarjeta arduino utilizada nos proporciona el protocolo serial asincrónico y utiliza el MAX232 para convertir los niveles de voltaje del microcontrolador al estándar RS-232.

Típicamente, la comunicación serial se utiliza para transmitir datos en formato ASCII. Para realizar la comunicación se utilizan 3 líneas: Tierra (referencia), Transmitir, y Recibir. Debido a que la transmisión es asincrónica, es posible enviar datos por una línea mientras se reciben datos por otra. Las características más importantes de esta comunicación son la velocidad de transmisión, bits de datos, bits de parada, y la paridad. Para que dos puertos se puedan comunicar, es necesario que las características sean iguales.

3.10 DIAGRAMAS DEL CIRCUITO DE POTENCIA

Este circuito fue utilizado para la activación de la electroválvula, banda transportadora, y lámparas indicadora. En el que el componente principal es un relé de 5V DC. Se diseñó el circuito en el software PCB wizard porque este nos permite realizar circuitos impresos de tal forma que podamos transportarlos a la baquelita. El diagrama eléctrico lo mostramos en la figura 46.

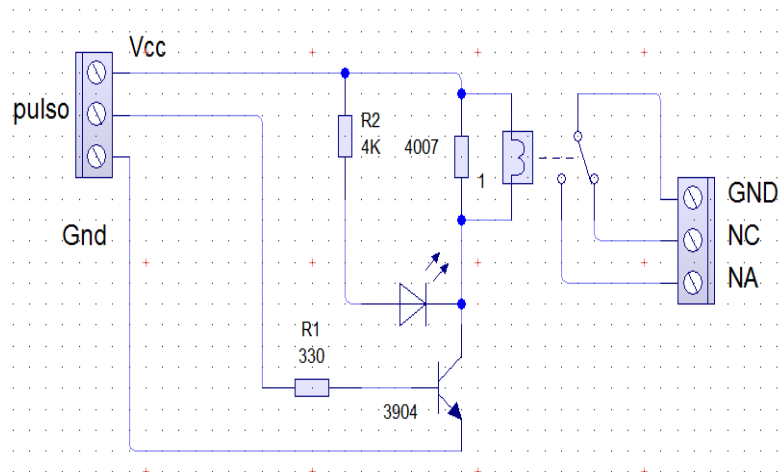


Figura 46. Diagrama esquemático del circuito de potencia
Fuente: Autor

El diagrama electrónico impreso y la placa de control utilizado para el funcionamiento del módulo se los muestra en el anexo 7.

3.12 CONTENIDO DE ELEMENTOS EN EL PANEL DE MANDO Y CONTROL

Tomando en cuenta las consideraciones anteriores y debido a los requerimientos didácticos se diseñó el panel de control compacto para facilitar el aprendizaje, en la figura 47 se observa dicho panel ensamblado el cual está conformado por los elementos que se detallan en la tabla 7.



Figura 47. Panel de control del Módulo
Fuente: Autor

Tabla 7 Elementos del panel de control

ELEMENTO	FUNCIÓN
LUZ PILOTO “ROJO”	Indicador visual de color de pieza
LUZ PILOTO “VERDE”	Indicador visual de color de pieza
LUZ PILOTO “AZUL”	Indicador visual de color de pieza
PULSADOR ON/OFF	Encendido y apagado del modulo
PULSADOR STOP NC “ROJO”	Paro de emergencia del modulo
FUENTE DE PODER	Alimentar de voltajes y corrientes a todo el Módulo Didáctico
TARJETA DE CONTROL DAQ (ARDUINO)	CONTROL DEL MODULO

Fuente: Autor

3.12 FUENTE DE PODER

Es la encargada de transformar la corriente alterna de la línea eléctrica comercial en corriente directa; la cuál es utilizada por los elementos electrónicos del módulo. Otras funciones son las de suministrar la cantidad de corriente y voltaje que los dispositivos requieren así como protegerlos de subidas de problemas en el suministro eléctrico como subidas de voltaje. En la tabla 8 se describen los voltajes y amperajes que no proporciona la fuente de voltaje dependiendo de los colores de sus cables.

Tabla 8. Características de la fuente de voltaje

COLORES DE CABLES	VOLTAJES	AMPERAJES
TOMATE	+3.3V	20 A
ROJO	+5V	30 A
AMARILLO	+12V	12 A
LILA	+5VSB	2 A
BLANCO	-5V	0.5 A

AZUL	-12V	0.8 A
VERDE	PS-ON	

Fuente: Manual de conexiones de la fuente de voltaje

3.13 DESCRIPCIÓN DE PUERTOS UTILIZADOS

Tanto las entradas como las salidas deben ser correctamente identificadas y numeradas para poder realizar la conexión respectiva a la placa, así como para poder asignar las direcciones correctas de cada una de las entradas y salidas, al desarrollar el programa en el software que se vaya a utilizar para el efecto.

La identificación de las entradas y salidas que se asignan a la placa, de los diferentes dispositivos utilizados en la realización del módulo se detallan en la tabla 9.

Tabla 9. Identificación de entradas y salidas asignadas

Entradas Analógicas	DESCRIPCIÓN
A0	Sensor QRD114 detección de piezas para capturar la imagen
A1,A2,A3,A4,A5	Libres
SALIDAS DIGITALES	
0	LIBRE
1	LIBRE
2	RELE Lámpara Indicadora ROJO
3	RELE Lámpara Indicadora VERDE
4	RELE Lámpara Indicadora AZUL
5	Control de Motor de Banda Transportadora
6	CONTROL DE SERVOMOTOR 1
7	CONTROL DE SERVOMOTOR 2
8	SENSOR INDUCTIVO
9	LIBRE
10	LIBRE
11	LIBRE
12	LIBRE
13	LIBRE

Fuente: Autor

3.14 IMPLEMENTACIÓN COMPLETA DEL MÓDULO DE RECONCIMIEN TO Y CLASIFICACIÓN

Anteriormente describimos las partes y componentes individualmente para la construcción del módulo de reconocimiento y clasificación de piezas ahora implementamos todos sus componentes la cual observaremos de mejor manera su ensamblaje en la figura 48 mientras que sus planos constructivos de diseño los observaremos en el anexo 9.



*Figura 48. Ensamblaje completo del módulo
Fuente: Autor*

CAPITULO 4

PRUEBAS DE FUNCIONAMIENTO DEL SISTEMA

4.1 INTRODUCCIÓN

En este capítulo se realizarán las pruebas que el estudiante puede realizar una vez implementados todos los sistemas que conforman el módulo didáctico, las cuales garanticen el óptimo funcionamiento de todos los dispositivos.

4.2 PRUEBA DE FUNCIONAMIENTO DE TARJETA CONTROLADORA

Para la realización de esta prueba se procede a activar desde la tarjeta controladora cada una de las salidas que activan las salidas digitales, las cuales comandan el funcionamiento de los actuadores. Los resultados obtenidos se detallan en la tabla 10.

Tabla 10. Resultado de prueba en tarjeta controladora (arduino)

SALIDA	ACTUADORES	RESULTADOS	
2	Relé de lámpara indicadora rojo	NORMAL	NORMAL
3	Relé de lámpara indicadora verde	NORMAL	NORMAL
4	Relé de lámpara indicadora azul	NORMAL	NORMAL
5	Relé de banda transportadora	NORMAL	NORMAL
6	Control servomotor 1	NORMAL	NORMAL
7	Control servomotor 2	NORMAL	NORMAL
8	Sensor inductivo	NORMAL	NORMAL

Fuente: Autor

4.3 PRUEBA DE FUNCIONAMIENTO DE SENSORES

Esta prueba fue realizada mediante la utilización de un multímetro digital para comprobar la continuidad en los sensores, lo que significa que de acuerdo al cumplimiento de una condición según sea el caso de montaje de cada uno de estos si existe continuidad, significa que el sensor enviara sin problemas la señal eléctrica correspondiente hacia la entrada analógica de nuestra tarjeta de control arduino. En el caso del Sensor óptico qrd1114 activa la cámara capturando la imagen y la envía al programa para procesarla. Para el caso sensor inductivo el cual es el encargado de diferenciar el tipo de material de las piezas, las mismas que deben pasar a una distancia máxima de 5mm atreves del sensor que es un rango óptimo para la detección de tipo de material y se procedió a verificar la continuidad entre sus cables. Los resultados observamos en la tabla 11

Tabla 11. Prueba de Sensores

SENSOR	FUNCIÓN	CONTINUIDAD	AJUSTE	CONTINUIDAD
QRD1114	PRESENCIA PARA CAMARA	SI	Posicionamiento	SI
INDUCTIVO	TIPO DE MATERIAL	SI	Posicionamiento	SI

Fuente: Autor

4.4 PRUEBAS DEL SISTEMA COMPLETO

Para realizar las pruebas del sistema completo se unió el alimentador de piezas la banda transportadora, la webcam, el gabinete de control y el depósito de piezas.

En esta se comprendió que la mejor forma de realizar las capturas de las piezas es por medio de un sensor óptico que de la señal a la cámara para que realice la toma.

Ya en la etapa de desarrollo del software se dio algunas ideas de funcionamiento del sistema implementado, sin embargo a modo de ilustrar el funcionamiento

completo y las pruebas del sistema se muestra en forma secuencial los resultados mediante algunas gráficas.

4.4.1 RESULTADOS EN LA ADQUISICIÓN DE LA IMAGEN

En la etapa de Adquisición de la imagen se utilizó la webcam con las características detalladas en el capítulo 2, algunas herramientas de adquisición de imágenes de procesamiento del lenguaje MATLAB versión 7.0. Se muestra el proceso de captura de la imagen utilizada donde se observa el montaje de los elementos necesarios para este fin (ver figura 49)

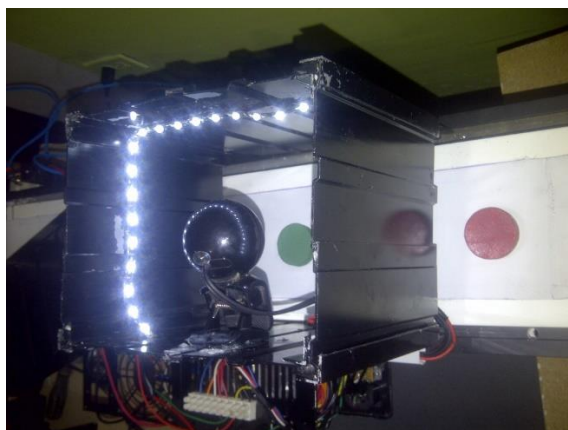


Figura 49. Adquisición de la imagen
Fuente: Autor

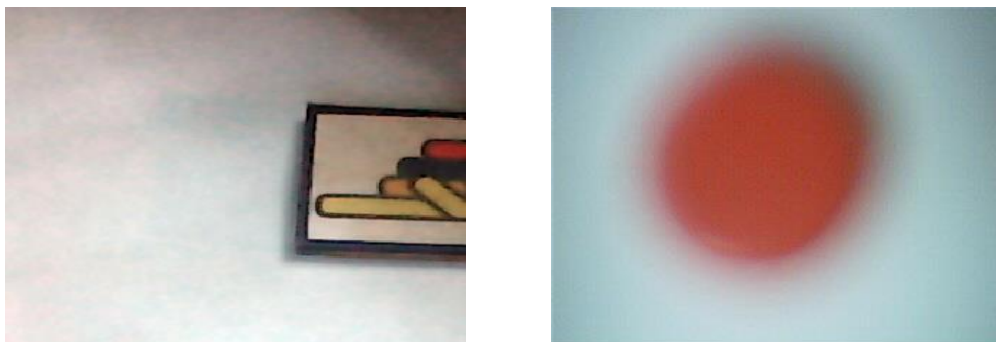
Las imágenes se capturaron en formato JPG por ser un formato que no tiene compresión y por lo tanto no distorsiona los niveles de color, como se observa en los resultados que muestra la figura 50.



Figura 50. Resultados de las imágenes captadas.
Fuente: Autor

Uno de los problemas frecuentes en la etapa de captación de la imagen es la inadecuada iluminación del objeto a reconocer, ya que es posible que las sombras se tomen como parte del objeto o los cambios de tono del color, hacen que el sistema de reconocimiento no reconozca el color real del objeto. Para evitar las sombras producidas por la iluminación, este sistema de reconocimiento se implementó dentro de una caja con adecuada iluminación. Otros puntos que se deben tomar en cuenta en la etapa de captación de la imagen son: el tamaño del objeto, el ángulo de captura, el enfoque de la cámara, entre otros.

Si la Cámara no alcanza a captar toda la pieza o no está bien enfocada, la imagen aparece muy borrosa y no se puede determinar la identidad de la pieza como se presenta en la figura 49. Para evitar este tipo de detalles, se colocó la Cámara a 20cm de la banda de tal modo que la captura óptima para su procesamiento.



*Figura 51. Imágenes mal capturadas por la webcam
Fuente: Autor*

4.4.2 MEDICIONES DE DESEMPEÑO REAL DEL SISTEMA

Al realizar algunas pruebas sobre el rendimiento del módulo implementado para demostrar el funcionamiento de un sistema de Visión Artificial se obtuvieron mediciones respecto a la cantidad de piezas que el sistema es capaz de analizar en un tiempo determinado; se obtuvieron los porcentajes de error y de aciertos que el sistema es capaz de proporcionar. Los datos son los siguientes:

Cantidad de piezas por minuto = 6

4.4.3 DETECCIÓN DE COLORES

Inicialmente en el presente proyecto se utilizó el espacio de color “RGB” el cual describe la pureza del color y brillo (*ver capítulo 1*) y se utilizó un clasificador para determinar el color del objeto. Los valores de color calculados en el espacio de color se muestran en la tabla 12.

Tabla 12. Resultados de las pruebas de colores

Población	núm.	Color	Resultados correctos	Resultados erróneos	Rendimiento%	Error%
20	7	Rojo	6	1	86%	14%
	7	Verde	7	0	100%	0
	6	Azul	5	1	83%	17%
				TOTAL	90%	10%

Fuente: Autor

4.4.4 PRUEBA POR FORMA GEOMÉTRICA

Los valores dependiendo de la iluminación hacia el objeto en el momento de la captura de la imagen. En este caso para cada figura geométrica existe una condición basada en el área de dicha pieza (tabla 4.4), la cual deben cumplir para evitar malos resultados causando problemas al momento de la clasificación.

Tabla 13. Condiciones que debe cumplir el programa en Matlab

CIRCULO	$(cociente > 3) \& (cociente < 5)$
TRIANGULO	$(cociente > 5) \& (cociente < 7)$
CUADRADO	$(cociente > 7)$

Fuente: Autor

Realizando las practicas con las población decidida de 20 piezas se obtuvo los resultados que mostrados en la tabla 14 los cuales son obtenidos de la ventana del programa Matlab llamada (Command Windows).

Tabla 14. Resultados obtenidos en Command Windows de Matlab

Total	Tipo	Cociente	Aciertos	Error
6	Cuadrado	7.393	X	
		9.349	X	
		6.434		X
		7.432	X	
		8.826	X	
		7.876		X
5	Circulo		X	
		3.256	X	
		4.864	X	
		3.687	X	
		4.367	X	
		4.612		
9	Triangulo		X	
		5.642	X	
		5.925	X	
		6.732	X	
		5.624	X	
		5.907	X	
		6.990	X	
		6.125	X	
		6.571		
		4.934		X
				TOTAL

Fuente: Autor

En la tabla 15 se presenta los resultados de las pruebas realizadas con la población de 20 piezas con 3 formas geométricas diferentes obteniendo un rendimiento óptimo en esta prueba.

Tabla 15. Resultados de las pruebas para determinar la forma geométrica

Población	núm.	Forma	Resultados correctos	Resultados erróneos	Rendimiento%	Error%
20	6	Cuadrado	6	0	100%	
	5	Circulo	5	0	100%	
	9	Triangulo	8	1	89%	11%
				TOTAL	96%	4%

Fuente: Autor

4.4.5 PRUEBAS POR TIPO DE MATERIAL

Al comprobar el paso de las piezas de diferente tipo de material (metal, madera) a través del sensor magnético se obtuvo como resultado los datos que se detalla en la siguiente tabla.

Tabla 16 Resultados de las pruebas realizadas para determinar el tipo de material de las piezas

Población	núm.	Tipo	Resultados correctos	Resultados erróneos	Rendimiento%	Error%
20	10	Metal	10	0	100%	
	10	Madera	10	0	100%	
				TOTAL	100%	0

Fuente: Autor

4.4.6 PRUEBAS DEL DEPÓSITO CLASIFICADOR

Se realizó varias pruebas para el movimiento que transmite los servomotores hacia el depósito dividido en 6 partes de tal forma que la programación de estos se posiciona exactamente en ángulos de 60 grados hasta completar el giro completo de 360 grados

Tabla 17. Resultados de las pruebas realizadas por el depósito clasificador

Población	núm.	DIVISIONES	Resultados correctos	Resultados erróneos	Rendimiento%	Error%
20	10	1	4	0	100%	
	10	2	3	0	100%	
		3	4	0	100%	
		4	3	0	100%	
		5	3	0	100%	
		6	3	0	100%	
				TOTAL	100%	0%

Fuente: Autor

4.5 ANÁLISIS DE RESULTADOS

Luego de realizadas las pruebas y obteniendo la valoración de las mismas, se puede decir que las 20 muestras usadas representan un número considerable para su análisis. El porcentaje de error relativamente bajo demuestra que el sistema implementado nos brinda unos resultados confiables; y mucho más considerando que siendo la cámara uno de los elementos significativos para el éxito de este proyecto, es configurable para tener una mayor exactitud.

Tabla 18. Análisis de resultados

PROCESOS	RENDIMIENTO TOTAL	ERROR TOTAL
DETECCION DE COLORES	90	10%
COMMAND WINDOWS	85	15%
FORMA GEOMETRICA	96	4%
TIPO DE MATERIAL	100	0
DEPOSITO DE PIEZAS	100	0
TOTAL	94,2%	5,85%

Fuente: Autor

Tomando en cuenta las pruebas realizadas en el sistema completo se tiene un error total de 5,8% el cual se podría considerar como un valor relativamente bajo el cual significa que el sistema es óptimo para las funciones requeridas las cuales son la realización de prácticas en los laboratorios.

4.6 ANÁLISIS DE RENDIMIENTO

Analizar como las piezas pasan por la banda transportadora desarrollando una trayectoria lineal, permite conocer el comportamiento del sistema para un mejor comportamiento de la cámara inteligente y conocer la cantidad de piezas por unidad de tiempo que se es capaz de clasificar. Si se considera que las electroválvulas en el punto de conmutación están en capacidad de activar el cilindro cada 10 segundos, lo que indica que el punto está en capacidad de conmutar 6PPM (PIEZAS POR MINUTO).

$$PPM = \left(\frac{1 \text{ PIEZA}}{10s} \right) \times \left(\frac{60seg}{1min} \right) = 6PPM$$

Se requiere calcular la velocidad angular del motor de la banda transportadora:

$$W = \left(\frac{100RPM * \left[\frac{(2\pi)}{60} \left(\frac{rad}{seg} \right) \right]}{1RPM} \right) = 10 \frac{rad}{seg} \quad \text{Ecuación 4.1}$$

Como el radio que describe la parte curva de la banda es de 22.5 mm = 0.0225 m, se tiene que la velocidad lineal de avance es de:

$$v = 0.0225m * 10 \frac{rad}{seg} = 0.225 \frac{m}{seg}$$

Es decir que la banda transportadora avanza 225 mm por segundo lo que indica un desplazamiento adecuado para evitar una acumulación excesiva de piezas en la banda transportadora y así no excede la frecuencia de inspección de la cámara de visión artificial.

CAPITULO 5

MANUAL DE USO

5.1 INTRODUCCION

El presente manual está diseñado para facilitar la utilización de la interface gráfica del sistema de inspección basado en Visión Artificial, comprende la descripción de las distintas funciones del software y las capacidades de respuesta de la aplicación. Además se propone prácticas que el estudiante puede realizar en éste módulo didáctico. La aplicación está perfilada para facilitar la entrada de datos y la percepción de la información por parte del usuario, por lo que cuenta únicamente con funciones definidas en pantalla sin la necesidad de ingresar a menús que puedan incrementar el tiempo de respuesta de dicho usuario a una solicitud del sistema, cuenta con diferentes campos de información para mantener al usuario actualizado en cuanto a lo que está sucediendo durante el análisis de las imágenes de las piezas a clasificar.

5.2 CONDICIONES PARA LA INSTALACION DEL SOFTWARE

- Windows 2000 o Superior
- 512 MB de memoria RAM
- Microprocesador de 1.8 GHz o superior.
- Matlab 6.5 o superior
- Toolbox de procesamiento de Imágenes
- Puerto USB

- Web cam
- Sistema de iluminación posterior difusa (Construido de acuerdo a las necesidades de la operación).

5.3 PERFIL DE USUARIO

- Conocimiento básico del sistema operativo Windows.
- Manejo de las herramientas de Matlab
- Haber recibido capacitación para operar el sistema de Visión Artificial.
- Manejo de plataforma de programación arduino.

5.4 INTERFACE USUARIO DEL SISTEMA DE VISIÓN

En esta parte se describirá en forma sintética las funcionalidades y el modo de operación que consta la interface del operador en el reconocimiento y clasificación de las piezas. En la figura 52 se enumera 6 regiones, las cuales serán descritas:

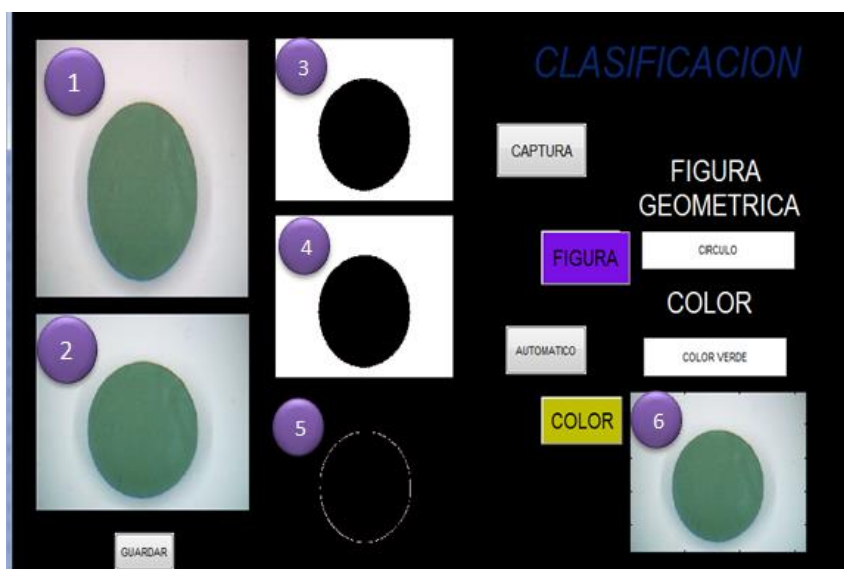


Figura 52. Descripción de regiones del HMI
Fuente: Autor




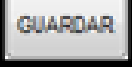
1.- En esta parte se visualiza la imagen de las piezas en tiempo real.

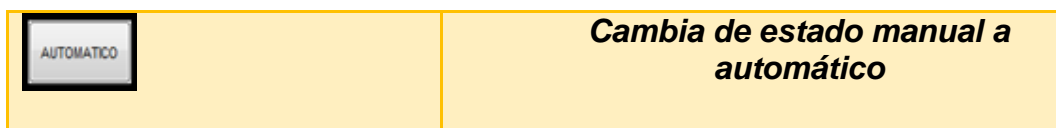
- 2.- Aquí se obtiene la imagen de la pieza capturada por la cámara, obteniendo información de las características de un objeto que ha sido reconocido.
- 3.- Empieza el procesamiento a la imagen original la binariza ósea que la transforma a blanco y negro.
- 4.- A la imagen binarizada se realiza un proceso de filtrado eliminando áreas no deseadas.
- 5.- Región donde se visualiza solo el contorno y calcula el perímetro del área del objeto procesado.
- 6.- En esta parte se identifica el distinto color de las piezas.

5.4.1 ESTADO MANUAL DEL SISTEMA

Si deseamos hacer algún tipo de pruebas para calibración del sistema, cámara de captura de imágenes, sensor de presencia, posicionamiento de clasificador sistema eléctrico, utilizaremos los botones del HMI para su funcionamiento en estado manual los cuales se describe en la tabla 19.

Tabla 19. Áreas descritas del HMI

	<p><i>Toma la foto para enviarla a procesar región de visualización de imagen</i></p>
	<p><i>Determina el tipo de pieza(cuadrado, circulo, triangulo)</i></p>
	<p><i>Determina el tipo de color(rojo, verde, azul)</i></p>
	<p><i>Guarda la imagen capturada en la misma carpeta donde está el programa.</i></p>



Fuente: Autor

5.5 INFORMACIÓN DE SEGURIDAD

En esta sección se indica las precauciones de seguridad básicas que se debe tomar en cuenta para antes de empezar a usar el módulo didáctico. Además se identifica las situaciones de peligro y advertencia las cuales se debe saber antes de operar, efectuar el mantenimiento o reparar. Debemos tomar en cuenta las siguientes precauciones para su instalación

Se debe fijar bien cada parte en una superficie estable no inclinada, para evitar caídas de las mismas y pérdidas de calibraciones de sus componentes.

Evitar dañar, romper y modificar el cable de alimentación.

No tirar del cable de alimentación con firmeza ni doblarlo más de lo necesario.

Verificar las conexiones como se describe en el diagrama eléctrico que se lo observa en el anexo 9.

No se debe desconectar la alimentación principal inmediatamente después de haberla conectado.

5.6 MANTENIMIENTO DEL MÓDULO DIDÁCTICO

Para el mantenimiento preventivo y/o correctivo será necesario quitar y poner las componentes del módulo, para esto se debe conocer la manera de sacar y volver a introducir adecuadamente los componentes así como las medidas básicas de seguridad el cual se explica a continuación.

Es indispensable contar con las herramientas y material adecuados para el mantenimiento, limpie adecuadamente el espacio de trabajo, quite de su alrededor lo que no le sea útil, también es recomendable tener a la mano un cuaderno para tomar notas y hacer pequeñas gráficas del desmontaje que va realizando.

Antes de desmontar los componentes que conforman el módulo es necesario revisarlos para saber específicamente como van ubicados, revisar el sistema neumático de piezas, banda transportadora, sistema de visión artificial, clasificador, tablero de control, así como verificar que el interface humano máquina (HMI), y por último el puerto serial necesario para la comunicación.

5.6.1 MANTENIMIENTO DEL SISTEMA NEUMÁTICO

Una vez desmontados todos los componentes primero verificamos el sistema neumático el cual con la planificación y realización de actividades de mantenimiento de rutina se podrían evitar muchos problemas del sistema neumático. La prevención empieza con el conocimiento del daño causado, los más comunes se los describe en la Tabla 20.

Tabla 20. Posibles causas de daños y soluciones del sistema neumático

Indicadores	Posibles causas	Opciones
Pérdidas	<ul style="list-style-type: none"> • Culata de cilindro rota o dañada • Presión del sistema demasiado alta • Vástago doblado o rayado • juntas dañadas o inadecuadas • Sellado defectuoso de la manguera y el acoplamiento • Conexión de la manguera mal apretada 	Inspección /reparación del componente
Desviación excesiva	<ul style="list-style-type: none"> • Hay que ajustar la válvula • Cilindro rayado • Juntas dañadas • Válvula rayada 	Inspección /reparación del componente
Funcionamiento ruidoso	<ul style="list-style-type: none"> • Enfriador de aire averiado • Nivel de aire bajo • Filtro obstruido • Bomba o motor desgastado • Válvula de seguridad averiada 	Llene el depósito hasta el nivel máximo Inspección /reparación del componente
	<ul style="list-style-type: none"> • Daño de origen externo 	
Movimiento excesivo de la manguera	<ul style="list-style-type: none"> • Alineación y fijación de la manguera deficientes • Aeración o cavitación 	Inspección /reparación del componente
Resultados del análisis	<ul style="list-style-type: none"> • Detección de altos niveles de desgaste en la muestra de aire 	Inspección
Muchas horas de servicio	<ul style="list-style-type: none"> • El sistema lleva muchas horas de servicio sin problemas 	Inspección

Fuente: Mantenimiento a Sistemas Neumáticos – Ing. J. Quiroga.

5.6.2 MANTENIMIENTO DE LA BANDA TRANSPORTADORA

Realice mantenimientos preventivos regularmente, una vez al mes o depende de su uso. Las partes móviles expuestas pueden causar heridas por tal razón se debe cortar la energía eléctrica antes de retirar cualquier parte o llevar a cabo cualquier mantenimiento.

Se empieza el mantenimiento preventivo lavando los rodillos y ejes con agua caliente y un limpiador químico ligero para eliminar el polvo, grasa y aceite lubricante. Se debe poner atención especial a la banda transportadora, puede usarse un limpiador especial pero también se puede usar jabón suave y agua. Nunca se debe emparar la banda y evite limpiadores que contienen alcohol, acetona, u otros químicos agresivos.

Inspeccionar la banda transportadora en forma regular, en busca de cortes o desgaste, estiramientos o roturas, y cualquier línea u cantos maltratados.

Inspeccione la banda para asegurarse que está sujeta y que no patinen. Con el tiempo, una banda puede estirarse y aflojarse. Los lubricantes pueden también debilitar la banda causando una mala tracción en la banda y el rodillo.

5.7 ESPECIFICACIÓN DE PINES UTILIZADOS EN LA TARJETA ARDUINO UNO.

En total se utilizan 8 pines salidas digitales y 1 entradas analógicas de los disponibles de la tarjeta arduino que son detallados a continuación. Se deberá conectar como se indica en la figura 51 por que ciertas conexiones de las borneras de color azul están conectadas internamente a la placa.

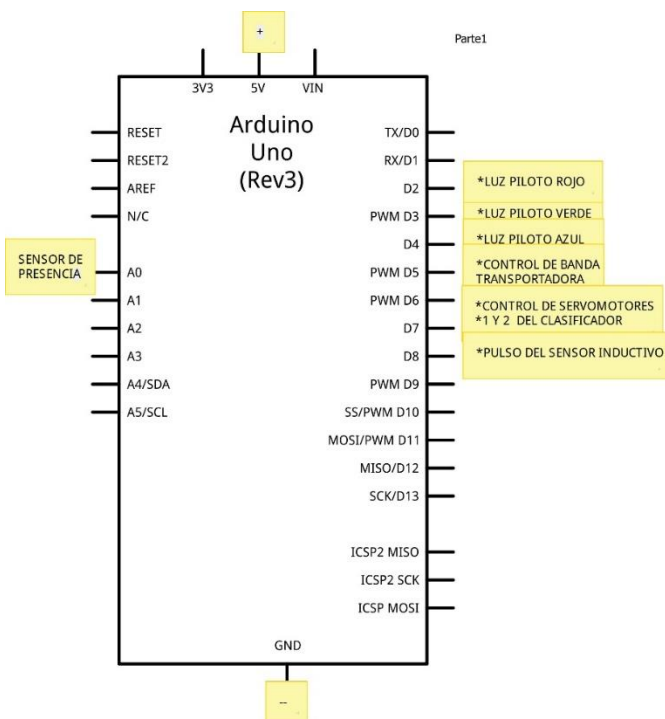


Figura 53. Diagrama de conexiones realizado en fritzing
Fuente: Autor

5.8 DESARROLLO DE GUÍA DE PRÁCTICAS

En esta parte se desarrolla una guía para que el alumno pueda realizar experimentos prácticos de visión artificial y simulaciones de control industrial, abordando cada una de las etapas que el módulo posee, hasta llegar a realizar prácticas que demanden la utilización del sistema en su totalidad.

5.8.1 ESQUEMATIZACION DE LAS PRÁCTICAS

Para la realización del esquema de las prácticas se toma en cuenta la misma denominación de los componentes que el módulo tiene, los cuales fueron descritos en capítulo 2. Debido a que el módulo ya se encuentra cableado y todos sus componentes están debidamente conectados hacia la tarjeta arduino uno, para establecer los requerimientos de cada una de las prácticas se debe referirse a la figura 51, la cual describe cada una de las entradas y salidas utilizadas de la tarjeta.

5.9 PRACTICA N° 1

LABORATORIO DE MECATRONICA		
PRACTICA N°	TEMA:	TIEMPO ESTIMADO DE PRÁCTICA
1	Accionamiento mediante la tarjeta controladora del cilindro neumático para ingreso de las piezas hacia la banda transportadora.	1 HORA

5.9.1 OBJETIVO

Implementar el cilindro neumático de simple efecto en una aplicación específica de alimentador de piezas, a través de un accionamiento de la electroválvula 3/2 la cual será activada o desactivada desde la tarjeta DAQ.

5.9.2 EQUIPO REQUERIDO

- 1 Unidad de mantenimiento.
- 1 cilindro de simple efecto (cilindro de alimentación)
- 1 electroválvula 3/2 vías
- Reguladores de caudal.
- Fuente de voltaje
- Tarjeta controladora (ARDUINO)

5.9.3 PROCEDIMIENTO

La figura 51 representa el esquema de conexiones libres para el accionamiento del cilindro de simple efecto de alimentación de piezas. La salida del cilindro debe producirse al accionar el pin por un determinado tiempo y retroceder al momento que ingrese la pieza a la banda. Para el accionamiento del cilindro se debe energizar el solenoide de la electroválvula 3/2 que está conectada a la salida digital deseada y para el retroceso.

5.9.4 VARIABLES A UTILIZAR

En la tabla se describe la salida digital la cual el practicante puede utilizar

SALIDAS	DESCRIPCION
DIGITAL 12	ELECTROVALVULA DE INGRESO DE PIEZAS

5.9.5 CONCLUSIONES Y RECOMENDACIONES

.....

PRECAUCIÓN:

Revise las conexiones neumáticas antes de alimentar el circuito con aire, recuerde que la presión es de 5 bares. Evite realizar actividades que no correspondan.

5.10 PRACTICA N° 2

LABORATORIO DE MECATRÓNICA		
PRACTICA N°	TEMA:	TIEMPO ESTIMADO DE PRÁCTICA
2	Captura de imagen mediante web-cam. En modo automático utilizando comunicación serial	90 min

5.10.1 OBJETIVO

Instaurar el funcionamiento de la web-cam ubicando de la forma adecuada para una captura óptima de las imágenes y acoplándola para la comunicación serial necesario para las posteriores prácticas.

5.10.2 EQUIPO REQUERIDO

- Web-cam
- Sistema de iluminación
- Sensor qrd1114
- Placa de control (ARDUINO).
- Cable conector USB de Arduino.
- Fuente de voltaje.

5.10.3 PROCEDIMIENTO

La figura 51 representa el esquema de conexiones de la tarjeta de control, se debe realizar un programa capaz de comunicarse con la tarjeta de control para enviar y recibir datos mediante su puerto serial.

En la tarjeta de control debe estar el código de activación del sensor, el cual envía un dato al programa realizado iniciando la captura con la cámara y guardando la imagen en la PC. Para activar el sensor se debe observar el diagrama de conexión en el que indica que entrada analógica está conectada el sensor qrd1114.

En el anexo 12 se establece la solución de las prácticas 1 y 2 con la finalidad de que tanto el alumno como el profesor las observen y puedan crear prácticas más complejas posteriormente.

5.10.4 VARIABLES A UTILIZAR

ENTRADAS	DESCRIPCIÓN
ANALOGICA A0	Recibe datos de sensor óptico

5.10.5 FUNCIONAMIENTO Y CONCLUSIONES

.....

.....

.....

5.11 PRACTICA N° 3

LABORATORIO DE MECATRONICA		
PRACTICA N°	TEMA:	TIEMPO ESTIMADO DE PRÁCTICA
3	Procesamiento de una imagen con una web-cam en tiempo real.	2 HORA

5.11.1 OBJETIVO

Reconocer y procesar una imagen capturada por la cámara en tiempo real y obtener una percepción más amplia de los elementos que intervienen en esta práctica.

5.11.2 EQUIPO REQUERIDO

- Web-cam
- Sistema de iluminación
- Sensor qrd1114
- Placa de control (ARDUINO).
- Cable conector USB de Arduino.
- Fuente de voltaje
- Banda transportadora

5.11.3 PROCEDIMIENTO

La figura 51 representa el esquema de conexiones de la tarjeta de control, se debe realizar un programa capaz de comunicarse con la tarjeta de control para enviar y recibir datos mediante su puerto serial. Se debe realizar un programa el cual capture una imagen al ingresar a través de la banda transportadora. En la tarjeta de control debe estar el código de activación del sensor, el cual envía un dato al programa realizado iniciando la captura con la cámara y guardando la imagen en la

PC. Al estar guardada la imagen el programa deberá tener la el código que realice la siguiente parte que es el procesamiento, donde interviene una serie de procesos para que la imagen pueda ser identificada correctamente. Para activar el sensor se debe observar el diagrama de conexiones en las que nos indica en que entrada analógica está conectada el sensor qrd1114.

El ejemplo de un programa para la captura de una imagen se obtiene en el *capítulo 2*.

5.11.4 VARIABLES A UTILIZAR

ENTRADAS	DESCRIPCIÓN
ANALOGICA A0	Recibe datos de sensor óptico

SALIDA	DESCRIPCIÓN
DIGITAL 9	ELECTROÁLVULA CILINDRO PARA INGRESO DE PIEZA

5.11.5 FUNCIONAMIENTO Y CONCLUSIONES

.....

.....

.....

5.12 PRACTICA N° 4

LABORATORIO DE MECATRONICA		
PRACTICA N°	TEMA:	TIEMPO ESTIMADO DE PRÁCTICA
4	Clasificación de piezas en base a su forma geométrica utilizando un sistema Mecatrónico de visión artificial	2 HORA

5.12.1 OBJETIVO

Utilizar un sistema de visión artificial para la clasificación de piezas mecanizadas en base a su forma geométrica tales como círculos, cuadrados, triángulos.

5.12.2 EQUIPO REQUERIDO

- Web-cam
- Sistema de ingreso de piezas
- Sistema de iluminación
- Sensor óptico qrd1114
- Placa de control (ARDUINO).
- Cable conector USB de Arduino.
- Fuente de voltaje
- Banda transportadora
- Deposito circular para la clasificación.

5.12.3 PROCEDIMIENTO

El modulo debe realizar las siguientes funciones: Ingreso de varias piezas de los tres tipos (círculos, cuadrados, triángulos) cada cierto tiempo determinado hacia la banda transportadora. Al momento de el paso de las piezas por el sensor óptico.

- La banda transportadora para
- La cámara captura la imagen
- Se envía a la computadora
- El programa debe realizar el reconocimiento de las piezas mediante el proceso de Visión Artificial.
- Al identificarse el tipo de figura geométrica, mover a las diferentes divisiones del depósito clasificador según se requiera la necesidad.

5.12.4 VARIABLES A UTILIZAR

ENTRADAS	DESCRIPCIÓN
----------	-------------

ANALOGICA A0	Recibe datos de sensor óptico
---------------------	-------------------------------

SALIDA	DESCRIPCION
DIGITAL 10	ELECTROVALVULA CILINDRO PARA INGRESO DE PIEZA
DIGITAL 2	INDICADOR VISUAL DE COLOR ROJO
DIGITAL 3	INDICADOR VISUAL DE COLOR VERDE
DIGITAL 4	INDICADOR VISUAL DE COLOR AZUL
DIGITAL 5	CONTROL DE BANDA TRANSPORTADORA
DIGITAL 6	DEPOSITO CLASIFICADOR
DIGITAL 7	DEPOSITO CLASIFICADOR

5.12.5 FUNCIONAMIENTO Y CONCLUSIONES

.....

.....

.....

5.13 PRACTICA N° 5

LABORATORIO DE MECATRONICA		
PRACTICA N°	TEMA:	TIEMPO ESTIMADO DE PRÁCTICA
5	Clasificación de piezas en base a su COLOR utilizando un sistema Mecatrónico de visión artificial	2 HORA

5.13.1 OBJETIVO

Utilizar un sistema de visión artificial para la clasificación de piezas mecanizadas en base a sus colores (RGB) rojo, verde, azul.

5.13.2 EQUIPO REQUERIDO

- Web-cam
- Sistema de ingreso de piezas
- Sistema de iluminación
- Sensor óptico qrd1114
- Placa de control (ARDUINO).
- computadora
- Cable conector USB de Arduino.
- Fuente de voltaje
- Banda transportadora
- Deposito circular para la clasificación.

5.13.3 PROCEDIMIENTO

El modulo debe realizar las siguientes funciones:

Ingreso de varias piezas de los tres colores (rojo verde, azul) cada cierto tiempo determinado hacia la banda transportadora mediante el sistema neumático.

Al momento del paso de las piezas por el sensor óptico el programa debe:

- Parar el movimiento de la banda transportadora
- Capturar la imagen con la web-cam
- Enviar datos a la computadora mediante comunicación serial
- Reconocer las piezas mediante el proceso de Visión Artificial.

- Identificar el color y mover a las diferentes divisiones del depósito clasificador según se requiera la necesidad.

5.13.4 VARIABLES A UTILIZAR

ENTRADAS		DESCRIPCIÓN
ANALOGICA A0		Recibe datos de sensor óptico
SALIDA	DESCRIPCIÓN	
DIGITAL 10	ELECTROVALVULA CILINDRO PARA INGRESO DE PIEZA	
DIGITAL 2	INDICADOR VISUAL DE COLOR ROJO	
DIGITAL 3	INDICADOR VISUAL DE COLOR VERDE	
DIGITAL 4	INDICADOR VISUAL DE COLOR AZUL	
DIGITAL 5	CONTROL DE BANDA TRANSPORTADORA	
DIGITAL 6	DEPOSITO CLASIFICADOR	
DIGITAL 7	DEPOSITO CLASIFICADOR	

5.13.5 FUNCIONAMIENTO Y CONCLUSIONES

.....

.....

.....

La resolución de estas prácticas propuestas se las presenta se las presenta en el anexo 5 en la que está el programa realizado en la tarjeta de control. Y en el anexo 8 el programa completo realizado en Matlab.

CONCLUSIONES Y RECOMENDACIONES

CONCLUSIONES

Se realizó la construcción del módulo didáctico de acuerdo a las normas técnicas requeridas de tal forma que en su ensamblaje se pueda realizar cambios rápidos y fiables, que se acoplen a nuevas aplicaciones de trabajo.

Se logró simular uno de los procesos más requerido en la industria ecuatoriana para la clasificación o detección de fallas de una pieza, que es la inspección por visión artificial la cual permitirá familiarizarse a los estudiantes con la industria.

Encontrar un sistema de iluminación adecuado para este proyecto fue muy complicado, de tal forma que se probó varias técnicas de iluminación hasta encontrar el sistema óptimo que para este caso fue la iluminación frontal.

El sistema construido posee una facilidad de manejo para el usuario pues se desarrolló un HMI amigable fácil de utilizarlo y flexible ante cualquier cambio o mejora, para la obtención de mejores resultados en las prácticas posteriores.

Se implementó un algoritmo que procesa y analiza imágenes, de tal forma que su procesamiento consiste en la implementación de filtros de extracción de bordes, eliminación de áreas no requeridas a la imagen binarizada.

RECOMENDACIONES

Este proyecto fue realizado con fines didácticos, razón por la cual los materiales usados en el mismo no son muy costosos, si se quiere implementar este proyecto en alguna industria relacionada con el transporte de objetos y su clasificación, se recomienda que se adquiriera una cámara de escaneo progresivo más sofisticada.

Para evitar las variaciones de luz que se producen en el día es necesario un sistema de iluminación adecuado según sean los requerimientos, ya que el sensor de la




cámara requiere un tiempo de adquisición muy corto para evitar los desenfoques, pero también necesita una buena iluminación para adquirir la imagen con el suficiente contraste

Para prácticas más complejas se necesita de una sincronización muy precisa con la cámara por lo que sería ideal el uso de una tarjeta de adquisición de imágenes con salidas digitales, a fin de que cumpla con todos los requerimientos que se presentan en una industria.

ANEXOS

ANEXO 1


RGB de la tabla de colores HSV

Color	Nombre del color	Maleficio	(R,G,B)	(H,S,V)
	Negro	# 000000	(0,0,0)	(0 ° , 0%, 0%)
	Blanco	# FFFFFFFF	(255 255 255)	(0 ° , 0%, 100%)
	Rojo	# FF0000	(255,0,0)	(0 ° , 100%, 100%)
	Cal	# 00FF00	(0,255,0)	(120 ° , 100%, 100%)
	Azul	# 0000FF	(0,0,255)	(240 ° , 100%, 100%)
	Amarillo	# FFFF00	(255,255,0)	(60 ° , 100%, 100%)
	Cian	# 00FFFF	(0,255,255)	(180 ° , 100%, 100%)
	Magenta	# FF00FF	(255,0,255)	(300 ° , 100%, 100%)
	Plata	# C0C0C0	(192 192 192)	(0 ° , 0%, 75%)
	Gris	# 808080	(128 128 128)	(0 ° , 0%, 50%)
	Granate	# 800000	(128,0,0)	(0 ° , 100%, 50%)
	Oliva	# 808000	(128,128,0)	(60 ° , 100%, 50%)
	Verde	# 008000	(0,128,0)	(120 ° , 100%, 50%)
	Púrpura	# 800080	(128,0,128)	(300 ° , 100%, 50%)
	Teal	# 008080	(0,128,128)	(180 ° , 100%, 50%)
	Marina de guerra	# 000080	(0,0,128)	(240 ° , 100%, 50%)

ANEXO 2

Catalogo FESTO de cilindros neumáticos

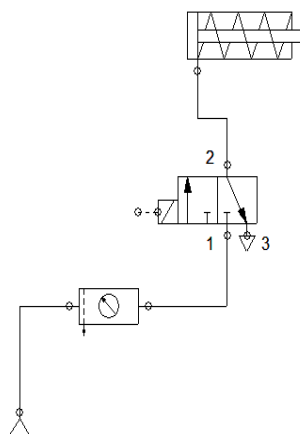
Cilindro normalizado

Tipo	Funcionamiento	Embolo Ø	Fuerza teórica con 6 bar en avance	Carrera	Amortiguación	Descripción
Cilindro normalizado, cilindro redondo DSNU, ESNU, DSN, ESN 	Doble efecto., Efecto de presión, Simple efecto	8 mm, 10 mm, 12 mm, 16 mm, 20 mm, 25 mm, 32 mm, 40 mm, 50 mm, 63 mm	19... 1.870,3 N	1... 500 mm	P: Anillos/placas de amortiguación elásticos en ambos lados, PPS: Amortiguación neumática autorregulable en las posiciones finales, PPy: Amortiguación neumática regulable en ambos lados	<ul style="list-style-type: none"> Cilindro normalizado DSNU, ESNU, DSN, ESN, émbolo Ø 8... 25 mm corresponde a ISO 6432 Cilindro redondo económico DSNU, ESNU, émbolo Ø 32... 63 mm Para la detección de las posiciones: DSNU, ESNU Sin detección de las posiciones: DSN, ESN Gran cantidad de variantes Gran rendimiento y duración Vástago con rosca interior o exterior → Internet: .../dsnu

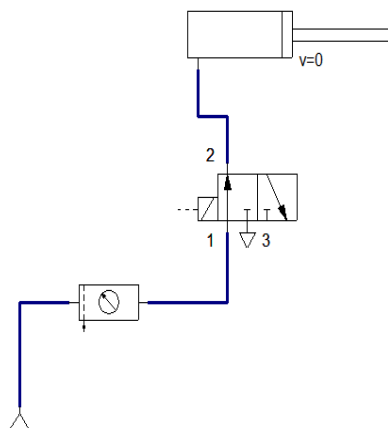
CIRCUITO NEUMATICO IMPLEMENTADO

REALIZADO EN FLUFSIM DE FESTO

SITEMA DESACTIVADO



SISTEMA ACTIVADO



ANEXO 3

Norma DIN ISO 8573-1

Calidad del aire comprimido ISO 8573-1:2001

Calidad Clase	SÓLIDOS			AGUA Punto de condensación de la presión °C	ACEITE Y VAPOR DE ACEITE mg/m ³	Calidad Clase
	0,1-0,5 micras	0,5-1 micras	1,0-5 micras			
0	Según especificaciones del usuario final o fabricante, y más estricta que la Clase 1					0
1	100	1	0	-70	0.01	1
2	100,000	1000	10	-40	0.1	2
3	No disponible	10,000	500	-20	1	3
4	No disponible	No disponible	1,000	3	5	4
5	No disponible	No disponible	20,000	7	No disponible	5
6	No disponible	No disponible	No disponible	10	No disponible	6

Eliminación de la humedad

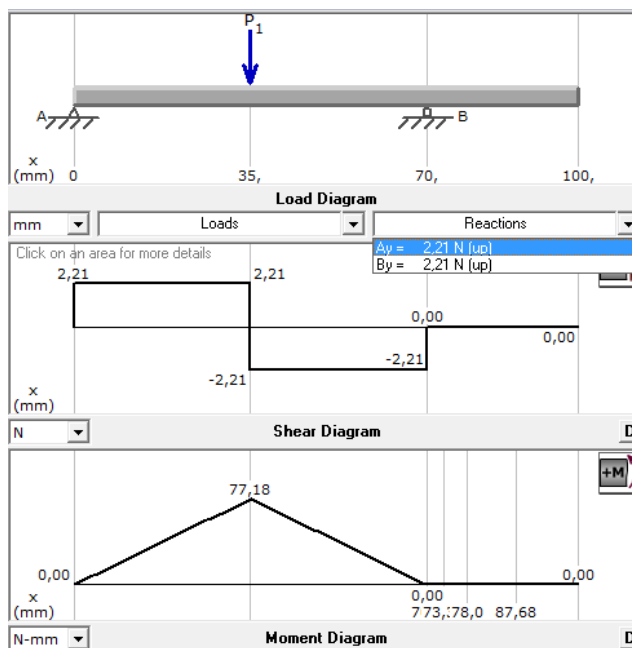
Calidad del aire comprimido ISO 8573-1:2001

Calidad Clase	SÓLIDOS			AGUA Punto de condensación de la presión °C	ACEITE Y VAPOR DE ACEITE mg/m ³	Calidad Clase
	0,1-0,5 micras	0,5-1 micras	1,0-5 micras			
0	Según especificaciones del usuario final o fabricante, y más estricta que la Clase 1					0
1	100	1	0	-70	0.01	1
2	100,000	1000	10	-40	0.1	2
3	No disponible	10,000	500	-20	1	3
4	No disponible	No disponible	1,000	3	5	4
5	No disponible	No disponible	20,000	7	No disponible	5
6	No disponible	No disponible	No disponible	10	No disponible	6

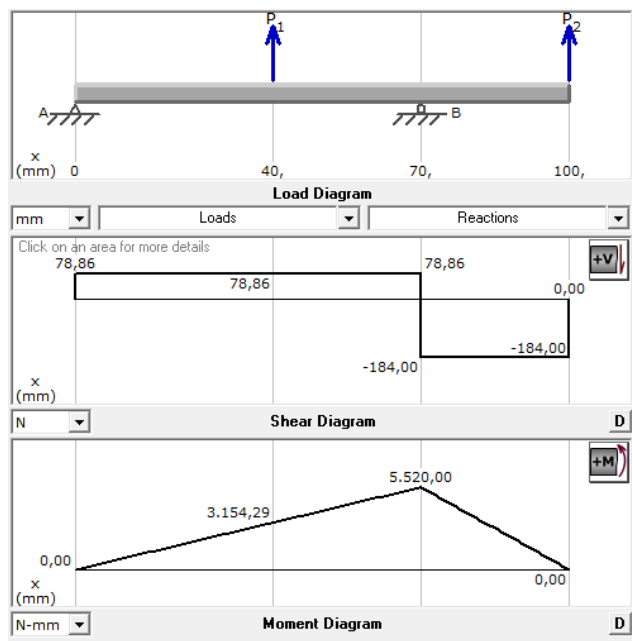
ANEXO 4

DIAGRAMAS DE CORTES Y MOMENTOS REALIZADOS EN EL SOFTWARE MDSOLID

Diagramas de cortes y momentos en el eje XY



Diagramas de cortes y momentos XZ

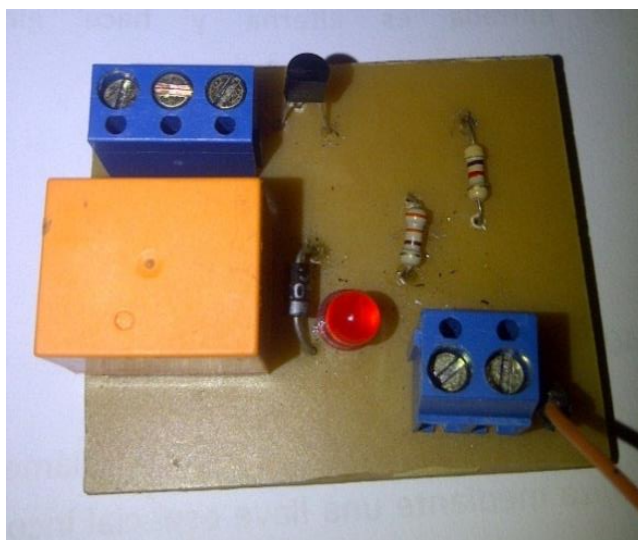
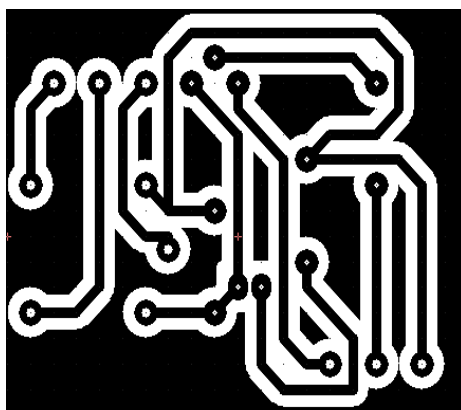


ANEXO 5

<pre> #include <Servo.h> // LIBRERIA DEL SERVO int analogPin = 0; // Pin analógico 0 Servo myservo; // creación de objeto servo1 Servo myservo2; // creación de objeto servo2 int val = 0; //declaración de la variable en la que se almacenará el valor leído por el conversor. int azul = 2; //COLOR AZUL int verde = 3; // COLOR VERDE int rojo = 4; // COLOR ROJO int led = 5; // BANDA TRANSPORADORA int led1 = 6; // ON/OFF ELECTROVALVULA void setup() { Serial.begin(9600); //Inicialización del módulo Serial. // PINES COMO SALIDAS pinMode(led, OUTPUT); pinMode(led1, OUTPUT); pinMode(azul, OUTPUT); pinMode(verde, OUTPUT); pinMode(rojo, OUTPUT); myservo.attach(5); // SERVO AL PIN5 myservo.write(0); // POS. INICIAL SERVO myservo2.attach(6); // SERVO AL PIN6 myservo2.write(0); } void loop() { val = analogRead(analogPin)/4; // lee el valor de tensión del pin //Serial.println(val); // envía el valor leído vía serial. delay(800); if(val <200) { //delay(25); digitalWrite(led, HIGH); // turn the LED on (HIGH is the voltage level) Serial.println('A'); delay(5000); digitalWrite(led, LOW); } { //Guardamos en dato el valor leído del puerto serie byte dato = Serial.read(); switch (dato) { //Si es '1' encendemos el led </pre>	<pre> case '1': myservo.write(60); digitalWrite(rojo,HIGH); delay(5000); digitalWrite(rojo, LOW); break; //Si es '2' apagamos el led case '2': myservo.write(120); digitalWrite(verde,HIGH); delay(5000); digitalWrite(verde, LOW); break; case '3': myservo2.write(0); digitalWrite(azul,HIGH); delay(3000); digitalWrite(azul, LOW); break; case '4': myservo2.write(60); digitalWrite(azul,HIGH); delay(3000); digitalWrite(azul, LOW); break; case '5': myservo2.write(120); digitalWrite(azul,HIGH); delay(3000); digitalWrite(azul, LOW); break; case '6': myservo2.write(180); digitalWrite(azul,HIGH); delay(3000); digitalWrite(azul, LOW); break; } } } </pre>
--	---

ANEXO 6

Circuito impreso de la placa de Control

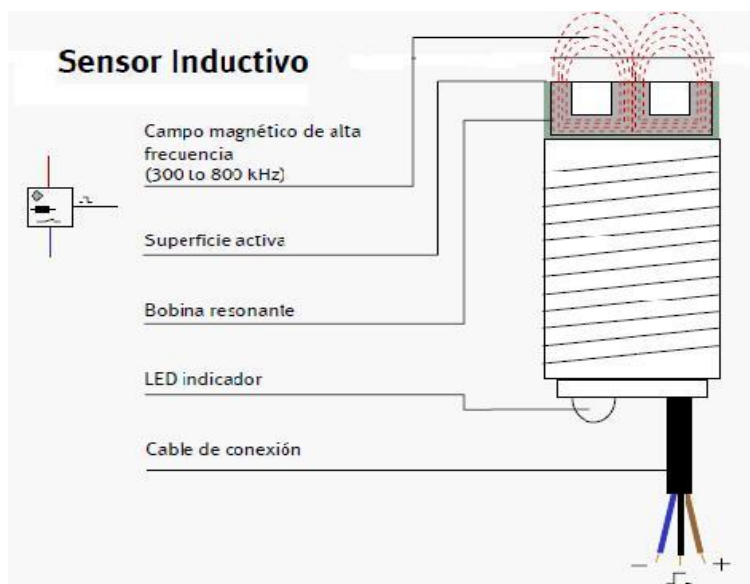


ANEXO 7

CARACTERISTICAS DEL SENSOR INDUCTIVO

tipo de alambre	Columna angular 3 dc alambre tipo(negro, de color marrón, azul)
interruptor de tipo de aspecto	tipo de columna angular
la teoría	sensor inductivo
tipo de salida	Npn no(normal abierto)
la distancia de detección	4mm
de suministro de voltaje	Dc 10-30v
corriente de salida	200ma
respuesta de frecuencia	500hz
el objeto de detectar	de hierro
la temperatura de funcionamiento	- 25& deg; a c +55& deg; c
plaza de tamaño de la cabeza	18*18*36mm/0.6 0.6 x x 1.4inch(w*h*l)
la longitud del cable	115-120cm/41"(aprox)
material externo	de plástico
peso neto	48g
contenido del paquete	x 1 interruptor de proximidad inductivo

PARTES INTERNAS DEL SENSOR INDUCTIVO



ANEXO 8

PROGRAMA PRINCIPAL REALIZADO EN MATLAB

```

%Con este programa se pretende mostrar la forma de capturar video e
imágenes en una interfaz gráfica

function varargout = videoGUI(varargin)

% Begin initialization code - DO NOT EDIT

gui_Singleton = 1;

gui_State = struct('gui_Name',       mfilename, ...
                  'gui_Singleton',  gui_Singleton, ...
                  'gui_OpeningFcn', @videoGUI_OpeningFcn, ...
                  'gui_OutputFcn',  @videoGUI_OutputFcn, ...
                  'gui_LayoutFcn',  [], ...
                  'gui_Callback',   []);

if nargin && ischar(varargin{1})
    gui_State.gui_Callback = str2func(varargin{1});
end

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

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

% Choose default command line output for videoGUI

handles.output = hObject;

```

```

handles.rgb = [];
handles.noback = [];

guidata(hObject, handles);

%** ACTIVA LA CAMARA DE VIDEO E INICIA LA PROYECCION DE LA IMAGEN

%La sentencia strcmp compara dos strings. Si son iguales retorna 1 y %caso
contrario, retorna 0. El código entre try y catch establece la %conexión de
video e inicia la reproducción. En caso de producirse un error en estas
líneas, se ejecuta el código entre catch y end, que coloca una imagen
('perfil.jpg') en el axes de captura de foto.

if strcmp(get(hObject,'Visible'),'off')
try
handles.videobj = videoinput('winvideo', 2);

% Update handles structure
start(handles.videobj);
guidata(hObject, handles);
vidRes = get(handles.videobj, 'VideoResolution');
nBands = get(handles.videobj, 'NumberOfBands');
hImage = image(zeros(vidRes(2), vidRes(1), nBands), 'Parent',
handles.video_cam);
preview(handles.videobj,hImage);
preview(handles.videobj);
catch
msgbox('NO HAY CÁMARA CONECTADA. Cargando foto.jpg.')
hImage = image(imread('perfil.jpg'), 'Parent',handles.video_cam);
axes(handles.videobj)

```

```

end

end

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

varargout{1} = handles.output;

%*****CAPTURAR LA IMAGEN*****

% --- Executes on button press in captura.
function captura_Callback(hObject, eventdata, handles)
try
handles.rgb = getsnapshot(handles.videobj);
catch
handles.rgb = imread('perfil.jpg');
end
% Update handles structure
imshow(handles.rgb)
guidata(hObject, handles);
image(handles.rgb, 'Parent', handles.fotografia);
%%objVideo.ReturnedColorspace='rgb'
axes(handles.fotografia)
axis off

% --- Executes on button press in guardar.

function guardar_Callback(hObject, eventdata, handles)

```

```

rgb = getimage(handles.fotografia);
if isempty(rgb), return, end
%guardar como archivo
fileTypes = supportedImageTypes;           % Función auxiliar.
[f,p] = uiputfile(fileTypes);
if f==0, return, end
fName = fullfile(p,f);
imwrite(rgb,fName);
msgbox(['Imagen guardada en ' fName]);

%Cambio al directorio donde se ha guardado la imagen (prescindible)

if ~strcmp(p,pwd)
    cd(p);
end

function fileTypes = supportedImageTypes
% Función auxiliar: formatos de imágenes.
fileTypes = {'*.jpg','JPEG (*.jpg)'; '*.tif','TIFF (*.tif)';...
 '*.bmp','Bitmap (*.bmp)'; '*.*','All files (*.*)'};

function figura_Callback(hObject, eventdata, handles)
% --- Executes during object creation, after setting all properties.
function figura_CreateFcn(hObject, eventdata, handles)
% hObject    handle to figura (see GCBO)
if ispc && isequal(get(hObject,'BackgroundColor'),
get(0,'defaultUiControlBackgroundColor'))
    set(hObject,'BackgroundColor','white');

```

```
end

% --- Executes during object creation, after setting all properties.
function colores_CreateFcn(hObject, eventdata, handles)

if ispc && isequal(get(hObject,'BackgroundColor'),
get(0,'defaultUicontrolBackgroundColor'))
    set(hObject,'BackgroundColor','white');
end

function ramiro_Callback(hObject, eventdata, handles)

function ramiro_CreateFcn(hObject, eventdata, handles)

if ispc && isequal(get(hObject,'BackgroundColor'),
get(0,'defaultUicontrolBackgroundColor'))

    set(hObject,'BackgroundColor','white');
end

% --- Executes on button press in auto.
function auto_Callback(hObject, eventdata, handles)

% hObject    handle to auto (see GCBO)

if get(hObject, 'String') == 'AUTOMATICO'
    set(hObject, 'String', 'MANUAL');
end
```

```
%*****+PUERTO SERIAL*****

warning off MATLAB:serial:fread:unsuccessfulRead
q=instrfind;

    if ~isempty(q)
        fclose (q);
        delete (q);
        clear q
    end

% crear puerto

q = serial ('COM3','BaudRate',9600);
q.InputBufferSize = 5000;

% abrir puerto serial
    fopen (q);
    for cc = 1:10
pause( 1 );
r = fread(q,1);

    if( r == 65)
        try

handles.rgb = getsnapshot(handles.videobj);
catch
handles.rgb = imread('perfil.jpg');
end
```



```
% Update handles structure

imshow(handles.rgb)

guidata(hObject, handles);

image(handles.rgb, 'Parent', handles.fotografia);

axes(handles.fotografia)

axis off

%%****CLASIFICACION EN BASE A SU COLOR

K=handles.rgb;

    size(K);

    image(K);

    R = K(:,:,1);

    G = K(:,:,2);

    B = K(:,:,3);

    mR = sum (sum(R));

    mG = sum (sum(G));

    mB = sum (sum(B));

    if((mR > mG) & (mR > mB))

        x=strcat('COLOR ROJO'),...

        fprintf (q, '1');

        set(handles.ramiro, 'String', x)

    elseif ((mG>mR) & (mG> mB))

        z=strcat('COLOR VERDE'),...

        fprintf (q, '2');
```

```

        set(handles.ramiro, 'String', z)
elseif( (mB > mR) & (mB > mG) )
y=strcat('COLOR AZUL'),...
        fprintf (q, '3');
set(handles.ramiro, 'String', y)

end

im=handles.rgb;
bn=im2bw(im);           %%convertir a blanco y negro la imagen
axes(handles.proc1)    %(primer imshow)
imshow(bn);
bn=bwareaopen(bn,1000); %%elimina areas menores a 1000 pixeles
axes(handles.proc2)    %(segundo imshow )
imshow(bn);
area = bwarea(bn);     %%area total del objeto
[g,tg] = edge(bn, 'sobel'); %%asignando bordes a la imagen
axes(handles.proc3)    %(tercer imshow )
imshow(g);
perimetro = bwarea(g); %% calculando el perimetro
k=perimetro^2;
cociente=k/area;      %%cociente
disp('cociente=');
disp (cociente)

%% determinando que figura es
if ((cociente>7) & (cociente<9))
    a=strcat('EL OBJETO ES UN TRIANGULO'),...
    set(handles.figura, 'String', a)

```

```
        fprintf(q, '4'); %envio del caracter "E" por el puerto serial

elseif ((cociente>11) & (cociente<23))

        a1=strcat ('EL OBJETO ES UN CIRCULO'),...

        set(handles.figura, 'String', a1)

        fprintf(q, '5');

elseif ((cociente>9) & (cociente<11))

        a2=strcat ('EL OBJETO ES UN CUADRADO'),...

        set(handles.figura, 'String', a2)

        fprintf(q, '6');

        end

end

end

%Cierro la conexión con el puerto serial y elimino las variables

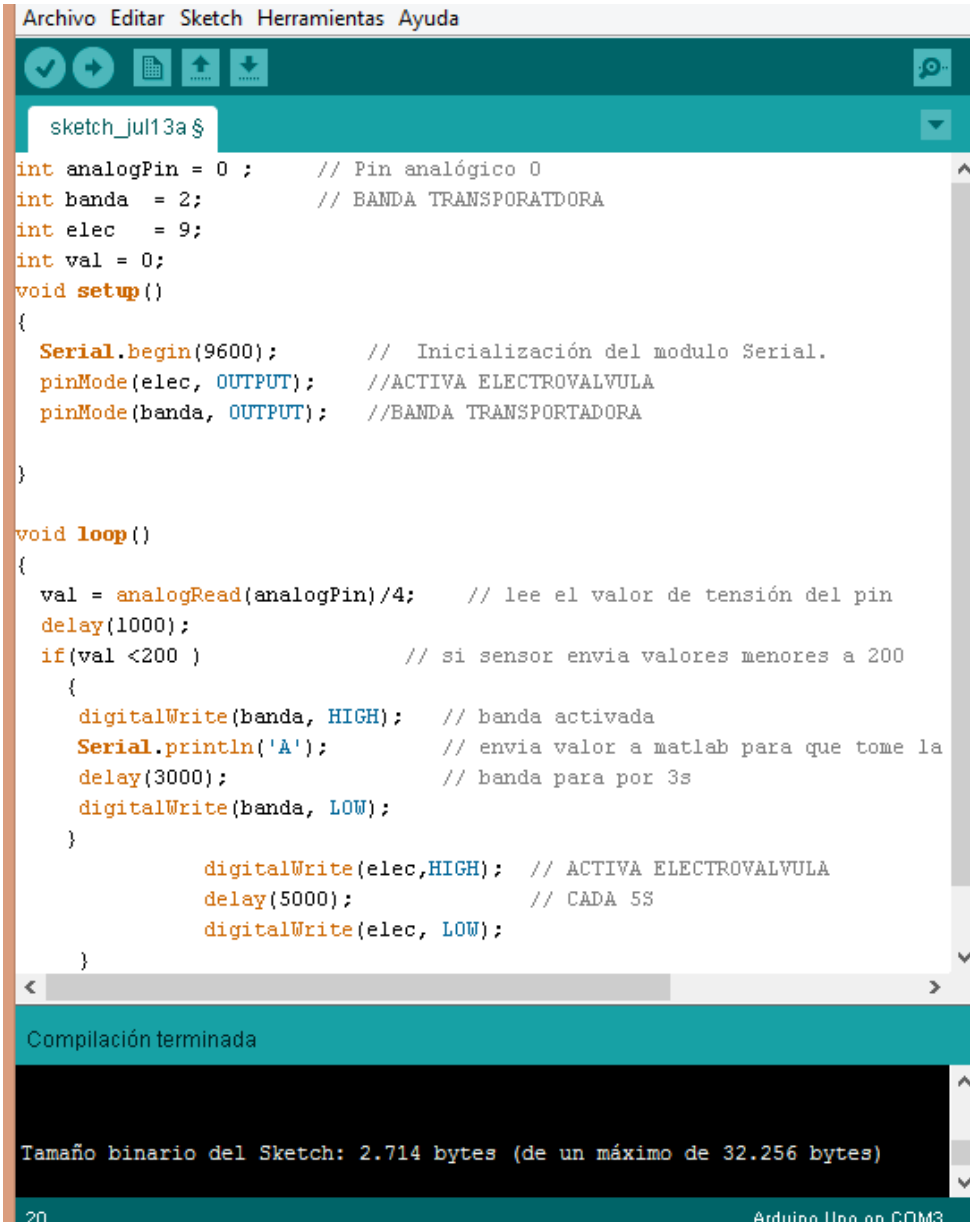
fclose(q);

delete(q);

clear q;

function activado_Callback(hObject, eventdata, handles)
```

ANEXO 10

PROGRAMA PRINCIPAL RESOLUCION DE LAS PRACTICAS
PROPUESTAS

```
Archivo Editar Sketch Herramientas Ayuda
sketch_jul13a $
int analogPin = 0 ; // Pin analógico 0
int banda = 2; // BANDA TRANSPORTADORA
int elec = 9;
int val = 0;
void setup()
{
  Serial.begin(9600); // Inicialización del modulo Serial.
  pinMode(elec, OUTPUT); //ACTIVA ELECTROVALVULA
  pinMode(banda, OUTPUT); //BANDA TRANSPORTADORA
}
void loop()
{
  val = analogRead(analogPin)/4; // lee el valor de tensión del pin
  delay(1000);
  if(val <200 ) // si sensor envia valores menores a 200
  {
    digitalWrite(banda, HIGH); // banda activada
    Serial.println('A'); // envia valor a matlab para que tome la
    delay(3000); // banda para por 3s
    digitalWrite(banda, LOW);
  }
  digitalWrite(elec,HIGH); // ACTIVA ELECTROVALVULA
  delay(5000); // CADA 5S
  digitalWrite(elec, LOW);
}
Compilación terminada
Tamaño binario del Sketch: 2.714 bytes (de un máximo de 32.256 bytes)
20 Arduino Uno on COM3
```