ESCUELA POLITÉCNICA NACIONAL

FACULTAD DE INGENIERÍA ELÉCTRICA Y ELECTRÓNICA

DISEÑO E IMPLEMENTACIÓN DE UN CONTROL SERVOVISUAL PARA UN ROBOT MANIPULADOR TIPO SCARA

PROYECTO PREVIO A LA OBTENCIÓN DEL TÍTULO DE INGENIERO EN ELECTRÓNICA Y CONTROL

MAYRA ALEXANDRA GUARNIZO LANDÁZURI

mayra.guarnizo@epn.edu.ec

DIRECTOR: Ing. PAULO CÈSAR LEICA ARTEAGA, PhD.

paulo.leica@epn.edu.ec

Quito, febrero 2019

DECLARACIÓN

Yo, Mayra Alexandra Guarnizo Landázuri, declaro bajo juramento que el trabajo aquí descrito es de mi autoría; que no ha sido previamente presentada para ningún grado o calificación profesional; y, que he consultado las referencias bibliográficas que se incluyen en este documento.

A través de la presente declaración cedo mis derechos de propiedad intelectual correspondientes a este trabajo, a la Escuela Politécnica Nacional, según lo establecido por la Ley de Propiedad Intelectual, por su Reglamento y por la normatividad institucional vigente.

_____

Mayra Alexandra Guarnizo Landázuri

CERTIFICACIÓN

Certifico que el presente trabajo fue desarrollado por Mayra Alexandra Guarnizo Landázuri, bajo mi supervisión.

_____

Ing. Paulo Leica, PhD

DIRECTOR DEL PROYECTO

AGRADECIMIENTO

Agradezco a Dios por darme la vida, por permitirme ser feliz junto a mis seres queridos.

A mi familia, por estar a mi lado, por darme alegrías, enseñarme a luchar por lo que deseo, gracias.

A mi novio, Daniel por compartir todas sus experiencias, por su apoyo incondicional.

A todo el equipo del laboratorio de Sistemas de Control, por ser un gran apoyo, por brindarme nuevos conocimientos, por dejarme formar parte de su grupo de trabajo, les agradezco por ser mí familia en la Escuela Politécnica Nacional.

Al Dr. Paulo Leica por los conocimientos brindados, por su paciencia y su colaboración, por confiar en mí para el desarrollo de este proyecto.

A mis amigos y profesores por compartir sus conocimientos y habilidades, por las experiencias vividas, gracias por su apoyo.

Mayra

DEDICATORIA

A mi papá Luis, a mi mamá Clara, a mis Hermanos Enma, Liliana y José Luis, por quienes siento toda mi admiración, respeto y cariño. Siempre han confiado en mí.

Mayra

CONTENIDO

Contenido i

CAPÍTULO 1 1

MARCO TEÓRICO 1

1.1 MANIPULADORES INDUSTRIALES 1

1.1.1 ROBOT MANIPULADOR INDUSTRIAL 1

1.2 MORFOLOGÍA DE UN ROBOT MANIPULADOR 1

1.2.1 ESTRUCTURA DE ROBOTS MANIPULADORES 1

1.3 MORFOLOGÍA DE UN ROBOT MANIPULADOR TIPO SCARA 4

1.3.1 ESTRUCTURA MECÁNICA DE UN ROBOT SCARA 4

1.3.2 SISTEMAS DE ACCIONAMIENTOS [3] 5

1.3.3 ACTUADORES 5

1.3.4 SENSORES 6

1.3.5 ELEMENTOS TERMINALES 8

1.3.6 REPRESENTACIÓN DE LA POSICIÓN Y ORIENTACIÓN [4] 8

1.3.7 CINEMÁTICA DEL ROBOT 12

1.4 VISIÓN ARTIFICIAL 12

1.4.1 MÉTODOS DE CAPTACIÓN DE LAS IMÁGENES 13

1.4.2 MODELADO DEL SISTEMA DE VISIÓN 15

1.4.3 ESTRUCTURA FÍSICA DE VISIÓN ARTIFICIAL 17

1.4.4 PROCESAMIENTO DE IMÁGENES 18

1.4.5 TIPOS DE PROCESAMIENTO DE DATOS 22

1.4.6 MODELOS DE COLOR 23

1.5 ESTABILIDAD DE LYAPUNOV 24

1.5.1 CONDICIONES DE ESTABILIDAD DE LYAPUNOV 26

1.6 ÍNDICE DE DESEMPEÑO ISE 26

1.7 MÉTODO DE MEDICIÓN DE DESEMPEÑO 27

1.7.1 DESVIACIÓN ESTÁNDAR.....27

1.8 CINEMÁTICA DEL ROBOT [1].....28

1.9.1 RESOLUCIÓN DEL MODELO CINEMÁTICO POR EL MÉTODO GEOMÉTRICO [1].....28

1.9.2 RESOLUCIÓN DEL MODELO CINEMÁTICO POR MATRICES DE TRANSFORMACIÓN HOMOGENEA [1] 29

1.9.3 RESOLUCIÓN DEL MODELO CINEMÁTICO APLICANDO EL ALGORITMO DE DENAVITH HARTENBERG [1] 30

1.9 MODELO DIFERENCIAL. MATRIZ JACOBIANA [1].....32

1.9.1 MATRIZ JACOBIANA ANALÍTICA.....32

1.9.2 MATRIZ JACOBIANA GEOMÉTRICA [1].....33

1.10 CONTROL SERVO VISUAL DE MANIPULADORES.....34

1.10.1 CÁMARA FIJA.....35

1.10.2 CÁMARA EN MANO.....36

1.10.3 EXTRACCIÓN DE CARACTERÍSTICAS DE LA IMAGEN.....36

1.11 ROBOT MANIPULADOR TIPO SCARA.....37

1.11.1 CONDICIÓN ACTUAL 38

1.12 COMPONENTES ELECTRÓNICOS 39

1.12.1 MICROCONTROLADOR ATMEGA 164P 39

CAPÍTULO 2 40

DESARROLLO DEL ALGORITMO DE CONTROL Y PROCESAMIENTO DE IMÁGENES 40

2.1 MODELO CINEMÁTICO DEL ROBOT MANIPULADOR TIPO SCARA ...40

2.2 DISEÑO DEL CONTROLADOR SERVO VISUAL 43

2.2.1 MODELO GEOMÉTRICO DE LA CÁMARA 43

2.2.2 JACOBIANO TOTAL.....46

2.2.3 DISEÑO DEL CONTROLADOR SERVO VISUAL 48

2.3 PROCESAMIENTO Y ADQUISICIóN DE IMÁGENES 51

2.3.1 ALGORITMO DE VISIÓN.....51

2.3.2 ADQUISICIóN DE IMAGEN 52

2.3.3 PREPROCESAMIENTO DE IMAGEN.....53

2.3.4 EXTRACCIÓN DE CARACTERÍSTICAS, RECONOCIMIENTO DE FORMA.....55

2.3.5 TRANSFORMACIÓN DE SISTEMAS DE COORDENADAS, RELACIÓN DEL PLANO DE LA IMAGEN (PIXEL) CON EL PLANO REAL (MILIMETRO) Y DETERMINACIÓN DEL ÁNGULO DE ROTACIÓN.....56

2.4 DISEÑO DEL HARDWARE Y SOFTWARE DEL ROBOT MANIPULADOR TIPO SCARA.....59

2.4.1 ANÁLISIS DEL ESTADO DE LAS PLACAS ELECTRÓNICAS 59

2.5 DISEÑO DEL SISTEMA DE CONTROL 60

2.5.1 PLACA DE CONTROL.....60

2.6 CONSTRUCCIÓN DE LA PLACA DE CONTROL.....62

2.6.1 PROGRAMA ISIS.....62

2.6.2 ARES.....63

2.6.3 PASOS DE CONSTRUCCIÓN DE LA PLACA DE CONTROL.....63

2.7 DESARROLLO DEL SOFTWARE.....64

2.7.1 POSICIONAMIENTO INICIAL DEL ROBOT SCARA 64

2.7.2 PROGRAMACIÓN DEL MICROCONTROLADOR ATMEGA 164P.....65

2.7.3 PROGRAMACIÓN DEL ENTORNO MATLAB.....71

CAPÍTULO 3..... 72

DESARROLLO DE UN HMI..... 72

3.1 DISEÑO E IMPLEMENTACIÓN DE LA INTERFAZ GRÁFICA PARA EL CONTROL SERVO VISUAL DE UN ROBOT MANIPULADOR TIPO SCARA. 72

3.1.1 ENTORNO GUI.....72

3.1.2 CONTROLES DE LA INTERFAZ GRAFICA 72

3.2 INTERFAZ GRÁFICA 73

3.1.3 DISEÑO DE LA INTERFAZ PARA EL CONTROLADOR SERVO VISUAL PARA UN ROBOT TIPO SCARA 73

CAPÍTULO 4 78

PRUEBAS Y RESULTADOS 78

4.1 ALGORITMO DE VISIÓN ARTIFICIAL 78

4.4.1 UMBRLIZACIÓN DE LA IMAGEN USANDO EL ESPACIO HSV 78

4.4.2 RECONOCIMIENTO DE IMÁGENES EN FUNCIÓN DE LA ILUMINACIÓN 79

4.4.3 RECONOCIMIENTO DE CARACTERÍSTICAS DE IMAGEN 82

4.4.4 ÁNGULO DE ORIENTACIÓN 82

4.2 DESEMPEÑO DEL CONTROLADOR SERVO VISUAL BASADO EN LAS CARACTERÍSTICAS DE LA IMAGEN. 83

4.2.1 PARÁMETROS DEL CONTROLADOR SERVO VISUAL 83

4.2.2 DESEMPEÑO DEL ROBOT MANIPULADOR TIPO SCARA 84

4.2.3 ÁREA DE TRABAJO DE LA CÁMARA CON EL ÁREA DE TRABAJO DEL ROBOT SCARA 85

4.3 RESULTADOS CONTROL SERVO VISUAL BASADO EN LAS CARACTERÍSTICAS DE IMAGEN 87

4.4 COSTOS DE IMPLEMENTACIÓN 93

CAPÍTULO 5 94

CONCLUSIONES Y RECOMENDACIONES 94

5.1. CONCLUSIONES 94

5.2. RECOMENDACIONES 95

REFERENCIAS BIBLIOGRÁFICAS 96

ANEXO A 1

MANUAL DE USUARIO 1

GUÍA DE INSTALACIÓN DE HADWARE 1

ANEXO B 1

GUÍA DE INSTALACIÓN DE SOFTWARE 1

B.1 PROGRAMA 1

B.2 INSTALACIÓN DEL PROGRAMA 1

B.3 UBICACIÓN DE LA INTERFAZ GRAFICA 1

ANEXO C 1

MANUAL DE USUARIO DE LA INTERFAZ GRÁFICA HMI 1

C.1 Comunicación Computadora- Robot 1

C.2 Control Lazo Abierto 1

C4 Gráficas de respuesta 4

ANEXO D 1

INDICE DE FIGURAS

Figura 1.1. Articulación Rotacional 2
Figura 1.2. Área de trabajo de un robot SCARA 3
Figura 1.3. Configuraciones básicas de robots industriales [2] 4
Figura 1.4. Robot SCARA de tres grados de libertad [3] 5
Figura 1.5. Disposición de un codificador óptico (encoder) incremental [3] 7
Figura 1.6. Encoder colocado en motor M3 [3] 7
Figura 1.7. Traslaciones en el espacio: (a) Eje XM; (b) Eje YM; (c) Eje ZM [4] 9
Figura 1.8. Representación gráfica de una matriz básica de traslación. 10
Figura 1.9. Rotaciones en el espacio respecto a: (a) eje X; (b) eje Y; (c) eje Z [4]. 11
Figura 1.10. (a) Imagen original, (b) Imagen digital, (c) Imagen píxeles 13
Figura 1.11. Representación gráfica de un píxel 14
Figura 1.12. Imagen en escala de grises 14
Figura 1.13. Imagen binarizada 15
Figura 1.14 Representación del Modelo de cámara Pin-hole [6] 16
Figura 1.15. Modelo de cámara Pin-hole sin versión de imagen [6] 17
Figura 1.16. Web Genius Facecam 320x 18
Figura 1.17. Segmentación de imagen, discontinuidad [8] 19
Figura 1.18. (a) Imagen binarizada, (b) Erosión [8] 20
Figura 1.19. Tipos de elementos estructurantes [9] 21
Figura 1.20. (a) Imagen binarizada, (b) Dilatación Binaria [8] 21
Figura 1.21. (a) Dilatación Binaria, (b) Apertura [8] 21
Figura 1.22. (a) Dilatación Binaria, (b) Cierre [8] 22
Figura 1.23. Reconocimiento de caracteres [9] 22
Figura 1.24. Representación del modelo HSV [11]. 23
Figura 1.25. Estabilidad del punto xa 25
Figura 1.26. Inestabilidad del punto xa 25
Figura 1.27. Estabilidad asintóticamente estable del punto xa 26
Figura 1.28. Curva de distribución normal [13] 28
Figura 1.29. Robots de 2 grados de libertad [1] 29
Figura 1.30. Relación entre la Jacobiana Analítica e Inversa. 32

Figura 1.31. Robot con cámara fija. 35
Figura 1.32. Robot con cámara en mano 36
Figura 1.33. Modelo de proyección de perspectiva. 37
Figura 1.34. Distribución de pines del micro controlador Atmega 164P. 39
Figura 2.1. Robot SCARA de tres grados de libertad 40
Figura 2.2. Esquema del controlador servo visual 43
Figura 2.3. Diagrama de bloques de la Cámara. 44
Figura 2.4. Diagrama de bloques del Jacobiano Total 47
Figura 2.5. Controlador Servo Visual. 49
Figura 2.6. Función tangente hiperbólica 51
Figura 2.7. Algoritmo de Visión 52
Figura 2.8. Adquisición de imagen mediante Matlab-Image Acquisition Tool. 53
Figura 2.9. Imagen real. 53
Figura 2.10. Imagen segmentada. 54
Figura 2.11. Imagen fragmentada. 55
Figura 2.12. Extracción de las características de la imagen. 55
Figura 2.13. Representación de pixeles en plano de la imagen. 57
Figura 2.14. Medición de lados. 57
Figura 2.15. Orientación, de Matlab. 58
Figura 2.16. Placa de control. 59
Figura 2.17. Placa de potencia 60
Figura 2.18. Pantalla inicial, programa ISIS 63
Figura 2.19. ARES: (a) Ubicación de los elementos; (b) Vista 3D de la placa. 63
Figura 2.20. Placa de control finalizada. 64
Figura 2.21. Diagrama de flujo, comunicación PC – micro controlador. 65
Figura 2.22. Diagrama de flujo de inicio, Motor 1. 67
Figura 2.23. Diagrama de flujo, Motor 2. 68
Figura 2.24. Diagrama de flujo, Pinza. 69
Figura 2.25. Trama de datos del micro controlador 70
Figura 2.26. Configuración interrupciones. 71
Figura 3.1. Entorno GUI para crear una interfaz. 72
Figura 3.2. Controles de interfaz de usuario 73
Figura 3.3. Panel de COMUNICACIÓN_PC_ROBOT. 74

Figura 3.4. Diagrama de las acciones que realiza el panel de COMUNICACIÓN_PC_ROBOT. 74
Figura 3.5. Panel ROBOT_SCARA, LAZO ABIERTO 75
Figura 3.6. Panel CONTROLADOR SERVO VISUAL, ROBOT_SCARA_CÁMARA_MANO 76
Figura 3.7. Diagrama de las tareas realizadas en el panel CONTROLADOR SERVO VISUAL 77
Figura 3.8. Interfaz de gráficas 77
Figura 4.1. Escoger espacio de color 78
Figura 4.2. (a) Imagen original, (b) Detección color azul, (c) Imagen binarizada . 79
Figura 4.3. (a) Imagen sin máscara HSV, (b) Imagen con máscara HSV 79
Figura 4.4. Detección con ambiente no controlado 80
Figura 4.5. Detección con ambiente controlado 80
Figura 4.6. Iluminación Directa 81
Figura 4.7. Imagen con mucha cantidad de luz 81
Figura 4.8. Coordenadas del centro de la imagen 82
Figura 4.9. Ángulo de orientación de la imagen 82
Figura 4.10. Área de trabajo robot con área de trabajo de la cámara 86
Figura 4.11. Objeto fuera del área de visión 86
Figura 4.12. Posición de la cámara 87
Figura 4.13. Características del objeto (punto rojo) y características deseadas (punto verde) 87
Figura 4.14. Evolución de las características en la imagen 88
Figura 4.15. Evolución del movimiento del robot alcanzando las características deseadas 88
Figura 4.16. Posición de la cámara en el plano de imagen (u, v) 89
Figura 4.17. Error de posición en el plano de imagen (u, v) 89
Figura 4.18.Angulo de Orientación 90
Figura 4.19. Acciones de control 90
Figura 4.20. Control servo visual sobre un objeto rotado 42 grados 91
Figura 4.21.Error de posición para un objeto rotado 42 grados 91
Figura 4.22. Posición para un objeto rotado 42 grados 92
Figura 4.23. Acciones de control para un objeto rotado 42 grados 92

Figura 4.24. Ángulo de orientación para un objeto rotado 42 grados..... 93
Figura A.1.1 Ubicación del robot SCARA..... 1

ÍNDICE DE TABLAS

Tabla 1.1. Características Hardware robot SCARA. 38
Tabla 1.2. Características del micro controlador Atmega 164P 39
Tabla 2.1. Valores de niveles de umbralización 54
Tabla 2.2. Distribución de pines y conexión, primer micro controlador. 61
Tabla 2.3. Distribución de pines y conexión, segundo micro controlador. 61
Tabla 2.4. Características del conversor USB TTL. 62
Tabla 2.5. Byte dirección de Motores 70
Tabla 2.6. Byte de Dirección de Motores 70
Tabla 4.1. Ángulos de Orientación de la imagen 83
Tabla 4.2. Constantes de calibración para el controlador Servo visual 83
Tabla 4.3. Datos de posiciones del robot SCARA 85
Tabla 4.4.Costos de Implementación 93

RESUMEN

El control visual requiere el uso de información visual obtenida de una o varias cámaras ya sean fijas o en mano, con la finalidad de controlar la posición y orientación del efector final del robot con respecto a un objeto o a un conjunto de características visuales del mismo. Existen robots que han sido equipados con sistemas de visión y que se encuentran en procesos de automatización a nivel industrial realizando ensamblaje de piezas, manipulación de objetos, entre otros.

El presente proyecto está enfocado en la implementación de un control servo visual en un robot manipulador tipo SCARA. Para cumplir con el objetivo de posicionar y orientar el efector final de este robot sobre un objeto estático deseado, se obtiene el modelo cinemático y se desarrolla un controlador servo visual basado en las características de la imagen, utilizando visión artificial por computador. El sensor usado para la adquisición de imágenes es una cámara ubicada en el efector final conocida como cámara en mano.

Se presenta el análisis de estabilidad del controlador usando la teoría de Lyapunov.

Se realiza un análisis del estado de funcionamiento del hardware y software del robot SCARA, para posteriormente realizar el diseño del nuevo sistema electrónico del robot en base a los requerimientos del análisis.

Con la finalidad de poder ver el desempeño del robot SCARA, se desarrolla una interfaz gráfica MATLAB-GUI, en el cual se facilita el uso de los programas realizados en MATLAB que hacen posible la implementación del controlador servo visual.

PRESENTACIÓN

En el presente proyecto se describe el diseño e implementación de un control servo visual para un robot manipulador tipo SCARA.

El Capítulo 1 presenta la definición de manipuladores industriales: robot manipulador industrial. Se menciona la morfología de un robot manipulador en general, como también la morfología del robot manipulador tipo SCARA, su estructura mecánica, sistema de accionamientos, actuadores, sensores, elementos terminales, representación de la posición y orientación y cinemática del robot. Se revisa la visión artificial, la estabilidad de Lyapunov, el índice de desempeño y la teoría del control servo visual a ser utilizados en el controlador y puesta en funcionamiento del sistema, para terminar, se realiza el análisis de funcionamiento actual del manipulador y los componentes electrónicos para poner en funcionamiento al robot SCARA.

El Capítulo 2 describe el desarrollo del algoritmo de control, la cinemática del robot SCARA donde se implementará el control servo visual basado en las características de la imagen, así como también el procesamiento y adquisición de imágenes donde la extracción de características, el reconocimiento de forma y la relación pixel – milímetro ayudan al reconocimiento del objeto deseado.

El diseño del sistema electrónico y de control se desarrolla en el Capítulo 3, analizando al sistema e identificando sus requerimientos, así como también el desarrollo del software y el diseño de la interfaz hombre-máquina.

El Capítulo 4 muestra los resultados obtenidos al implementar el presente proyecto, cumpliendo con los objetivos planteados.

Las conclusiones y recomendaciones se presentan en el Capítulo 5.

CAPÍTULO 1
MARCO TEÓRICO

En esta sección se presentan conceptos generales acerca del robot manipulador y su estructura, clasificación y cinemática. A continuación, se trata el tema de la visión artificial. Luego, se define los conceptos matemáticos que permitirán el desarrollo de los modelos cinemáticos y controlador servo visual.

1.1 MANIPULADORES INDUSTRIALES

1.1.1 ROBOT MANIPULADOR INDUSTRIAL

La Asociación Internacional de Estándares (ISO) define al robot manipulador como:

"Robot manipulador industrial (ISO): Manipulador de 3 o más ejes, con control automático, reprogramable, multi-aplicación, móvil o no, destinado a ser utilizado en aplicaciones de automatización industrial. Incluye al manipulador (sistema mecánico y accionadores) y al sistema de control (software y hardware de control y potencia)" [1].

1.2 MORFOLOGÍA DE UN ROBOT MANIPULADOR

Se conoce como morfología de un robot manipulador al conjunto de elementos que forman su estructura tales como [2]: tipos de articulaciones, estructuras básicas, orientación del efector final y efectores finales.

1.2.1 ESTRUCTURA DE ROBOTS MANIPULADORES

Un robot manipulador es un brazo articulado formado por un conjunto de eslabones consecutivos unidos entre dos por las articulaciones, las mismas que definen los grados de libertad que va a tener el robot.

1.2.1.1 Tipos de articulaciones

Las articulaciones pueden ser de tipo: esférica, planar, tornillo, rotacional, prismática y cilíndrica. De las cuales se hará mención la articulación rotacional, ya que se usa en el presente trabajo, como se muestra en la Figura 1.1.

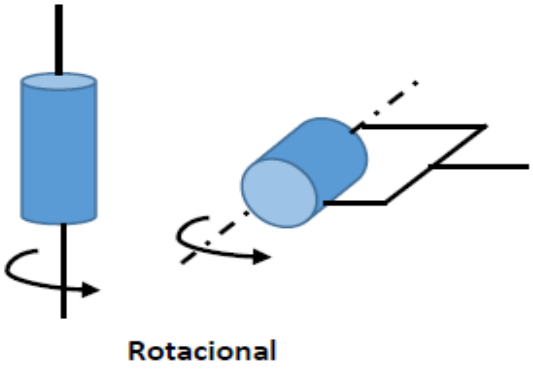


Figura 1.1. Articulación Rotacional

1.2.1.1.1 Articulación rotacional

Produce un movimiento de rotación entre dos eslabones formando un ángulo de rotación.

1.2.1.2 Espacio de tarea

Es un parámetro importante dentro del diseño del robot, ya que se encuentra definido como el grupo de puntos que pueden ser alcanzados por el efector final, es decir, es la superficie de desenvolvimiento del robot, donde se reconocen localización, posición y orientación del efector final, generalmente expresadas en coordenadas cartesianas.

El robot SCARA presentado en este trabajo, posee tres grados de libertad, dos rotacionales y una prismática, permite el movimiento dentro de los límites dados por los ángulos o distancias máximos de este robot.

El área de trabajo se encuentra definida como se muestra en la Figura 1.2

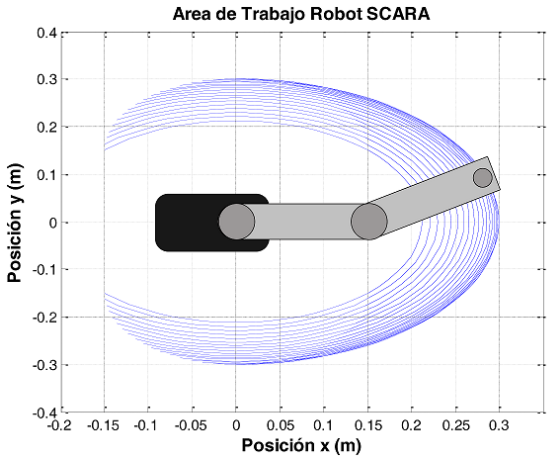


Figura 1.2. Área de trabajo de un robot SCARA

1.2.1.3 Estructuras básicas

La estructura de un robot manipulador se puede clasificar de acuerdo al espacio de trabajo y a la facilidad que tenga para ocupar posiciones determinadas.

- Robot Cartesiano

  Conocido como configuración (PPP), tiene tres ejes de control lineales y perpendiculares, como se observa en la Figura 1.3 (a).

- Robot Cilíndrico

  Configuración (RPP), cuya base tiene un movimiento rotacional y sus dos ejes perpendiculares, como se observa en la Figura 1.3 (b).

- Robot Esférico

  Presenta dos ejes rotacionales y un eje planar, como se observa en la Figura 1.3 (c).

- Robot Angular

  Presenta sus tres ejes rotacionales, como se observa en Figura 1.3 (d).

- Robot SCARA

  Presenta una configuración (RRP), tiene dos ejes paralelos rotacionales, y un eje lineal paralelo al plano de desplazamiento vertical. Se los utiliza para realizar tareas de montaje en un plano. [2]

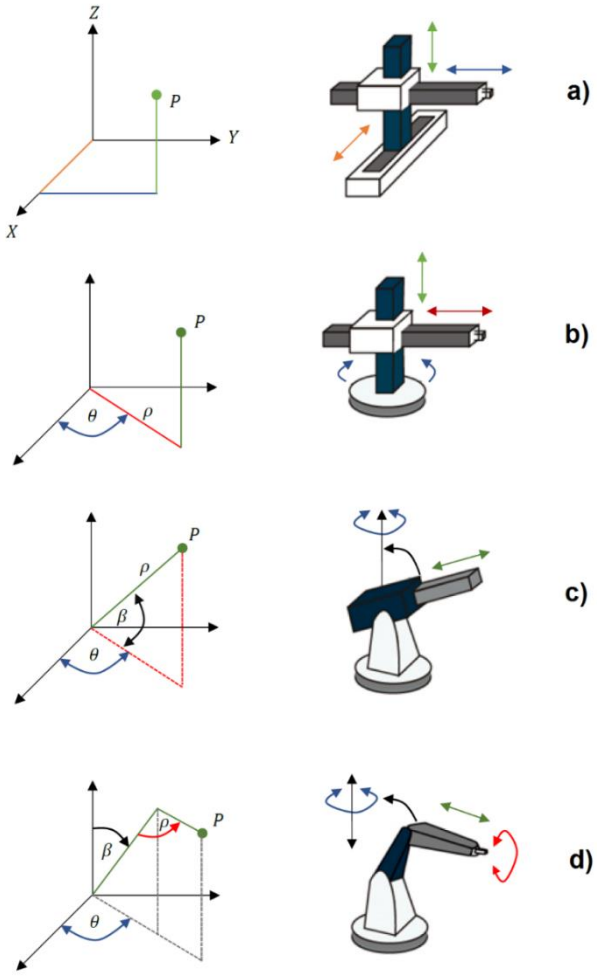


Figura 1.3. Configuraciones básicas de robots industriales [2]

1.3 MORFOLOGÍA DE UN ROBOT MANIPULADOR TIPO SCARA

El robot manipulador tipo SCARA (*Selective Compliance Assembly Robotic Arm)*, tiene como ventaja un reducido número de grados de libertad, su arquitectura es la más usada para el ensamblaje de piezas electrónicas entre otras aplicaciones similares.

1.3.1 ESTRUCTURA MECÁNICA DE UN ROBOT SCARA

El SCARA de este trabajo presenta una configuración (RRP), son dos las articulaciones rotacionales denominadas θ_1 y θ_2, cuyos ángulos de rotación varían entre $-\pi/2$ y $\pi/2$. Las dimensiones del brazo a_2 y antebrazo a_3 son de 0.15 m. La tercera articulación q_3 permite el desplazamiento de la articulación prismática en un

plano perpendicular al movimiento de las otras articulaciones. La longitud del cuerpo d_1 es de 0.12 m, medida desde la base del robot hasta la unión del cuerpo con el brazo, considerando la nomenclatura de Denavith Hartenberg, como se observa en la Figura 1.4,

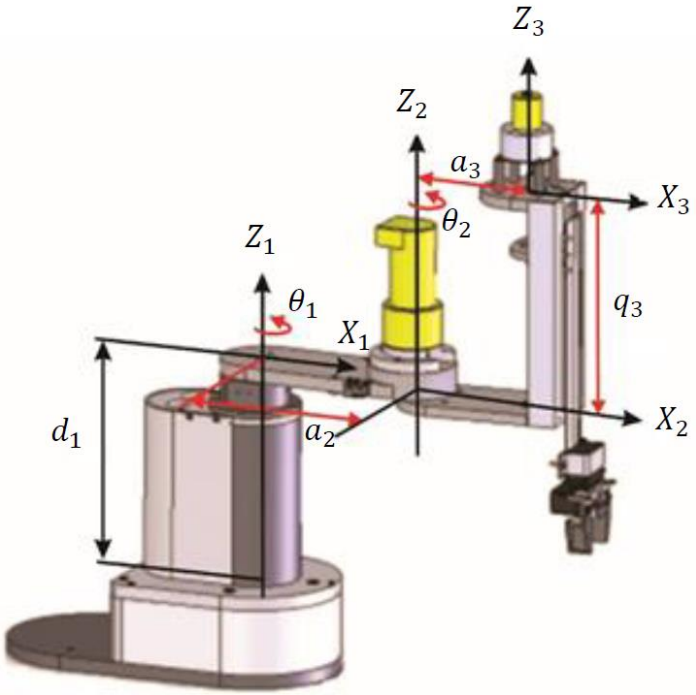


Figura 1.4. Robot SCARA de tres grados de libertad [3]

1.3.2 SISTEMAS DE ACCIONAMIENTOS [3]

Los sistemas de accionamientos son utilizados en robots que requieran combinar gran precisión con alta velocidad (como es el presente robot SCARA): al usar un accionamiento directo se tiene un posicionamiento rápido y preciso, el aumento de la controlabilidad del sistema y la simplificación del sistema mecánico.

1.3.3 ACTUADORES

La importancia de los actuadores en un robot es generar el movimiento de los eslabones según las ordenes enviadas por un controlador. Se pueden clasificar en:

- Actuadores Eléctricos. - Este tipo de actuador es el más usado en robots industriales, cuya acción permita mover las articulaciones de los robots. Dentro de estos actuadores encontramos tres categorías diferentes: motores de corriente continua, motores de corriente alterna y motores paso a paso.

- Actuadores Neumáticos. - Pueden ser lineales y rotativos siendo movidos por el aire. Las aplicaciones de estos actuadores se los puede observar en válvulas neumáticas con movimiento lineal y motores neumáticos con movimiento de rotación continua.

- Actuadores Hidráulicos. - Basa su funcionamiento en la presión ejercida por un líquido, sirven especialmente en aplicaciones donde requieran carga pesada.

Los motores utilizados en el robot SCARA, tienen un accionamiento directo, son motores DC sin escobillas, con imanes permanentes y cuyos ejes van conectados sin intermedios a cada una de las articulaciones RRP del robot, debido a que cada motor cuenta con su propia caja reductora [3].

1.3.4 SENSORES

Las tareas asignadas a un robot deben ser en su mayoría precisas, inteligentes, con un nivel de respuesta bastante rápida, que tenga un conocimiento de su estado y el de su entorno.

- Sensores Internos. - Conocimiento de posición, velocidad y aceleración de las articulaciones del robot.
- Sensores Externos. - Conocimiento del estado de su entorno como la visión.

Los sensores utilizados en el robot SCARA, son sensores internos como los de posición (encoders) y presencia (finales de carrera) y sensores externos como los de visión (cámara), los mismos que darán información continua al sistema de control sobre la posición del efector final y sus eslabones.

1.3.4.1 Sensores de posición

1.3.4.1.1 Sensores de posición en cuadratura

Para obtener la posición angular de los motores, se utilizan codificadores ópticos en cuadratura (acoplados de fábrica), formados por un disco transparente con

divisiones blancas y negras que están marcadas radialmente y equidistantes entre sí, posee tres canales desfasados 90º con una resolución de 0.003º (según datos del fabricante); un emisor de luz pasa a través de estas marcas hasta llegar a un elemento foto-receptor para generar un pulso y conocer la posición del eje, como se observa en la Figura 1.5.

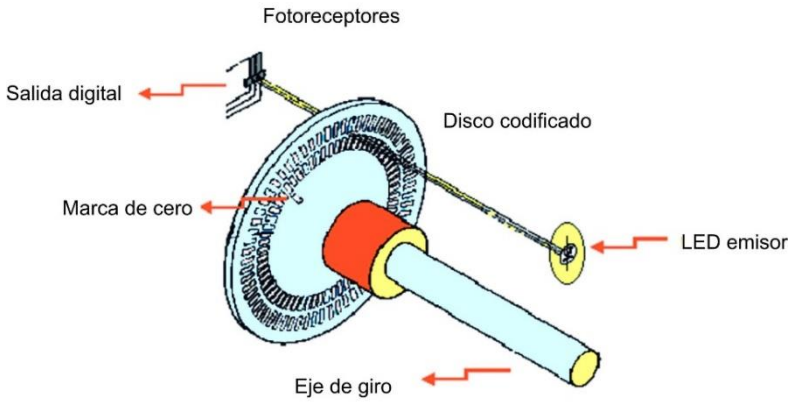


Figura 1.5. Disposición de un codificador óptico (encoder) incremental [3]

1.3.4.1.2 Sensores de posición con interrupción luminosa

Están compuestos por un emisor y detector de luz (fototransistor), la velocidad de respuesta es mayor al de foto-resistores. Al interponerse un objeto deja de incidir el haz luminoso cambiando su valor.

El sensor se encuentra acoplado al eje del tercer motor del robot SCARA que produce un movimiento vertical del último eslabón, es de un pulso por revolución equivalente a un desplazamiento de 1.6 mm aproximadamente, como se muestra en la Figura 1.6.

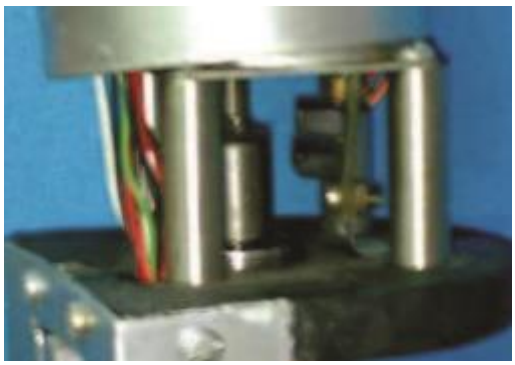


Figura 1.6. Encoder colocado en motor M3 [3]

1.3.4.2 Sensores de presencia

Estos sensores activan o desactivan automáticamente un mecanismo eléctrico al que se encuentre conectado; tienen un radio de acción determinado en donde detecta o no la presencia del objeto. Se utiliza un interruptor abierto o cerrado cambiando su estado.

En el robot SCARA desarrollado se usa cinco sensores de presencia tipo interruptor, los cuales se encuentran colocados estratégicamente, que sirven para dar información al sistema de control sobre la posición final del brazo y antebrazo como también la partida y llegada del efector final.

1.3.4.3 Sensores de visión

Los sensores de visión pertenecen al grupo de los sensores externos que brindan información sobre el entorno del robot. Utilizan imágenes capturadas por una cámara para establecer la presencia, orientación y precisión de las piezas.

1.3.5 ELEMENTOS TERMINALES

Son dispositivos adicionales que permiten la interacción del robot con el entorno; los actuadores finales se utilizan para describir la herramienta que está unida a la muñeca para que el robot cumpla aplicaciones particulares.

Los efectores finales pueden ser pinzas usadas para la sujeción y toma de objetos durante el ciclo de trabajo del robot, y las herramientas para realizar una tarea específica sobre las piezas de trabajo.

1.3.6 REPRESENTACIÓN DE LA POSICIÓN Y ORIENTACIÓN [4]

Para la localización de un robot en el espacio, es necesario conocer la posición y orientación adecuada en el espacio, para lo cual se requiere determinar un sistema de referencia el cual permita describir la posición y orientación del efector final del robot (pinza).

Partiendo de este propósito, las coordenadas homogéneas y la matriz de transformación homogénea son empleadas en robótica para determinar conjuntamente en una sola matriz la posición y orientación de un objeto respecto

de un sistema de referencia. Entonces la matriz de transformación homogénea está dada por la ecuación (1.1).

$$T = \begin{bmatrix} R_{3x3} & p_{3x1} \\ f_{1x3} & w_{1x1} \end{bmatrix} = \begin{bmatrix} Rotación & Traslación \\ Perspectiva & Escalado \end{bmatrix} \quad (1.1)$$

En robótica, la sub matriz f_{1x3}, que se refiere a la transformación de perspectiva, es nula porque representa a la proyección de un objeto visto por un lente; mientras que la sub-matriz w_{1x1}, es igual a la unidad ya que representa al escalado global.

$$T = \begin{bmatrix} R_{3x3} & p_{3x1} \\ 0 & 1 \end{bmatrix} \quad (1.2)$$

Entonces, la matriz de transformación definida por la ecuación (1.2) representa la posición y orientación de un sistema girado y trasladado a los ejes principales de un sistema asociado M', con respecto a un sistema fijo de referencia M.

1.3.6.1 Traslación

Una matriz de transformación puede presentar traslaciones básicas, como se observa en la Figura 1.7.

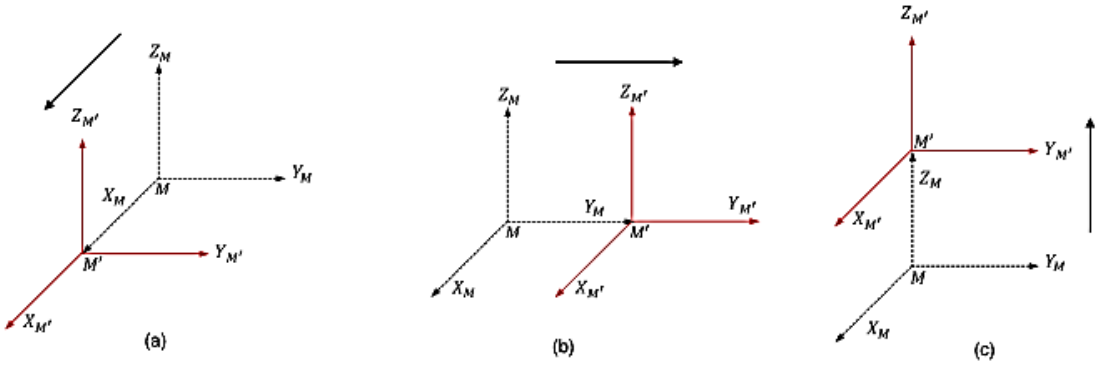


Figura 1.7. Traslaciones en el espacio: (a) Eje $\boldsymbol{X_M}$; (b) Eje $\boldsymbol{Y_M}$; (c) Eje $\boldsymbol{Z_M}$ [4]

La matriz de básica de traslación está definida por la ecuación (1.3), en donde la matriz de transformación homogénea resultante estará compuesta por una submatriz de rotación que será la identidad, ya que únicamente se trasladó un vector con respecto al sistema fijo o posición inicial.

$$Trans(p_x, p_y, p_z) = \begin{bmatrix} 1 & 0 & 0 & p_x \\ 0 & 1 & 0 & p_y \\ 0 & 0 & 1 & p_z \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{1.3}$$

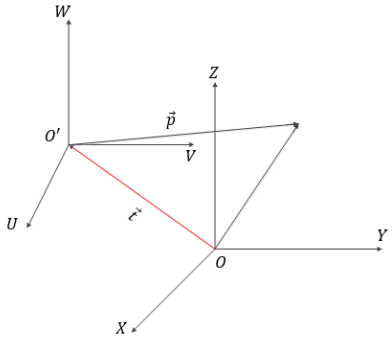


Figura 1.8. Representación gráfica de una matriz básica de traslación.

En la Figura 1.8 el sistema $O'UVW$ está trasladado un vector t_{UVW} con respecto al sistema $OXYZ$, para calcular el vector de translación p en el sistema $OXYZ$ se emplea la ecuación (1.4).

$$p = \begin{bmatrix} 1 & 0 & 0 & p_x \\ 0 & 1 & 0 & p_y \\ 0 & 0 & 1 & p_z \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} t_U \\ t_V \\ t_W \\ 1 \end{bmatrix} = \begin{bmatrix} t_U + p_x \\ t_V + p_y \\ t_W + p_z \\ 1 \end{bmatrix} \tag{1.4}$$

1.3.6.2 Rotación

Las matrices de rotación se utilizan para definir la rotación sobre el eje x, y, z, es decir, ayudan a representar un punto con respecto a un nuevo sistema de referencia.

Se puede definir tres matrices homogéneas básicas de rotación considerando el eje de dicha rotación con sus respectivos ángulos $(\propto, \beta, \gamma)$. Estas rotaciones se observan en la Figura 1.9.

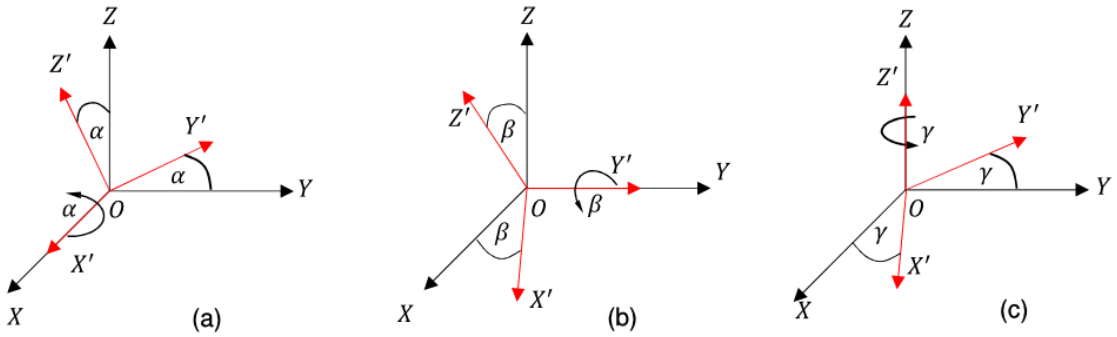


Figura 1.9. Rotaciones en el espacio respecto a: (a) eje X; (b) eje Y; (c) eje Z [4].

Donde ∝ es el ángulo de orientación del sistema ($OX'Y'Z'$), cuando el eje OX' coincide con el eje OX, representada por la siguiente matriz.

$$Rot(\alpha) = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & \cos\alpha & -\sin\alpha & 0 \\ 0 & \sin\alpha & \cos\alpha & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (1.5)$$

Donde β es el ángulo de orientación del sistema ($OX'Y'Z'$), cuando el eje OY' coincide con el eje OY, representada por la siguiente matriz.

$$Rot(\beta) = \begin{bmatrix} \cos\beta & 0 & \sin\beta & 0 \\ 0 & 1 & 0 & 0 \\ -\sin\beta & 0 & \cos\beta & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (1.6)$$

Donde γ es el ángulo de orientación del sistema ($OX'Y'Z'$), cuando el eje OZ' coincide con el eje OZ, representado por la siguiente matriz.

$$Rot(\gamma) = \begin{bmatrix} \cos\gamma & -\sin\gamma & 0 & 0 \\ \sin\gamma & \cos\gamma & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (1.7)$$

1.3.6.3 Composición de matrices homogéneas

Una transformación total puede descomponerse en transformaciones simples consecutivas si lo que cambia es la posición del objeto con respecto a un sistema

de referencia y/o de rotación, si lo que se produce es un giro del objeto con respecto al sistema de referencia.

Es relevante aclarar que no es lo mismo trasladar y girar con respecto a un sistema de referencia, que girar y trasladar con respecto al mismo sistema de referencia.

En este caso, primero se traslada el objeto a la posición del efector final del robot y luego se realiza una rotación del extremo final del robot sobre la posición que ocuparía en ese instante.

1.3.7 CINEMÁTICA DEL ROBOT

La cinemática del robot estudia el movimiento del mismo respecto a un sistema de referencia, en donde el principal interés es la descripción analítica del movimiento espacial del robot, así como también las relaciones entre la posición y orientación de su extremo final.

1.3.7.1 Cinemática directa

Como ya se mencionó, la cinemática directa permite conocer la posición u orientación del extremo final del robot en donde las variables fijan la posición y orientación de sus articulaciones tomando valores determinados.

1.3.7.2 Cinemática inversa

La cinemática inversa consiste en encontrar valores que deben adoptar las coordenadas del robot, determinando que el extremo se posicione y oriente con respecto a una ubicación espacial.

1.3.7.3 Cinemática de movimiento

Se trata de la relación entre las velocidades del movimiento de las articulaciones y las del extremo, en donde esta relación esta expresada mediante la matriz Jacobiana.

1.4 VISIÓN ARTIFICIAL

La visión artificial es un campo de la “Inteligencia Artificial”, que proporciona información a las computadoras mediante el análisis y procesamiento de imágenes.

La visión artificial tiene como objetivo obtener información visual de la captura de imágenes tomadas del entorno de trabajo para extraer las características de la imagen, las mismas que son el resultado de las proyecciones en el plano de la imagen de las características físicas del objeto.

Hoy en día, el uso de la visión artificial cumple un papel importante para el desarrollo e implementación de controladores en donde el análisis, procesamiento y extracción de características permitan a los robots tener mejores respuestas ante las tareas asignas, por ejemplo, en el ensamblaje de piezas, manipulación de objetos, maquinado, medicina y realización de tareas conjuntas humano-robot [5].

1.4.1 MÉTODOS DE CAPTACIÓN DE LAS IMÁGENES

1.4.1.1 Imagen digital

Es una representación binaria de la imagen original, en donde cada punto de la imagen original es representado por un número entero positivo.

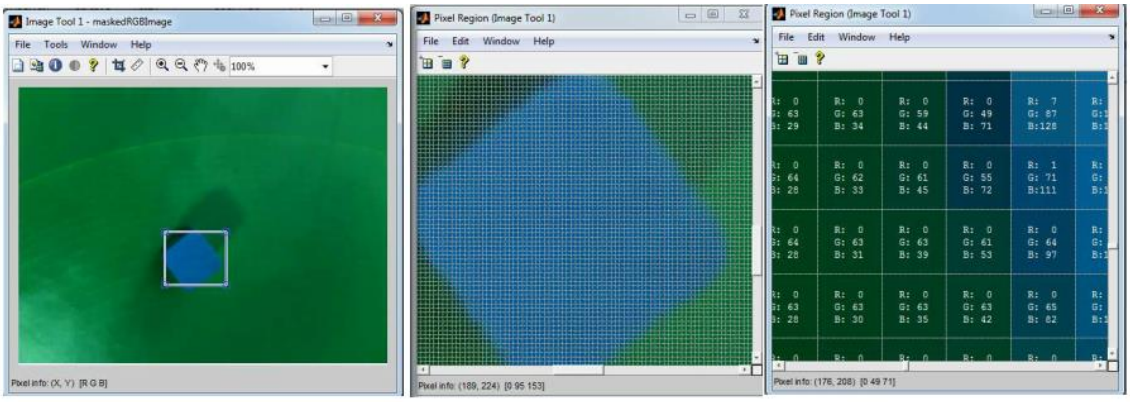


Figura 1.10. (a) Imagen original, (b) Imagen digital, (c) Imagen píxeles

1.4.1.2 Píxel

Arreglo bidimensional de valores de brillo y color. Tienen una ubicación fila-columna y un valor asignado obtenido a partir de una escena tridimensional.

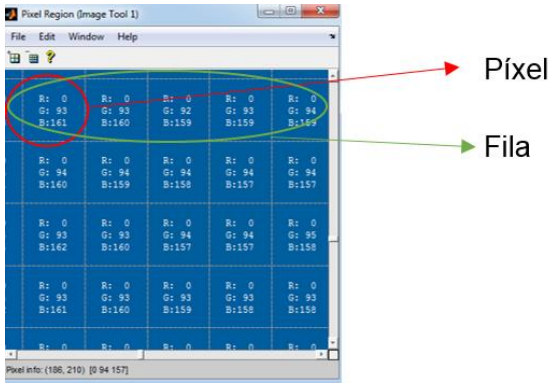


Figura 1.11. Representación gráfica de un píxel

1.4.1.3 Nivel de grises

Es la digitalización de la imagen en donde cada pixel representa un valor individual de luminancia en una escala que se extiende entre blancos y negros.

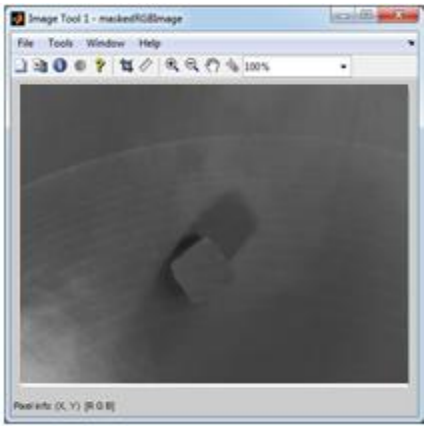


Figura 1.12. Imagen en escala de grises

1.4.1.4 Imagen binaria

Consiste en la binarización de la imagen original, en donde la información obtenida se reduce a dos únicos valores verdadero y falso, siendo vistos en una imagen digital como unos y ceros conocidos como blanco y negro.

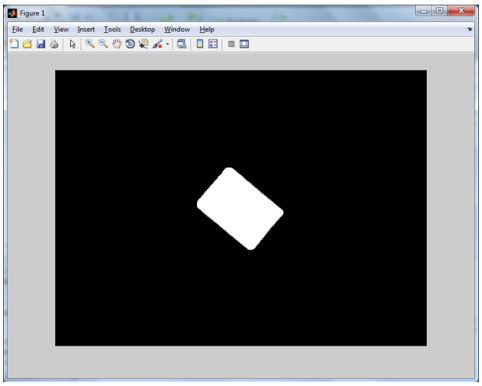


Figura 1.13. Imagen binarizada

1.4.1.5 Escena

Es un área de memoria en donde se guarda todos los parámetros tomados de un objeto, pueden ser: cámara utilizada, imágenes patrón memorizados, datos a visualizar, entradas y salidas de control, entre otros [5].

1.4.2 MODELADO DEL SISTEMA DE VISIÓN

1.4.2.1 Modelo de una cámara

Es el modelo geométrico asociado al proceso de información de una imagen con la cámara. Este modelo está caracterizado por un cierto número de parámetros los cuales determinan las propiedades intrínsecas de la cámara y su situación en el mundo. Cuando las propiedades intrínsecas no cambian y la situación de la cámara es la misma, al utilizar la matriz de proyección es posible localizar la cámara en un entorno a partir de las imágenes obtenidas del mismo, así como obtener las características intrínsecas de la cámara.

1.4.2.2 Calibración de la cámara

La calibración precisa de la cámara permite obtener distancias en el mundo real a partir de las imágenes tomadas del mismo, es un proceso para obtener los parámetros intrínsecos y extrínsecos que definen el modelo de la cámara, tanto lineal como no lineal.

1.4.2.2.1 Parámetros Intrínsecos y Extrínsecos

Los parámetros extrínsecos miden la posición y la orientación de la cámara respecto al sistema de coordenadas establecido para la escena. Mientras que los parámetros intrínsecos describen la geometría y óptica del conjunto cámara y tarjeta de adquisición de imágenes.

1.4.2.3 Formación de la imagen

En el proceso de formación de imagen la cámara posee un lente que forma una proyección bidimensional (2D) de la escena sobre el plano de la imagen, lugar donde está localizado el sensor de visión. Esta proyección causa la pérdida de la información de profundidad en donde cada punto sobre el plano de la imagen corresponde a un rayo en el espacio tridimensional (3D).

1.4.2.4 Modelo de cámara Pin-hole

Este modelo describe la relación matemática que existe entre las coordenadas de un punto en 3D y su proyección en el plano de imagen; posee un pequeño agujero en el cual por la propagación de la luz que lo atraviesa, se crea una imagen del espacio externo en el lado opuesto de la caja, como se observa en Figura 1.14.

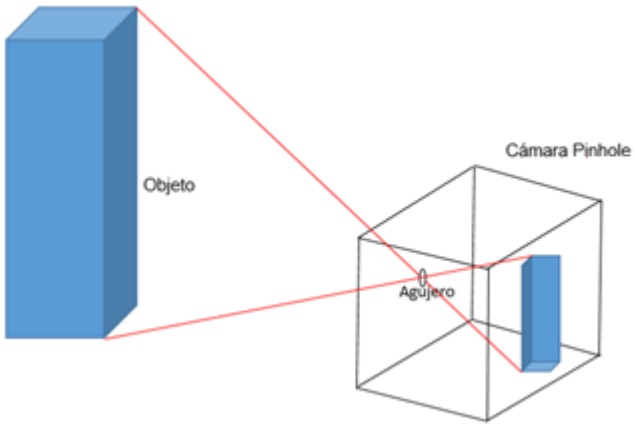


Figura 1.14 Representación del Modelo de cámara Pin-hole [6]

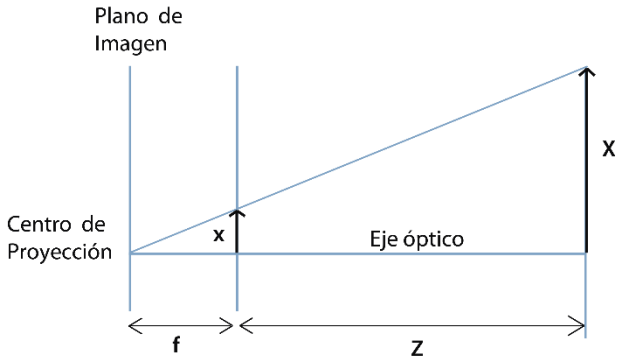


Figura 1.15. Modelo de cámara Pin-hole sin versión de imagen [6]

Una vez encontrado el modelo equivalente de una cámara Pin-hole como se muestra en la Figura 1.15 se puede realizar geometría en los triángulos para determinar su semejanza de la cual se obtiene la ecuación (1.8).

Dónde:

$$\frac{f}{x} \approx \frac{Z}{X} \tag{1.8}$$

f: Es la distancia focal

Z: Distancia desde la cámara hacia el objeto

X: Tamaño real de la imagen

x: Tamaño de la imagen proyectada

La distancia focal de la cámara f es constante, es decir, no depende de la distancia a la que se encuentre la imagen. Para obtener la distancia focal se procede a colocar la imagen con tamaño real conocido X frente a la cámara a una distancia conocida Z y utilizando el cálculo del área se calcula la altura del objeto x, como se observa en la ecuación (1.9).

$$f = \frac{Z\,x}{X} \tag{1.9}$$

1.4.3 ESTRUCTURA FÍSICA DE VISIÓN ARTIFICIAL

Las etapas de la visión artificial son el sistema de formación de imágenes y el sistema de procesamiento de imágenes. En el sistema de formación de imágenes

encontramos iluminación, captación y adquisición de la imagen, mientras que el sistema de procesamiento desarrolla algoritmos para transformar la imagen adquirida en información.

1.4.4 PROCESAMIENTO DE IMÁGENES

Se conoce que las imágenes capturadas obtienen información completa que debe ser procesada de forma digital para extraer informaciones utilizables para aplicaciones determinadas. Estos procesamientos presentan las siguientes etapas.

1.4.4.1 Adquisición de imágenes

Para desarrollo de este proyecto, en la obtención de imagen se usará la cámara Web Genius Facecam 320x como se observa en la Figura 1.16.

Figura 1.16. Web Genius Facecam 320x

Las características de esta cámara son las siguientes:

- Tipo de conexión: USB
- Compatibilidad: USB 1.1 /1.0 /2.0
- Giro: 360º horizontal y +-25º vertical
- Sensor de imagen: CMOS 640x480 (VGA)
- Graba archivo en formato JPEG (fotos) y WMV (video en alta calidad)
- Resolución: Imagen fija 8MP
- Resolución de video: 30fps
- Dimensiones (ancho x alto x profundidad): 5.1x4.3x3.9 cm

- Peso: 69 g
- Soporte de software: Windows 8,7, Vista, XP, Linux 2.3.21 y Mac 10.4.9

1.4.4.2 Análisis de imágenes

1.4.4.2.1 Pre procesamiento

Esta etapa presenta técnicas para el mejoramiento de la imagen digitalizada, como por ejemplo la reducción del ruido. En esta etapa de pre procesamiento se consideran dos métodos en el dominio del espacio que operan directamente sobre los pixeles y métodos en el dominio de la frecuencia como la transformada de Fourier espacial.

1.4.4.2.2 Segmentación [7]

Es el proceso de división de una imagen para obtener regiones u objetos de interés. Los objetos se extraen de la escena para luego ser analizados y reconocidos, las técnicas utilizadas pueden ser:

a) *Discontinuidad* cuando se obtiene los pixeles del contorno que permite detectar puntos aislados, líneas y aristas (bordes).

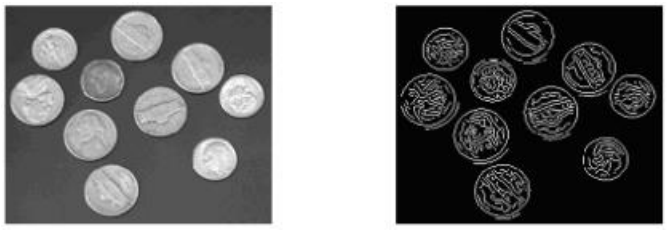


Figura 1.17. Segmentación de imagen, discontinuidad [8]

b) *Similitud cuando se* obtienen las regiones por alguna propiedad común de los pixeles, que permite construir regiones por división y fusión, por crecimiento o por umbralización, esta técnica es utilizada para separar los objetos respecto del fondo de la imagen capturada, cuando se aplica un umbral la imagen queda binarizada como se describe en la ecuación

$$I(x,y) = \begin{cases} 1 \; f(x,y) > T \\ 0 \;\; f(x,y) < T \end{cases} \tag{1.10}$$

Donde

$f(x,y)$: Función que retorna nivel de gris del pixel.

T: Umbral.

$I(x, y)$: Imagen binarizada.

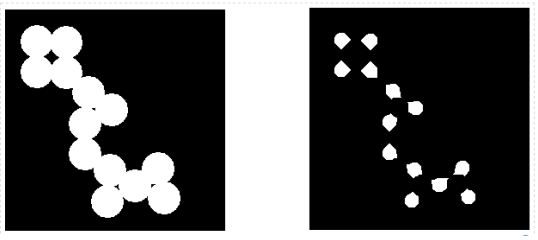
1.4.4.2.3 Descripción

Consiste en extraer las características de un objeto para reconocerlo. Los descriptores deben ser independientes de la localización, orientación y cambios de escala de los objetos de una escena, y deben contener suficiente información para distinguir un objeto de otro.

Para el desarrollo de los algoritmos para el procesamiento de imágenes tenemos:

Erosión binaria: La transformación de la erosión es el resultado de comprobar si el elemento estructurante $\boldsymbol{B}$ está completamente incluido dentro de la imagen $\boldsymbol{X}$. Cuando no ocurre el resultado de la erosión es el conjunto vacío. La erosión supone una degradación de la imagen, como se observa en la Figura 1.18, pues consiste en definir una geometría determinada por el elemento estructurante y pasarlo sobre la imagen. Un elemento estructurante es una figura geométrica conocida, el tamaño y forma se elige de acuerdo con la morfología sobre el que va a interseccionar y en función de la obtención de formas que se desea extraer, en la Figura 1.19 se muestra algunos tipos de elementos estructurantes empleados [8].

$$\varepsilon_B(X) = \{x | B_x \subseteq X\} \tag{1.11}$$



(a) (b)

Figura 1.18. (a) Imagen binarizada, (b) Erosión [8]

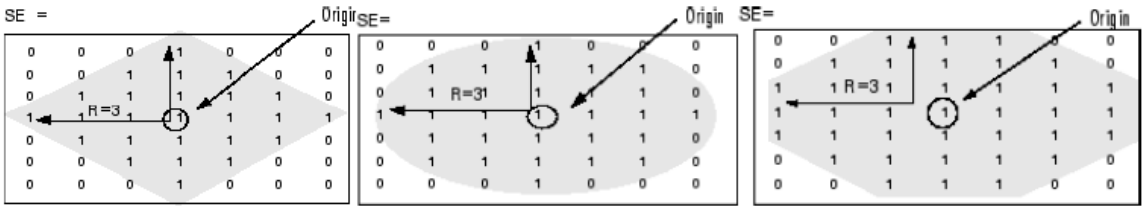
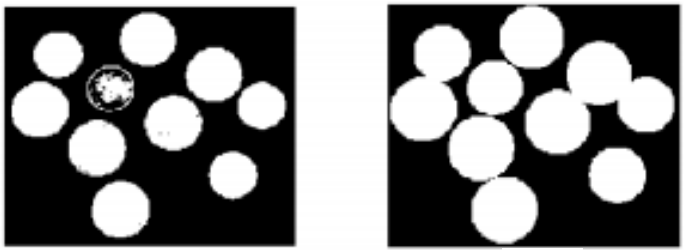


Figura 1.19. Tipos de elementos estructurantes [9]

Dilatación binaria: El resultado de la dilatación es el conjunto de elementos tal que al menos algún elemento del conjunto estructurante $\boldsymbol{B}$ está contenido en el conjunto $\boldsymbol{X}$, cuando $\boldsymbol{B}$ se desplaza sobre el conjunto $\boldsymbol{X}$ [8]. La dilatación puede interpretarse como el valor máximo del entorno de vecindad definido por el elemento estructurante [8], como se observa en la Figura 1.20.

$$\delta_B(X) = \{x | X \cap B_x \neq \emptyset\} \tag{1.12}$$



(a) (b)

Figura 1.20. (a) Imagen binarizada, (b) Dilatación Binaria [8]

a) Apertura: Esta técnica nos ayuda a suavizar los contornos de una imagen y la eliminación de pequeños salientes. Sirve para la eliminación de zonas estrechas en un elemento estructural, como se observa en la Figura 1.21 [8].

(a) (b)

Figura 1.21. (a) Dilatación Binaria, (b) Apertura [8]

b) Cierre: Rellena pequeños huecos para eliminar espacios no deseados en una imagen digital, como se observa en la Figura 1.22 [8].

(a) (b)

Figura 1.22. (a) Dilatación Binaria, (b) Cierre [8]

1.4.4.2.4 Reconocimiento

Procedimiento que identifica y clasifica los objetos en grupos, en donde se identifica los objetos segmentados en función de un conocimiento previo para etiquetarlos, como se observa en la Figura 1.23.

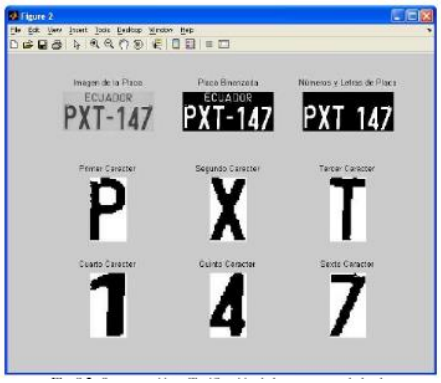


Figura 1.23. Reconocimiento de caracteres [9]

1.4.4.2.5 Interpretación

Asociación de un significado al conjunto de objetos reconocidos en una escena.

1.4.5 TIPOS DE PROCESAMIENTO DE DATOS

Estas etapas se analizan de acuerdo con el tamaño de datos que vamos a requerir para nuestro procesamiento y pueden ser:

a) Procesamiento de bajo nivel

En las primeras etapas de este procesamiento se maneja altos volúmenes de datos.

b) Procesamiento de alto nivel

En las etapas finales del procesamiento la cantidad de datos es significativamente menor, pero con requerimiento mayor de inteligencia.

1.4.6 MODELOS DE COLOR

1.4.6.1 Modelo RGB

El espacio de color RGB formado por los colores primarios rojo, verde y azul, representa una imagen mostrada en un monitor. El espacio RGB establece un valor de intensidad para cada pixel entre 0 (negro) y 255 (blanco) que comprende los pixeles del espacio RGB de una imagen en color.

1.4.6.2 Modelo HSV [10]

El espacio HSV (Hue/Matiz - Saturation/Saturación – Value/Valor) deriva del espacio RGB y representa los colores combinando tres valores.

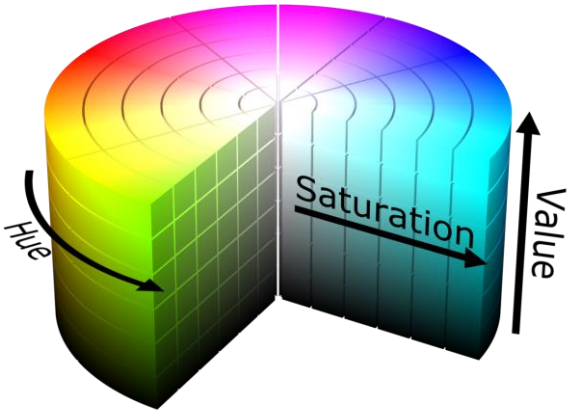


Figura 1.24. Representación del modelo HSV [11].

- **Tono (H):** Se refiere a la frecuencia dominante del color dentro del espectro visible y viene en grados.

- **Saturación (S):** Se refiere a la cantidad del color o a la pureza de este, va de un color claro a un color más vivo, expresado en %.
- **Brillo (V):** Es la intensidad de luz de un color y viene en %.

Se usa este espacio de color considerando la zona de color más grande, conservando la forma de la imagen; esta técnica nos permite desarrollar algoritmos de procesamiento de imágenes, en donde la percepción de color sea parecida a la forma como la visualizan los seres humanos.

1.5 ESTABILIDAD DE LYAPUNOV

La estabilidad es una de las características más importantes de los sistemas no lineales o sistemas variantes en el tiempo.

El sistema es estable en el sentido de Lyapunov, si todas las soluciones que se inicien en las cercanías del punto de equilibrio permanecen en dichas cercanías, caso contario es inestable. Mientras que es asintóticamente estable si las soluciones tienden al mismo punto conforme el tiempo transcurre [12].

La ecuación (1.13) describe las condiciones iniciales para definir el punto de equilibrio:

$$\frac{dx(t)}{d(t)} = F\big(x(t)\big); t \geq 0; x(t) \in \mathcal{R}^n; f: \mathcal{R}^n \times \mathcal{R}^n \rightarrow continua \tag{1.13}$$

Se asume que existe solución única $S(t, t_0, x_0)$ para cada condición inicial $x(t_0) = x_0$.

El estado x_a es un punto de equilibrio si:

Entonces la solución única esta expresada por:

$$S(t, t_0, x_a) = x_a; \ \forall t \geq t_0 \geq 0 \tag{1.14}$$

Entonces se tiene el punto de equilibrio:

$$f[t, \boldsymbol{O}] \equiv \boldsymbol{O}; \ \forall t \geq 0 \tag{1.15}$$

$$\text{Si } x(t_0) = \boldsymbol{O}, \ S(t, t_0, \boldsymbol{O}) = \boldsymbol{O}; \ \forall t \geq t_0 \geq 0 \tag{1.16}$$

- *Sistema estable*: Se dice que un punto de equilibrio x_a es estable si, para cada $\varepsilon > 0;\ t_0 \geq 0, \exists\ \delta(t_0, \varepsilon) > 0$ tal que:

$$\| x_0 \| < \delta(t_0, \varepsilon) \rightarrow \| S(t, t_0, x_0) \| < \varepsilon,\ \forall t \geq 0 \tag{1.17}$$

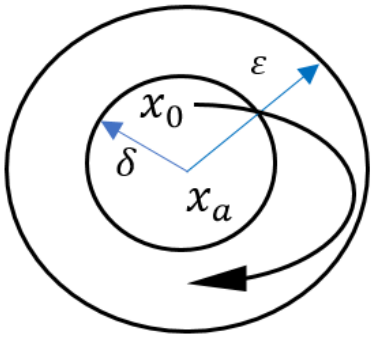


Figura 1.25. Estabilidad del punto $\boldsymbol{x_a}$

- *Sistema inestable*, un punto de equilibrio x_a es inestable, cuando el tiempo incrementa, sale del conjunto solución $S(t, t_0, x_0)$ como se observa en la Figura 1.26

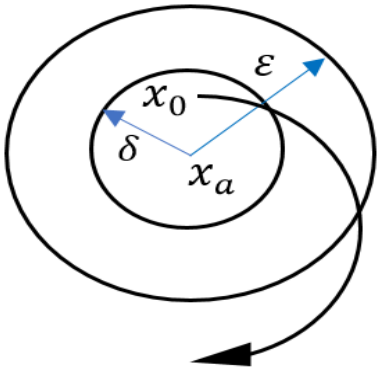


Figura 1.26. Inestabilidad del punto $\boldsymbol{x_a}$

- *Sistema asintóticamente estable*: un punto de equilibrio x_a es asintóticamente estable, cuando el tiempo se incrementa, y se encierra dentro del conjunto solución δ.

$$\delta(t_0) > {}^{0}/_{x_0} \| < \delta(t_0) \rightarrow |S(t, t_0, x_a)| \rightarrow 0;\ \ t \rightarrow \infty \tag{1.18}$$

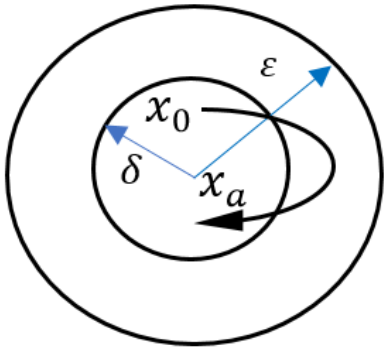


Figura 1.27. Estabilidad asintóticamente estable del punto x_a

1.5.1 CONDICIONES DE ESTABILIDAD DE LYAPUNOV

Se define a un sistema dinámico, autónomo y un sistema lineal o no lineal variante en el tiempo, tal que:

$$\dot{x} = f(x,t) \tag{1.19}$$

Para demostrar la estabilidad del sistema se busca una función de Lyapunov, la cual debe cumplir que sea definida positiva, es definida positiva cuando cumple las siguientes condiciones:

$$V(x,t) > 0\,;t \geq t_0 \tag{1.20}$$

$$V(0,t) = 0\,;t \geq t_0 \tag{1.21}$$

La estabilidad del sistema se examinará de acuerdo con la derivada de la función de Lyapunov $\dot{V}(x)$ [12]:

- Si $\dot{V}(x) < 0$, la energía disminuye y llega a un valor pequeño, en donde el punto de equilibrio es estable, y si llega a cero es asintóticamente estable.
- Si $\dot{V}(x) > 0$, la energía presenta un incremento indefinido por lo que el sistema es inestable[12].
- Si $\dot{V}(x) = 0$, la energía no crece ni disminuye, es decir es estable.

1.6 ÍNDICE DE DESEMPEÑO ISE

Para analizar el desempeño del controlador se evaluará utilizando las características del ISE (Integral del cuadrado del error). Este criterio penaliza

grades errores y discrimina en respuestas excesivamente amortiguadas y sub amortiguadas.

La definición del ISE viene dada por:

$$ISE_{ejex} = \int (e_x)^2 (t) dt \tag{1.22}$$

$$ISE_{ejey} = \int (e_y)^2 (t) dt \tag{1.23}$$

1.7 MÉTODO DE MEDICIÓN DE DESEMPEÑO

1.7.1 DESVIACIÓN ESTÁNDAR

La desviación estándar que es un índice numérico de dispersión de un conjunto de datos.

$$\sigma = \sqrt{\frac{\sum_{i=1}^{N}(x_i - \bar{x})}{N}} \tag{1.24}$$

Donde:

σ: Desviación estándar.

x_i: Valor del conjunto de datos adquiridos para encontrar su desviación estándar

$\bar{x}$: Media aritmética de todos los datos, descrita por la ecuación

$$\bar{x} = \frac{1}{N} \sum_{i=1}^{N} x_i \tag{1.25}$$

N: Número de datos, por punto muestreado.

La distribución normal se encuentra representada por la campana de Gauss, como se observa en la Figura 1.8, cuando los valores de la distribución normal se encuentran dentro de los límites establecidos se dice que opera con eficacia y se pueden realizar trabajos de alta exigencia con respecto a la variable controlada.

Cuando supera los límites establecidos se dice que se está trabajando con ciertas limitaciones debidas a los recursos empleados, es poco eficaz y el tipo de trabajo debe ser poco exigido con respecto a la variable controlada.

Cuando no cumple con las características de seguir una distribución normal, se recomienda realizar un nuevo análisis de datos, no se puede realizar ningún trabajo sobre la variable controlada.

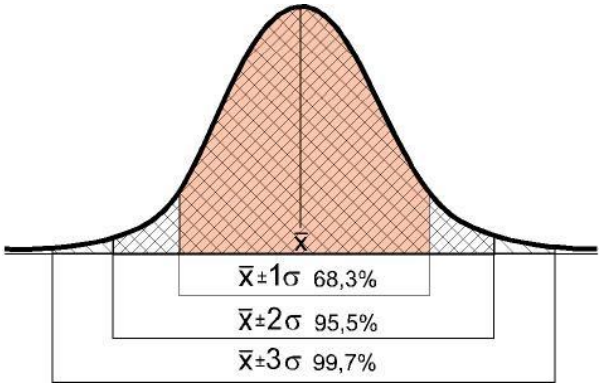


Figura 1.28. Curva de distribución normal [13]

1.8 CINEMÁTICA DEL ROBOT [1]

La resolución del problema cinemático directo permite conocer cuál es la posición y orientación que adopta el extremo del robot cuando cada una de las variables toma valores determinados que fijen su orientación y posición. [1]

La obtención del modelo cinemático directo puede ser abordado por dos enfoques diferentes denominados métodos geométricos y métodos basados en sistemas de referencia.

1.9.1 RESOLUCIÓN DEL MODELO CINEMÁTICO POR EL MÉTODO GEOMÉTRICO [1]

Este método es adecuado para robots que tengan pocos grados de libertad, este método consiste en encontrar las relaciones que permiten conocer la localización espacial del extremo del robot a partir de los valores de sus coordenadas articulares. Por ejemplo, robots de dos grados de libertad como se observa en la Figura 1.29.

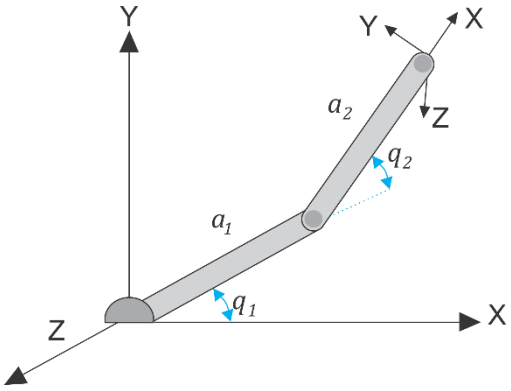


Figura 1.29. Robots de 2 grados de libertad [1]

Por geometría se obtienen las posiciones finales del extremo del robot expresadas en las ecuaciones (1.26), (1.27), (1.28).

$$x = l_1 \cos q_1 + l_2 \cos(q_1 + q_2) \tag{1.26}$$

$$y = l_1 \operatorname{sen} q_1 + l_2 \operatorname{sen}(q_1 + q_2) \tag{1.27}$$

$$z = 0 \tag{1.28}$$

1.9.2 RESOLUCIÓN DEL MODELO CINEMÁTICO POR MATRICES DE TRANSFORMACIÓN HOMOGENEA [1]

Los robots manipuladores se consideran un conjunto de n articulaciones y n eslabones, donde el conjunto de estos dos términos le otorgan un grado de libertad al robot. Para determinar la ubicación de un elemento i con respecto al elemento $i-1$, se puede asociar a cada uno un sistema de referencia; que en conjunto con las transformaciones homogéneas, representan en forma sencilla, rotaciones y traslaciones relativas entre elementos, para ubicar el elemento final del robot en la posición deseada.

Las matrices de transformación homogéneas contienen las coordenadas de traslación y rotación a los eslabones consecutivos de un robot y se denotan como ${}^{i-1}A_i$ y cuando la misma considera todas las articulaciones del robot se la representa como T. Por ejemplo, ${}^{0}A_1$ denota la orientación y posición del sistema 1 (primer eslabón móvil) con respecto al sistema ubicado en la base fija del robot o sistema 0, la disposición de estos elementos se encuentra en la ecuación (1.8),

donde se observa la posición del i-ésimo eslabón y las coordenadas de rotación de los ángulos en torno a los ejes X, Y y Z.

1.9.3 RESOLUCIÓN DEL MODELO CINEMÁTICO APLICANDO EL ALGORITMO DE DENAVITH HARTENBERG [1]

El algoritmo de Denavith Hartenberg se emplea para definir los sistemas de coordenadas para cada eslabón en la cadena cinemática, con los siguientes parámetros:

- Rotación alrededor del eje $\boldsymbol{z}_{i-1}$ un ángulo $\boldsymbol{\theta}_i$.
- Traslación a lo largo del eje $\boldsymbol{z}_{i-1}$ una distancia $\boldsymbol{d}_i$, vector $\boldsymbol{d}_i(0,0,d_i)$.
- Traslación a lo largo del eje $\boldsymbol{x}_i$ una distancia $\boldsymbol{a}_i$, vector $\boldsymbol{a}_i(a_i,0,0)$.
- Rotación alrededor del eje $\boldsymbol{x}_i$ un ángulo $\boldsymbol{\alpha}_i$.

Dado que el producto de matrices no es conmutativo, las transformaciones se han de realizar en el orden indicado, de este modo se tiene que:

$$^{i-1}A_i = \boldsymbol{Rotz}(\theta_i)\,\boldsymbol{T}(0,0,d_i)\boldsymbol{T}(a_i,0,0)\boldsymbol{Rotx}(\alpha_i) \tag{1.29}$$

y realizando el producto entre matrices se obtiene que:

$$^{i-1}A_i = \begin{bmatrix} cos(\theta_i) & -sen(\theta_i) & 0 & 0 \\ sen(\theta_i) & cos(\theta_i) & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & d_i \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & a_i \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & cos(\alpha_i) & -sen(\alpha_i) & 0 \\ 0 & sen(\alpha_i) & cos(\alpha_i) & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{1.30}$$

$$^{i-1}A_i = \begin{bmatrix} cos(\theta_i) & -cos(\alpha_i)sen(\theta_i) & sen\alpha_i sen(\theta_i) & a_i\, cos(\theta_i) \\ sen(\theta_i) & cos(\alpha_i)cos(\theta_i) & -sen\alpha_i cos(\theta_i) & a_i sen(\theta_i) \\ 0 & sen(\alpha_i) & cos(\alpha_i) & d_i \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{1.31}$$

Donde $\theta_i, d_i, a_i, \alpha_i$ son los parámetros de Denavith Hartenberg del eslabón i. A partir de la ecuación (1.31) y de la obtención de los parámetros de Denavith Hartenberg se pueden obtener las matrices $^{i-1}A_i$ para relacionar todos los eslabones del robot. Para asegurar las matrices es necesario que los sistemas de cada eslabón se hayan escogido siguiendo las siguientes normas:

DH1. Numerar los eslabones comenzando con el número 1 para el primer eslabón móvil y acabando con n para el último eslabón móvil. Se enumerará como eslabón 0 a la base del robot.

DH2. Numerar cada articulación comenzando con el número 1 para el primer grado de libertad y acabando en n.

DH3. Localizar el eje de cada articulación. Si es rotativa corresponderá a su propio eje de giro mientras que de ser prismática corresponderá al eje a lo largo del cual se produce la traslación.

DH4. Para i de 0 a $n-1$ situar el eje z_i sobre el eje de la articulación $i+1$.

DH5. Situar el origen del sistema de la base $\{S_0\}$ en cualquier punto del eje z_0. Los ejes x_0 e y_0 se situarán para cumplir con un sistema dextrógiro (se mueve en sentido de las agujas del reloj).

DH6. Para i de 1 a $n-1$, situar el origen del sistema $\{S_i\}$en la intersección del eje z_i con la línea normal común a z_{i-1} y z_i. Si ambos ejes se cortasen se situaría $\{S_i\}$ en el punto de corte. Si fuesen paralelos $\{S_i\}$ se situaría en la articulación $i+1$.

DH7. Situar x_i en la línea normal común a z_{i-1} y z_{i-1}.

DH8. Situar y_i de modo que forme un sistema dextrógiro con x_i y z_i.

DH9. Situar el sistema $\{S_n\}$ en el extremo del robot de modo que z_n coincida con la dirección z_{n-1} y x_n sea normal a z_{n-1} y z_n.

DH10. Obtener θ_i como el ángulo que haya de girar en torno a z_{i-1} para que x_{i-1} y x_i queden paralelos.

DH11. Obtener d_i como la distancia, medida a lo largo de z_{i-1}, que habría que desplazar $\{S_{i-1}\}$ para que x_i y x_{i-1} queden alineados.

DH12. Obtener a_i como la distancia, medida a lo largo de x_i, (que ahora coincidiría con x_{i-1}) que habría de desplazar el nuevo $\{S_{i-1}\}$ para que su origen coincidiese con $\{S_i\}$.

DH13. Obtener α_i como el ángulo que haya de girar en torno a x_i, para que el nuevo $\{S_{i-1}\}$ coincidiese totalmente con $\{S_i\}$.

DH14. Hallar las matrices de transformación ${}^{i-1}\boldsymbol{A_i}$ definidas en la ecuación (1.31).

DH15. Hallar la matriz de transformación que relaciona el sistema de la base fija del robot con el efector final del mismo de acuerdo a la ecuación (1.32).

$$\boldsymbol{T} = {}^{0}\boldsymbol{A}_1\,{}^{1}\boldsymbol{A}_2\,{}^{n-1}\boldsymbol{A}_n \tag{1.32}$$

DH16. La matriz $\boldsymbol{T}$ presentada en la ecuación (1.32) define la orientación y posicionamiento del efector final referido a la base fija del robot, en función de todas las coordenadas angulares.

1.9 MODELO DIFERENCIAL. MATRIZ JACOBIANA [1]

El modelo cinemático del robot busca las relaciones entre las velocidades de las coordenadas articulares como también de las posición y orientación del extremo operativo del robot.

1.9.1 MATRIZ JACOBIANA ANALÍTICA

La Jacobiana Analítica relaciona las velocidades articulares con las velocidades de localización del extremo operativo del robot.

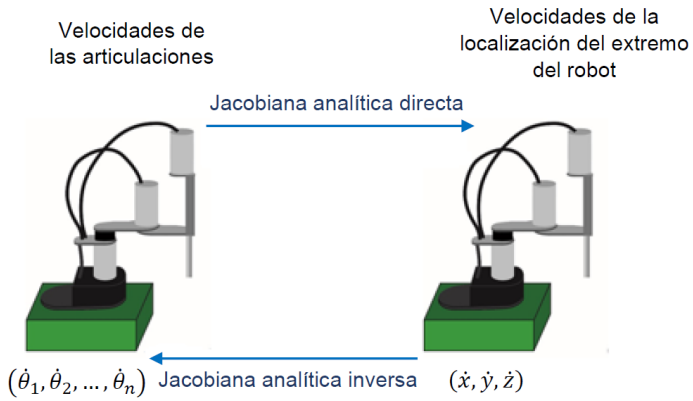


Figura 1.30. Relación entre la Jacobiana Analítica e Inversa.

Cada uno de los elementos pertenecientes a la Jacobiana analítica depende de los valores de las coordenadas articulares en donde, el valor de la Jacobiana cambiará de acuerdo con los puntos del espacio articular.

$$\begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{z} \\ \dot{\emptyset} \\ \dot{\theta} \\ \dot{\psi} \end{bmatrix} = J_a \begin{bmatrix} \dot{q_1} \\ \vdots \\ \vdots \\ \dot{q_n} \end{bmatrix} \; con \; J_a = \begin{bmatrix} \frac{\partial x_e}{\partial q_1} & \cdots & \frac{\partial x_e}{\partial q_n} \\ \vdots & \ddots & \vdots \\ \frac{\partial z_e}{\partial q_1} & \cdots & \frac{\partial z_e}{\partial q_n} \end{bmatrix} \tag{1.33}$$

Donde

x_e: Posición del efector final en el eje x .

y_e: Posición del efector final en el eje y.

z_e: Posición del efector final en el eje z.

q_n: Vector de posición angular de las articulaciones

1.9.2 MATRIZ JACOBIANA GEOMÉTRICA [1]

La Jacobiana analítica presenta la relación de la velocidad de las articulaciones con la velocidad de la variación de posición y orientación del extremo final del robot.

Otra posible relación de interés es la que se establece entre las velocidades articulares con las velocidades lineales v, y angulares w del extremo del robot con referencia a la base del robot $\{S_0\}$, obtenidas de manera directa a partir de la matriz de transformación ${}^{i-1}\boldsymbol{A}_i$, que define el modelo cinemático del robot.

La matriz Jacobiana geométrica que relaciona las velocidades articulares con las velocidades de traslación y rotación del extremo del robot está representada por la matriz (1.34)

$$J = \begin{cases} \begin{bmatrix} {}^0\boldsymbol{z}_{i-1} \times {}^{i-1}\boldsymbol{p}_n \\ {}^0\boldsymbol{z}_{i-1} \end{bmatrix}, \text{si el eslabón i es de rotación} & (1.34) \\ \\ \begin{bmatrix} {}^0\boldsymbol{z}_{i-1} \\ 0 \end{bmatrix}, \text{si el eslabón i es de traslación} & (1.35) \end{cases}$$

Se denomina ${}^0\boldsymbol{z}_i$ al vector unitario orientado según el eje de la articulación $i+1$, definido en el sistema de coordenadas de la base del robot $\{S_0\}$ (como se definió en las reglas DH3 Y DH4 del algoritmo de Denavith Hartenberg).

Las matrices ${}^{i-1}\boldsymbol{A}_i$ contienen la información de los vectores directores y origen del sistema $\{\boldsymbol{S}_i\}$ en la base $\{\boldsymbol{S}_{i-1}\}$, como se observa en la ecuación (1.36)

$$ {}^{i-1}\boldsymbol{A}_i = \begin{bmatrix} {}^{i-1}\boldsymbol{n}_i & {}^{i-1}\boldsymbol{o}_i & {}^{i-1}\boldsymbol{a}_i & {}^{i-1}\boldsymbol{p}_i \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{1.36} $$

Por tanto, la matriz ${}^0\boldsymbol{A}_i = \begin{bmatrix} {}^0\boldsymbol{n}_i & {}^0\boldsymbol{o}_i & {}^0\boldsymbol{a}_i & {}^0\boldsymbol{p}_i \\ 0 & 0 & 0 & 1 \end{bmatrix}$ contendrá la información de los vectores directores y origen del sistema $\{\boldsymbol{S}_i\}$ en la base $\{S_0\}$. De modo que ${}^0\boldsymbol{z}_i$ estará definido por los tres primeros elementos de la tercera columna de ${}^0\boldsymbol{A}_i$. (Al ser ${}^0\boldsymbol{A}_0$ la matriz identidad y ${}^0\boldsymbol{z}_0$ será el vector (0, 0,1)). Por ejemplo, la ecuación (1.37)

$$ {}^0\boldsymbol{z}_i = {}^0\boldsymbol{A}_i(1{:}3,3) \tag{1.37} $$

Donde, la notación $(i{:}j,k)$ indica los elementos i a j de la columna k.

Se denominará ${}^i\boldsymbol{p}_n$ al vector que va desde el origen del sistema $\{S_i\}$ hasta el extremo del robot (origen del sistema $\{S_n\}$) expresado en el sistema de la base del robot $\{S_0\}$. Puesto que la cuarta columna de la matriz de transformación ${}^0\boldsymbol{A}_n$ contiene las coordenadas del extremo del robot en el sistema $\{S_0\}$ y la cuarta columna del ${}^0\boldsymbol{A}_i$ contiene las coordenadas del origen del sistema $\{S_i\}$ en el sistema $\{S_0\}$, ${}^i\boldsymbol{p}_n$ se obtendrá restando las cuartas columnas de ${}^0\boldsymbol{A}_n$ y ${}^0\boldsymbol{A}_i$, como se observa en la ecuación (1.38):

$$ {}^i\boldsymbol{p}_n = {}^0\boldsymbol{A}_n(1{:}3,4) - {}^0\boldsymbol{A}_i(1{:}3,4) \tag{1.38} $$

Definidos los vectores ${}^0\boldsymbol{z}_i$ y ${}^i\boldsymbol{p}_n$, se obtienen los términos para definir la matriz Jacobiana geométrica.

1.10 CONTROL SERVO VISUAL DE MANIPULADORES

El control servo visual de robots manipuladores tiene como funciones principales la captura de imágenes visuales del entorno de trabajo, y obtener información de este

espacio sin tener un modelo del mismo. A estos entornos se los conoce como entornos no estructurados o parcialmente estructurados.

El control servo visual presenta ventaja con respecto a un control visual en lazo abierto: principalmente, ser independiente de la precisión de los sensores de visión, así como de la precisión del robot. Sin embargo, en lazo cerrado aumenta las posibilidades de inestabilidad complicando su desarrollo [11].

La información se obtiene a partir del procesamiento de imágenes obtenidas de una o dos cámaras y probablemente complementada con información de profundidad.

En cuanto al montaje de la cámara en el robot, se la puede ubicar en el extremo operativo del robot (cámara en mano), o bien fija observando el espacio de trabajo.

1.10.1 CÁMARA FIJA

Captura imágenes del robot y de su espacio de trabajo, es decir, los marcos de referencia robot y cámara es constante, ofrece un amplio campo de visibilidad para facilitar la posición de su efector final, como se observa en la Figura 1.31.

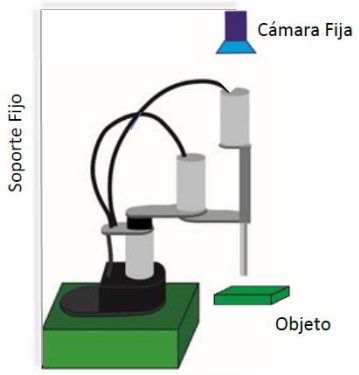


Figura 1.31. Robot con cámara fija.

La mayoría de sistemas robóticos que utilizan realimentación visual por así decirlo, trabajan con una cámara fija externa al sistema robot – objeto, para obtener la captura de la imagen del robot y del espacio de trabajo.

Su principal desventaja es la obstrucción ya que el robot puede interponerse entre la cámara y el objeto provocando errores en la tarea asignada.

1.10.2 CÁMARA EN MANO

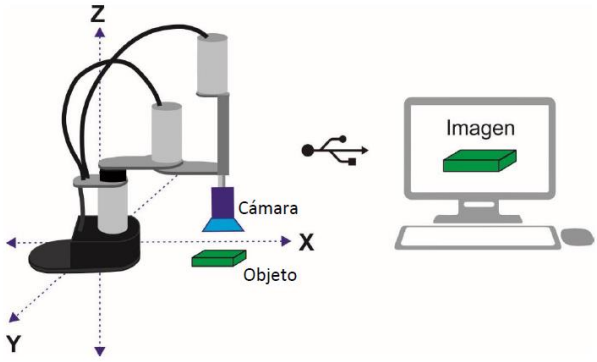


Figura 1.32. Robot con cámara en mano

Al usar una cámara en mano se evita la obstrucción del objeto de interés, como se observa en la Figura 1.32, por parte del robot manipulador; tiene información directa del espacio de trabajo y una mayor resolución visual facilitando la orientación del efector final, este tipo de realimentación visual obtiene las características de la imagen mediante las posiciones relativas entre la cámara y el objeto. La imagen obtenida sobre el objeto a manipular depende del movimiento del robot.

Es algunas aplicaciones puede tener las dos configuraciones como ayuda para realizar su tarea, la cámara móvil puede estar colocada sobre otro manipulador observando al primero.

Las desventajas de esta configuración es la calibración en línea de la cámara el cálculo de la posición articular del robot en su proyección frente a la cámara por estar en un movimiento al extremo del robot.

1.10.3 EXTRACCIÓN DE CARACTERÍSTICAS DE LA IMAGEN

La información que se extrae de las imágenes es conocida como parámetros de imagen (perímetro del contorno, longitud de la línea, entre otros), el mismo que es un valor real de la característica de imagen (contorno, una línea, un punto, entre otros), se entiende por característica de imagen a la relación cuantificable en la imagen, siendo resultado de la proyección en el plano de imagen de las características físicas de un objeto, como se observa en la Figura 1.33.

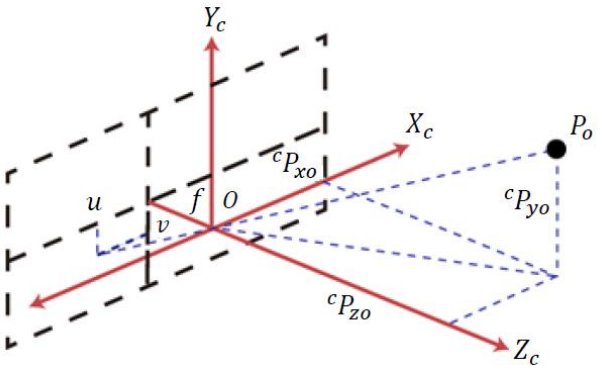


Figura 1.33. Modelo de proyección de perspectiva.

Donde:

u, v: Características de la imagen.

P_o: Posición del objeto.

${}^cP_{xo}, {}^cP_{yo}, {}^cP_{zo}$: Posición del objeto visto desde la cámara en coordenadas x, y, z

X_c, Y_c, Z_c: Ejes coordenados de la cámara.

El objetivo de trabajar con las características de la imagen es reducir información de la salida de la cámara de video a una cantidad de datos manipulables y representativos, es decir, extraer la esencia de la escena.

La selección del mejor conjunto de características para obtener un comportamiento óptimo del sistema de control, esto se cumple cuando:

- El error es pequeño entre valores iniciales y deseados de las características
- Se restrinja todos los grados de libertad del robot en donde la matriz Jacobiana sea $2p > n$, se minimice la no linealidad de la matriz jacobiana.

1.11 ROBOT MANIPULADOR TIPO SCARA

En el presente proyecto se utilizó un robot manipulador tipo SCARA de tres grados de libertad que se encuentra en el Laboratorio de Control y Sistemas. Este robot manipulador presenta una estructura mecánica en configuración RRP, cada articulación tiene un actuador eléctrico (motores DC con caja reductora incluida) debido a su sencillez, precisión y una relativa facilidad de control, como se describe en (Capítulo 1, sección 1.3.1). También presenta sensores de posición y presencia,

para dar información de la posición del efector final y de sus respectivas articulaciones.

1.11.1 CONDICIÓN ACTUAL

En esta sección se menciona el estado del hardware del robot, la parte electrónica para su debido control como son placa de potencia, placa de control, motores de cada una de las articulaciones, encoders y finales de carrera.

Análisis del hardware

a) *Placa de control.* Estaba constituida por una placa de terminales provenientes de los actuadores y sensores del robot SCARA que eran conectados a la PC por una tarjeta de adquisición de datos (PCI 6221-M).
b) *Placa de potencia.* El circuito de potencia constituido en la placa contiene drivers (L298N) para que este circuito integrado trabaje como un conversor DC-DC de 4 cuadrantes en donde la corriente y voltaje al motor puedan invertir su dirección y garanticen el cambio de sentido de giro de cada motor DC del robot SCARA.
c) *Fuente de alimentación.* El robot SCARA necesita de tres voltajes de polarización y por ende se usan tres fuentes de alimentación.

Al hacer un análisis de todo el hardware se encontró las siguientes novedades:

Tabla 1.1. Características Hardware robot SCARA.

ELEMENTO DEL HARDWARE	ESTADO/FUNCIONALIDAD
Placa de control	Se encuentra en buen estado, para propósitos y requisitos de software no se puede utilizar.
Placa de potencia	Placa en buen estado, funcional.
Motores DC	Motores en buen estado y funcionales.
Encoders	En buen estado y funcionales.
Finales de carrera	En buen estado y funcionales.
Estructura mecánica	Funcional.

1.12 COMPONENTES ELECTRÓNICOS

1.12.1 MICROCONTROLADOR ATMEGA 164P

La Figura 1.34 muestra la distribución de pines que presente este microcontrolador.

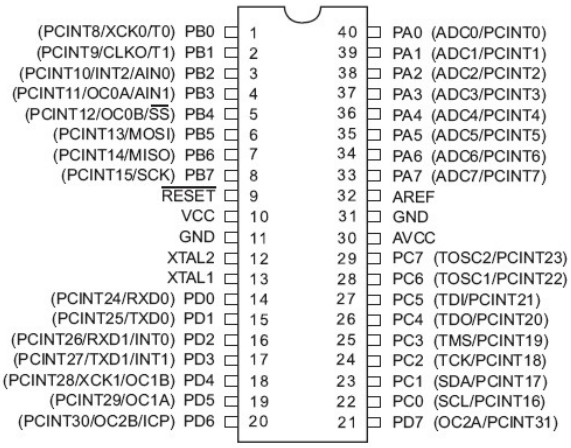


Figura 1.34. Distribución de pines del micro controlador Atmega 164P.

La Tabla 1.2 muestra las principales características de este micro controlador.

Tabla 1.2. Características del micro controlador Atmega 164P.

CARACTERÍSTICA	Valor	Unidades
Consumo de Corriente	4.8	mA
Voltaje de Alimentación	2.7-5.5	V
Frecuencia de operación	8	MHz
PERIFÉRICOS	**Número de pines**	
Conversor Análogo Digital	8	
Puerto de comunicación USART	2	
Pines de Entrada y Salida	32	
Salidas PWM	6	

CAPÍTULO 2
DESARROLLO DEL ALGORITMO DE CONTROL Y PROCESAMIENTO DE IMÁGENES

En este capítulo se estudiará la cinemática del robot manipulador tipo SCARA donde se implementará el control servo visual basado en las características de la imagen, cuya prioridad será posicionar y orientar el efector final hacia el objeto deseado. A continuación, se implementará la adquisición y procesamiento de imágenes. Finalmente, se diseñará el circuito de control.

2.1 MODELO CINEMÁTICO DEL ROBOT MANIPULADOR TIPO SCARA

Se entiende por cinemática el estudio del movimiento sin considerar las fuerzas que lo producen. Por lo tanto, se trata de estudiar tanto las propiedades geométricas como las temporales del movimiento [1].

Para este caso de estudio utilizaremos un manipulador de tres grados de libertad como se menciona en el (Capítulo 1, sección 1.3.1) y se observa en la Figura 2.1.

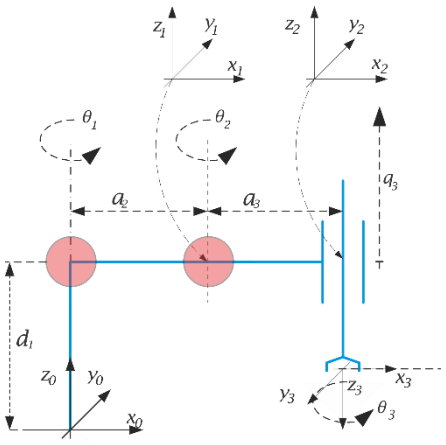


Figura 2.1. Robot SCARA de tres grados de libertad

Donde:

d_1: Distancia de la base superior del robot al primer eslabón (eje hombro).

a_2: Longitud del ante brazo.

a_3: Longitud del brazo.

θ_1: Ángulo de rotación de a_2.

θ_2: Ángulo de rotación de a_3.

q_3: Desplazamiento del eje Z.

x_e, y_e, z_e: Posiciones finales del efector final (pinza) con referencia al origen de coordenadas.

Las relaciones geométricas del manipulador son:

$$x_e = a_2 \cos\theta_1 + a_3 \cos(\theta_1 + \theta_2) \tag{2.1}$$

$$y_e = a_2 \operatorname{sen}\theta_1 + a_3 \operatorname{sen}(\theta_1 + \theta_2) \tag{2.2}$$

$$z_e = q_3 + d_1 \tag{2.3}$$

La cinemática del manipulador $\dot{h}$ está dada por:

$$\dot{h} = J_h \, \dot{q}_e \tag{2.4}$$

Se define a la matriz $h = [\, x_e \;\; y_e \,]^T$ y $q_e = [\, \theta_1 \;\; \theta_2 \,]^T$, entonces se tiene que: $\dot{h} = [\, \dot{x}_e \;\; \dot{y}_e \,]^T$; $\dot{q_e} = [\, \dot{\theta}_1 \;\; \dot{\theta}_2 \,]^T$ el jacobiano del manipulador h , definido de la siguiente manera:

$$J_h = \begin{bmatrix} \frac{\partial x_e}{\partial \theta_1} & \frac{\partial x_e}{\partial \theta_2} & \frac{\partial x_e}{\partial \theta_3} \\ \frac{\partial y_e}{\partial \theta_1} & \frac{\partial y_e}{\partial \theta_2} & \frac{\partial y_e}{\partial \theta_3} \\ \frac{\partial z_e}{\partial \theta_1} & \frac{\partial z_e}{\partial \theta_2} & \frac{\partial z_e}{\partial \theta_3} \end{bmatrix} \tag{2.5}$$

Si se deriva parcialmente la ecuación (2.1) se obtiene:

$$\frac{\partial x_e}{\partial \theta_1} = -a_2 \sin\theta_1 - a_3 \sin\theta_1 \cos\theta_2 - a_3 \cos\theta_1 \sin\theta_2 \tag{2.6}$$

$$\frac{\partial x_e}{\partial \theta_2} = a_2 \cos\theta_1 + a_3 \cos\theta_1 \cos\theta_2 - a_3 \sin\theta_1 \sin\theta_2 \quad (2.7)$$

Se toma (2.2) y aplicando derivada parcial:

$$\frac{\partial y_e}{\partial \theta_1} = -a_3 \cos\theta_1 \sin\theta_2 - a_3 \sin\theta_1 \cos\theta_2 \quad (2.8)$$

$$\frac{\partial y_e}{\partial \theta_2} = -a_3 \sin\theta_1 \sin\theta_2 + a_3 \cos\theta_1 \cos\theta_2 \quad (2.9)$$

Partiendo de (2.3) y aplicando derivada parcial,

$$\frac{\partial z_e}{\partial \theta_1} = 0 \quad (2.10)$$

$$\frac{\partial z_e}{\partial \theta_2} = 0 \quad (2.11)$$

El Jacobiano total del manipulador h es:

$$J_h = \begin{bmatrix} J_{11} & J_{12} \\ J_{21} & J_{22} \\ J_{31} & J_{32} \end{bmatrix} \quad (2.12)$$

Donde:

$$J_{11} = -a_2 \sin\theta_1 - a_3 \sin\theta_1 \cos\theta_2 - a_3 \cos\theta_1 \sin\theta_2 \quad (2.13)$$

$$J_{12} = -a_3 \cos\theta_1 \sin\theta_2 - a_3 \sin\theta_1 \cos\theta_2 \quad (2.14)$$

$$J_{21} = a_2 \cos\theta_1 + a_3 \cos\theta_1 \cos\theta_2 - a_3 \sin\theta_1 \sin\theta_2 \quad (2.15)$$

$$J_{22} = -a_3 \sin\theta_1 \sin\theta_2 + a_3 \cos\theta_1 \cos\theta_2 \quad (2.16)$$

$$J_{31} = 0 \quad (2.17)$$

$$J_{32} = 0 \quad (2.18)$$

De tal modo que (2.4) queda expresada como se muestra en la ecuación (2.19)

$$\begin{bmatrix} \dot{x}_e \\ \dot{y}_e \end{bmatrix} = \begin{bmatrix} J_{11} & J_{12} \\ J_{21} & J_{22} \\ J_{31} & J_{32} \end{bmatrix} \begin{bmatrix} \dot{\theta}_1 \\ \dot{\theta}_2 \end{bmatrix} \quad (2.19)$$

Siendo $\begin{bmatrix} \dot{\theta_1} & \dot{\theta_2} \end{bmatrix}^T$, las entradas del manipulador SCARA.

2.2 DISEÑO DEL CONTROLADOR SERVO VISUAL

El controlador servo visual tendrá la estructura como se observa en la Figura 2.2, en donde el modelo del robot definido en el (Capítulo 2, sección 1) en unión con el modelo geométrico de la cámara forman parte del modelo del robot.

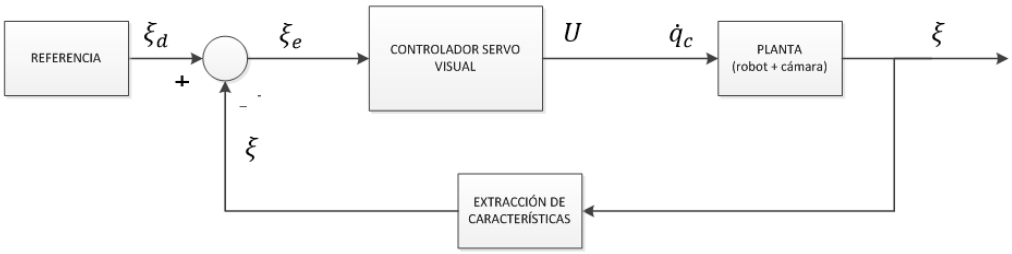


Figura 2.2. Esquema del controlador servo visual

ξ_d: Vector de características de imagen (u_d, v_d) deseadas.

ξ_e: Vector del error de características de imagen (u_e, v_e).

U: Ley de control.

$\dot{q}_c$: Vector de posiciones angulares del controlador θ_1 y θ_2

ξ: Vector de características de imagen (u, v)

2.2.1 MODELO GEOMÉTRICO DE LA CÁMARA

La Figura 2.3 muestra el diagrama de bloques del modelo geométrico de la cámara.

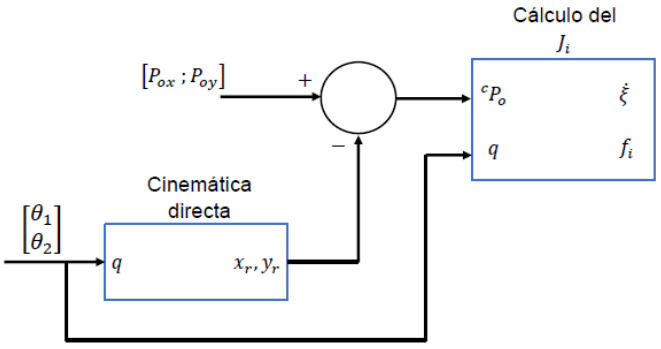


Figura 2.3. Diagrama de bloques de la Cámara.

Donde:

P_{ox}: Posición del objeto en x.

P_{oy}: Posición del objeto en y.

x_r: Posición del robot en x.

y_r: Posición del robot en y.

q: Posición articular θ_1 y θ_2

${}^cP_o$: Posición del objeto (x, y, z) con respecto a la cámara.

$\dot{\xi}$: derivada del vector de características de imagen (u, v)

f_i: Vector de características de imagen (u, v).

J_i: Jacobiano de la imagen.

Para encontrar el modelo de la cámara, se detalla las siguientes ecuaciones:

$$\xi = \begin{bmatrix} u \\ v \end{bmatrix} = -\frac{\alpha\lambda}{{}^cP_{zo}} \begin{bmatrix} {}^cP_{xo} \\ {}^cP_{yo} \end{bmatrix} = i({}^wP_c, {}^wP_o) \tag{2.20}$$

$$\dot{\xi} = \frac{\partial i}{\partial {}^wP_c} {}^w\dot{P}_c + \frac{\partial i}{\partial {}^wP_c} {}^w\dot{P}_o \tag{2.21}$$

Donde:

${}^wP_o$: Posición del objeto en el mundo.

${}^{w}P_{c}$: Postura de la cámara en el mundo.

α: Factor de escala igual a $72727\ (pixeles/m)$.

λ: Longitud focal igual a $0.016\ m$.

${}^{c}R_{w}$: Matriz de transformación del sistema de coordenadas del mundo a la cámara.

Para un objeto fijo ${}^{w}\dot{P}_{o} = 0$, se tiene:

$$\dot{\xi} = \frac{\partial i}{\partial {}^{w}P_{c}}\,{}^{w}\dot{P}_{c} \tag{2.22}$$

$$\dot{\xi} = -\alpha\lambda \begin{bmatrix} \frac{\partial {}^{c}P_{xo}\,{}^{c}P_{zo}{}^{-1}}{\partial {}^{c}P_{xo}} & \frac{\partial {}^{c}P_{xo}\,{}^{c}P_{zo}{}^{-1}}{\partial C_{P_{yo}}} & \frac{\partial {}^{c}P_{xo}\,{}^{c}P_{zo}{}^{-1}}{\partial {}^{c}P_{zo}} \\ \frac{\partial {}^{c}P_{yo}\,{}^{c}P_{zo}{}^{-1}}{\partial {}^{c}P_{xo}} & \frac{\partial {}^{c}P_{yo}\,{}^{c}P_{zo}{}^{-1}}{\partial C_{P_{yo}}} & \frac{\partial {}^{c}P_{yo}\,{}^{c}P_{zo}{}^{-1}}{\partial {}^{c}P_{zo}} \end{bmatrix} \begin{bmatrix} {}^{c}\dot{P}_{xo} \\ {}^{c}\dot{P}_{yo} \\ {}^{c}\dot{P}_{zo} \end{bmatrix} \tag{2.23}$$

$$\dot{\xi} = -\frac{\alpha\lambda}{{}^{c}P_{zo}} \begin{bmatrix} 1 & 0 & -\frac{{}^{c}P_{xo}}{C_{P_{zo}}} \\ 0 & 1 & -\frac{{}^{c}P_{yo}}{{}^{c}P_{zo}} \end{bmatrix} \begin{bmatrix} {}^{c}\dot{P}_{xo} \\ {}^{c}\dot{P}_{yo} \\ {}^{c}\dot{P}_{zo} \end{bmatrix} \tag{2.24}$$

Para el movimiento de un punto de un marco móvil respecto de uno fijo:

$$\frac{\partial}{\partial t} = \left({}^{c}P_{xo}, {}^{c}P_{yo}, {}^{c}P_{zo}\right)^{T} = {}^{c}R_{w}(q)\left\{-{}^{w}_{c}\omega \times \left({}^{w}P_{o} - {}^{w}P_{corg}(q)\right) + \left({}^{w}\dot{P}_{o} - {}^{w}V_{c}\right)\right\} \tag{2.25}$$

$$\begin{bmatrix} {}^{c}\dot{P}_{xo} \\ {}^{c}\dot{P}_{yo} \\ {}^{c}\dot{P}_{zo} \end{bmatrix} = \begin{bmatrix} -1 & 0 & 0 & 0 & -{}^{c}P_{zo} & {}^{c}P_{yo} \\ 0 & -1 & 0 & {}^{c}P_{zo} & 0 & -{}^{c}P_{yo} \\ 0 & 0 & -1 & -{}^{c}P_{yo} & {}^{c}P_{xo} & 0 \end{bmatrix} \begin{bmatrix} {}^{c}R_{w}(q) & \mathbf{0} \\ \mathbf{0} & {}^{c}R_{w}(q) \end{bmatrix} \begin{bmatrix} {}^{w}V_{c} \\ {}^{w}\omega_{c} \end{bmatrix} + {}^{c}R_{w}(q)\,{}^{w}\dot{P}_{o} \tag{2.26}$$

Como ${}^{w}\dot{P}_{o} = 0$, se tiene:

$$\begin{bmatrix} {}^{c}\dot{P}_{xo} \\ {}^{c}\dot{P}_{yo} \\ {}^{c}\dot{P}_{zo} \end{bmatrix} \tag{2.27}$$

$$= \begin{bmatrix} -1 & 0 & 0 & 0 & -{}^{c}P_{zo} & {}^{c}P_{yo} \\ 0 & -1 & 0 & {}^{c}P_{zo} & 0 & -{}^{c}P_{yo} \\ 0 & 0 & -1 & -{}^{c}P_{yo} & {}^{c}P_{xo} & 0 \end{bmatrix} \begin{bmatrix} {}^{c}R_{w}(q) & \mathbf{0} \\ \mathbf{0} & {}^{c}R_{w}(q) \end{bmatrix} \begin{bmatrix} {}^{w}V_{c} \\ {}^{w}\omega_{c} \end{bmatrix}$$

Reemplazando la ecuación (2.27) en (2.24):

$$\dot{\xi} = \frac{\alpha\lambda}{{}^{c}P_{zo}} \begin{bmatrix} -1 & 0 & \frac{{}^{c}P_{xo}}{{}^{c}P_{zo}} & \frac{{}^{c}P_{xo}\,{}^{c}P_{zo}}{{}^{c}P_{zo}} & -\frac{{}^{c}P_{zo}{}^{2} + {}^{c}P_{xo}{}^{2}}{{}^{c}P_{zo}} & {}^{c}P_{yo} \\ 0 & -1 & \frac{{}^{c}P_{yo}}{{}^{c}P_{zo}} & \frac{{}^{c}P_{zo}{}^{2} + {}^{c}P_{yo}{}^{2}}{{}^{c}P_{zo}} & -\frac{{}^{c}P_{xo}{}^{2} + {}^{c}P_{yo}{}^{2}}{{}^{c}P_{zo}} & -{}^{c}P_{xo} \end{bmatrix} \tag{2.28}$$

$$\begin{bmatrix} {}^{c}R_{w}(q) & \mathbf{0} \\ \mathbf{0} & {}^{c}R_{w}(q) \end{bmatrix} \begin{bmatrix} {}^{w}V_{c} \\ {}^{w}\omega_{c} \end{bmatrix}$$

$$\dot{\xi} = \begin{bmatrix} \frac{\alpha\lambda}{{}^{c}P_{zo}} & 0 & -\frac{u}{{}^{c}P_{zo}} & -\frac{uv}{\alpha\lambda} & \frac{\alpha^2\lambda^2 + u^2}{\alpha\lambda} & -v \\ 0 & \frac{\alpha\lambda}{{}^{c}P_{zo}} & -\frac{v}{{}^{c}P_{zo}} & -\frac{\alpha^2\lambda^2 + v^2}{\alpha\lambda} & \frac{uv}{\alpha\lambda} & u \end{bmatrix} \tag{2.29}$$

$$\begin{bmatrix} {}^{c}R_{w}(q) & \mathbf{0} \\ \mathbf{0} & {}^{c}R_{w}(q) \end{bmatrix} \begin{bmatrix} {}^{w}V_{c} \\ {}^{w}\omega_{c} \end{bmatrix}$$

De esto se puede rescribir a la ecuación J_i como

$$\dot{\xi} = J_i(\xi, {}^{c}P_{zo}) = \begin{bmatrix} {}^{c}R_{w}(q) & \mathbf{0} \\ \mathbf{0} & {}^{c}R_{w}(q) \end{bmatrix} \begin{bmatrix} {}^{w}V_{c} \\ {}^{w}\omega_{c} \end{bmatrix} \tag{2.30}$$

Donde $J_i(\xi, {}^{c}P_{zo})$ es el jacobiano de la imagen:

$$J_i(\xi, {}^{c}P_{zo}) = \begin{bmatrix} \frac{\alpha\lambda}{{}^{c}P_{zo}} & 0 & -\frac{u}{{}^{c}P_{zo}} & -\frac{uv}{\alpha\lambda} & \frac{\alpha^2\lambda^2 + u^2}{\alpha\lambda} & -v \\ 0 & \frac{\alpha\lambda}{{}^{c}P_{zo}} & -\frac{v}{{}^{c}P_{zo}} & -\frac{\alpha^2\lambda^2 + v^2}{\alpha\lambda} & \frac{uv}{\alpha\lambda} & u \end{bmatrix} \tag{2.31}$$

2.2.2 JACOBIANO TOTAL

La Figura 2.4 muestra el diagrama de bloques para la obtención del Jacobiano Total.

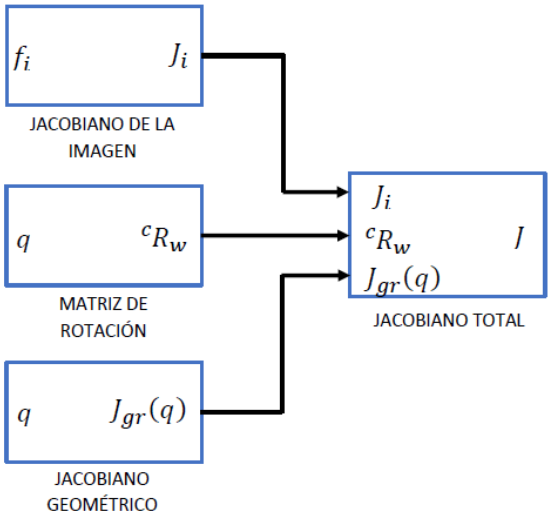


Figura 2.4. Diagrama de bloques del Jacobiano Total

Donde:

q: Posición articular θ_1 y θ_2.

f_i: Vector de características de imagen (u, v).

J_i: Jacobiano de la imagen.

${}^cR_w$: Matriz de transformación del sistema de coordenadas del mundo a la cámara.

$J_{gr}(q)$: Jacobiano geométrico del robot.

J: Jacobiano total.

Para determinar el Jacobiano total J, se considera la posición de la cámara ubicada en el extremo operativo del robot manipulador tipo SCARA. Se parte del Jacobiano geométrico para obtener las velocidades lineales y angulares de la cámara mediante el Jacobiano del robot por las velocidades articulares.

Para obtener la matriz de rotación se calcula mediante la transformación de coordenadas del mundo al de la cámara R_{wc}, como se escribe en la ecuación (2.31) y posteriormente se define la matriz de rotación de la cámara R_z respecto al eje z, como se muestra en la ecuación (2.32).

$$R_{wc} = \begin{bmatrix} \cos(q_1 + q_2) & -\sin(q_1 + q_2) & 0 \\ \sin(q_1 + q_2) & \cos(q_1 + q_2) & 0 \\ 0 & 0 & 1 \end{bmatrix} \tag{2.32}$$

La ecuación (2.32) muestra la rotación de la cámara con respecto al eje z, partiendo de la matriz de rotación mencionada en el (Capítulo 1, sección 1.3.6), quedando definida

$$R_z = \begin{bmatrix} 1 & 0 & 0 \\ 0 & -1 & 0 \\ 0 & 0 & -1 \end{bmatrix} \tag{2.33}$$

Por lo tanto, la ecuación de transformación del sistema de coordenadas del mundo a la cámara queda definida como:

$${}^{c}R_w = R_z R_{wc} \tag{2.34}$$

$${}^{c}R_w = \begin{bmatrix} \cos(q_1 + q_2) & -\sin(q_1 + q_2) & 0 \\ \sin(q_1 + q_2) & \cos(q_1 + q_2) & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 \\ 0 & -1 & 0 \\ 0 & 0 & -1 \end{bmatrix} \tag{2.35}$$

$${}^{c}R_w = \begin{bmatrix} \cos(q_1 + q_2) & \sin(q_1 + q_2) & 0 \\ \sin(q_1 + q_2) & -\cos(q_1 + q_2) & 0 \\ 0 & 0 & -1 \end{bmatrix} \tag{2.36}$$

$$\dot{\xi} = Ji(\xi, {}^{c}P_{zo}) \begin{bmatrix} {}^{c}R_w(q) & \mathbf{0} \\ \mathbf{0} & {}^{c}R_w(q) \end{bmatrix} J_{gr}(q)\dot{q} \tag{2.37}$$

Donde la matriz **0**, conocida como matriz nula es un artificio matemático para completar el número de filas y de columnas requeridas para las operaciones matemáticas de matrices.

Quedando la expresión simplificada como se muestra en la ecuación (2.38):

$$\dot{\xi} = J(q, {}^{W}P_O)\dot{q} \tag{2.38}$$

2.2.3 DISEÑO DEL CONTROLADOR SERVO VISUAL

El controlador servo visual tendrá la estructura de la Figura 2.5. Partiendo del modelo se propone una función de Lyapunov que demuestre la estabilidad del robot manipulador SCARA.

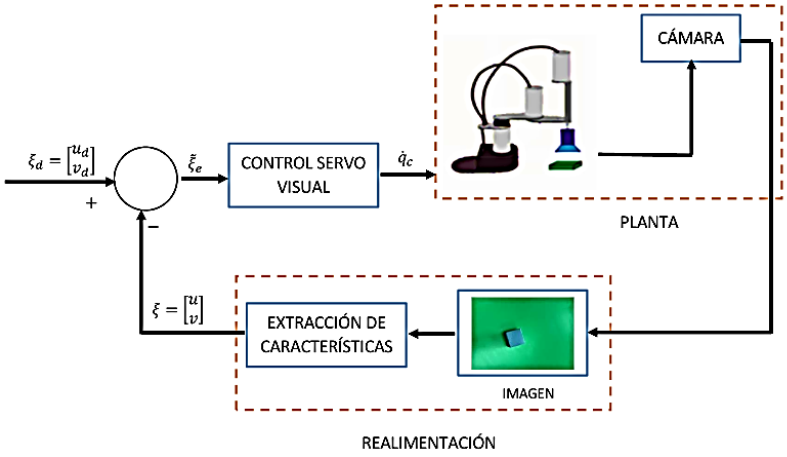


Figura 2.5. Controlador Servo Visual.

El error puede definirse mediante la ecuación (2.39).

$$\xi_e = \xi_d - \xi = \begin{bmatrix} u_d - u \\ v_d - v \end{bmatrix} \tag{2.39}$$

Donde ξ_e corresponde al error de características, ξ_d es la referencia de las características y ξ_c corresponde a la característica de la cámara.

Derivando, se obtiene que:

$$\dot{\xi}_e = \dot{\xi}_d - \dot{\xi} = \begin{bmatrix} \dot{u}_d - \dot{u} \\ \dot{v}_d - \dot{v} \end{bmatrix} \tag{2.40}$$

La candidata de Lyapunov será:

$$V = \frac{1}{2}\tilde{\xi}^T\tilde{\xi} \tag{2.41}$$

Derivando V, se obtiene la ecuación (2.42)

$$\dot{V} = \tilde{\xi}^T\tilde{\xi} \tag{2.42}$$

Reemplazando (2.40) en (2.42):

$$\dot{V} = \tilde{\xi}^T(\dot{\xi}_d - \dot{\xi}) \tag{2.43}$$

Reemplazando (2.38) en (2.43):

$$\dot{V} = \tilde{\xi}^T(\dot{\xi}_d - J(q, {}^W P_O)\dot{q}) \tag{2.44}$$

La ecuación (2.44) debe cumplir con las condiciones de estabilidad descritas en el (Capítulo 1, sección 1.5), por lo que se propone la siguiente ley de control:

$$\dot{q}_c = J^{-1}(\dot{\xi}_d + k_1\tilde{\xi}) \tag{2.45}$$

Reemplazando (2.45) en (2.44) se tendrá:

$$\dot{V} = \tilde{\xi}^T(\dot{\xi}_d - J(J^{-1}(\dot{\xi}_d + k_1\tilde{\xi}))) \tag{2.46}$$

$$\dot{V} = \tilde{\xi}^T(\dot{\xi}_d - \dot{\xi}_d - k_1\tilde{\xi}) \tag{2.47}$$

$$\dot{V} = -k_1\tilde{\xi}^T\tilde{\xi} \tag{2.48}$$

Como se puede ver en la ecuación (2.48), se cumple que:

$$\dot{V} < 0 \; para \; todo \; k_1 > 0 \tag{2.49}$$

El controlador cumple que para todo valor de k_1 mayor a cero, la función de Lyapunov escogida V será estable, por lo tanto, el error $\tilde{\xi}$ tiende a cero conforme el tiempo tienda a infinito.

Se diseña saturadores con la finalidad de suavizar las señales de control que serán enviadas a los actuadores del robot, la ley de control queda definida de la siguiente manera:

$$U = J^{-1}(\dot{\xi}_d + K_P\tanh(k_2\tilde{\xi})) \tag{2.50}$$

Donde la constante K_p determinará la amplitud del error y la constante K_2 determina la pendiente entre los puntos positivos y negativos del error, como se observa en la Figura 2.6.

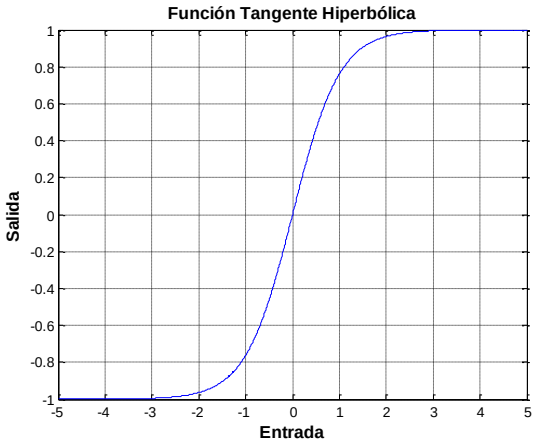


Figura 2.6. Función tangente hiperbólica

La Figura 2.6 muestra el resultado de la función tangente hiperbólica, en donde, la respuesta negativa de la función corresponde a los valores negativos de la señal de entrada, así como también si los valores de la función son positivos corresponde a los valores positivos de la señal de entrada, por lo que no afecta a la estabilidad del controlador y se mantiene el signo del error. Por ende se logra saturar la acción de control sin alterar su estabilidad con $K_p > 0$ y $k_2 > 0$.

2.3 PROCESAMIENTO Y ADQUISICIÓN DE IMÁGENES

2.3.1 ALGORITMO DE VISIÓN

El procesamiento y adquisición de imagen presentan una secuencia de etapas que nos permiten obtener información necesaria para dar soluciones a aplicaciones propuestas. Las etapas del algoritmo de visión se muestran en la Figura 2.7.

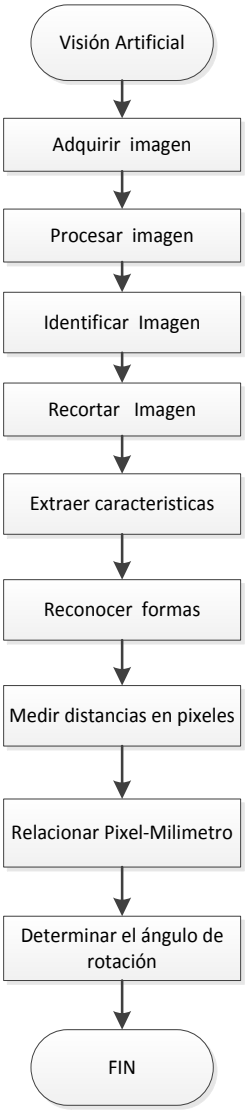


Figura 2.7. Algoritmo de Visión

2.3.2 ADQUISICIÓN DE IMAGEN

En la adquisición de la imagen se usa la herramienta de Matlab (Image Acquisition Tool) como se observa la Figura 2.8. En el proceso de visión artificial, se debe recordar escoger un sensor de visión (cámara) adecuado para la captura de imágenes, como se menciona en el (Capítulo 1, sección 1.4.4.1).

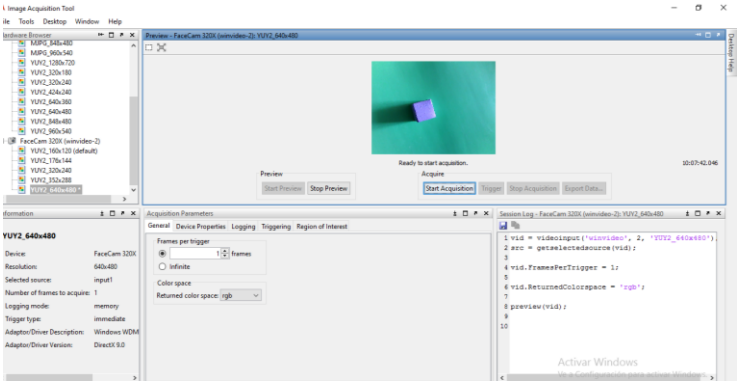


Figura 2.8. Adquisición de imagen mediante Matlab-Image Acquisition Tool.

Se inicia el proceso de adquisición en (Imagen Acquisition Tool) con la selección del nombre de la cámara (FaceCam 320x), la selección de la resolución de la imagen, en este caso, 640x480pixeles y la generación de un código fuente, como se observa en la Figura 2.8. Existen otros parámetros que se puede utilizar como son: la región de interés, selección del espacio de color, entre otros. Para la grabación y captura de imagen se utiliza comandos propios de Matlab, en la Figura 2.9 se muestra la captura de imagen a color RGB.

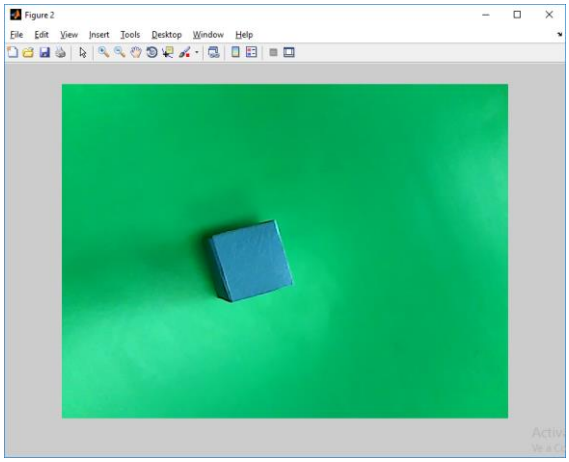


Figura 2.9. Imagen real.

2.3.3 PREPROCESAMIENTO DE IMAGEN

2.3.3.1 Umbralización Fija

En la Figura 2.10 se presenta la imagen segmentada, en donde se cambia la imagen al espacio HSV y se realiza la segmentación por umbralización, es decir, convertir una imagen en escala de grises a una nueva con solo dos niveles, cuyo

fin es separar el objeto del fondo, como se menciona en el (Capítulo 1, sección 1.4.6.2).

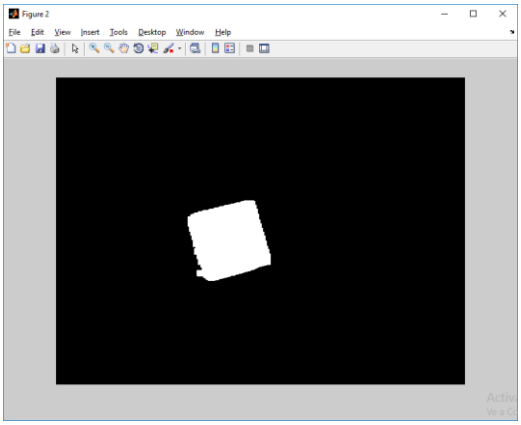


Figura 2.10. Imagen segmentada.

La idea de umbralización es diferenciar objetos utilizando niveles de intensidad de acuerdo con:

$$P(x,y) = \begin{cases} 1 & si \;\; Umin \leq I(x,y) \leq Umax \\ 0 & si \qquad\quad Otro \end{cases} \tag{2.51}$$

Donde,

$P(x,y)$: Imagen binaria.

$I(x,y)$; Imagen original en el espacio HSV.

$Umin$: Rango de umbralización mínimo.

$Umáx$: Rango de umbralización máximo.

En la Tabla 2.1 se presenta los valores de los niveles de umbralización que se va a utilizar.

Tabla 2.1. Valores de niveles de umbralización

COLOR	RANGO	H	S	V
AZUL	$Umin$	0.481	0.000	0.000
	$Umáx$	0.995	1.000	1.000

En la Figura 2.11a se observa la imagen segmentada con pequeñas áreas que no pertenecen al objeto debido a factores externos como son la iluminación, entre otros, es por eso por lo que se necesita otro tipo de proceso de segmentación como es dilatación y erosión estudiada en el (Capítulo 1, sección 1.4.4.2.3).

En la Figura 2.11b se observa la imagen segmentada sin áreas no deseadas y para comenzar con el procesamiento de imagen.

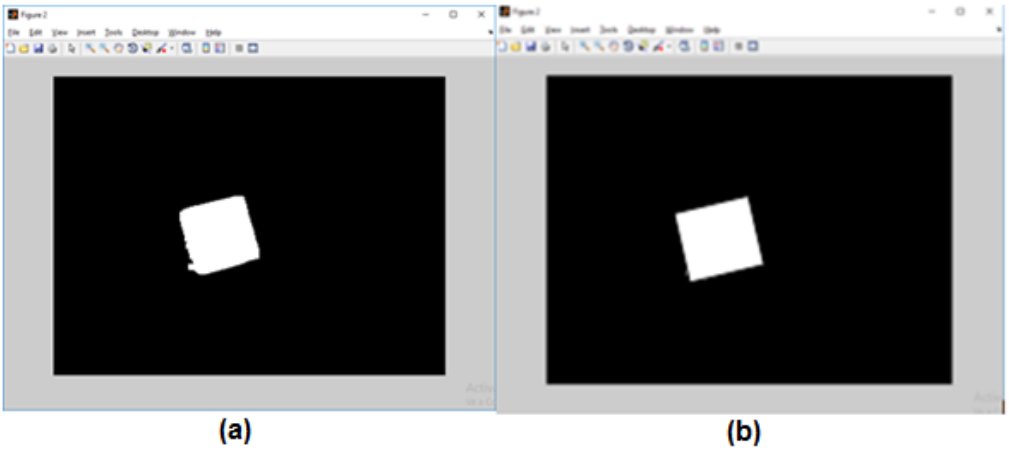


Figura 2.11. Imagen fragmentada.

2.3.4 EXTRACCIÓN DE CARACTERÍSTICAS, RECONOCIMIENTO DE FORMA

2.3.4.1 Extracción de características de la imagen

Cuando se tiene la imagen binarizada, se realiza el procesamiento y la obtención de las caracteristicas de la imagen, para extraer las caracteristicas de la imagen se usa la propiedad *"regionprops"* →"*Centroid"* de Matlab, como se observa en la Figura 2.12.

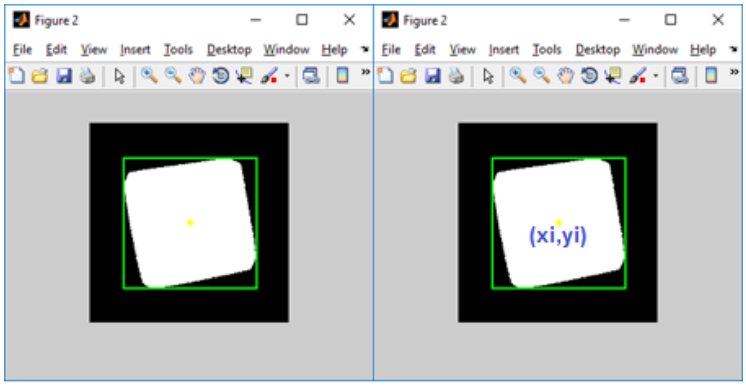


Figura 2.12. Extracción de las características de la imagen.

2.3.4.2 RECONOCIMIENTO DE FORMA

Para el reconocimiento de nuestro patrón (cubo), se realiza el procedimiento del análisis de imagen como se explica en el (Capítulo 1, sección 1.4.4.2), el patrón es un conjunto de características que permite buscar el objeto deseado, basadas en las características extraídas. Los procesos que ayudan con la asignación de características similares son:

- La clasificación geométrica (en donde los patrones deben ser graficarles como la geometría de forma, valores numéricos, puntos de atracción, etc.).
- La clasificación estadística (donde los patrones se basan en probabilidades y estadísticas [10]).

En este proyecto se ejecuta el algoritmo de visión artificial con el proceso de clasificación geométrica, buscando áreas que se asemejen al área del patrón establecido, para luego encontrar su centroide, como se explica en el (Capítulo 2, sección 2.3.4.1).

El objeto localizado dentro del plano de la imagen está representado en pixeles. Esta localización en el plano real tiene un valor representado en centímetros, por ende, se necesita realizar una relación pixel-milímetro.

2.3.5 TRANSFORMACIÓN DE SISTEMAS DE COORDENADAS, RELACIÓN DEL PLANO DE LA IMAGEN (PIXEL) CON EL PLANO REAL (MILIMETRO) Y DETERMINACIÓN DEL ÁNGULO DE ROTACIÓN.

2.3.5.1 Transformación de coordenadas de pixeles a coordenadas cartesianas

Para realizar la transformación de coordenadas, se debe tener en cuenta la representación de los pixeles en el plano de la imagen como se observa en la Figura 2.13.

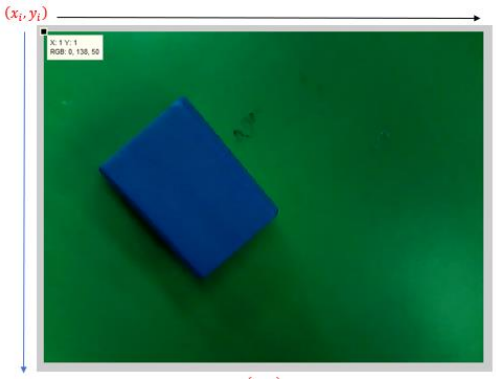


Figura 2.13. Representación de pixeles en plano de la imagen.

Para representar los pixeles en el plano de la imagen se usa los ejes coordenados (x_i, y_i) : el eje x_i representa el ancho de la imagen y va de izquierda a derecha, mientras el eje y_i representa el alto de la imagen y crece hacia abajo.

2.3.5.2 Relación del plano real con el plano de la imagen

Para la localización del objeto en el área de trabajo del robot SCARA, es necesario realizar una transformación de pixeles a milímetros, así como también la transformación de coordenadas mencionadas en el (Capítulo 2, sección 2.3.5.1).

El procedimiento propuesto consiste en medir las distancias de los lados de nuestro objeto en el plano de la imagen, como se puede observar en la Figura 2.14.

Figura 2.14. Medición de lados.

La relación del plano real con el plano de la imagen consiste en buscar una relación matemática (pixel-milímetro) que permita obtener una sola unidad de medida.

Se define como relación milímetro por pixel en la siguiente ecuación:

$$Relación_{ejex} = \frac{distancia\ en\ milímetros\ de\ X}{distancia\ en\ píxeles\ de\ X} \tag{2.52}$$

$$Relación_{ejex} = \frac{38[mm]}{223[pixel]} \approx 0.17[mm/px]$$

$$Relación_{ejey} = \frac{distancia\ en\ milímetros\ de\ Y}{distancia\ en\ píxeles\ de\ Y} \tag{2.53}$$

$$Relación_{ejey} = \frac{28[mm]}{156[pixel]} \approx 0.17[mm/px]$$

Entonces, se puede realizar las transformaciones de las coordenadas del centro de la imagen, que se usará como características de la imagen.

En el diseño del modelo cinemático, las longitudes del robot las expresamos en metros [m]. Entonces, para poder trabajar en el controlador servo visual que incluye modelo cinemático y modelo cámara, es necesario cambiar de unidades a la relación, es decir, pasar de [mm/pixel] a [m/pixel].

2.3.5.3 DETERMINACIÓN DEL ÁNGULO DE ROTACIÓN

Para determinar el ángulo de rotación de la imagen, se utiliza la propiedad “Orientation” del comando “regionprops” de Matlab. Esta propiedad consiste en trazar imaginariamente una elipse que contenga el ROI (región de interés), en donde se mide el ángulo que forma el eje x de la imagen con el eje mayor de la elipse, como se observa en la Figura 2.15.

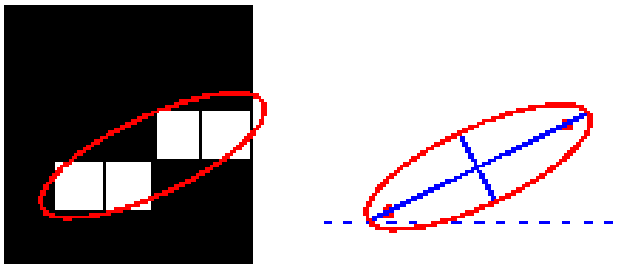


Figura 2.15. Orientación, de Matlab.

2.4 DISEÑO DEL HARDWARE Y SOFTWARE DEL ROBOT MANIPULADOR TIPO SCARA

2.4.1 ANÁLISIS DEL ESTADO DE LAS PLACAS ELECTRÓNICAS

El robot SCARA, ubicado en el Laboratorio de Control y Sistemas, posee una placa de potencia y una placa de control, las mismas que fueron revisadas y analizadas para verificar su estado y su funcionamiento.

2.4.1.1 Placa de control

El robot manipulador tipo SCARA ubicado en el Laboratorio de Control y Sistemas, poseía una tarjeta de adquisición de datos denominada PCI 6221M de National Instruments. Esta tarjeta estaba conectada directamente al mainboard de la Pc; posee un zócalo que es acoplado a un bloque de terminales en donde ingresan todas las señales análogas y digitales de los sensores que tiene el robot SCARA.

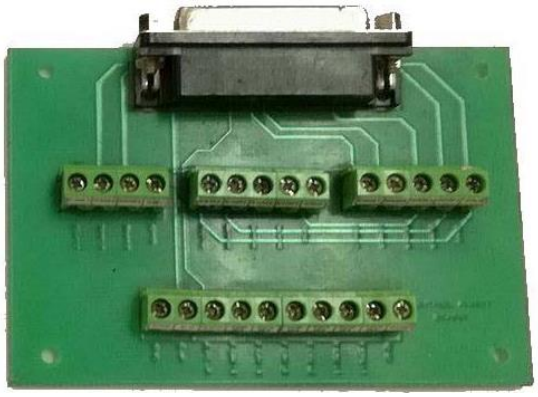


Figura 2.16. Placa de control.

Externo a esta tarjeta de adquisición en la placa de potencia se encontró un PIC que controlaba la pinza ubicada en el efector final del robot manipulador SCARA.

2.4.1.2 Placa de potencia

La placa de potencia del robot SCARA se encuentra en buen estado, las pistas y elementos estaban bien conectados y en perfecto estado Figura 2.17.

El control de los motores se realiza utilizando puentes H (denominado L298), esta placa tiene una alimentación eléctrica desde la red,

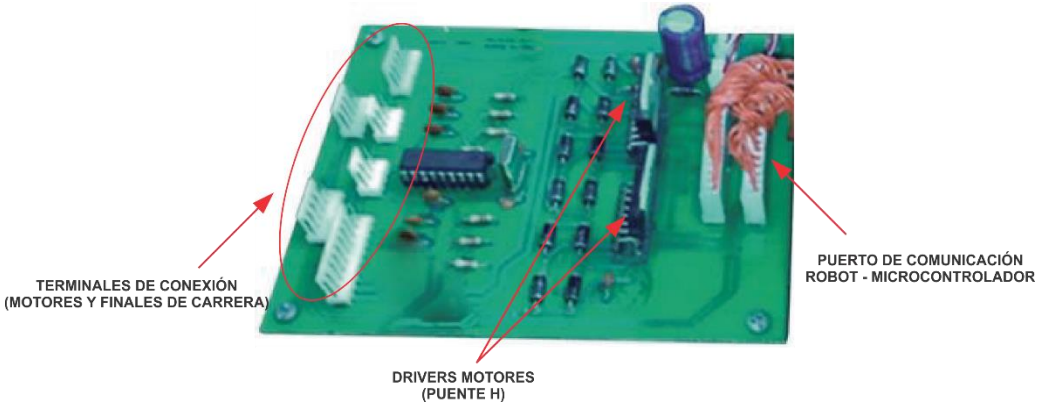


Figura 2.17. Placa de potencia

2.5 DISEÑO DEL SISTEMA DE CONTROL

Una vez realizado el análisis de las placas de potencia y la placa de control, se realiza el diseño de la nueva placa de control, como también la construcción de la misma.

2.5.1 PLACA DE CONTROL

En la placa de control se asigna tareas como el procesamiento de datos, la toma de decisiones y la comunicación. Consta de las siguientes partes:

- Micro controlador.
- Comunicación.
- Conector DB25.
- Fuente de alimentación.

2.5.1.1 Selección del micro controlador

La tarea del micro controlador es realizar el procesamiento de los datos obtenidos de los sensores y el accionamiento de los actuadores. Para establecer las características del micro controlador, se debe conocer el número necesario de entradas y salidas, así como también contar con la memoria suficiente para almacenar el programa que controla los actuadores, realizar el procesamiento de los sensores y una comunicación serial bidireccional con una PC.

De acuerdo con las características, el micro controlador escogido es el Atmega 164P. En la Tabla 2.2 se indica la distribución de pines y la conexión con el resto del hardware.

Tabla 2.2. Distribución de pines y conexión, primer micro controlador.

Atmega 164p	Conectado a	Características
Vin	5V	Fuente de Alimentación
Gnd	Tierra	Fuente de Alimentación
Puerto A	A0, A1, …A7	Entradas de los encoders y finales de carrera
Puerto B	B0, B1, …B7	Salidas PWM y contadores de pulsos
Puerto C	C0, C1, …C7	Salidas PWM y contadores de pulsos
Puerto D	D0, D1, …D7	Activación de interrupciones externas para la pinza.

En la Tabla 2.3, se presenta la distribución de pines del segundo micro controlador para el control de la pinza del manipulador.

Tabla 2.3. Distribución de pines y conexión, segundo micro controlador.

Atmega 164p	Conectado a	Características
Vin	5V	Fuente de Alimentación
Gnd	Tierra	Fuente de Alimentación
Puerto A	A0, A1, …A7	Entradas de los finales de carrera
Puerto B	B0, B1, …B7	Salidas PWM y contadores de desplazamiento

2.5.1.2 Selección del conversor USB TTL

Para lograr la comunicación con la computadora se utiliza comunicación serial entre microprocesador y PC usando un conversor USB TTL, en la Tabla 2.4 se describe las características del conversor.

Tabla 2.4. Características del conversor USB TTL.

CARACTERÍSTICAS	DESCRIPCIÓN
VELOCIDAD DE TRANSMISIÓN	9600 baudios
PARIDAD	Sin paridad
BITS DE PARADA	0 bits
NÚMERO DE BITS DE COMUNICACIÓN	8 bits

2.5.1.3 Conector DB25

Es un conector análogo de 25 clavijas que se usa para conexiones en paralelo, ya que permite una transmisión asíncrona de datos según la norma RS-232. En la placa de control se requiere de este conector para establecer conexiones con los periféricos de entrada y salida de la placa de potencia.

2.6 CONSTRUCCIÓN DE LA PLACA DE CONTROL

El circuito esquemático, en el que se observa los pines de conexión de la placa de control se encuentra en el anexo C.

2.6.1 PROGRAMA ISIS

Este programa permite dibujar y simular el circuito completo del sistema de control. Los elementos usados en el programa ISIS tienen su propia librería con su forma física que facilita la realización de un circuito impreso en el ARES. La pantalla inicial para realizar el circuito de control se muestra en la Figura 2.18.

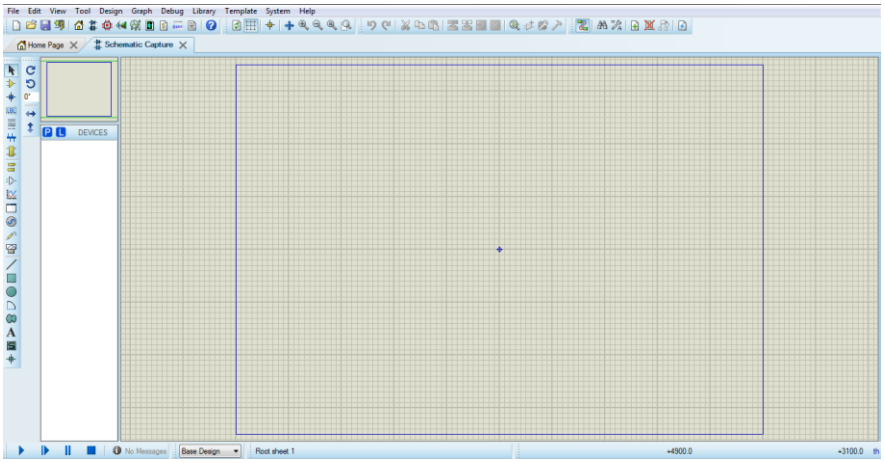


Figura 2.18. Pantalla inicial, programa ISIS

Para representar elementos que no tienen en ISIS su propia librería se utiliza borneas o conectores para obtener su forma física en el diseño del circuito impreso en el ARES.

2.6.2 ARES

Los esquemas electrónicos diseñados en ISIS pasan a ARES (Figura 2.19) para empezar el trazado de las pistas que unen a cada uno de los elementos dibujados en ISIS, es decir, lo que se hace en el ARES es facilitar el diseño de las placas de un circuito impreso.

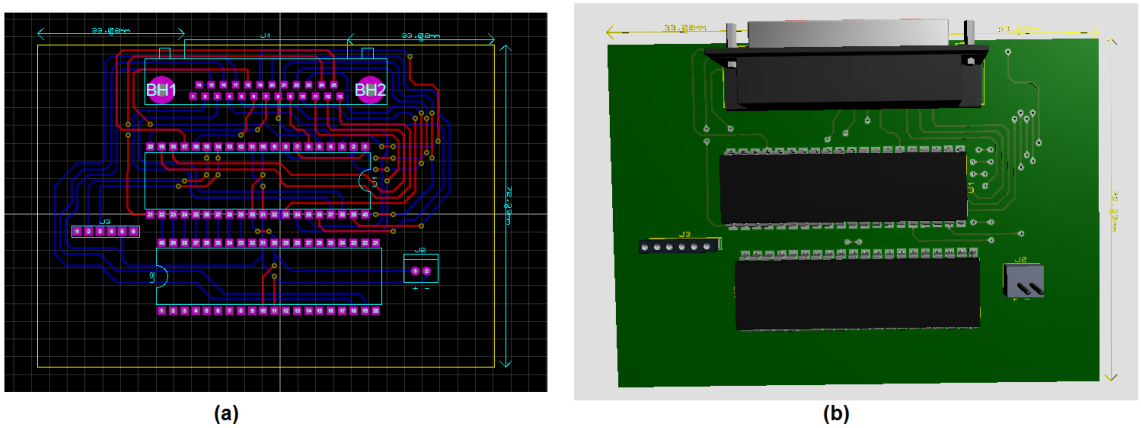


Figura 2.19. ARES: (a) Ubicación de los elementos; (b) Vista 3D de la placa.

2.6.3 PASOS DE CONSTRUCCIÓN DE LA PLACA DE CONTROL

Ya realizado el circuito impreso en ARES, se procede a pasarlo a una baquelita, posteriormente soldar sus elementos y por último colocarla en el robot SCARA.

En el trabajo de grabado de la PCB (circuito impreso) se utilizó servicios de la empresa Smelektronik, con la finalidad de obtener una placa de control con acabado profesional. Una vez soldados los elementos se procede hacer pruebas de funcionamiento y por último coloca la placa al robot SCARA, como se observa en la Figura 2.20.

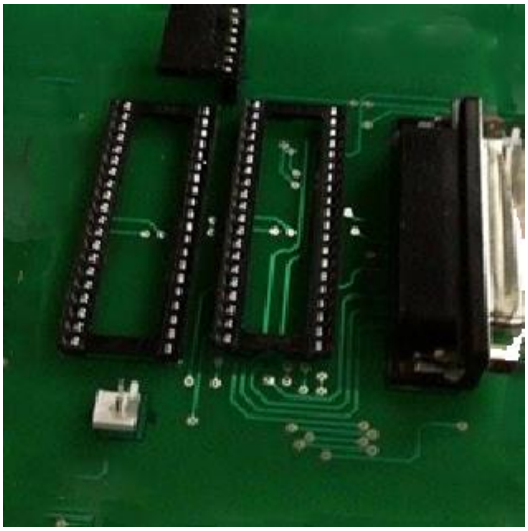


Figura 2.20. Placa de control finalizada.

2.7 DESARROLLO DEL SOFTWARE

Una vez analizado y rediseñado el hardware del sistema, se requiere realizar un software adecuado para el funcionamiento del mismo. El micro controlador debe ser programado para que pueda realizar las tareas analizadas anteriormente descritas.

2.7.1 POSICIONAMIENTO INICIAL DEL ROBOT SCARA

La posición inicial del robot SCARA es importante en el desarrollo del programa, para conocer el inicio de los contadores de pulsos obtenidos de los encoders, así como también el ángulo y el punto inicial del efector final con respecto a la base del robot debe tener las siguientes coordenadas $P_i = [P_x = 30, P_y = 0, P_z = 12]$, conociéndose que el brazo debe estar completamente extendido con el efector final en la parte superior del eje Z.

2.7.2 PROGRAMACIÓN DEL MICROCONTROLADOR ATMEGA 164P

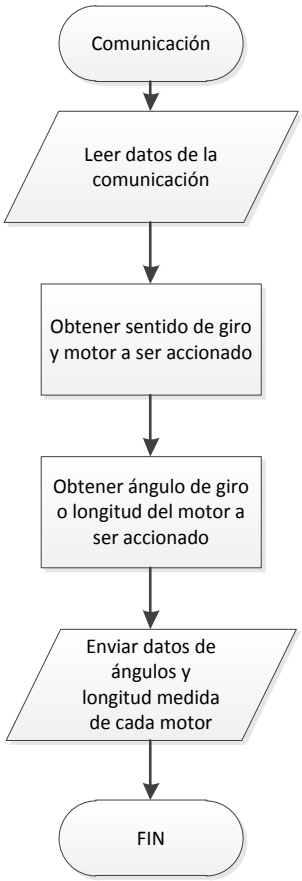


Figura 2.21. Diagrama de flujo, comunicación PC – micro controlador.

El micro controlador Atmega 164p, se encargará de procesar la información enviada de los sensores (encoders, finales de carrera y cámaras) y control de los motores, como también se usará para realizar una comunicación con la interfaz del computador. Para la comunicación PC - micro controlador, la Figura 2.21 muestra el diagrama de bloques con el funcionamiento de ésta.

El robot manipulador tipo SCARA tiene 3 articulaciones y cada una de estas utiliza 1 salida tipo PWM. Las articulaciones q_1 y q_2 utilizan el Timer0 del microcontrolador el cual es de 8 bits, los motores de las articulaciones poseen encoders con un desfase de 90º entre las señales de cada uno de los canales (A y B), por lo que se necesita realizar un contador de pulsos que se encargue de realizar el incremento o decremento del valor, tomando en cuenta el sentido de giro de cada motor. Los contadores de pulso tomaran una posición cero cuando se cumpla la rutina de

inicialización como se explica (Capítulo 2, sección 2.7.1), por ende, se usará una interrupción externa del micro controlador.

La pinza usa 2 salidas PWM, la primera se utiliza para el movimiento rotacional del actuador, mientras que la segunda abre y cierra la pinza.

La Figura 2.22 muestra el diagrama de flujo para la configuración de variables de entrada y salida, y el funcionamiento del motor 1.

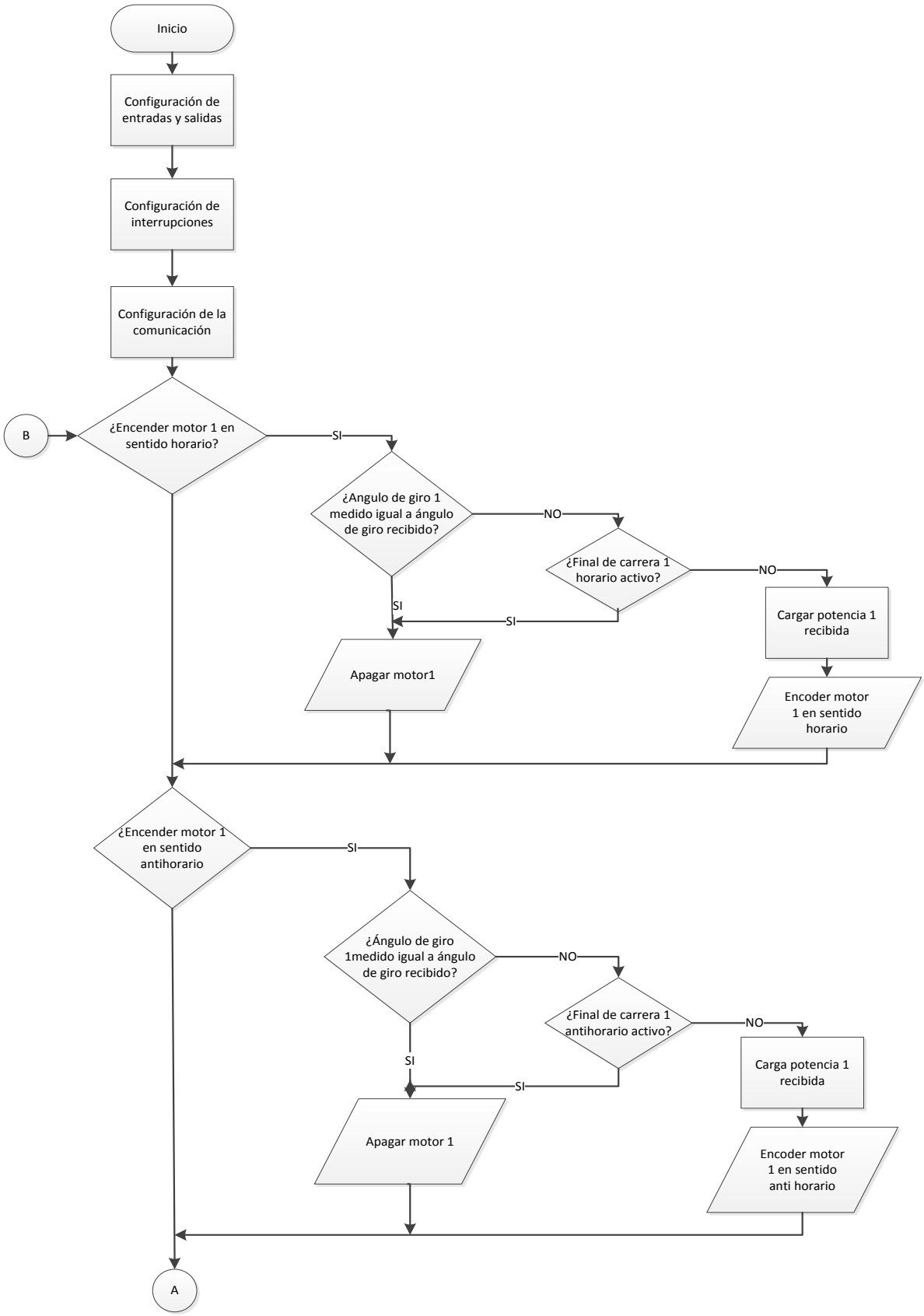


Figura 2.22. Diagrama de flujo de inicio, Motor 1.

La Figura 2.23 muestra el diagrama de flujo del funcionamiento del motor 2 del robot manipulador.

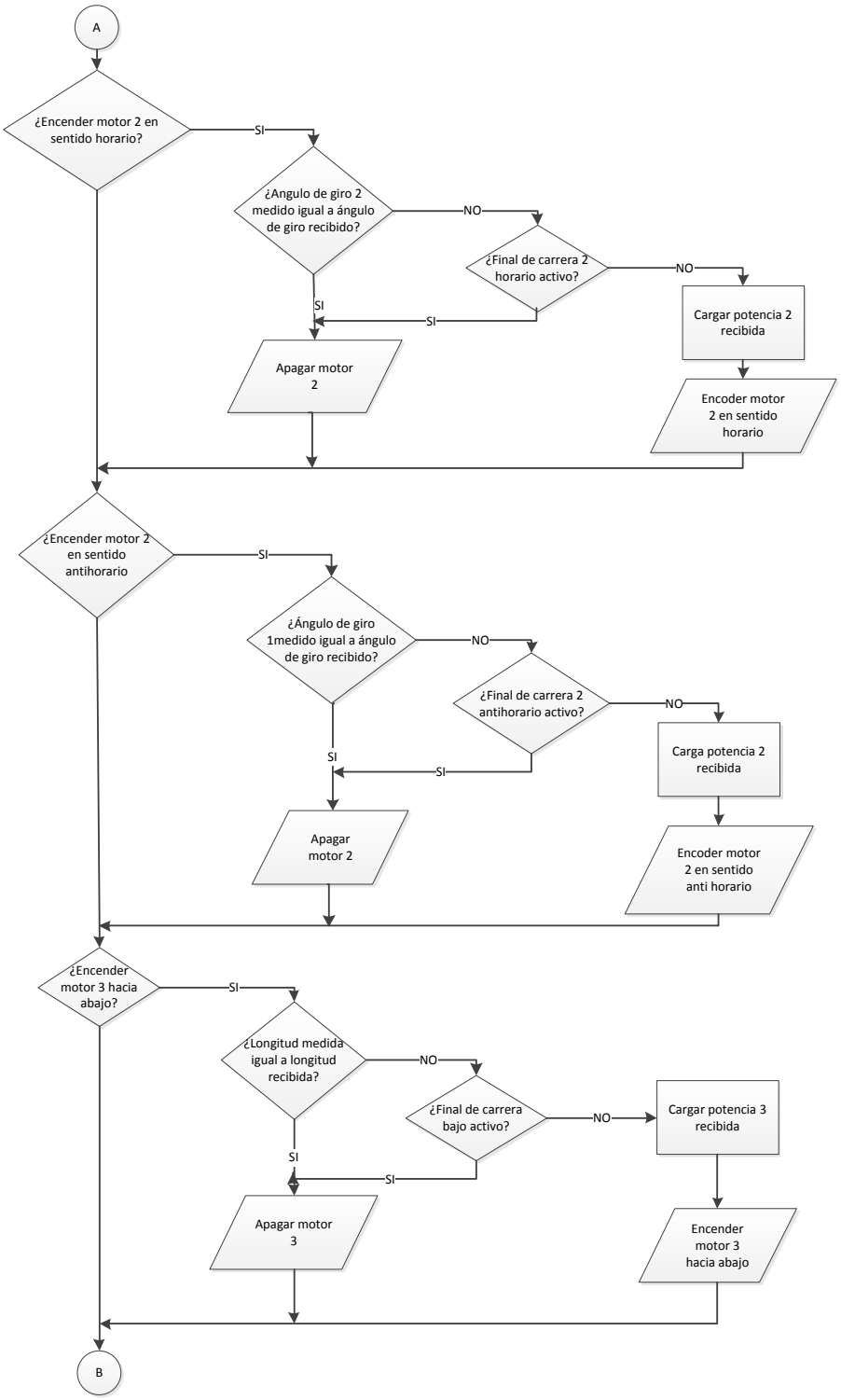


Figura 2.23. Diagrama de flujo, Motor 2.

Para la apertura de la pinza, la lógica de control se presenta el diagrama de flujo en la Figura 2.24.

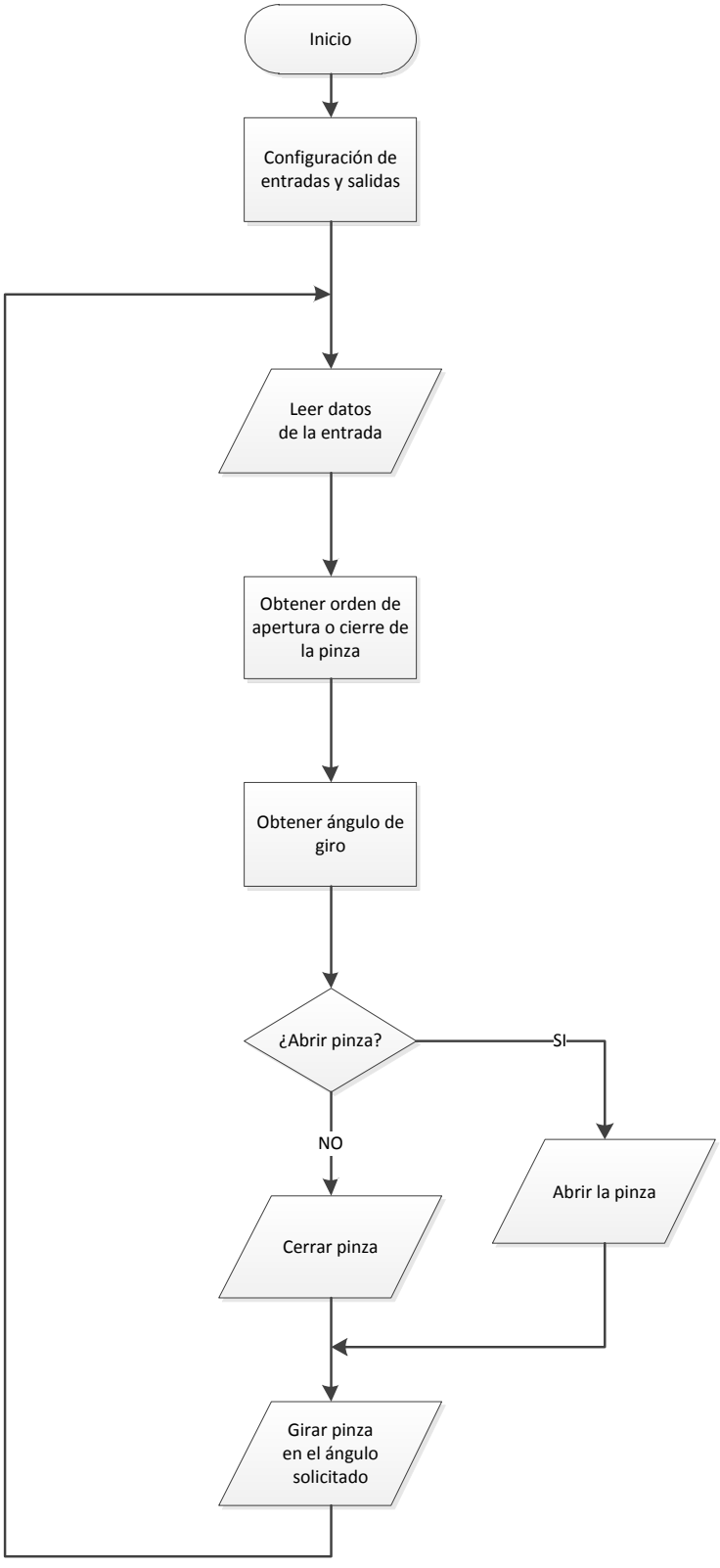


Figura 2.24. Diagrama de flujo, Pinza.

En la trama de datos el computador envía las direcciones para el movimiento de los motores, mientras que el micro controlador envía al computador las señales de los sensores internos (encoders) y externos (finales de carrera).

La trama de datos para la posición de los motores se presenta en la Figura 2.25.

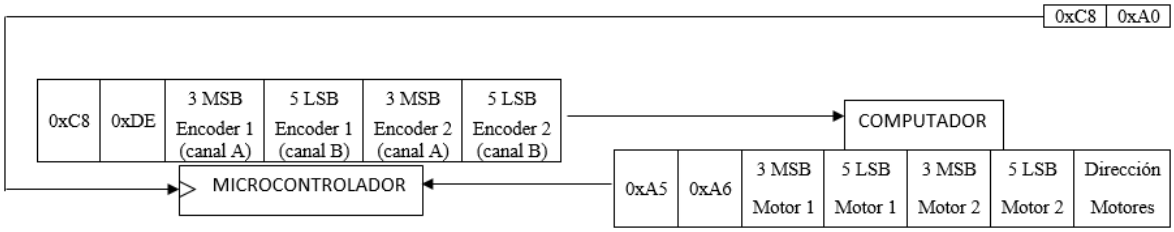


Figura 2.25. Trama de datos del micro controlador

Tabla 2.5. Byte dirección de Motores

DIRECCIÓN DE MOTORES							
Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
-	-	SAM2	SHM2	CM2	SAM1	SHM1	CM1

En la Tabla 2.5 se describe el byte de dirección de motores, cada bit se encuentra descrito como se observa en la Tabla 2.6.

Tabla 2.6. Byte de Dirección de Motores

CMX	SHMX	SAMX	ACCIÓN
0	0	0	Motor en marcha
0	0	1	Giro anti horario
1	1	0	Giro horario
1	1	1	Motor detenido

La Figura 2.26 la configuración para cada una de las interrupciones del micro controlador.

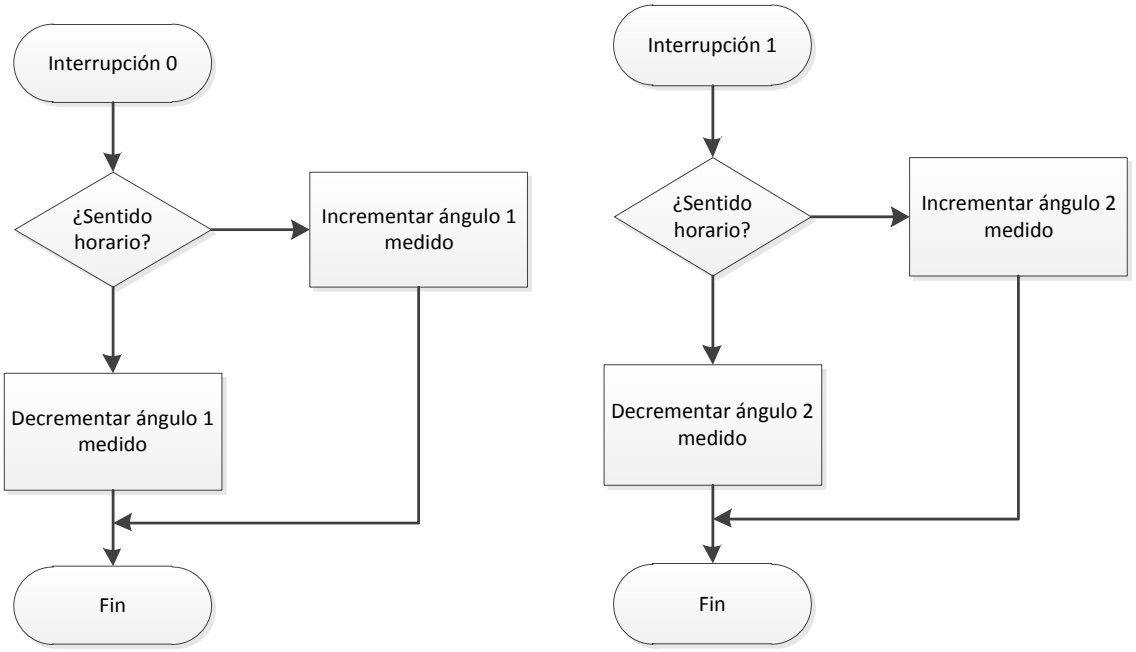


Figura 2.26. Configuración interrupciones.

2.7.3 PROGRAMACIÓN DEL ENTORNO MATLAB

El controlador diseñado en Matlab tiene datos que deben ser enviados al micro controlador para que el robot SCARA funcione, el entorno de Matlab y sus comandos permiten su envío, usando el puerto serial de la computadora.

CAPÍTULO 3

DESARROLLO DE UN HMI

3.1 DISEÑO E IMPLEMENTACIÓN DE LA INTERFAZ GRÁFICA PARA EL CONTROL SERVO VISUAL DE UN ROBOT MANIPULADOR TIPO SCARA.

El programa GUI de Matlab permite el desarrollo de una interfaz gráfica que facilite al usuario el manejo del programa.

3.1.1 ENTORNO GUI

Representan un frontal gráfico que automatizan una tarea o un cálculo, presentando su entorno de trabajo como se observa en la Figura 3.1.

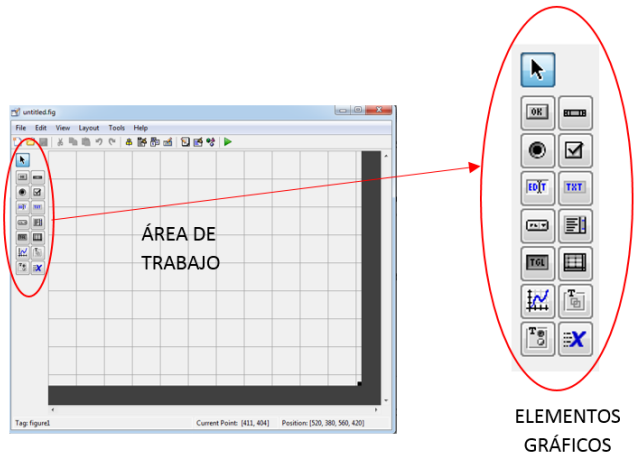


Figura 3.1. Entorno GUI para crear una interfaz.

3.1.2 CONTROLES DE LA INTERFAZ GRAFICA

Las GUI incluyen controles como menús, barras de herramientas, botones, controles deslizantes y elementos del entorno gráfico, como se observa en la Figura 3.2.

ELEMENTO-CONTROL	DESCRIPCIÓN
Select	Seleccionar el elemento o control
Push Button	Invoca un evento inmediatamente
Slider	Usado para representar un rango de valores
Radio Button	Indica una opción que puede ser seleccionada
Check Box	Indica el estado de una opción
Edit Text	Caja para editar texto
Static Text	Muestra el string de texto en una caja
Pop-up Menu	Provee una lista de opciones
Listbox	Provee una lista dentro d una caja
Toggle Button	Representa dos estados “on” u “off”
Axes	Muestra un gráfico en una caja
Panel	Muestra botones

Figura 3.2. Controles de interfaz de usuario

3.2 INTERFAZ GRÁFICA

Interfaz se encuentra estructurada de la siguiente manera:

- Panel de comunicación
- Panel ROBOT_SCARA, LAZO ABIERTO
- Panel CONTROLADOR SERVO VISUAL, ROBOT_SCARA_CÁMARA_MANO.
- CONTROL SERVO VISUAL PARA UN ROBOT TIPO SCARA-DESEMPEÑO y gráficas.

3.1.3 DISEÑO DE LA INTERFAZ PARA EL CONTROLADOR SERVO VISUAL PARA UN ROBOT TIPO SCARA

3.1.3.1 Panel de COMUNICACIÓN_PC_ROBOT

La interfaz para el controlador servo visual para un robot manipulador tipo SCARA se realizó con la finalidad de poder interactuar con el robot a través de la computadora. En la Figura 3.3 se observa la ventana correspondiente a este panel, en donde se observa la opción de conectar al robot con la interfaz y el posicionamiento inicial del robot.

Figura 3.3. Panel de **COMUNICACIÓN_PC_ROBOT.**

La Figura 3.4 presenta un diagrama de las principales acciones que se realiza en el panel de **COMUNICACIÓN_PC_ROBOT.**

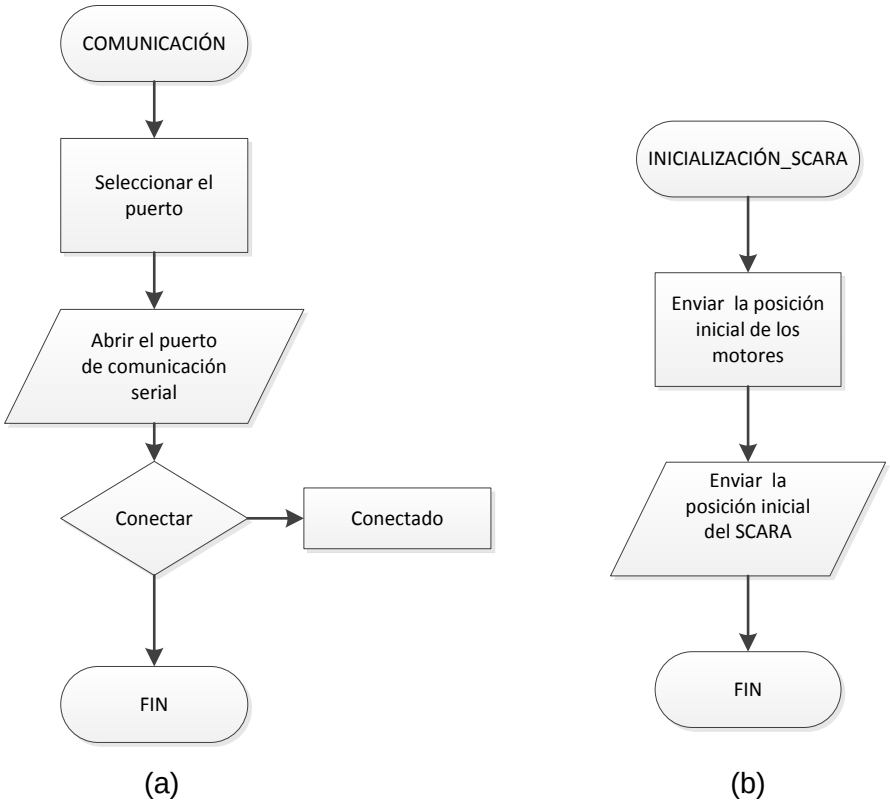


Figura 3.4. Diagrama de las acciones que realiza el panel de **COMUNICACIÓN_PC_ROBOT**.

3.1.3.2 Panel ROBOT_SCARA, LAZO ABIERTO

En la Figura 3.5 se observa el panel **LAZO ABIERTO** el cual presenta slider que representan las velocidades para dar movimiento a los motores, se usan botones

para enviar los valores negativos o sentido anti horario y positivos o sentido horario que corresponden de -180 grados a 180 grados respectivamente para cada motor, así como también un botón para enviar a la posición inicial al robot, es decir, 0 grados.

Los indicadores representados en VALOR_ÁNGULOS permiten observar si el robot está funcionando, como también los valores de las variables que forman parte del sistema.

Figura 3.5. Panel **ROBOT_SCARA, LAZO ABIERTO**

Para el movimiento del efector final, se presentan elementos actuadores como los slider para la rotación de la pinza cuyo valor corresponde de 0 grados a 90 grados, ya que el motor implementado para el efector final es un servo motor.

Se usan botones para enviar los valores de los ángulos de rotación, como también para la función abrir y cerrar pinza, los indicadores VALOR_ÁNGULOS, muestran los valores de las variables.

3.1.3.3 Panel CONTROLADOR SERVO VISUAL, ROBOT_SCARA_CÁMARA_MANO.

El panel **CONTROLADOR SERVO VISUAL** permite implementar el controlador servo visual en el robot manipulador tipo SCARA, en este panel se puede elegir el tipo de cámara Face Cam o Web Cam, tal como se aprecia en la Figura 3.6.

En el panel **Video_Tiempo_Real** se visualiza la posición del objeto en el área de trabajo del robot y de la cámara, si el objeto esta fuera del campo de visión no existe movimiento en el robot. Se representa en forma de simulación la visualización del área de trabajo del robot y el área de trabajo de la cámara en donde se observa el movimiento del robot.

Para visualizar los valores de la posición y orientación que tiene el efector final se usan indicadores para conocer las posiciones finales del robot.

Se usan botones para escoger el cuadrante en el que se encuentra el objeto deseado dentro del área de trabajo, así como también un botón para aplicar el controlador servo visual y un botón para abrir la interfaz de desempeño y graficas de errores.

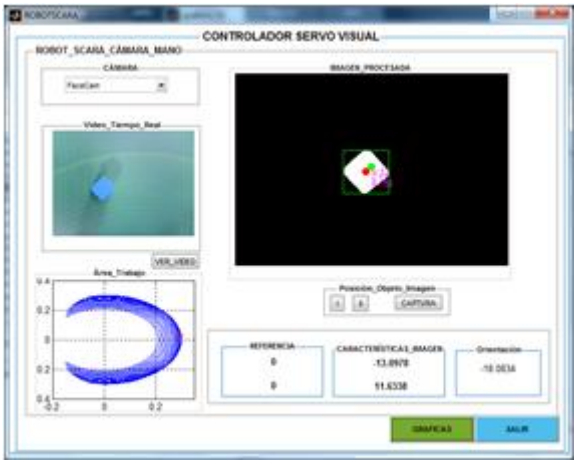


Figura 3.6. Panel **CONTROLADOR SERVO VISUAL, ROBOT_SCARA_CÁMARA_MANO**

El diagrama de bloques de las principales tareas realizadas en este panel se observa en la Figura 3.7.

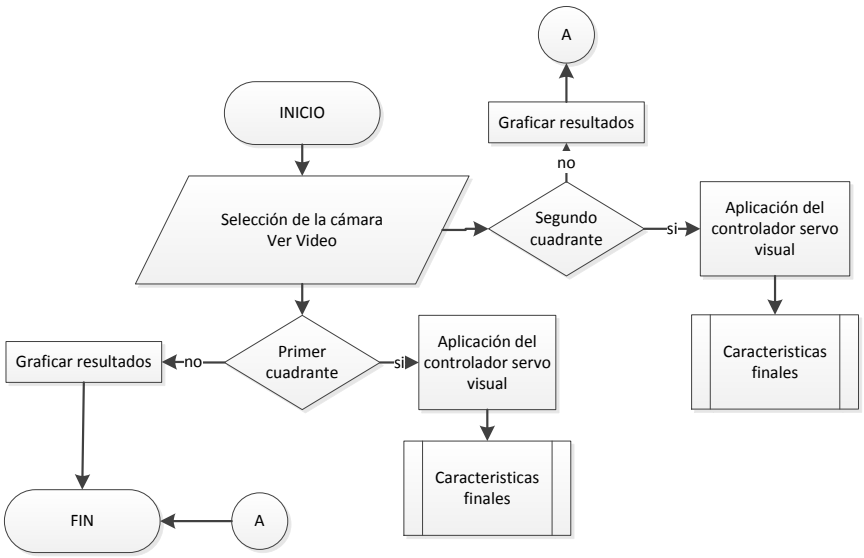


Figura 3.7. Diagrama de las tareas realizadas en el panel **CONTROLADOR SERVO VISUAL**

3.1.3.4 CONTROL SERVO VISUAL PARA UN ROBOT TIPO SCARA-DESEMPEÑO y gráficas.

Como se puede apreciar en la Figura 3.8 se puede seleccionar la posición del efector final en el plano de la imagen, así como también los errores, la gráfica de la orientación del efector final y las acciones de control.

En esta interfaz se puede visualizar el desempeño del controlador aplicando el ISE para cada uno de los ejes del plano de la imagen.

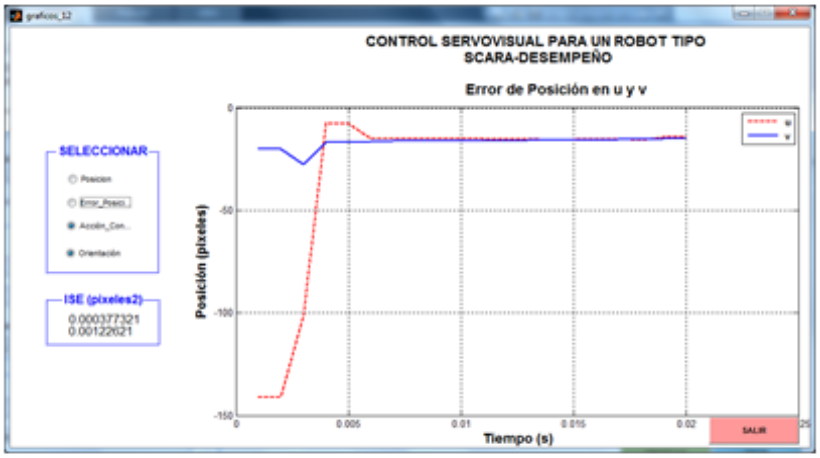


Figura 3.8. Interfaz de gráficas

CAPÍTULO 4

PRUEBAS Y RESULTADOS

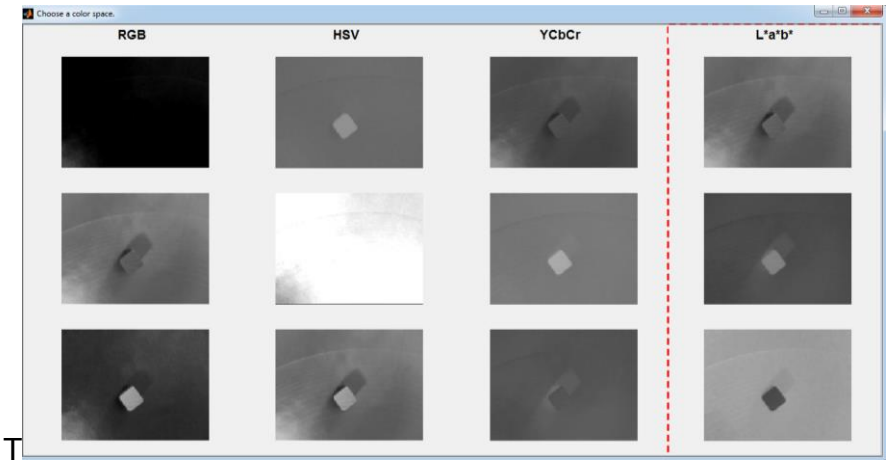
En el presente capítulo se muestran las pruebas y resultados del controlador servo visual basado en las características de la imagen de un robot manipulador tipo SCARA diseñado en el Capítulo 2.

4.1 ALGORITMO DE VISIÓN ARTIFICIAL

En la etapa de visión artificial se describen los resultados obtenidos para encontrar las características del objeto en el plano de la imagen, como también los resultados obtenidos con diferentes ángulos de orientación.

4.4.1 UMBRLIZACIÓN DE LA IMAGEN USANDO EL ESPACIO HSV

La imagen en espacio RGB se transformó a un espacio HSV usando “Image Processing Toolbox” de Matlab, este Toolbox permite al usuario transformar las imágenes reales a diferentes espacios de color como son: RGB, HSV, YCbCr L*a*b*, como se observa en la Figura 4.1,



T

Figura 4.1. Escoger espacio de color

En este proyecto se usa la transformación a espacio de color HSV, como se observa la Figura 4.2, donde la imagen real adquirida es filtrada de forma natural, el parámetro tono (H) indica a que color se hace referencia, este valor oscila entre 0 y 1 en Matlab siendo el valor para el color azul desde (0.45-0.65), la saturación y el brillo no son de importancia ya que representan al color rojo y verde, y solo nos interesa filtrar el color azul.

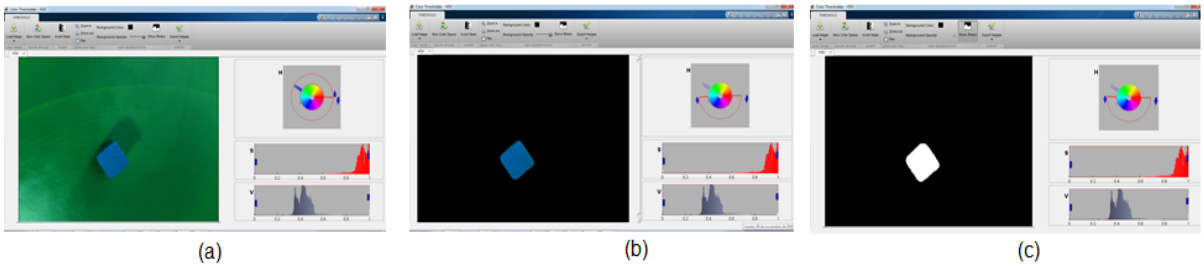


Figura 4.2. (a) Imagen original, (b) Detección color azul, (c) Imagen binarizada

En la Figura 4.3 se observa la imagen real, en la cual se colocó la tapa inferior de un esfero de color rojo sobre el objeto azul. Esta prueba se realizó para comprobar la utilización de la máscara realizada en “Image Processing Toolbox” de Matlab, con la finalidad de obtener la extracción de características del objeto en la imagen.

Figura 4.3. (a) Imagen sin máscara HSV, (b) Imagen con máscara HSV

4.4.2 RECONOCIMIENTO DE IMÁGENES EN FUNCIÓN DE LA ILUMINACIÓN

4.4.2.1 TIPOS DE AMBIENTES DENTRO DE LA VISIÓN ARTIFICIAL

La visión por computadora presenta problemas cuando del ambiente se trata, ya que este depende de factores como: iluminación, reflejo del objeto sobre una superficie, brillo, objetos en la superficie sobre la cual se va a trabajar, cantidad de luz, entre otros. En este proyecto los problemas que se han presentado son los de ambiente ya que aparecen aleatoriamente, este tipo de ambiente se conoce como ambiente no controlado, como se observa en la Figura 4.4.

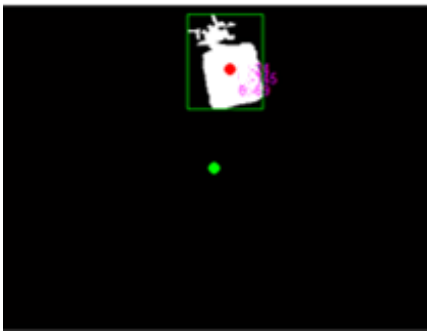


Figura 4.4. Detección con ambiente no controlado

4.4.2.2 Iluminación Directa

La iluminación, cantidad de luz, superficie y sombras son los principales problemas que afectan a los parámetros de tono (H), saturación (S) y brillo (V).

En este proyecto se presentaron los mismos problemas, y se propuso trabajar en un ambiente controlado por iluminación para disminuir las alteraciones en los parámetros tono (H), saturación (S) y brillo (V).

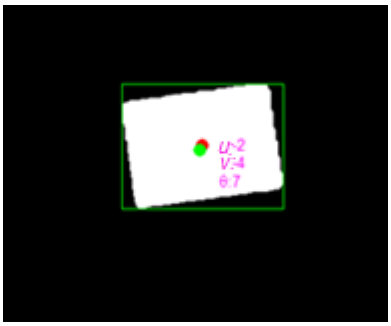


Figura 4.5. Detección con ambiente controlado

La iluminación directa es la más utilizada, ya que la luz apunta directamente al objeto con cierta inclinación respecto al eje de la cámara.

Este método se ocupa cuando el objeto presenta poco reflejo. Dependiendo de la aplicación y objetivo de la adquisición de imagen, se debe aplicar de manera correcta para no crear sombras sobre el objeto o en la superficie [12].

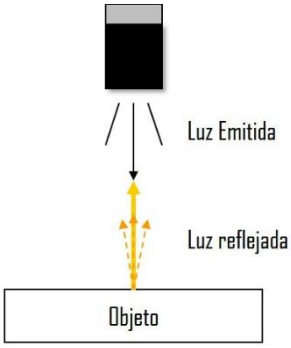


Figura 4.6. Iluminación Directa

La solución al aparecimiento de sombras es colocar sobre la superficie de trabajo una cartulina de color verde que sirve para disminuir las sombras y reflejos del objeto. Esto ocurre porque la cámara de video digital es más sensible al color verde, por lo que se necesita menos luz.

La existencia de ventanas en el entorno de trabajo es otro problema que se presenta en este proyecto debido a la cantidad de luz que incide sobre el objeto a detectar.

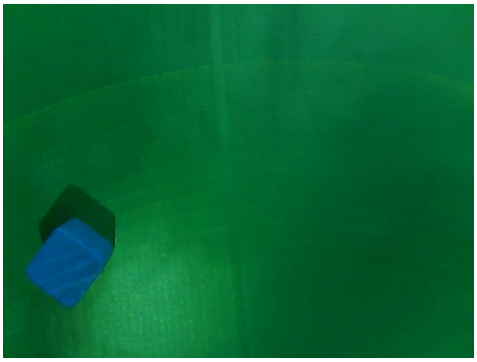


Figura 4.7. Imagen con mucha cantidad de luz

Una posible solución es ajustar los parámetros de la máscara tomando en cuenta la cantidad de luz ingresada en el entorno de trabajo, no se recomienda calibrar los parámetros ya que se podrían obtener errores en la detección.

4.4.3 RECONOCIMIENTO DE CARACTERÍSTICAS DE IMAGEN

En el procesamiento de imágenes se detectó el color, y en este proyecto se tiene la imagen de un rectángulo, en el reconocimiento de características de imagen se usa el comando "*regionprops"* →"*Centroid"* de Matlab.

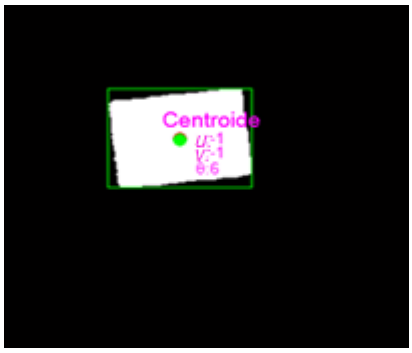


Figura 4.8. Coordenadas del centro de la imagen

En la Figura 4.8 se aplica la propiedad "*Centroid"* para encontrar las coordenadas del centro del objeto detectado que es el centro de la imagen.

4.4.4 ÁNGULO DE ORIENTACIÓN

Para determinar el ángulo de orientación del efector final, se consideró que la cámara está ubicada en la pinza, presentándose la configuración cámara en mano.

Para determinar el ángulo de orientación del efector final del robot SCARA, se usó la propiedad "*Orientation"* del comando "*regionprops"* de Matlab, como se explica en el (Capítulo 2, sección 2.3.5.3).

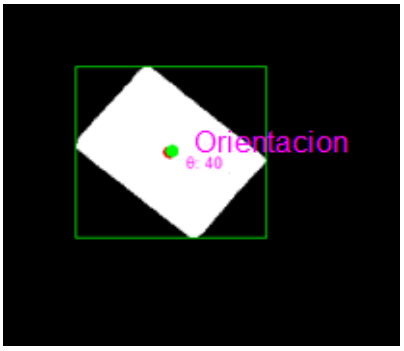


Figura 4.9. Ángulo de orientación de la imagen

En la Figura 4.9 se muestra la orientación del objeto dentro de la imagen, la medición del ángulo de orientación realizada en un experimento de 3 pruebas

presenta una desviación estándar de $\pm\ 0.002°$, como se puede observar en la Tabla 4.1, por lo que no se necesita realizar algún ajuste.

Tabla 4.1. Ángulos de Orientación de la imagen

Ángulo orientación plano real (grados)	Ángulo orientación plano de imagen (grados)	Error absoluto	Error relativo (%)
70	69.44	0. 008	0.8
90	89.19	0.009	0.9
42	41.92	0.0019	0.19

4.2 DESEMPEÑO DEL CONTROLADOR SERVO VISUAL BASADO EN LAS CARACTERÍSTICAS DE LA IMAGEN.

4.2.1 PARÁMETROS DEL CONTROLADOR SERVO VISUAL

Las constantes de calibración propuestas para el controlador servo visual se muestran en la Tabla 4.2, que tienen la ecuación descrita en el (Capítulo 2, sección 2.2.1).

$$U = J^{-1}(\dot{\xi}_d + K_P \tanh(k_2 \tilde{\xi}))$$

Tabla 4.2. Constantes de calibración para el controlador Servo visual

CONSTANTE	VALOR
K_p	[-1 0 ; 0 -1]
k_2	[0.00005 0 ; 0 0.00001]

Los valores de las constantes propuestas en la Tabla 4.2 se obtuvieron realizando pruebas de funcionamiento y obteniendo el mejor desempeño del controlador servo visual.

4.2.2 DESEMPEÑO DEL ROBOT MANIPULADOR TIPO SCARA

El desempeño del robot manipulador SCARA como planta, se la obtiene con la medición de la repetibilidad de una tarea asignada, es decir la capacidad que el robot tiene para regresar a una posición establecida el número de veces que sean necesarias, los mismos que deben ser similares.

En este proyecto para determinar la repetibilidad del robot manipulador SCARA, se realiza la tarea de posicionamiento hacia un punto perteneciente dentro del área de trabajo del robot, partiendo de la posición inicial como se explicó en el (Capítulo 2, sección 2.7.1), como se puede observar en la Tabla 4.3.

Los datos mostrados en la Tabla 4.3 representan las posiciones enviadas al robot para cada uno de los ejes, posiciones pertenecientes al área de trabajo del robot SCARA definida en el (Capítulo 1, sección 1.2.1.2).

Los cálculos del valor promedio de la desviación estándar para cada uno de las coordenadas de las posiciones finales del robot, permitirán establecer la repetibilidad del robot.

La cantidad de datos de las pruebas realizadas con el mismo número de muestras obtenidas de todas las posiciones establecidas en el área de trabajo servirán para aplicar las ecuaciones (1.21) y (1.24) para cada coordenada.

Los resultados de la desviación estándar mostrados en la Tabla 4.3 describen un valor de desviación estándar de 0,9 centímetros para eje $\boldsymbol{x}$ y 0,15 centímetros para el eje $\boldsymbol{y}$, lo que demuestra que el robot presenta una repetibilidad de seis veces la desviación estándar, es decir, 2,1 centímetros, para asegurar que el robot tenga la probabilidad más alta de cumplir la tarea asignada.

Tabla 4.3. Datos de posiciones del robot SCARA

Posición Deseada eje x (m)	Posición medida eje x (m)	$x_i - \bar{x}$	Posición deseada eje y (m)	Posición medida eje y (m)	$y_i - \bar{y}$
-0,15	-0,1414	-0,208	-0,15	-0,12	-0,405
-0,1	-0,096	-0,175	-0,1	-0,087	-0,372
-0,05	-0,0506	-0,113	-0,05	-0,054	-0,339
0	-0,0052	-0,108	0	-0,021	-0,306
0,05	0,049	-0,075	0,05	0,032	-0,273
0,1	0,098	0,022	0,1	0,85	-0,24
0,15	0,147	0,26	0,15	0,18	-0,207
0,2	0,2	0,172	0,2	0,19	-0,174
0,25	0,245	0,190	0,25	0,244	-0,141
0,3	0,294	0,224	0,3	0,277	-0,108

4.2.3 ÁREA DE TRABAJO DE LA CÁMARA CON EL ÁREA DE TRABAJO DEL ROBOT SCARA

En este proyecto se trabajar con el área de visión de la cámara que representa la resolución de la imagen, en este caso, 640x480 pixeles, para fines didácticos se coloca el área de trabajo de la cámara sobre el área de trabajo del robot SCARA, visto desde la cámara, como se observa en la Figura 4.10.

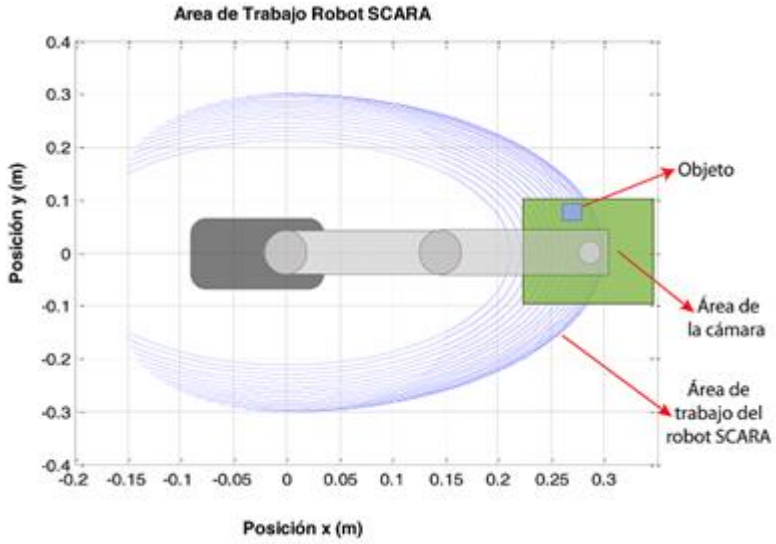


Figura 4.10. Área de trabajo robot con área de trabajo de la cámara

La Figura 4.11 representa el área de trabajo del robot en conjunto con la cámara, el área de visión de la cámara es indispensable para la ubicación del objeto (rectángulo de color azul) ya que este puede estar fuera de la ventana de visión y el robot no tendrá ningún punto al cual posicionarse.

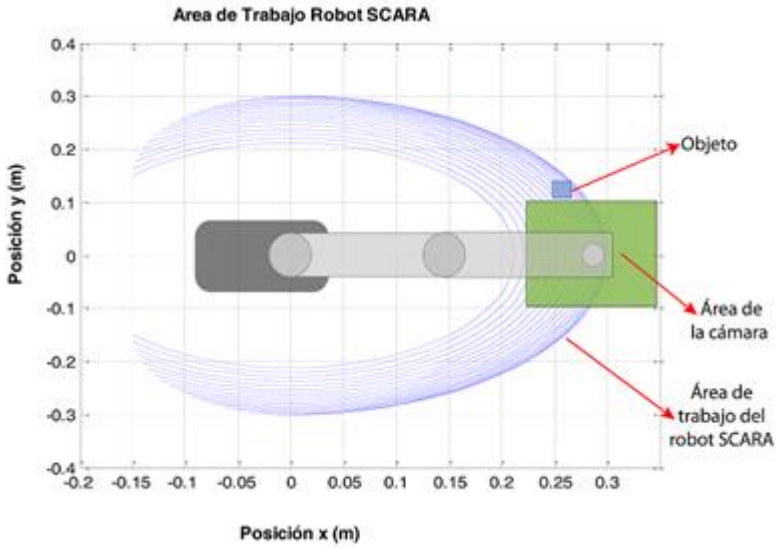


Figura 4.11. Objeto fuera del área de visión

El robot SACARA tiene una rutina de inicialización de brazo extendido, como se menciona en el (Capítulo 2, sección 2.7.1), es decir, la cámara tendrá la posición del efector final $P_i = [P_x = 30, P_y = 0, P_z = 12]$, pues ahora si enviamos al robot una posición cualquiera de su área de trabajo, la cámara tendrá la posición del efector

final, por ejemplo $P_i = [P_x = 0{,}26, P_y = 0{,}13, P_z = 0{,}12]$, en donde el área de visión de la cámara se reflejará sobre el área de trabajo del robot como se observa en la Figura 4.12

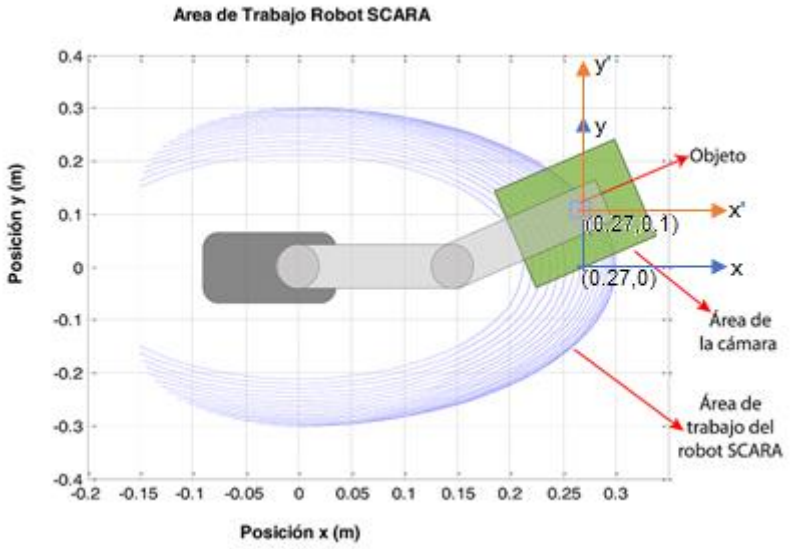


Figura 4.12. Posición de la cámara

4.3 RESULTADOS CONTROL SERVO VISUAL BASADO EN LAS CARACTERÍSTICAS DE IMAGEN

En la Figura 4.13 las características visuales del objeto, medidas en pixeles, son utilizadas directamente en la realimentación del esquema de control de forma que el movimiento deseado del robot se obtiene haciendo converger las características actuales (u,v) a las deseadas (u_d, v_d).

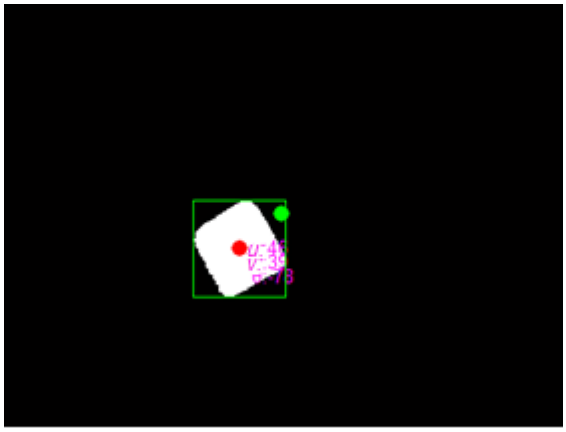


Figura 4.13. Características del objeto (punto rojo) y características deseadas (punto verde)

Se utiliza el control servo visual basado en las características de la imagen para controlar el robot de forma que las características visuales adquiridas en cada instante de muestreo, como se observa en la Figura 4.14, coincidan finalmente en las características deseadas.

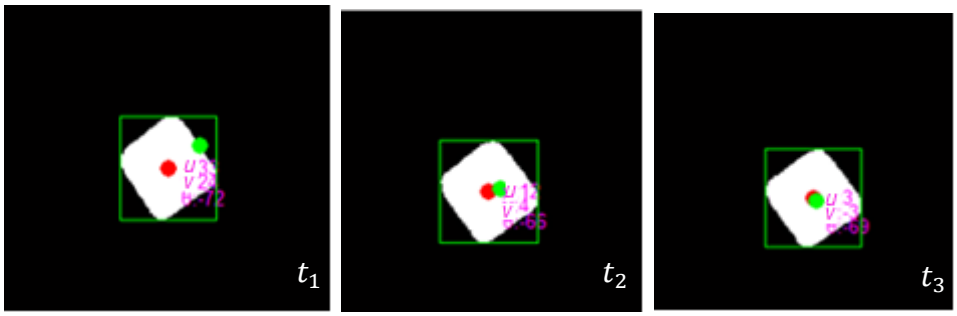


Figura 4.14. Evolución de las características en la imagen

En la Figura 4.15 se representa la evolución del movimiento deseado del robot cuando converge desde las características actuales a las deseadas.

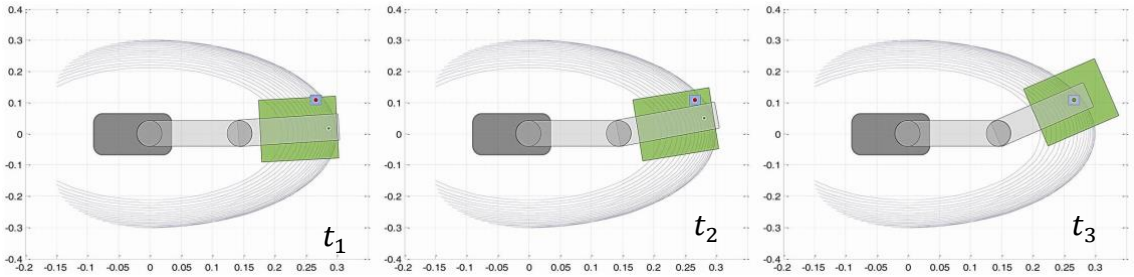


Figura 4.15. Evolución del movimiento del robot alcanzando las características deseadas

En la Figura 4.16, se muestra la trayectoria de la cámara en la que se puede apreciar como la cámara alcanza la posición final deseada, en este caso se propuso la posición deseada el centro de la imagen $(0,0)$ pixeles.

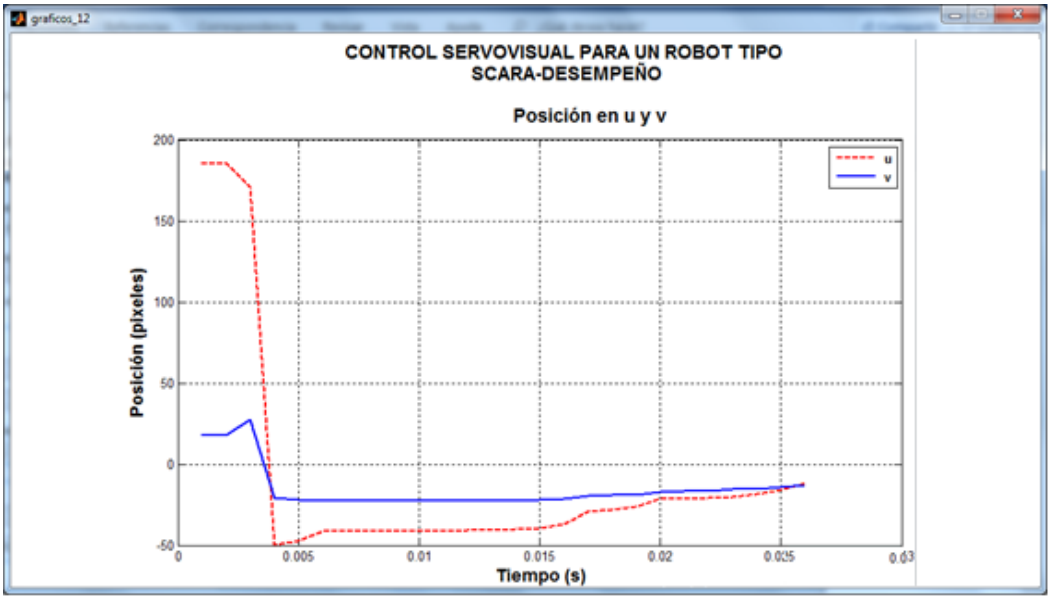


Figura 4.16. Posición de la cámara en el plano de imagen $(\mathbf{u}, \mathbf{v})$

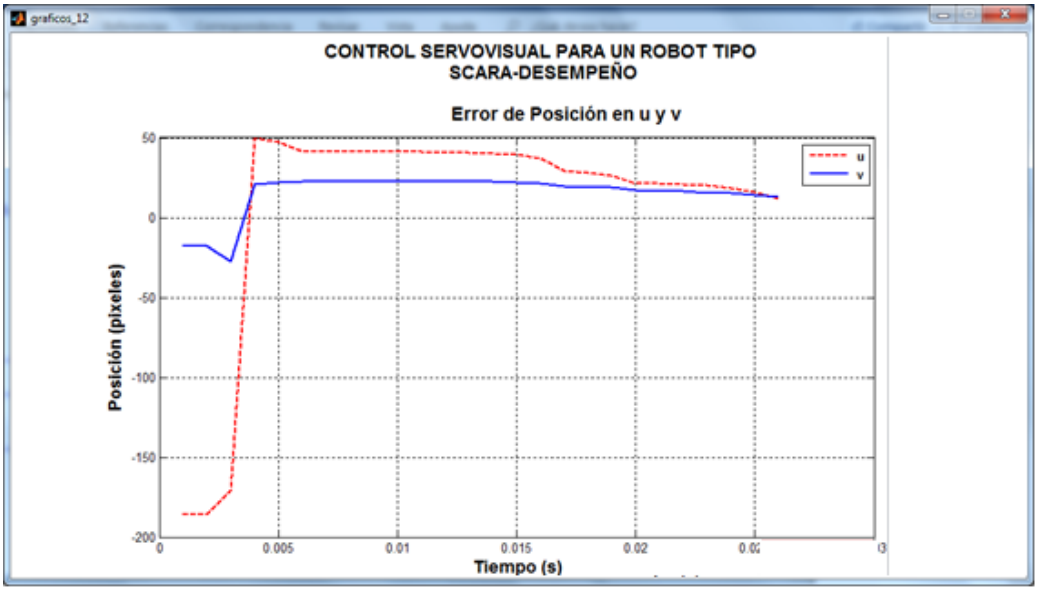


Figura 4.17. Error de posición en el plano de imagen $(\mathbf{u}, \mathbf{v})$

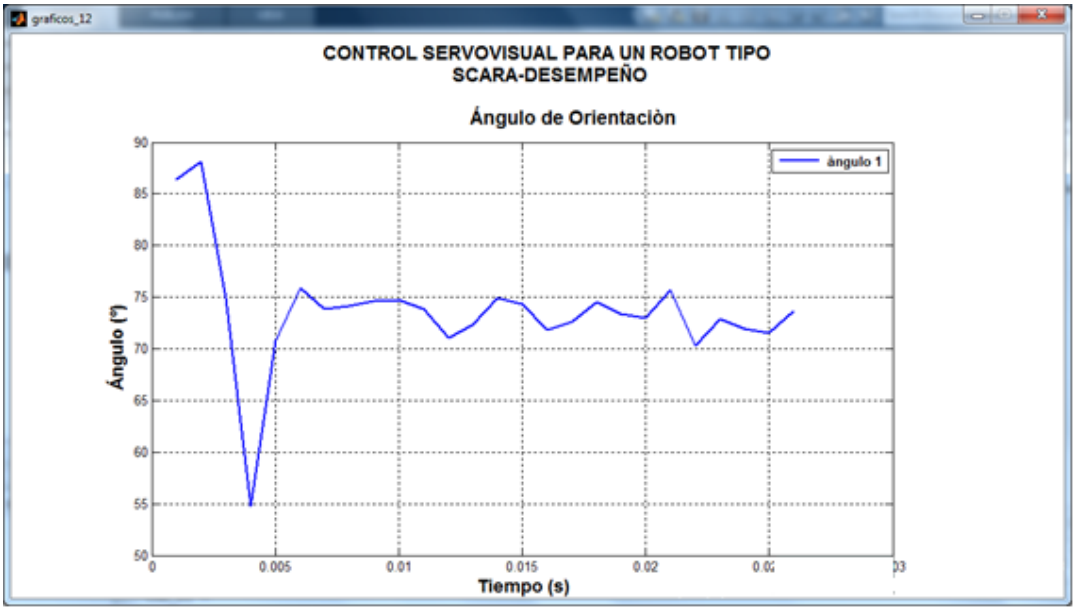


Figura 4.18. Angulo de Orientación

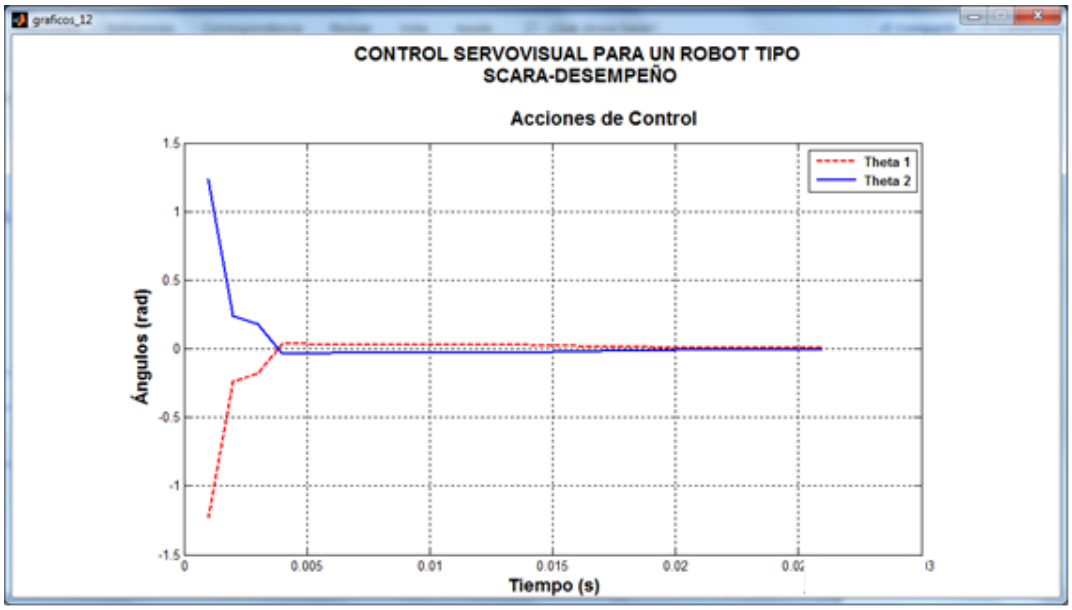


Figura 4.19. Acciones de control

En la tarea del control servo visual el hecho que las características visuales consideradas converjan a sus valores deseados como se observa en la Figura 4.17, no asegura que la posición final de la cámara coincida con la posición final deseada debido a la posible existencia de errores de calibración. De los datos obtenidos en

la implementación se puede indicar que el error de orientación máximo es de 0,3 grados como se observa en la Figura 4.18 y el de posición máxima para el eje $u =$ 0.29 pixeles y para el eje $v =$0.32 pixeles como se observa en la Figura 4.17, demostrando que el controlador diseñado cumple con todas las condiciones haciendo que el robot no pierda información de la imagen y logre posicionarse sin dificultades.

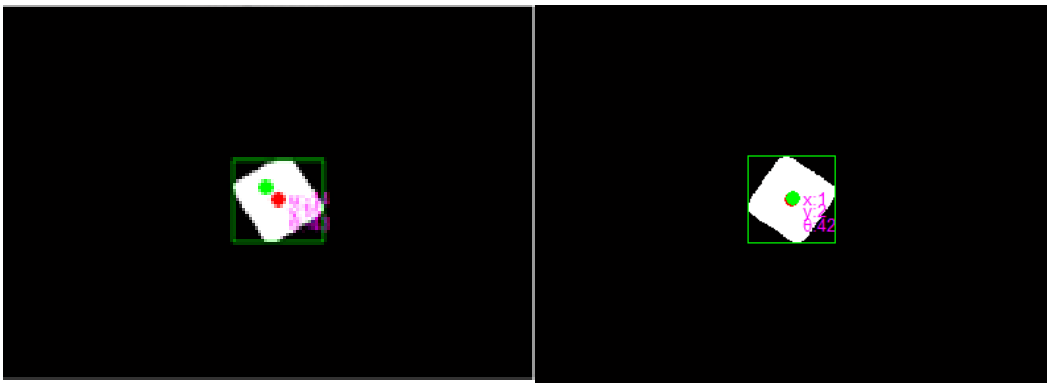


Figura 4.20. Control servo visual sobre un objeto rotado 42 grados

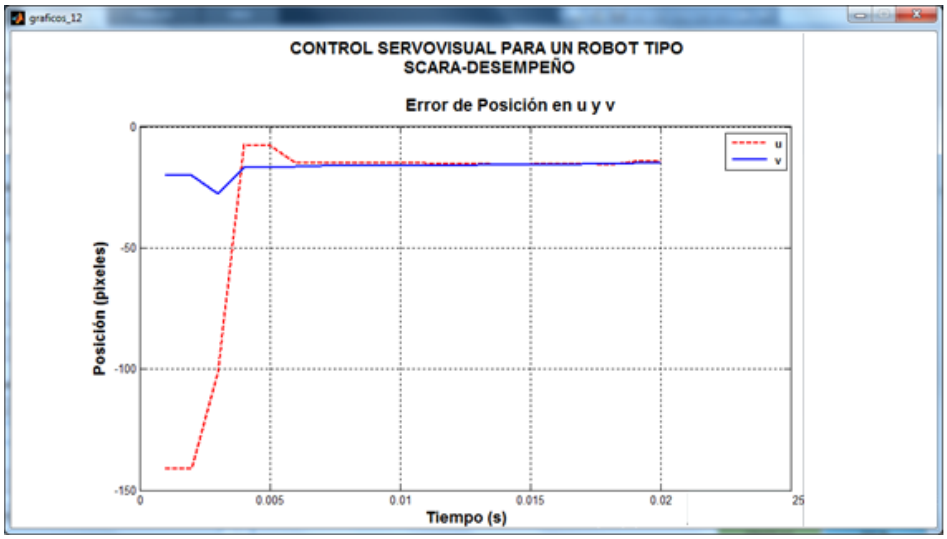


Figura 4.21.Error de posición para un objeto rotado 42 grados

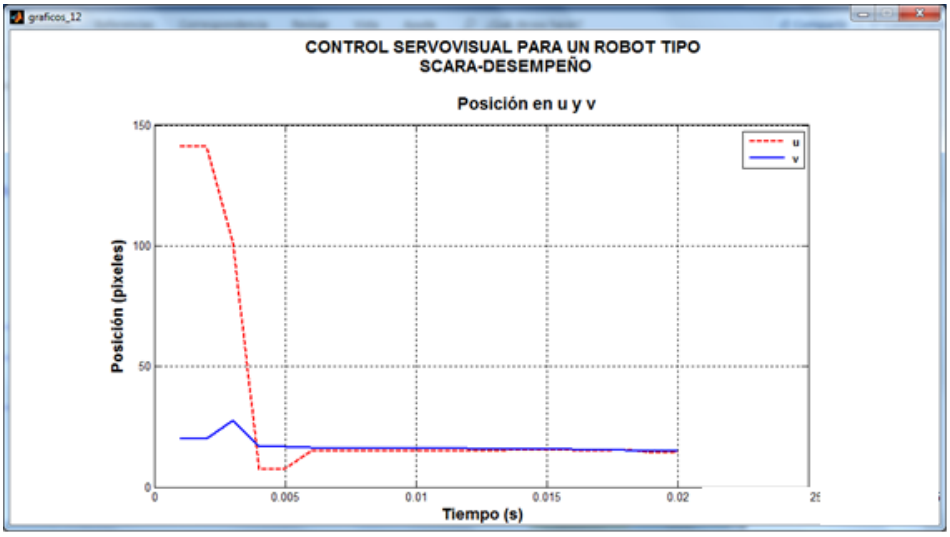


Figura 4.22. Posición para un objeto rotado 42 grados

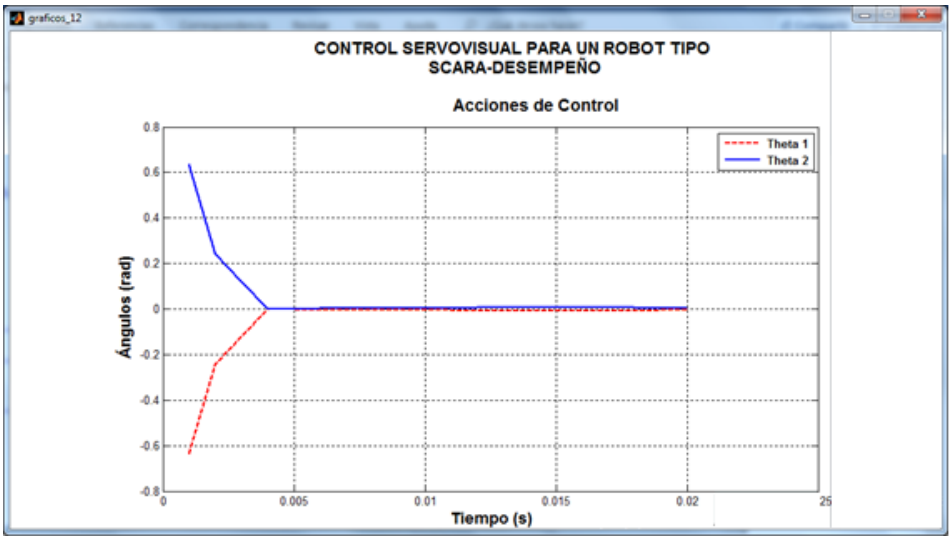


Figura 4.23. Acciones de control para un objeto rotado 42 grados

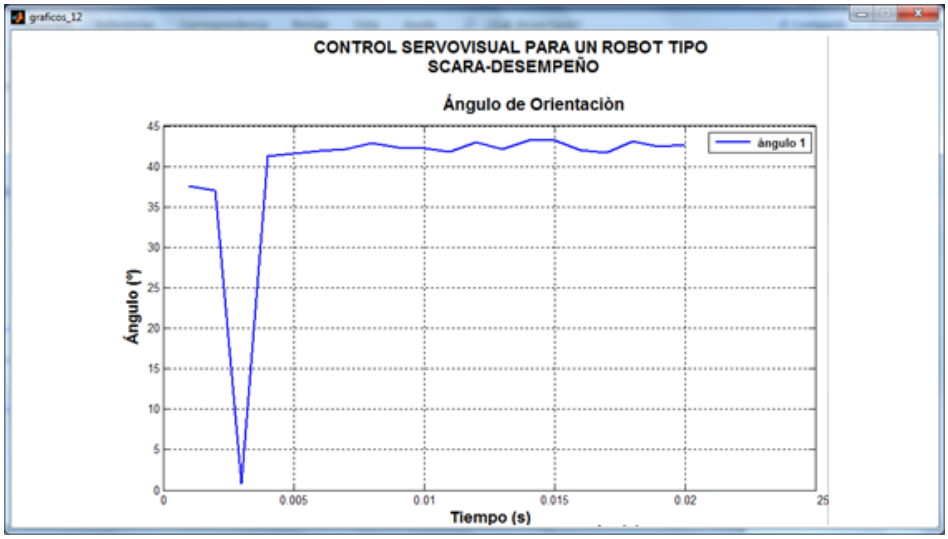


Figura 4.24. Ángulo de orientación para un objeto rotado 42 grados

4.4 COSTOS DE IMPLEMENTACIÓN

En la implementación del control servo visual, el robot manipulador SCARA tuvo costos de restauración y costos de ingeniería. Los costos detallados se observan en la Tabla 4.4.

Tabla 4.4. Costos de Implementación

Cantidad	Descripción	Precio unitario	Total
1	Placa de Control para el Manipulador	60.00	60.00
1	Web Genius Facecam 320x	11.50	11.50
1	Conector db25	2.50	2.50
2	Microcontroladores Atmega 164p	4.50	9.00
1	Metro de cable	0.50	0.50
1	Bornera 2 pines	0.80	0.80
400	Horas de Ingeniería	7.00	2800.0
	Total		2884.30

CAPÍTULO 5

CONCLUSIONES Y RECOMENDACIONES

5.1. CONCLUSIONES

- Se estudió el funcionamiento del robot manipulador SCARA disponible en el Laboratorio de Sistemas de Control, lo cual, permitió determinar el estado actual del robot, y mediante este estudio realizar el diseño del hardware y software necesario para la implementación de los algoritmos diseñados.
- Se estudió y determinó el modelo cinemático del robot manipulador tipo SCARA, una vez obtenido el modelo matemático se realizaron pruebas de simulación, concluyendo que el modelo simulado es similar al modelo del robot real.
- Se estudió e implemento una técnica de reconocimiento de objetos por color usando el espacio de color HSV en el programa computacional Matlab, para crear una máscara de imagen del objeto deseado y facilitar el procesamiento, se verifico mediante pruebas experimentales el correcto funcionamiento de la técnica de reconocimiento de imágenes.
- Se diseñó e implementó un controlador servo visual basado en las características de la imagen, empleando la técnica de cámara en mano en un robot manipulador tipo SCARA, se observó el buen desempeño del controlador, mediante pruebas experimentales donde el efector final del robot logró posicionarse y orientarse sobre el objeto deseado.
- Se diseñó una interfaz gráfica para el control servo visual del robot manipulador SCARA usando el editor de diseño GUIDE- Matlab, el cual permitió realizar la simulación y las pruebas experimentales de los algoritmos diseñados.
- Se realizaron pruebas de simulación y experimentales para evaluar el rendimiento del controlador implementado, logrando obtener un error de hasta 10 píxeles, dependiendo de la posición y del ángulo de rotación de la imagen

5.2. RECOMENDACIONES

- Se recomienda que en aplicaciones de visión por computador se procure la utilización de cámaras de alta definición cuando el sistema de visión requiera un mayor procesamiento o detalle.
- Se recomienda realizar un estudio del ambiente de trabajo para aplicar técnicas de visión, existen muchos problemas con la iluminación cuando se trata de la adquisición de imágenes, puede ingresar ruido en el momento del procesamiento de imágenes.
- Se recomienda utilizar un fondo oscuro en el área de trabajo, esto con la finalidad de disminuir la velocidad de procesamiento de imagen, así como también el tamaño de la imagen. Esto permite que los pixeles de las matrices formadas sean pequeñas y se facilite su uso.

REFERENCIAS BIBLIOGRÁFICAS

[1] A. Barrientos, L. Peñín, C. Balaguer, R. Aracil, “Fundamentos de Robótica”, 2ªed, Madrid, McGraw-Hill, 2007, pp. 17-18.

[2] A. Ollero B., “Robótica, Manipuladores y Robots Móviles”, España, Editorial Marcombo, 2001, pp. 19-25.

[3] Y. Boada A., L. Morales, “Construcción, Ensamblaje y Control de un Robot de Tres Grados de Libertad Tipo SCARA (SELECTIVE COMPLIANCE ASSEMBLY ROBOT ARM)”, Quito, mayo 2010.

[4] P. Leica, Desarrollo de Algoritmos de Inteligencia Artificial Aplicado a Sistemas Distribuidos Robotizados. Escuela Politécnica Nacional. Quito. Febrero 2008**.**

[5] I. Barranco, “Manejo de color”, Metodología para un sistema de visión artificial, 2001. [Online]. Available: http://posgrado.itlp.edu.mx/barranco/VisionCurso/color.pdf.

[6] S. Fernando “Taller de robótica y visión para estudios de grado”, Master, Centro internacional de postgrado, Universidad de Oviedo, 2013. [Online]. Available: http://digibuo.uniovi.es/dspace/bitstream/10651/18286/11/TFM%20-%20Fernando%29Soto.pdf.

[7] Universidad Politécnica de Madrid, Apuntes de visión artificial. [Online]. Available: http://www.elai.upm.es/moodle/course/view.php?id=37.

[8] Universidad Politécnica de Madrid, Apuntes de visión artificial. [Online]. Available: http://www.elai.upm.es/moodle/mod/resource/view.php?id=445

[9] Departamento de Educación, Universidades e Investigación, “Aplicación práctica de la visión artificial en el control de procesos industriales”, Ministerio de Educación, Gobierno de España, Unión Europea. [Online]. Available: http://visionartificial.fpcat.cat/wp-content/uploads/UD_1_didac_Conceptos_previos.pdf.

[10] I. García, “Visión Artificial y Procesamiento Digital de Imágenes usando Matlab”. [Online]. Available: https://vdocuments.mx/vision-artificial-y-procesamiento-digital-de-imagenes-usando-matlab-561ee4a542ecc.html

[11] J. M. Artigas, P. Capilla, “Tecnología del color”. [Online]. Available: http://www2.elo.utfsm.cl/~elo328/pdf1dpp/PDI13_Color_1dpp.pdf

[12] A. Guayasamin, “Diseño y simulación de algoritmos de control tipo PID y SMC para seguimiento de trayectoria de un manipulador aéreo”. Escuela Politécnica Nacional. Quito. Agosto 2018**.**

[13] B. López. M.; F. Roberti; A. Pérez: Control Servo-Visual de Robots Manipuladores un enfoque hacia la industria, Revista Épsilon, NO. 19, julio-diciembre 2012.

[14] INFAIMON. (2014). Iluminación. [Online]. Available: http://www.crit.upc.edu/JCEE2005/pdf_ponencies/Presentacio_Infaimon.pdf.

[15] M. Trujillo. (2010) Distribución Normal [Online]. Available: https://jesusgarciaj.com/2010/01/22/la-curva-de-distribucion-normal/

ANEXO A

MANUAL DE USUARIO

GUÍA DE INSTALACIÓN DE HADWARE

Colocar la lona o cartulina de color verde sobre una superficie sólida, puede ser una mesa de escritorio.

Ubicar la caja verde sobre la lona o cartulina verde tomando en cuenta el centro del área de trabajo trazada en la lona, como se observa en la Figura A.1

Colocar el robot SCARA sobre la caja verde, posicionando de forma manual al robot con el brazo extendido $P_i = [P_x = 30, P_y = 0, P_z = 12]$ siguiendo las marcas que se encuentran en la caja, como se observa en la Figura A.1.

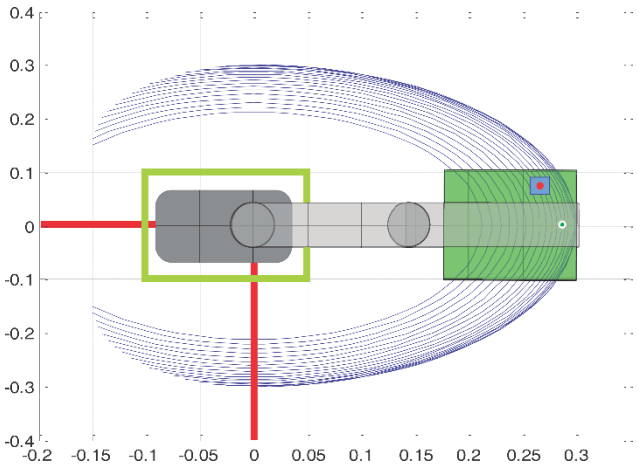


Figura A.1.1 Ubicación del robot SCARA

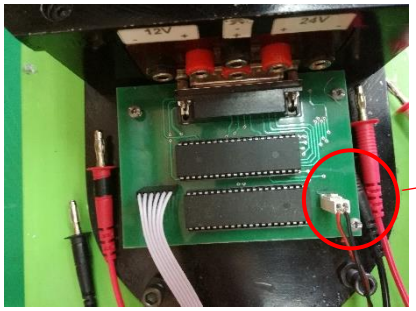
A.2

Posicionar el objeto sobre el área de trabajo dibujada en la lona o cartulina verde, como se observa en la Figura A.2.

Figura A.2. Posición del objeto sobre el espacio de trabajo

A.3

Utilizar una fuente DC para energizar el robot SCARA con los valores marcados en la placa de control, recordar que la placa de control tiene una alimentación de 5 voltios, como se observa en la Figura A.3a.



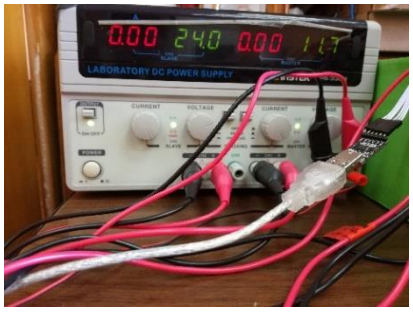
Alimentación microcontroladores

(a)

Alimentación Robot SCARA

-12 V+	-5 V +	-24 V+

(b)



(c)

Figura A.3. (a)Conector de alimentación micro controladores Fuente DC; (b) Etiquetas de información; (c) Fuente DC.

A.4

Conectar la extensión USB de la cámara al computador.

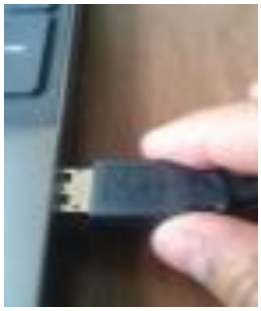


Figura A.4. Cable de extensión USB de la cámara

A.5

Conectar el módulo de comunicación USB serial de la placa de control a la extensión USB del computador.

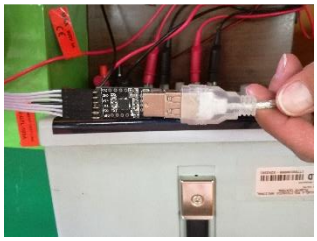


Figura A.5. Cable connexion USB serial

ANEXO B
GUÍA DE INSTALACIÓN DE SOFTWARE

B.1 PROGRAMA

El programa base que debe instalarse en el computador para este proyecto es: Matlab 2014a.

B.2 INSTALACIÓN DEL PROGRAMA

Copiar la carpeta “CONTROL_SERVOVISUAL” que se encuentra en la unidad de almacenamiento al Disco local (C:) del computador.

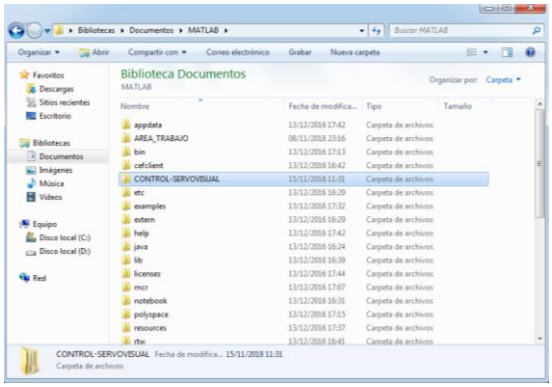


Figura B.1. Ubicación de la carpeta “CONTROL_SERVOVISUAL”

B.3 UBICACIÓN DE LA INTERFAZ GRAFICA

Ya ubicado el programa en el Disco local (C:), abrir la carpeta “CONTROL_SERVOVISUAL”→“CONTROL_SERVOVISUAL.hmi”→HMI_CONTROL_SERVO_VISUAL.m

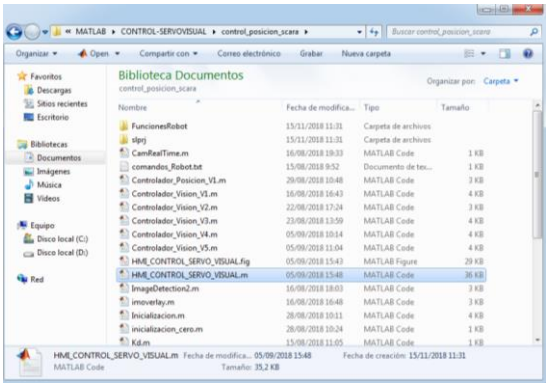


Figura B.2. Ubicación de la interfaz grafica

ANEXO C
MANUAL DE USUARIO DE LA INTERFAZ GRÁFICA HMI

La interfaz HMI_CONTROL_SERVO_VISUAL.m, permite al usuario realizar pruebas en lazo abierto, control en lazo cerrado y observar las gráficas de resultados.

C.1 COMUNICACIÓN COMPUTADORA- ROBOT

Para el uso de la interfaz gráfica, en primer lugar, se tiene el panel COMUNICACIÓN_PC_ROBOT mostrada en la Figura C.1, seleccionar el serial port y escoger el COM adecuado, una vez que se elige se debe presionar el botón conectar, se recibe el mensaje conectado. Se presiona el botón home para enviar al robot a una posición para dar inicio a la rutina de inicialización presionando el botón inicio.

Figura C.1. Serial Port

C.2 CONTROL LAZO ABIERTO

Una vez iniciada la tarea computador y micro controlador se procede a seleccionar una de las ventanas en las que se desea trabajar, se presentan dos opciones LAZO ABIERTO Y CONTROL EN LAZO CERRADO.

En el caso de que se elija LAZO ABIERTO, operar en forma manual al robot, el usuario deberá usar los botos sliders para el manejo de los motores y ángulos de rotación de las articulaciones, como se muestra en la Figura C.2a. Para el envío de datos, presionar ENVIAR.

Figura C.2a Lazo Abierto

Las sliders operan las articulaciones, variando el ancho de pulso de la PWM del M1 y M2.

Ambas sliders varían entre -90° y 90° y sus valores se observan en “Posiciones Angulares”.

PINZA

La ventana PINZA tiene las opciones de abrir y cerrar, tiene una slider para rotar la pinza en un intervalo de 0° a 90°, como se observa en la Figura C.2b.

Figura C.2b. Abrir y Cerrar Pinza, Rotación

C.3 Control Lazo Cerrado

En el caso que se elija CONTROL EN LAZO CERRADO, como se ve en la Figura C.3a.

Figura C.3a Control lazo cerrado.

Para el uso de la interfaz gráfica CONTROL LAZO CERRADO, en primer lugar se debe seleccionar la cámara: FaceCam y WebCam, como se observa en la **¡Error! o se encuentra el origen de la referencia.**C.3b, luego se presiona el botón EMPEZAR donde se visualiza un mensaje CARGANDO VIDEO.

Figura C.3b. Selección de la cámara.

Si el usuario desea capturar esta imagen, se debe presionar el botón CAPTURA ubicado en la parte media. Luego, aparece en la zona Objeto la imagen fragmentada y en la esquina inferior izquierda aparecen los valores de las características del objeto, como se muestra en la Figura C.3c.

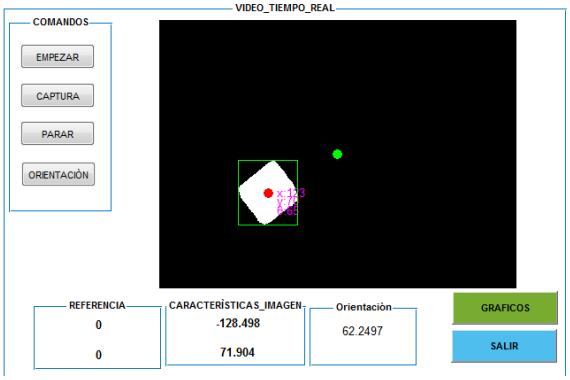


Figura C.3c. CAPTURA de imagen

Terminado el procesamiento de imagen, los valores mostrados en uc, vc y Orientación se envían al controlador, y desde luego, al robot.

En la Figura C.3d, se observa en el panel AREA_DE_TRABAJO, la gráfica indica en tiempo real la posición del robot dentro de su área de trabajo, el punto rojo representa al robot mientras que la zona azul el área de trabajo del robot SCARA.

Figura C.3d.ÁREA_DE_TRABAJO.

C4 GRÁFICAS DE RESPUESTA

La Figura C.4 muestra las curvas de posición en el plano XY, el error producido en la posición de la pinza y los ángulos de rotación de las articulaciones del robot SCARA. Para visualizar cada una de éstas, se debe presionar la opción requerida. En esta pantalla se puede observar el valor del índice de rendimiento ISE producido para este controlador, ubicado en la parte inferior izquierda.

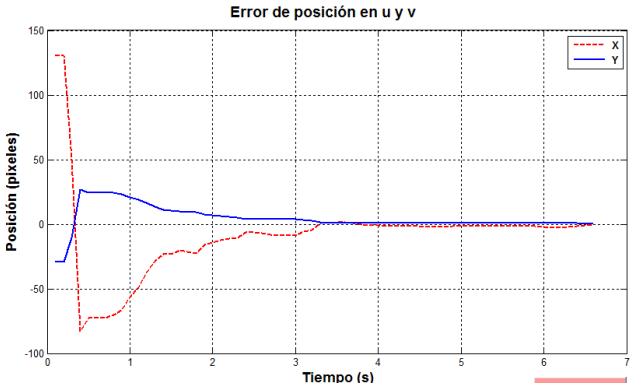


Figura C.4. Pantalla ERRORES.

Se debe seleccionar cualquier de las opciones a continuación mencionadas:

Posición: muestra la gráfica de la posición en el plano de imagen (u, v).

Error Posición: muestra la gráfica del error de posición en el plano de imagen(u, v).

ISE: muestra el valor del índice de desempeño del controlador.

ANEXO D

Configuración de pines del controlador ubicados en la placa de control.

Tabla D.1. Configuración de pines

PIN	NOMBRE	ENTRADA/ SALIDA	DESCRIPCION
40	A0	pin 21	Final de carrera antihorario motor 3
39	A1	pin 20	Final de carrera horario motor 3
38	A2	pin3	Final de carrera antihorario motor 2
37	A3	pin 2	Final de carrera horario motor 2
36	A4	pin 5	Final de carrera antihorario motor 1
35	A5	pin 4	Final de carrera horario motor 1
34	A6	pin 11	Sentido horario motor 1
33	A7	pin 10	Sentido antihorario motor 1
1	B0	pin 22	Arriba motor 3 - SALIDA
2	B1	pin 23	Abajo motor 3 - SALIDA
3	B2	pin 19	Encoder motor 3
4	B3	pin 8	Sentido antihorario motor 2
5	B4	pin 18	Velocidad motor 3
6	B5	pin 9	Sentido horario motor 2
7	B6	pin 22	Arriba motor 3 - ENTRADA
8	B7	pin 23	Abajo motor 3 - ENTRADA
22	C0	micro 2	Bit 0 ángulo pinza

23	C1	micro 2	Bit 1 ángulo pinza
24	C2	micro 2	Bit 2 ángulo pinza
25	C3	micro 2	Bit 3 ángulo pinza
26	C4	micro 2	Bit 4 ángulo pinza
27	C5	micro 2	Bit 5 ángulo pinza
28	C6	micro 2	Bit 6 ángulo pinza
29	C7	micro 2	Abrir/cerrar pinza
14	D0	TX adaptador	RX
15	D1	RX adaptador	TX
16	D2	pin 6	Encoder canal A motor 1
17	D3	pin 25	Encoder canal A motor 2
18	D4	pin 12	Velocidad motor 2
19	D5	pin 13	Velocidad motor 1
20	D6	pin 7	Encoder canal B motor 1
21	D7	pin 24	Encoder canal B motor 2
		pin 15	Negativo
		pin 14	Negativo
		pin 1	Libre
		pin 17	Girar pinza micro 2
		pin 16	Abrir pinza micro 2

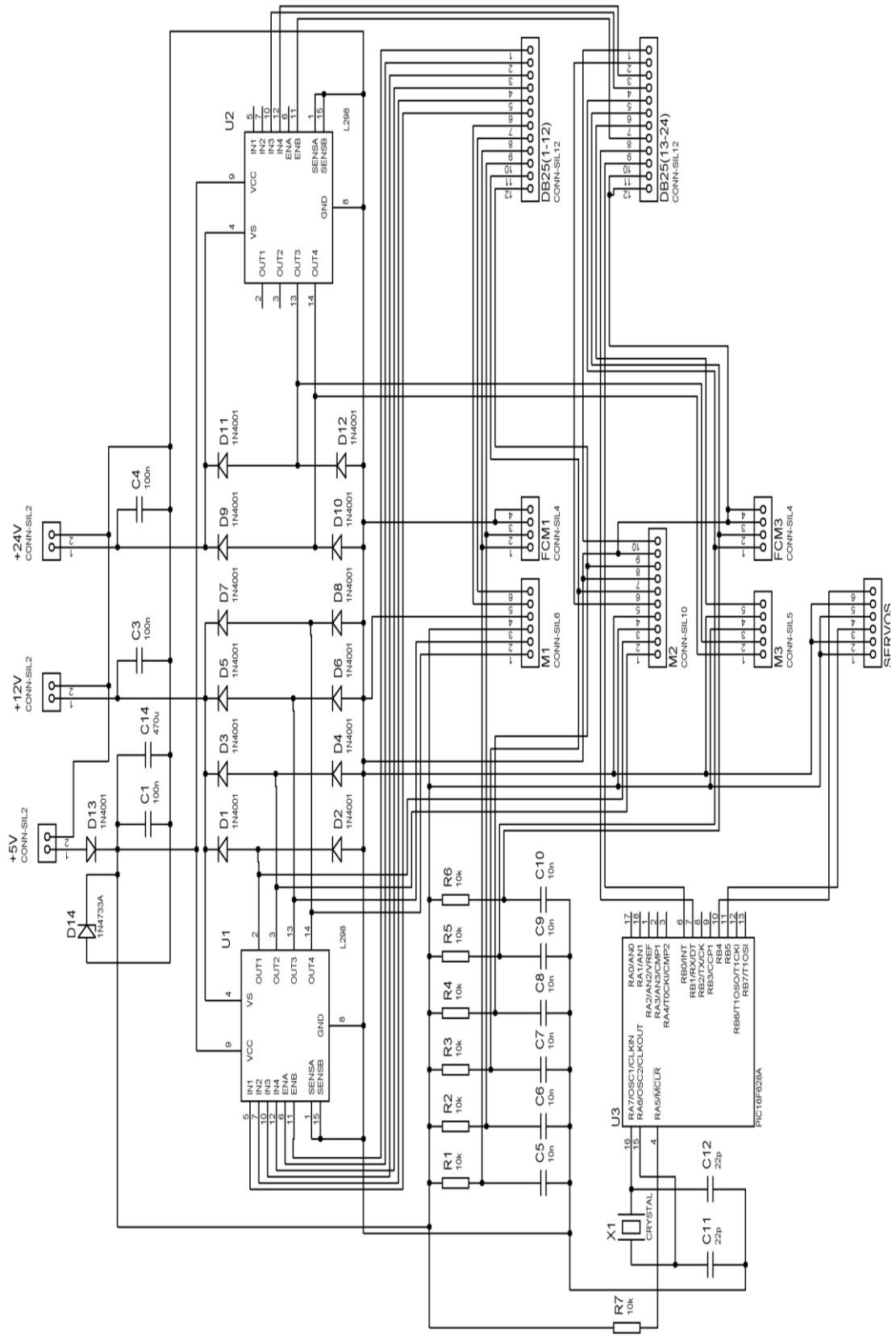


Figura D.1 Esquemático placa_1 para el control del robot SCARA [3]

Tabla D.2 Listado de elementos para PCB_1 [3]

CANT.	REFERENCIA	DESCRIPCIÓN
3	+5V, +12V, +24V	Conector para fuentes de poder
3	C1,C3,C4	Capacitor 100nF
6	C5,…,C10	Capacitor 10nF
12	D1,…,D12	Diodos Schotty SD101B
2	U1,U2	Driver L298N
7	R1,…,R7	Resistencias 10KΩ, ¼W
2	C11,C12	Capacitores 22pF
1	C14	Capacitor 470uF
1	X1	Cristal 4MHz
1	U3	Microcontrolador 16F628A
3	FCM1, FCM2 y FCM3	Conector finales de carrera para motores M1, M2 y M3 respectivamente
2	M1, SERVOS	Conector para alimentación y líneas de control de motores M1 y servomotores de pinza respectivamente
1	M2	Conector para alimentación y líneas de control de motor M2
1	M3	Conector para alimentación y líneas de control de motor M3
2	DB25(1-12) DB25(13-24)	Conector para acceso a variables del robot SCARA

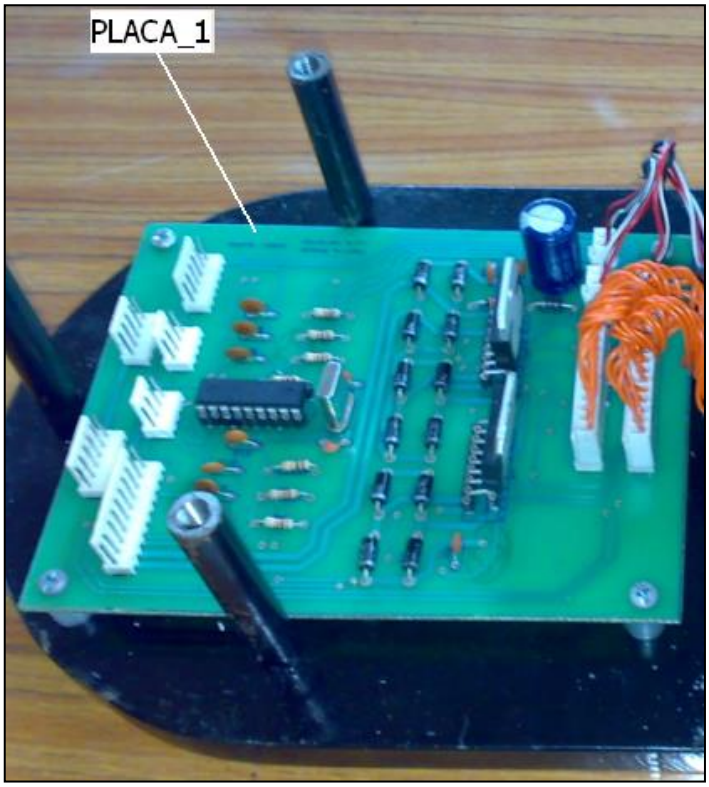


Figura D.2 Tarjeta PCB_1 construida [3]

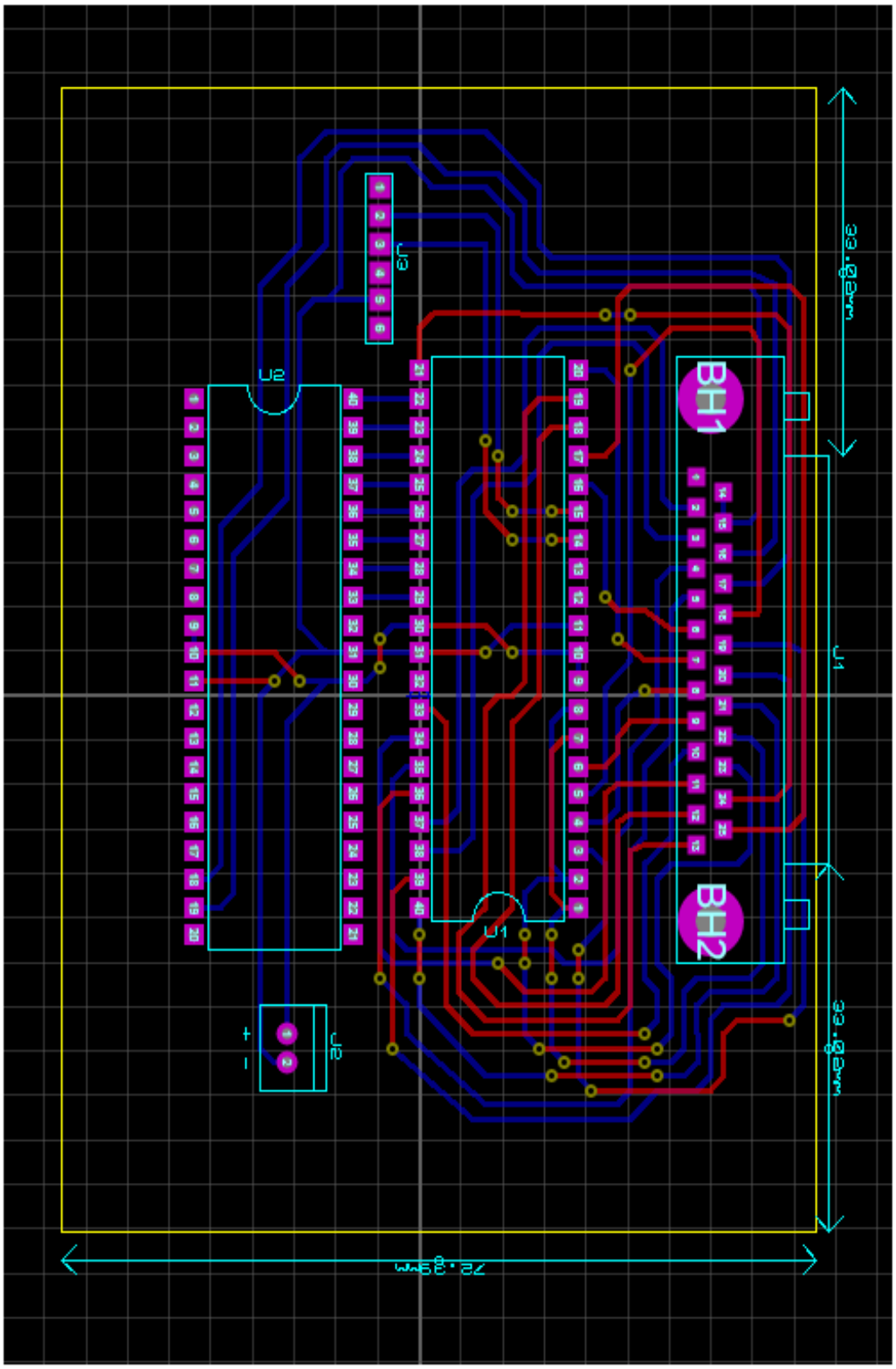


Figura D.3. ARES: Ubicación de los elementos de la placa de control