

**SISTEMA ROBÓTICO DE BIN PICKING 3D GUIADO POR VISIÓN PARA EL
COBOT BAXTER**

LUIS DAVID SAAVEDRA GRIMALDO

Trabajo de grado para optar al título de

INGENIERÍA MECATRÓNICA

Gustavo Moreno Hincapié

Magíster en Ingeniería Biomédica



UNIVERSIDAD EIA

INGENIERÍA MECATRÓNICA

ENVIGADO

2019

CONTENIDO

	pág.
INTRODUCCIÓN	10
1. PRELIMINARES	11
1.1 Planteamiento del problema.....	11
1.2 Objetivos del proyecto	13
1.2.1 Objetivo General	13
1.2.2 Objetivos Específicos.....	13
1.3 Marco de referencia	14
1.3.1 Antecedentes	14
1.3.2 Marco teórico	17
2. METODOLOGÍA.....	32
2.1 Planeación.....	32
2.2 Desarrollo de concepto	33
2.3 Diseño a nivel de sistema.....	38

La información presentada en este documento es de exclusiva responsabilidad de los autores y no compromete a la EIA.

2.3.1 Diseño de hardware	38
□ Condiciones del espacio de trabajo.....	38
2.3.2 Diseño de software	49
3. PRESENTACIÓN Y DISCUSIÓN DE RESULTADOS.....	76
4. CONCLUSIONES.....	81
5. CONSIDERACIONES FINALES	82
6. BIBLIOGRAFÍA.....	83

La información presentada en este documento es de exclusiva responsabilidad de los autores y no compromete a la EIA.

LISTA DE TABLAS

	pág.
Tabla 1 Rango de trabajo articulaciones de doblez	27
Tabla 2 Rango de trabajo articulaciones de giro.....	28
Tabla 3 Velocidad de las articulaciones.....	28
Tabla 4 Rigidez en la flexión	28
Tabla 5 Torque Máximo	29
Tabla 6 Especificaciones miscelaneas	29
Tabla 7 Lista de necesidades.....	33
Tabla 8 Lista de alternativas	34
Tabla 9 Matriz morfológica	36
Tabla 10 Tabla de parámetros Denavit-Hartenberg para Baxter	55
Tabla 11 Tabla de parámetros necesarios	55
Tabla 12 Resultados de la prueba de selección.....	76

La información presentada en este documento es de exclusiva responsabilidad de los autores y no compromete a la EIA.

LISTA DE FIGURAS

	pág.
Figura 1 Matthias Nieuwenhuisen et al. (2013). Mobile Bin Picking with an Anthropomorphic Service Robot.....	17
Figura 2 Representación de la distancia Euclidiana	18
Figura 3 Abid K. (2013). Introduction to SURF (Speeded-Up Robust Features) 19	
Figura 4 Movimed. (2018). ¿Qué es la triangulación laser?.....	20
Figura 5 Hernández, Martinsanz, & Mata-García, (2011). Esquema del algoritmo de correspondencia estéreo basada en segmentos, con imágenes de entrada.	21
Figura 6 MathWorks, (2019). Ilustración del espacio de color del HSV.	22
Figura 7 OpenCV,(2013). Comparación de distintas técnicas de binarizado para una imagen....	24
Figura 8 OpenCV, (2013). Resultado de la aplicación de filtrado gaussiano a una imagen	25
Figura 9 Rethink Robotics. (2015). Hardware Specifications – Joint names	26
Figura 10 Rethink Robotics. (2015). Hardware Specifications – Link Lengths.....	27
Figura 11 Rethink Robotics. (2015). Hardware Specifications – Range of Motion.....	27
Figura 12 Rethink Robotics. (2015). Hardware Specifications – Range of Motion.....	28

La información presentada en este documento es de exclusiva responsabilidad de los autores y no compromete a la EIA.

Figura 13 Esquema de caja negra	33
Figura 14 Esquema de caja transparente.....	33
Figura 15 Ranking "Qué" vs "Como" basado en la casa de la calidad	36
Figura 16 Ministerio de trabajo e inmigración, (2003). Tamaño máximo recomendable de una carga.....	39
Figura 17 Logitech, (2018). Cámara Web C270.....	41
Figura 18 Pieza 1 diseñada para pruebas	42
Figura 19 Pieza 2 diseñada para pruebas	42
Figura 20 Soporte para cámara web ensamblado	43
Figura 21 Soporte para cámara web desensamblado	44
Figura 22 SCHMALZ (2016). Ventosas de fuelle	45
Figura 23 SCHMALZ (2016). Diámetro de la ventosa	46
Figura 24 Ventosa de fuelle Festo	47
Figura 25 Ventosa y generador de vacío adaptado al manipulador izquierdo del Baxter	47
Figura 26 Robot Baxter con las tuberías neumáticas montadas	48
Figura 27 Conexiones eléctricas y neumáticas para la electroválvula 5/2 monoestable	49

La información presentada en este documento es de exclusiva responsabilidad de los autores y no compromete a la EIA.

Figura 28 (II, 2017) . Diagrama de cinemática para el manipulador izquierdo con 6 grados de libertad y $\Theta_3 = 0$	54
Figura 29 (II, 2017) . Vista frontal de Baxter	55
Figura 30 (II, 2017) . Detalle de la vista superior de Baxter	55
Figura 31 Simulación en Matlab para la posición de 0° en todas las articulaciones del manipulador izquierdo de Baxter	60
Figura 32 Simulación en Matlab para una posición aleatoria en el espacio con orientación del efector final a 90°	61
Figura 33 Etiquetas diseñadas para el algoritmo de visión artificial	63
Figura 34 Umbral adaptativo gaussiano aplicado a la escena capturada por la cámara RGB	64
Figura 35 Resultado del cálculo de centroide e identificación de etiquetas en la escena	64
Figura 36 Recorte del logo de la etiqueta que será usado en la implementación de SURF.....	65
Figura 37 Recorte de la etiqueta encontrada para realizar la identificación del color	65
Figura 38 Cambio del espacio de color a HSV aplicado a la figura 37	66
Figura 39 Binarización de imagen aplicada a la figura 38.....	66
Figura 40 Imagen de referencia para algoritmo SURF.....	67

La información presentada en este documento es de exclusiva responsabilidad de los autores y no compromete a la EIA.

Figura 41 Resultado del emparejamiento de dos etiquetas identificadas en la escena con la imagen de referencia común.....	68
Figura 42 Sistemas coordenados asignado al Baxter y al espacio de trabajo	71
Figura 43 Posición inicial de extracción.....	73
Figura 44 Posición final de sujeción.....	74
Figura 45 Interfaz de usuario	75

La información presentada en este documento es de exclusiva responsabilidad de los autores y no compromete a la EIA.

RESUMEN

Para mantener la competencia en la industria se hace necesario producir de forma más rápida y eficiente. Las tareas de selección y clasificación, en ocasiones hacen parte de los procesos de producción, siendo estas a menudo realizadas manualmente. Debido a esto se plantea como proyecto de trabajo de grado la implementación de un método para la automatización de procesos de selección y clasificación de objetos

Este documento presenta un sistema robótico de Bin Picking 3D guiado por visión para el cobot Baxter. La aplicación básica para estos sistemas está en líneas de producción. Este sistema se elaboró como una solución para la que podría ser una de las aplicaciones de bin picking más sencillas, denominada bin picking estructurado. En el presente documento se encuentra el avance parcial realizado para la evaluación del trabajo de grado denominado “Sistema robótico de Bin Picking 3D guiado por visión para el cobot Baxter”. El proyecto consiste en la implementación de un sistema robótico guiado por visión artificial, para la selección y clasificación de piezas que se encuentra en una escena, siendo Bin Picking la denominación usada para este tipo de procesos.

La metodología usada está basada en Karl T. Ulrich, sin embargo, para implementarla se eliminaron algunos puntos en los que se trabajaba con estudio de mercados, debido a que el proyecto no pretende abordar estos temas.

El resultado final será el que contempla el objetivo general de proyecto “Desarrollar un sistema de Bin Picking 3D mediante visión artificial haciendo uso del cobot Baxter” para el cual se plantearon una serie de objetivos específicos, los cuales se pueden encontrar más adelante.

La información presentada en este documento es de exclusiva responsabilidad de los autores y no compromete a la EIA.

INTRODUCCIÓN

El presente documento corresponde al informe parcial del trabajo de grado “Sistema robótico de Bin Picking 3d guiado por visión para el cobot Baxter”, se desarrolla desde planteamiento del problema hasta la selección de los componentes necesarios para construir el sistema.

En el planteamiento del problema se describe la razón por la cual es crucial generar soluciones que mejoren la productividad de los procesos de ensamblaje industrial, siendo la clasificación de objetos parte de estos, se presentan los objetivos que se alcanzaran al finalizar la implementación del sistema, se desarrolla el marco de referencia donde se encuentran contemplados los antecedentes y el marco de teórico, por último, en la metodología se describe el proceso para determinar los componentes que harán parte del sistema construido.

La información presentada en este documento es de exclusiva responsabilidad de los autores y no compromete a la EIA.

1. PRELIMINARES

1.1 PLANTEAMIENTO DEL PROBLEMA

El ensamblaje industrial es uno de los procesos más importantes y representativos de los procesos de manufactura, estas operaciones adicionan hasta el 30% de los costos de fabricación de un producto final, por esto el ensamblaje es un punto crucial en la competitividad de la industria en general, sin embargo, a pesar de la importancia que representan, estas operaciones se realizan manualmente, lo que eleva en muchos casos los costos de producción. (Wu, Jiang, & Song, 2015) (Rowland, 1995)

Para mantener la competitividad global, la industria manufacturera debe concentrarse en flexibilidad y re-configurabilidad para producir piezas personalizadas en una forma rápida y eficiente. Algunas de estas operaciones llegan a ser realizadas por robots en las líneas de producción, muchos de estos robots están limitados a una sola tarea debido a una ausencia de percepción sensorial, lo que impide la realización de tareas más sofisticadas en entornos no estructurados. (Sansoni, Bellandi, Leoni, & Docchio, 2014) (Wu, Jiang, & Song, 2015)

Según lo anterior, uno de los principales problemas del ensamblaje industrial está en la falta de habilidad de los robots para trabajar en entornos no estructurados donde ni la forma ni la ubicación de los objetos es predecible, siendo la clasificación de objetos una de las operaciones que presentan este tipo de entornos.

La información presentada en este documento es de exclusiva responsabilidad de los autores y no compromete a la EIA.

La clasificación y el conteo de objetos son unas de las aplicaciones más requeridas tanto en sistemas de transporte como en líneas de producción. Los sistemas de transporte y las líneas de producción son comúnmente usados en muchas industrias, incluyendo la automotriz, agricultura, computación, electrónica, procesamiento de alimentos, farmacéutica, química, embotelladora y envasadora, lavado y limpieza, acabado de impresión, embalado entre otros. (Jovanovic, Mijanovic, Stojanovic, & Lekic, 2015).

Debido a lo anterior, para mantener la competencia en la industria se hace necesario desarrollar tecnologías que faciliten las tareas de clasificación, no solo teniendo en cuenta los diversos sectores en los que se realizan estas operaciones, sino que además, al analizar algunas de las tareas que se pueden realizar en una típica celda de ensamble, se concluye que la clasificación de objetos hace parte de las actividades que componen al ensamblaje industrial. (J.Cecil, Powell, & Vasquez, 2007) (Vonasek, Vick, & Saska1, 2017)

Luego de lo planteado se genera la siguiente cuestión ¿Es posible desarrollar un sistema robótico que permita realizar tareas de clasificación en entornos no estructurados con mayor rapidez y eficiencia?

1.2 OBJETIVOS DEL PROYECTO

1.2.1 Objetivo General

Desarrollar un sistema de Bin Picking 3D mediante visión artificial haciendo uso del cobot Baxter

1.2.2 Objetivos Específicos

- Realizar la identificación de parámetros de los diferentes elementos que integran el sistema.
- Implementar algoritmos para el procesamiento de imágenes que permitan obtener información 3D de la escena.
- Integrar el algoritmo de procesamiento de imágenes con el control de movimiento del cobot Baxter
- Desarrollar una interfaz de usuario en la que se pueda interactuar con diferentes ejemplos de prueba.
- Verificar el funcionamiento del sistema construido mediante pruebas de selección y clasificación de objetos.

La información presentada en este documento es de exclusiva responsabilidad de los autores y no compromete a la EIA.

1.3 MARCO DE REFERENCIA

1.3.1 Antecedentes

El bin picking es uno de los más desafiantes problemas en robótica ya que requiere de una integración compleja de métodos para la detección de objetos, estimación de posición, agarre y planificación de movimientos. (Vonasek, Vick, & Saska1, 2017)

El bin picking se refiere al problema de que un robot recoja automáticamente objetos de un contenedor, la cuestión clave es identificar los objetos de la escena y determinar sus poses (ubicaciones y orientaciones) en el espacio 3D. A pesar de que es un problema que ha sido estudiado por muchos años pocas soluciones existen en la industria. (Xu, Pu, Zeng, & Zha, 2012)

Se han generado diversas publicaciones de carácter científico en las que se presentan métodos para abordar los sistemas de Bin Picking como una aplicación completa o alrededor de los componentes que conforman estos sistemas.

Para la construcción de sistemas Bin Picking basados en nubes de puntos las cuestiones a resolver son las siguientes: (1) Adquisición de la escena a través de visión, (2) Segmentación de escena, (3) Reconocimiento y estimación de pose de los objetos segmentados. (Sansoni, Bellandi, Leoni, & Docchio, 2014)

Para la adquisición de información de la escena algunos autores como Cheng-Hei Wu et al., Shingeharu Miyata et al. y Matthias Nieuwenhuisen et al., proponen el uso del sensor Kinect como

La información presentada en este documento es de exclusiva responsabilidad de los autores y no compromete a la EIA.

un dispositivo para la captura imágenes RGB-D y generación de la nube de puntos de los objetos en la escena (Miyata & Yashiki, 2014), (Nieuwenhuisen, y otros, 2013), (Wu, Jiang, & Song, 2015). Otra técnica utilizada es la triangulación laser, esta se usa para capturar mediciones tridimensionales al vincular una fuente de iluminación laser a una cámara 3D, la luz del proyector laser rebotara de regreso a un sensor sensible a la posición o una matriz CMOS o CCD en la cámara. (Suliga, 2016) (Sansoni, Bellandi, Leoni, & Docchio, 2014)

Como parte del proceso de adquisición de información de la escena se realiza un filtrado para eliminar puntos de la nube debido a que en ocasiones estas son muy grandes, los métodos de filtrado son variados, algunos autores proponen realizar un análisis de las vecindades de un punto tomado como referencia, para esto se da un punto q y se asigna una cantidad n de puntos permisibles, el algoritmo eliminara entonces los puntos mas lejanos al punto q , posteriormente (Sansoni, Bellandi, Leoni, & Docchio, 2014), Cheng Hei Wu et al. propone el uso de filtros de cuadrícula voxel, en donde se crea una cuadrícula 3D sobre la nube de puntos y luego todos los puntos que se encuentran dentro de cada voxel son reemplazados por su centroide (Wu, Jiang, & Song, 2015).

Para la segmentación de la escena, la nube de puntos adquirida se dividirá en sub nubes, cada subnube pertenecerá a un objeto identificado en la escena, Giovanna Sansoni et al. realiza la segmentación de la escena usando filtros basados en umbrales, para esto previamente se debieron asignar los puntos q en los cuales se aplicara el umbral, que corresponde a un radio r , lo puntos por fuera del umbral son eliminados, se aplica un segundo filtro con características similares pero

La información presentada en este documento es de exclusiva responsabilidad de los autores y no compromete a la EIA.

esta vez los puntos dentro del umbral se almacenan en clústeres individuales, con esto se asigna una sub nube a cada objeto en la escena. (Sansoni, Bellandi, Leoni, & Docchio, 2014).

Para la estimación de la pose de los objetos en la escena, Cheng Hei Wu et al. propone la estimación de posición basada en modelos CAD, en este método se generan nubes de puntos para los modelos CAD de las piezas involucradas en la escena, obtenida la nube de puntos se calculan los descriptores de características de la nube de puntos del modelo CAD y de la nube de puntos de la escena entonces los descriptores de características del modelo CAD se emparejaran por pares con los descriptores de características de la escena, una vez identificados los descriptores similares entre las nubes se realizan una serie transformaciones con el objetivo de alinear los descriptores del modelo CAD con la escena, conociendo la pose de la nube del modelo CAD conocemos entonces la pose del objeto en la escena (Wu, Jiang, & Song, 2015)

Otros métodos usan una combinación de técnicas específicas de visión artificial, Kyekyung Kim et al. propone un método que combina el reconocimiento y la detección de múltiples características en un objeto, estereoscopia y un algoritmo SIFT, esto permite estimar la localización, la distancia y ángulo superficial de rotación de la pieza. (Kim, Kang, Kim, Lee, & Kim, 2015), dando solución a la adquisición de escena, segmentación y estimación de pose.

La información presentada en este documento es de exclusiva responsabilidad de los autores y no compromete a la EIA.

1.3.2 Marco teórico

- **Bin picking:**

El bin picking se refiere al problema de que un robot recoja automáticamente objetos de un contenedor, la cuestión clave es identificar los objetos de la escena y determinar sus poses (ubicaciones y orientaciones) en el espacio 3D. (Xu, Pu, Zeng, & Zha, 2012)

- **Segmentación:**

Se trata de la separación y organización de los objetos de una escena tomando como base características homogéneas de la nubes de puntos, con respecto a una serie de criterios previamente establecidos (Sarmiento, 2012)

- **Nubes de puntos:**

Una nube de puntos es un gran conjunto de puntos adquiridos mediante escáneres láser 3D u otras tecnologías para crear representaciones tridimensionales de las estructuras existentes. (Autodesk, 2019)

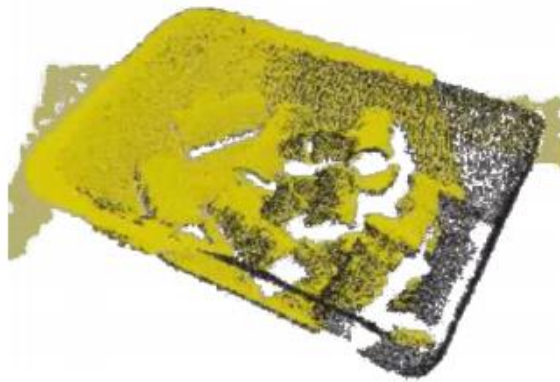


Figura 1 Matthias Nieuwenhuisen et al. (2013). Mobile Bin Picking with an Anthropomorphic Service Robot [Figura]. Recuperado de IEEE international conference on Robotics and automation (ICRA).

La información presentada en este documento es de exclusiva responsabilidad de los autores y no compromete a la EIA.

- **Distancia Euclidiana:**

Es la distancia descrita entre dos puntos de un espacio euclídeo, esta se deduce a partir del teorema de Pitágoras.

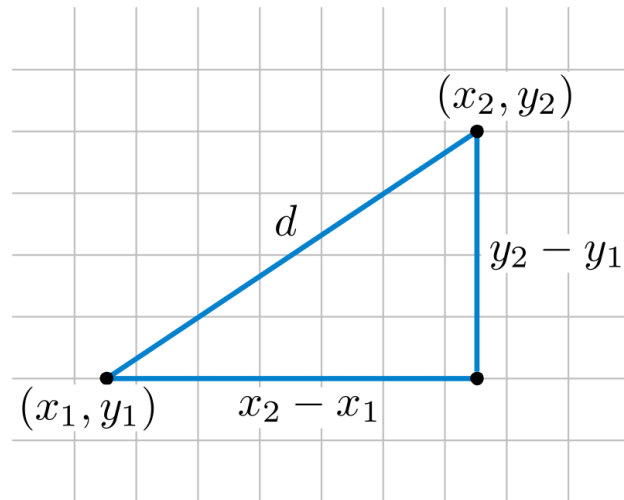


Figura 2 (Belk, 2007). Representación de la distancia Euclidiana [Figura]

- **SURF (Speeded-Up Robust Features):**

Es una técnica de extracción de puntos de interés de una imagen. Mediante SURF, para los puntos seleccionados de la imagen se obtiene un conjunto de descriptores muy distintivos e invariantes ante cambios en escala, iluminación y perspectiva. Por ello, el emparejamiento de estos puntos entre dos imágenes distintas es altamente robusto. El coste computacional que presenta es, además, inferior en comparación con otros extractores similares como SIFT. La extracción de los puntos la realiza detectando en primer lugar los posibles puntos de interés y su localización dentro de la imagen. Es mucho más rápido que el método SIFT, ya que los keypoints contienen muchos menos descriptores. (Rahman, 2013)



Figura 3 Abid K. (2013). *Introduction to SURF (Speeded-Up Robust Features)* [Figura]. Recuperado de *OpenCV2 tutorials*).

- **Keypoint matching**

Cuando se comparan las características extraídas de dos imágenes con el fin de hallar coincidencias de una imagen dentro de otra aparece el concepto de keypoint matching. Se trata de encontrar la mejor coincidencia de candidatos para cada punto clave en una imagen. El vecino más cercano se define como el punto clave con una distancia euclidiana mínima para el vector descriptor invariante, sin embargo, muchas características de una imagen no tendrán una coincidencia correcta en la base de datos de referencia porque surgen del desorden de fondo. Por lo tanto, sería útil tener una forma de descartar características que no tienen una buena coincidencia con la base de datos, David Lowe autor del método SIFT propuso un método que descartaría el 90% de los matches incorrectos mientras descartaba menos del 5%

La información presentada en este documento es de exclusiva responsabilidad de los autores y no compromete a la EIA.

de matches correctos, el método es simple y se trata de descartar todos los matches cuya relación de distancia sea superior a 0.8.

- **Triangulación láser:**

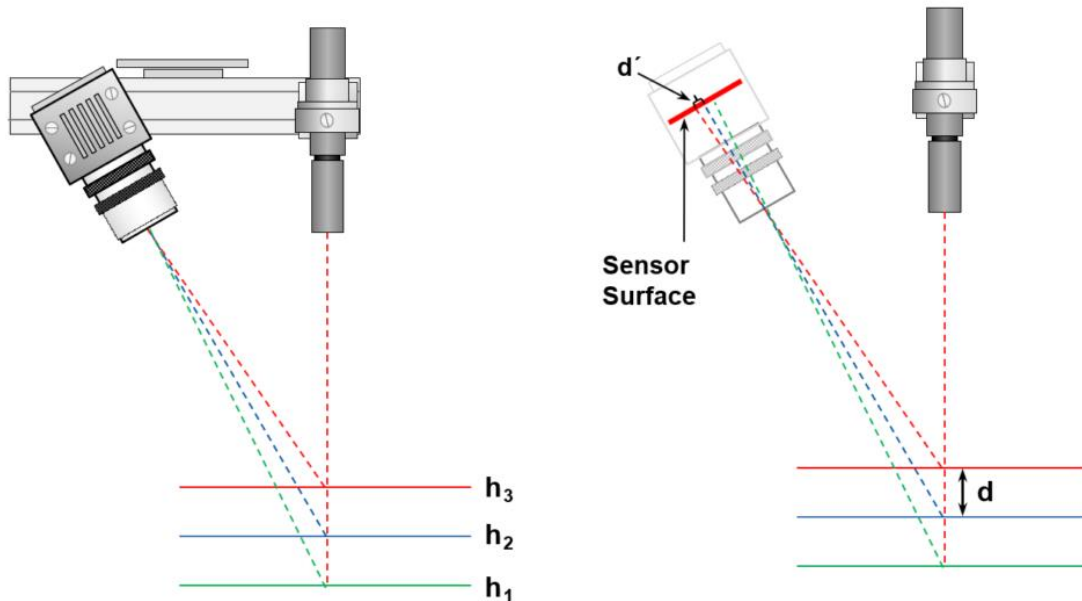


Figura 4 Movimed. (2018). ¿Qué es la triangulación láser? [Figura]. Recuperado de <http://www.movimed.com/knowledgebase/what-is-laser-triangulation/>

La triangulación láser es una técnica de visión artificial utilizada para capturar mediciones tridimensionales al vincular una fuente de iluminación láser con una cámara. El rayo láser y la cámara apuntan al objetivo de inspección (como se muestra en la Figura 4), pero al adoptar un desplazamiento angular conocido entre la fuente de láser y el sensor de la cámara, es posible medir las diferencias de profundidad utilizando la trigonometría. (Movimed, 2018)

- **Visión estereoscópica:** La visión estereoscópica constituye un procedimiento para la obtención de la forma de los objetos en la escena. En este caso la forma se determina a través de la distancia de los objetos en relación con un sistema de referencia por lo que se trata de un

La información presentada en este documento es de exclusiva responsabilidad de los autores y no compromete a la EIA.

método para la obtención de la tercera dimensión. (Hernández, Martinsanz, & Mata-García, 2011)

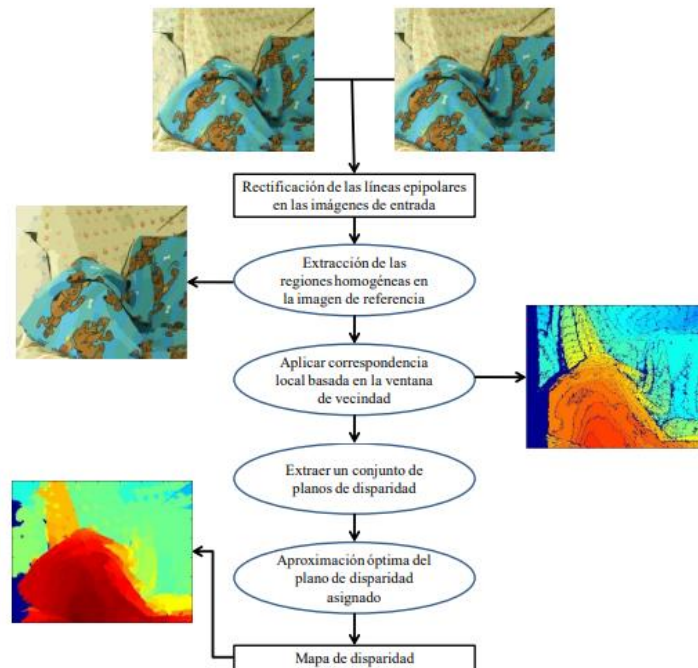


Figura 5 Hernández, Martinsanz, & Mata-García, (2011). Esquema del algoritmo de correspondencia estéreo basada en segmentos, con imágenes de entrada. [Figura]. Recuperado de Técnicas de procesamiento de imágenes estereoscópicas

- **Espacio de color HSV:**

El espacio de color HSV (tono, saturación, valor) es utilizado a menudo por personas que están seleccionando colores (por ejemplo, de pinturas o tintas) de una rueda de color o paleta, porque se corresponde mejor a cómo la gente experimenta el color que el espacio de color RGB hace.

La información presentada en este documento es de exclusiva responsabilidad de los autores y no compromete a la EIA.

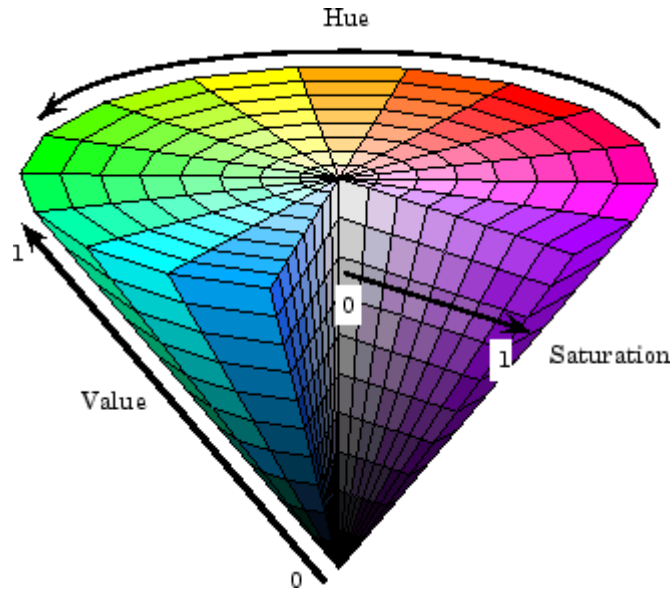


Figura 6 MathWorks, (2019). Ilustración del espacio de color del HSV. [Figura]. Recuperado de MathWorks

Una breve explicación que ayudará a comprender mejor el espacio de color HSV será la siguiente: Como el matiz varía de 0 a 1,0, los colores correspondientes varían de rojo a través de amarillo, verde, cian, azul, magenta y de nuevo a rojo, por lo que en realidad hay valores rojos en 0 y 1,0. Como la saturación varía de 0 a 1,0, los colores correspondientes (matices) varían de insaturados (tonos de gris) a completamente saturado (sin componente blanco). Como valor, o brillo, varía de 0 a 1,0, los colores correspondientes se vuelven cada vez más brillantes. (MathWorks, 2019)

- **Binarización:**

Es una técnica que consiste en la realización de un barrido en la matriz de la imagen digital, con el fin de que un proceso produzca la reducción de la escala de grises a dos únicos valores. Negro (= 0) y blanco (= 255), o lo que es lo mismo, un sistema binario de ausencia y presencia

La información presentada en este documento es de exclusiva responsabilidad de los autores y no compromete a la EIA.

de color 0-1. La comparación de cada píxel de la imagen viene determinada por el umbral de sensibilidad (valor $T = \text{Threshold}$). (Magro, 2013)

A continuación, se presentarán algunas técnicas usadas en la binarización de imágenes.

- **Umbral simple:**

En esta técnica de binarización el programador definirá el umbral de binarización que se aplicará a una imagen en escala de grises, el resultado tal que si el valor de un píxel supera este umbral su valor cambiará a 255 y si está por debajo será 0, o en otro caso es posible obtener el resultado inverso. (OpenCV, 2013)

- **Umbral adaptativo**

Es una técnica de binarización de imágenes, la cual cuenta con librería en OpenCV para facilitar su implementación, en esta técnica el algoritmo determina el umbral para un píxel en función de una pequeña región a su alrededor, por lo tanto, se obtienen diferentes umbrales para diferentes regiones de una misma imagen, lo que da mejores resultados para entornos con iluminación variable (OpenCV, 2013)

Además de lo anteriormente expuesto el método posee dos formas para decidir cómo se calcula el valor del umbral, son las siguientes

- **ADAPTIVE_THRESH_MEAN:** El valor umbral es la media del área del vecindario.
- **ADAPTIVE_THRESH_GAUSSIAN:** El valor umbral es la suma ponderada de los valores del vecindario donde los pesos son una ventana gaussiana.

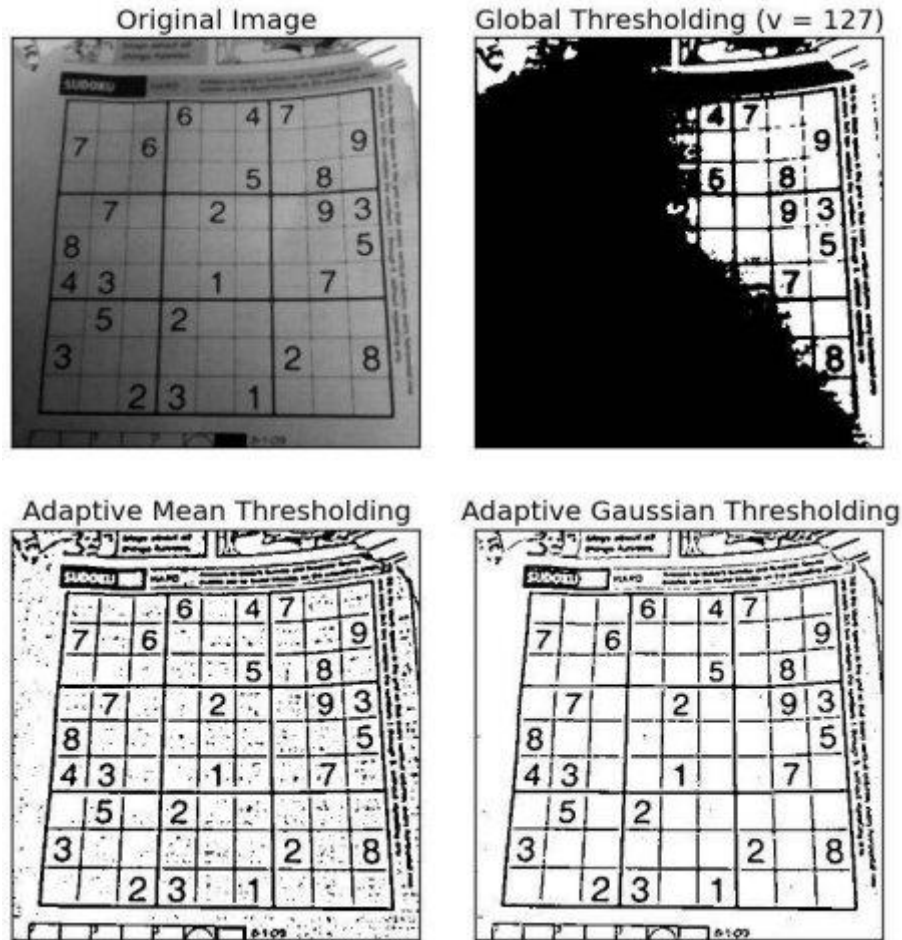


Figura 7 OpenCV, (2013). Comparación de distintas técnicas de binarizado para una imagen [Figura]. Recuperado de OpenCV Docs

- **Suavizado de imagen:**

El desenfocado de la imagen se logra convolucionando la imagen con un núcleo de filtro de paso bajo. Este tipo de técnicas útil para eliminar el ruido de alta frecuencia (por ejemplo, ruido, bordes) de la imagen, lo que da como resultado que los bordes se desdibujen cuando se aplica este filtro. En ocasiones aplicar este tipo de enfoques mejora los resultados obtenidos en la binarización de una imagen.

La información presentada en este documento es de exclusiva responsabilidad de los autores y no compromete a la EIA.

A continuación, se presentará una de las técnicas usadas para el suavizado de imágenes la cual fue implementada en el desarrollo del presente trabajo de grado.

o **Filtrado gaussiano**

En este enfoque, en lugar de un filtro de caja que consiste en coeficientes de filtro iguales, se utiliza un núcleo gaussiano. Deberíamos especificar el ancho y la altura del núcleo, que debería ser positivo e impar. También debemos especificar la desviación estándar en las direcciones X e Y, σ_X y σ_Y respectivamente

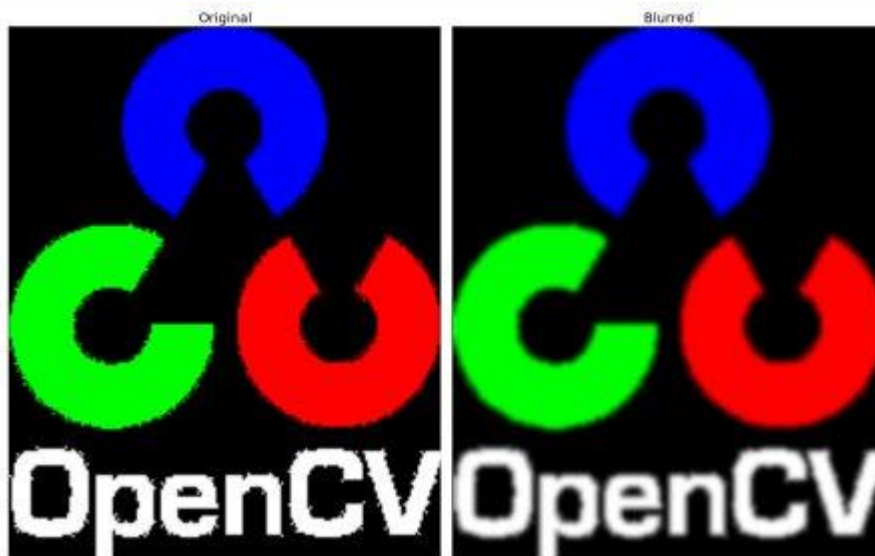


Figura 8 OpenCV, (2013). Resultado de la aplicación de filtrado gaussiano a una imagen [Figura]. Recuperado de OpenCV Docs

- **Baxter:**

Baxter es un robot antropomórfico humanoide con dos brazos con 7 grados de libertad cada uno y tecnologías de detección de vanguardia, incluye detección y control de fuerza, par y posición en cada junta, cámaras con soporte para aplicaciones de visión artificial integradas,

La información presentada en este documento es de exclusiva responsabilidad de los autores y no compromete a la EIA.

elementos de entrada y salida de usuario como un apantalla montada en la cabeza botones y mandos. (Robotics, 2015)

Especificaciones

○ Nombre de las articulaciones

Las articulaciones Baxter fueron nombradas y enumerados para efectos prácticos en la programación de la siguiente forma (S= Hombro, E= Codo, W= Muñeca)

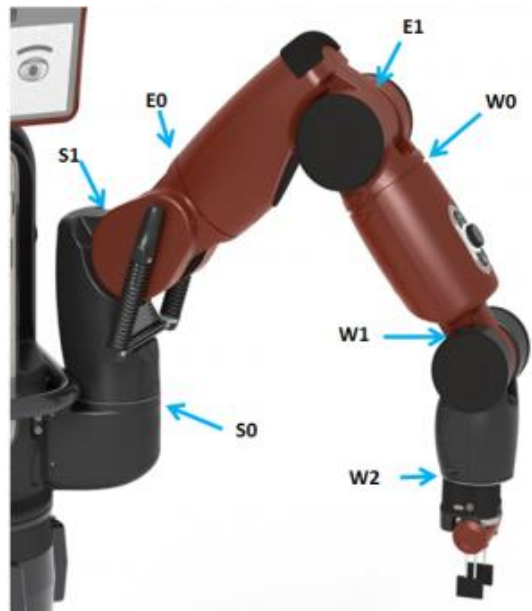


Figura 9 Rethink Robotics. (2015). Hardware Specifications – Joint names [Figura]. Recuperado de http://sdk.rethinkrobotics.com/wiki/Hardware_Specifications

○ Distancia entre articulaciones

La distancia entre las articulaciones esta medida en mm, desde el centro de una articulación al centro de la siguiente, se puede observar en la siguiente imagen.

La información presentada en este documento es de exclusiva responsabilidad de los autores y no compromete a la EIA.

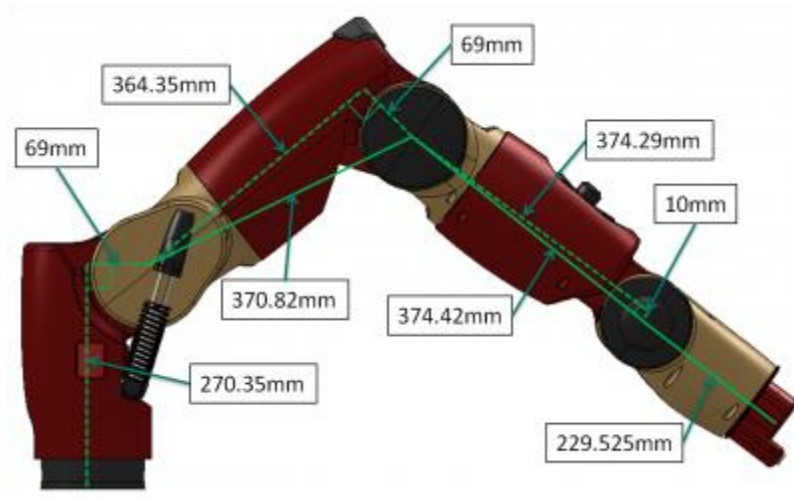


Figura 10 Rethink Robotics. (2015). Hardware Specifications – Link Lengths [Figura]. Recuperado de http://sdk.rethinkrobotics.com/wiki/Hardware_Specifications

○ **Rango de movimiento – Articulaciones de doblez**

Se presenta el rango de trabajo de las articulaciones de doblez, en la figura 11 se puede ver a que articulaciones hace referencia.

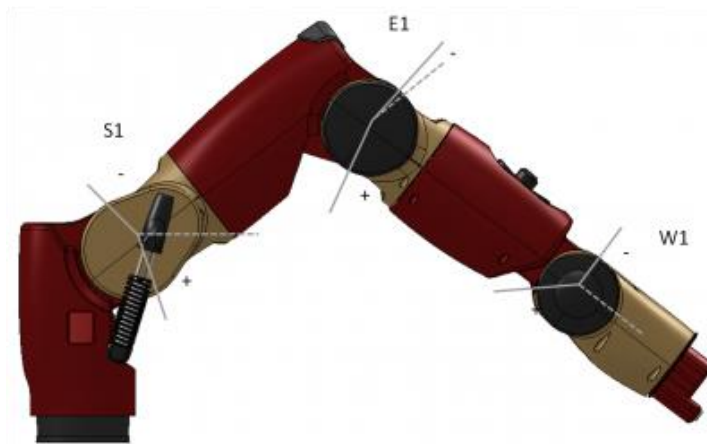


Figura 11 Rethink Robotics. (2015). Hardware Specifications – Range of Motion [Figura]. Recuperado de http://sdk.rethinkrobotics.com/wiki/Hardware_Specifications

La información presentada en este documento es de exclusiva responsabilidad de los autores y no compromete a la EIA.

Tabla 1

Tabla rango de trabajo articulaciones de doblez

Articulación	(Grados) Límite máximo	Límite máximo	Rango	(Radianes) Límite mínimo	Máximo límite	Rango
S1	-123	+60	183	-2.147	+1.047	3.194
E1	-2.864	+150	153	-0.05	+2.618	2.67
W1	-90	+120	210	-1.5707	+2.094	3.6647

Tabla 1. Rethink Robotics. (2015). *Hardware Specifications – Range of Motion* [Tabla]. Recuperado de http://sdk.rethinkrobotics.com/wiki/Hardware_Specifications

○ **Rango de movimiento – Articulaciones de giro**

Se presenta el rango de trabajo de las articulaciones de giro en la figura 12 se puede ver a que articulaciones hace referencia.

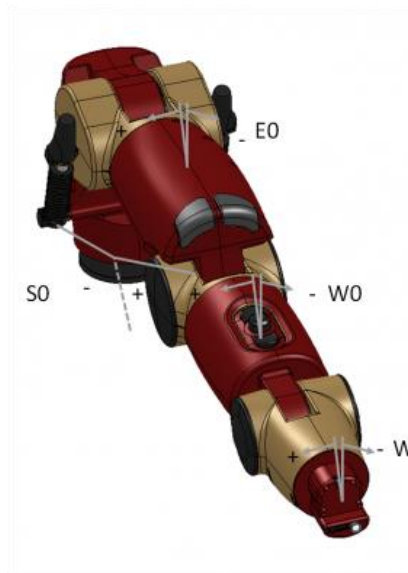


Figura 12 Rethink Robotics. (2015). *Hardware Specifications – Range of Motion* [Figura]. Recuperado de http://sdk.rethinkrobotics.com/wiki/Hardware_Specifications

La información presentada en este documento es de exclusiva responsabilidad de los autores y no compromete a la EIA.

Tabla 2

Tabla rango de trabajo articulaciones de giro

Articulación	(Grados) Límite máximo	Límite máximo	Rango	(Radianes) Límite mínimo	Máximo límite	Rango
S0	-97.494	+97.494	194.998	-1.7016	+1.7016	3.4033
E0	-174.987	+174.987	349.979	-3.0541	+3.0541	6.1083
W0	-175.25	+175.25	350.5	-3.059	+3.059	6.117
W2	-175.25	+175.25	350.5	-3.059	+3.059	6.117

Tabla 2. Rethink Robotics. (2015). *Hardware Specifications – Range of Motion* [Tabla]. Recuperado de http://sdk.rethinkrobotics.com/wiki/Hardware_Specifications

- **Máxima velocidad de las articulaciones**

Tabla 3

Velocidad de las articulaciones

Articulación	Velocidad máxima
S0	2.0
S1	2.0
E0	2.0
E1	2.0
W0	4.0
W1	4.0
W2	4.0

Tabla 3. Rethink Robotics. (2015). *Hardware Specifications – Range of Motion* [Tabla]. Recuperado de http://sdk.rethinkrobotics.com/wiki/Hardware_Specifications

- **Rigidez en la flexión de la articulación**

Tabla 4

Rigidez en la flexión

Articulación	Rigidez
Pequeñas flexiones (W0, W1, W2)	3.4deg @ 15Nm (~250Nm/rad)
Grandes flexiones (S0, S1, E0, E1)	3.4deg @ 50Nm (~843Nm/rad)

La información presentada en este documento es de exclusiva responsabilidad de los autores y no compromete a la EIA.

Tabla 4. Rethink Robotics. (2015). *Hardware Specifications – Range of Motion* [Tabla]. Recuperado de http://sdk.rethinkrobotics.com/wiki/Hardware_Specifications

- **Torque máximo**

Tabla 5

Torque Máximo

Articulación	Torque máximo
S0,S1,E0,E1	50Nm
W0,W1,W2	15Nm

Tabla 5. Rethink Robotics. (2015). *Hardware Specifications – Range of Motion* [Tabla]. Recuperado de http://sdk.rethinkrobotics.com/wiki/Hardware_Specifications

- **Especificaciones misceláneas**

Tabla 6

Especificaciones misceláneas

Descripción	Especificaciones
Resolución de pantalla	1024 x 600 pixeles
Precisión posicional	+/- 5 mm
Carga útil máxima	5 lb / 2.2 kg
Fuerza de agarre	35 N

- **ROS (Robot Operating System):** Es un marco flexible para escribir software de robot. Es una colección de herramientas, bibliotecas y convenciones que tienen como objetivo simplificar la tarea de crear un comportamiento robótico complejo y robusto en una amplia variedad de plataformas robóticas. (ROS, 2013)

- **TCD (Transformada cinemática directa):**

El estudio de la cinemática de manipuladores se relaciona, entre otras cosas, con la manera en que cambian las ubicaciones de una serie de tramas agregadas en el mecanismo, a medida que se articula el mismo. El tema central de la TCD es el método para calcular la posición y la orientación

La información presentada en este documento es de exclusiva responsabilidad de los autores y no compromete a la EIA.

del efector final del manipulador relativo a la base del mismo, como una función de las variables de las articulaciones. (Craig, 2006)

- **TCI (Transformada cinemática inversa):**

Se trata del estudio del problema de cinemática en el que, dada la posición y orientación deseadas de la herramienta respecto a la estación, como se obtendrán el conjunto de ángulos de articulación que logren satisfacer el resultado deseado. (Craig, 2006)

2. METODOLOGÍA

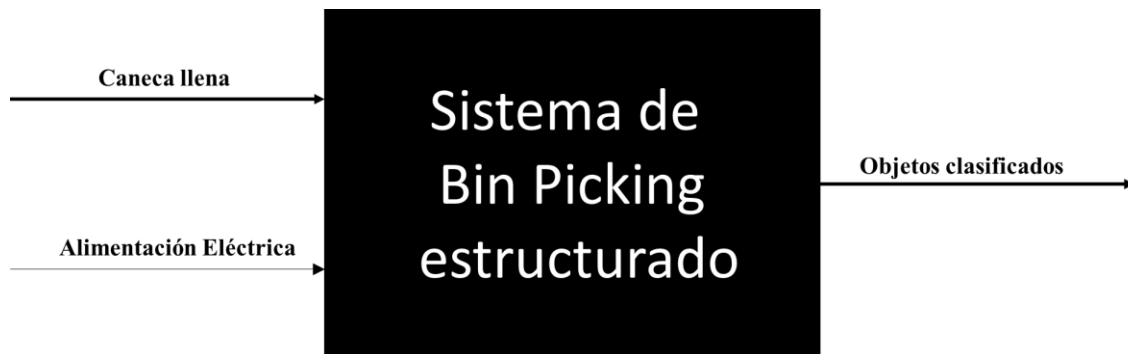
Para el desarrollo de la presente tesis de grado, se usará la metodología de Karl T. Ulrich, sin embargo, debido a que el proyecto no se orientará en obtener un producto para incursionar en un mercado, las secciones metodológicas orientadas a la evaluación y análisis de mercados de esta metodología no serán desarrolladas.

El desarrollo metodológico obtenido de la metodología de diseño de Karl T. Ulrich consta de las cinco fases, a continuación, se presentarán la fase de planeación y desarrollo del concepto hasta la selección de componentes.

2.1 Planeación

En la etapa de planeación se define la arquitectura del sistema, para ello se hace uso de los esquemas de caja negra y caja transparente, estos presentan de una forma generalizada los diferentes componentes que la conforman y su interacción.

En la figura 13 se presenta el esquema de caja negra en donde se consideran las entradas y salidas del sistema.



La información presentada en este documento es de exclusiva responsabilidad de los autores y no compromete a la EIA.

Figura 13 Esquema de caja negra

Posteriormente se definen los elementos que hacen parte de la arquitectura del sistema y la manera en que estos interactúan entre ellos, el resultado se encuentra en la figura 14, en el esquema de caja transparente.

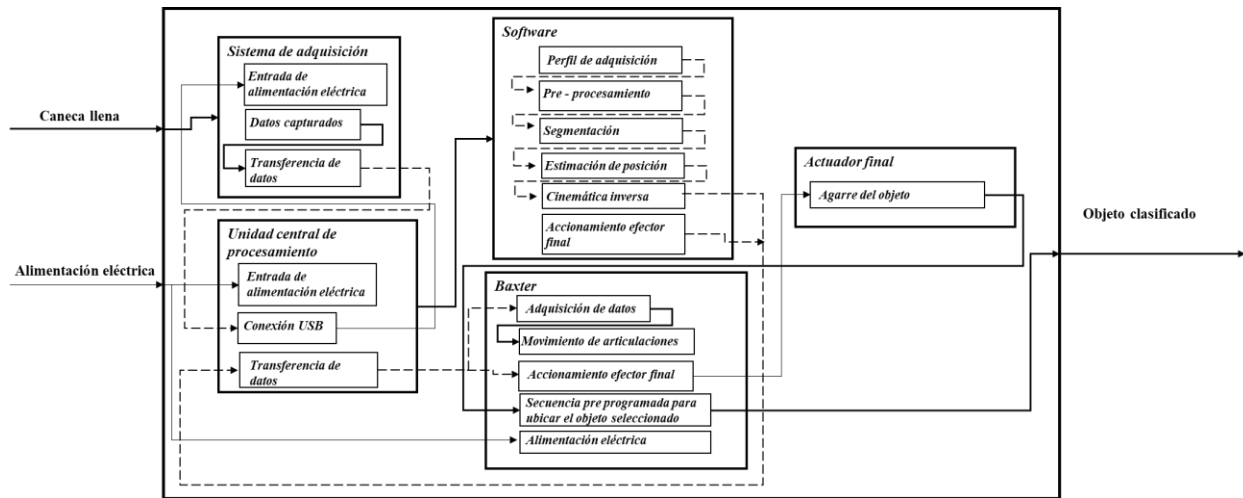


Figura 14 Esquema de caja transparente

2.2 Desarrollo de concepto

Para el desarrollo del concepto se utiliza como base la HoQ (Casa de la calidad), de su estructura se usaron los elementos que podrían aportar más valor en el contexto que se está trabajando, estos elementos fueron la lista de necesidades de usuario que se puede interpretar como los “Qué” del proyecto y un listado de distintas alternativas de solución relacionadas con la arquitectura del proyecto, que se pueden interpretar como los “Cómo”. En la tabla 6 se presenta la lista de necesidades obtenida y la tabla 7 la lista de alternativas según arquitectura planteada.

La información presentada en este documento es de exclusiva responsabilidad de los autores y no compromete a la EIA.

Tabla 7

Lista de necesidades

	Dispositivo	Requerimientos de usuario	Importancia
1	Unidad central de procesamiento	Buena capacidad de procesamiento	5
2	Unidad central de procesamiento	Manipulación de hardware sencilla	3
3	Unidad central de procesamiento	Compacto	4
4	Unidad central de procesamiento	Suficientes entradas para periféricos	5
5	Unidad central de procesamiento	Conexión ethernet	5
6	Unidad central de procesamiento	Modular	2
7	Unidad central de procesamiento	Bajo costo	5
8	Unidad central de procesamiento	Tamaño reducido	1
9	Software	Debe existir suficiente documentación	5
10	Software	Fácil integración entre los distintos paquetes de software	3
11	Software	Open source	3
12	Estimación de posición de objetos en la escena	Bajo costo	5
13	Estimación de posición de objetos en la escena	No sobredimensionada	5
14	Estimación de posición de objetos en la escena	Exactitud	5
15	Estimación de posición de objetos en la escena	Información adquirida fácil de procesar	4
16	Estimación de posición de objetos en la escena	Segmentación de la escena sencilla	5
17	Interfaz	Estética	1
18	Interfaz	Interfaz amigable con el usuario	3
19	Efactor final	Fiabilidad en el agarre	5
20	Efactor final	Facilidad de agarre	4
21	Efactor final	Evitar colisiones con objetos adyacentes	4

La información presentada en este documento es de exclusiva responsabilidad de los autores y no compromete a la EIA.

Tabla 8

Lista de alternativas

Unidad Central de Procesamiento	Ordenador personal (laptop)
	ODROID-H2
	Raspberry Pi 3
Software	Windows
	Ubuntu
	Matlab
	Python
Estimación de posición de objetos en la escena	Cámaras
	Kinect
	Triangulación laser
Interfaz	GUI
	CMD
Actuador Final	Ventosas
	Pinzas

Entonces haciendo uso de los dos elementos anteriores se realiza un ranking al comparar los “Qué” vs los “Cómo” y se obtiene el resultado que se puede observar en la figura 15.

Como se puede apreciar en la matriz de comparación, el ranking esta realizado analizando la relación entre la arquitectura y las necesidades del usuario, ya que no resultaría lógico relacionar una computadora personal con la necesidad de evitar colisiones con objetos adyacentes en la escena.

El ranking por si solo muestra cual sería la selección más óptima para un bloque independiente de la arquitectura, pero no si la interacción entre los distintos bloques es posible, por esta razón se realiza una matriz morfológica donde se definen las distintas soluciones alcanzables con las alternativas disponibles.

La información presentada en este documento es de exclusiva responsabilidad de los autores y no compromete a la EIA.

Lista de necesidades		Unidad Central de Procesamiento			Software				Estimación de posición de objetos en la escena			Interfaz		Actuador Final		
		Ordenador personal (laptop)	ODROID-H2	Raspberry Pi 3	Windows	Ubuntu	Matlab	Python	Camaras	Kinect	Triangulacion laser	GUI	CMD	Ventosas	Pinzas	
Requerimientos de usuario	Importancia															
Buena capacidad de procesamiento	5	9	9	3												
Manipulación de hardware sencilla	3	9	3	3												
Compacto	4	9	3	2												
Suficientes entradas para perifericos	5	9	9	9												
Conexión ethernet	5	9	9	9												
Modular	2	9	9	9												
Bajo costo	5	3	9	9												
Tamaño reducido	1	3	9	9												
Debe existir suficiente documentación	5				9	9	3	9								
Facil integracion entre los distintos paquetes de software	3				9	9	9	9								
Open source	3				1	9	1	9								
Bajo costo	5								9	3	1					
No sobredimensionada	5								9	9	1					
Exactitud	5								3	9	9					
Información adquirida facil de procesar	4								9	9	3					
Segmentación de la escena sencilla	5								9	3	1					
Estetica	1											9	1			
Interfaz amigable con el usuario	3											9	3			
Fiabilidad en el agarre	5													3	9	
Facilidad de agarre	4													9	1	
Evitar colisiones con objetos adyacentes	4													9	1	
Absoluto		234	228	198	75	99	45	99	186	156	72	36	10	87	53	
Relativo		11,70	11,40	9,90	3,75	4,95	2,25	4,95	9,30	7,80	3,60	1,80	0,50	4,35	2,65	
Ranking		1	2	3	2	1	3	1	1	2	3	1	2	1	2	

Figura 15 Ranking "Qué" vs "Como" basado en la casa de la calidad

Tomando como base el ranking individual para cada componente de la arquitectura realizado en la casa de la calidad, se plantean las rutas que permitirían obtener las 3 mejores soluciones.

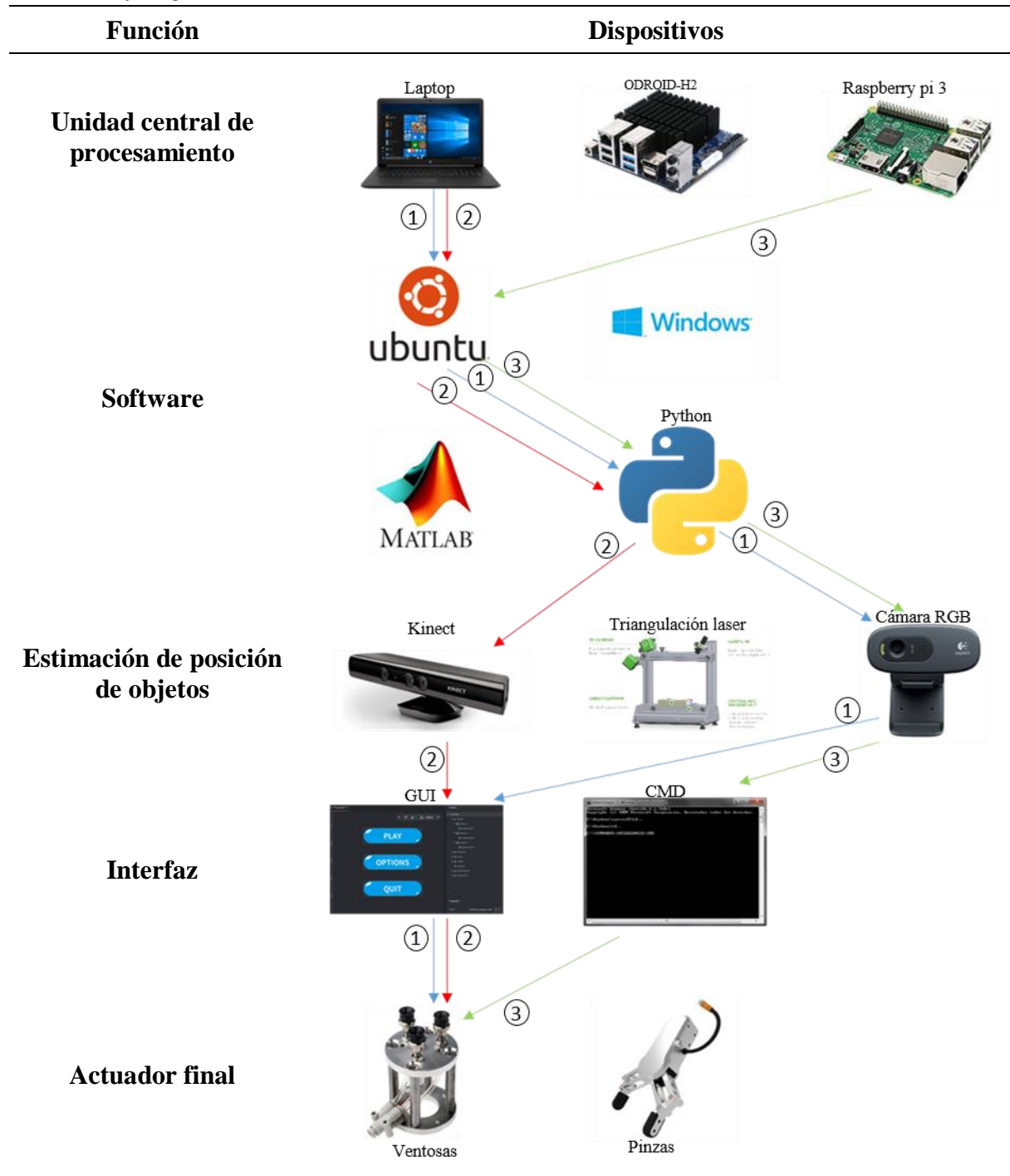
En la tabla 8 las rutas están separadas por color y enumeradas de la siguiente manera:

- Ruta Azul Solución 1
- Ruta Roja Solución 2
- Ruta Verde Solución 3

La información presentada en este documento es de exclusiva responsabilidad de los autores y no compromete a la EIA.

Tabla 9

Matriz morfológica



La información presentada en este documento es de exclusiva responsabilidad de los autores y no compromete a la EIA.

2.3 Diseño a nivel de sistema

2.3.1 Diseño de hardware

- **Condiciones del espacio de trabajo**

Se debe considerar que Baxter es un robot colaborativo, que puede trabajar en entornos cercanos al personal de operación, por esta razón las condiciones del espacio de trabajo deben adecuarse para que sean las óptimas para el desarrollo de actividades por parte de los operarios.

Existen guías técnicas donde se detallan las características que debe poseer un espacio de trabajo con el objetivo de prevenir de riesgos para las personas, relativos a la manipulación de cargas manuales, de donde se pueden extraer las condiciones óptimas para configurar un espacio de trabajo.

A continuación se definirán el máximo peso y tamaño de la carga que transportara un operario hasta el área de trabajo del Baxter

- **El peso de la carga**

Se consideran como cargas todos los objetos que tengan un peso igual o superior a 3kg, como indicación general, el peso máximo que se recomienda no sobrepasar (en condiciones ideales de manipulación) es de 25 kg. Sin embargo, si la población expuesta son mujeres, trabajadores jóvenes o mayores no se deberían manejar cargas superiores a 15 kg. (Ministerio de trabajo e inmigración, 2003)

- **El tamaño de la carga**

La información presentada en este documento es de exclusiva responsabilidad de los autores y no compromete a la EIA.

Este se refiere al tamaño máximo de la carga que puede transportar un trabajador, es conveniente que la anchura de la carga no supere la anchura de los hombros (60 cm aproximadamente). La profundidad de la carga no debería superar los 50 cm.

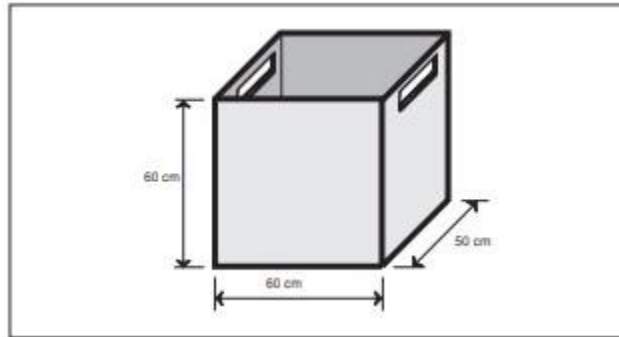


Figura 16 Ministerio de trabajo e inmigración, (2003). Tamaño máximo recomendable de una carga [Figura]. Recuperado de guía técnica para la manipulación manual de cargas

Debido a que se desea realizar bin-picking, se usará el tamaño de la base de la carga anteriormente descrita (60x50 cm) como el espacio de trabajo máximo que analizará la cámara rgb.

- **Selección de la cámara**

Se usará como parámetro de selección el tamaño del sensor que debe tener la cámara a usar, para calcularlo se usaran la altura H, a la que se debe colocar la cámara para ver todo el espacio de trabajo el cual sería 60 cm X 50 cm, como se describió en el ítem anterior, definido como el máximo ancho y largo de la carga que lleva un operario.

Debido a que el soporte que se construirá para la cámara (se discutirán las características de este mas adelante) tiene una altura máxima aproximada de 90 cm, será este valor el usado para el cálculo del tamaño de sensor necesario.

La información presentada en este documento es de exclusiva responsabilidad de los autores y no compromete a la EIA.

Haciendo uso de la ecuación 1 es posible calcular la distancia focal necesaria, teniendo como parámetros de entrada las características del espacio de trabajo a capturar y la distancia focal de la cámara.

$$b = \frac{B*f}{D} \quad (\text{Ec. 1})$$

Donde:

f: Distancia focal

b: Tamaño del sensor

B: Ancho del objeto a capturar

D: Distancia de trabajo

Sin embargo, es necesario conocer la distancia focal de la cámara a utilizar, se usará entonces la distancia focal de una cámara que se encuentra disponible en los laboratorios de la Universidad EIA cuya referencia es Logitech C270, la cual tiene como distancia focal 4 mm.

$$b = \frac{600mm*4mm}{900mm} = 2.7 \text{ mm} \quad (\text{Ec. 2})$$

De la ecuación 2 se concluye que el tamaño de sensor adecuado es de aproximadamente 2.7 mm, debido a que el fabricante no suministra el tamaño de sensor, se realiza un procedimiento para calcularlo a partir de la ecuación 1, el procedimiento consiste en medir en mm el área del espacio de trabajo que puede capturar la cámara a cierta altura conocida y usando la ecuación 1 se calcula el tamaño del sensor de la cámara, de dicho procedimiento se obtiene que el tamaño del sensor es

La información presentada en este documento es de exclusiva responsabilidad de los autores y no compromete a la EIA.

de 2.85 mm, esto significa que a 90 cm del espacio de trabajo, la cámara lograra observar un espacio de trabajo mayor al deseado, sin embargo de ser necesario este problema se puede solucionar por medio de programación con Opencv realizando un recorte a la imagen capturada.



Figura 17 Logitech, (2018). Cámara Web C270 [Figura]. Recuperado de <https://www.logitech.com/es-roam/product/hd-webcam-c270>

Debido a cuestiones comerciales, de adquisición y como la cámara disponible cumple con las condiciones necesarias, se usará esta para la realización del trabajo.

- **Elementos de trabajo**

El objetivo final será que se puedan clasificar objetos de cualquier geometría, pero que puedan ser fácilmente alcanzables, es decir que sin tener que variar la orientación final del manipulador se logre sacarlos del espacio de trabajo. Teniendo lo anterior en cuenta se fabricarán algunas piezas con geometrías sencillas, las cuales se usarán para realizar pruebas.

A continuación, se muestran algunas de las piezas que usaran, estas fueron diseñadas con el objetivo de tener características fijas, que ayuden caracterizar el funcionamiento del algoritmo implementado.

La información presentada en este documento es de exclusiva responsabilidad de los autores y no compromete a la EIA.

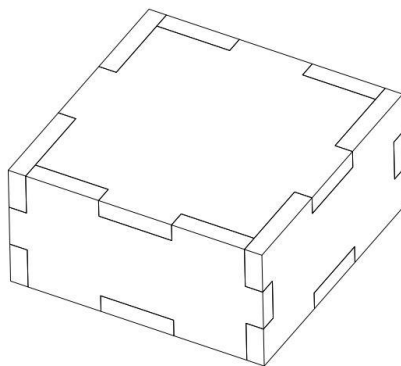


Figura 18 Pieza 1 diseñada para pruebas

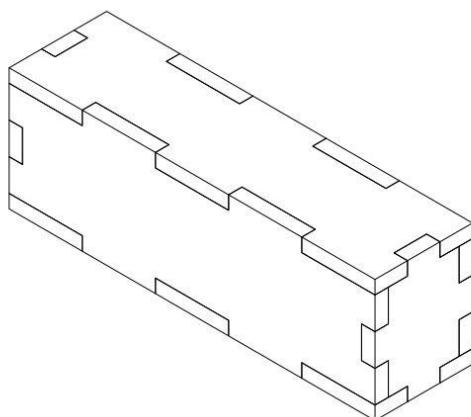


Figura 19 Pieza 2 diseñada para pruebas

- **Construcción de soporte para la cámara**

Para el diseño del soporte de la cámara se plantearon 3 necesidades que deberían ser cumplidas:

- Brindar la posibilidad de cambiar la altura configurada de la cámara.
- Permitir asignar una posición fija e invariante en el tiempo.
- Sera fácilmente desmontable y almacenable.

La información presentada en este documento es de exclusiva responsabilidad de los autores y no compromete a la EIA.

El sistema diseñado logro satisfacer las necesidades planteadas, permitiendo establecer cambios de altura desde los 5 cm hasta 90 cm aproximadamente, en pasos de 3 cm, adicionalmente el soporte permitía cambiar la posición de la cámara en una dirección adicional, lo que da mayor libertad de configuración al área de trabajo.



Figura 20 Soporte para cámara web ensamblado

La información presentada en este documento es de exclusiva responsabilidad de los autores y no compromete a la EIA.

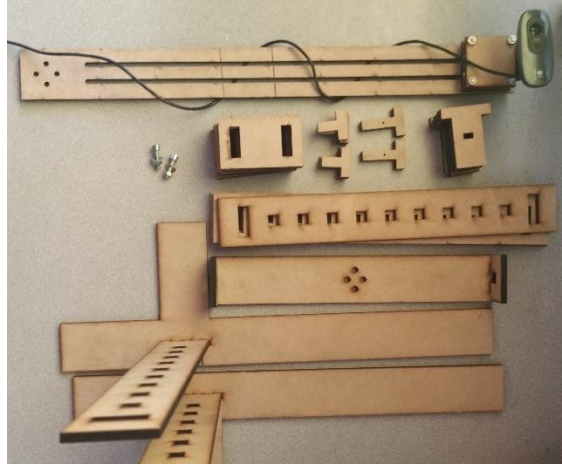


Figura 21 Soporte para cámara web desensamblado

- **Dimensionamiento de las ventosas**

- **Selección del tipo de ventosa**

Debido a que se dependerá de un algoritmo de visión artificial para la determinación de la altura a la que se encuentra una pieza es inevitable incurrir en errores a la hora de estimar dicha distancia en la escena, con el fin de tener un margen de tolerancia en la estimación de la altura se hizo uso de ventosas de fuelle, estas proveerán una tolerancia en la altura, igual a la diferencia entre el tamaño del fuelle recogido y tamaño total elongado. (SCHMALZ, 2016)



La información presentada en este documento es de exclusiva responsabilidad de los autores y no compromete a la EIA.

Figura 22 SCHMALZ (2016). Ventosas de fuelle [Figura]. Recuperado de <https://www.schmalz.com/es/saber-de-vacio/el-sistema-de-vacio-y-sus-componentes/ventosas-de-vacio/formas-de-las-ventosas/>

Ventajas de las ventosas de fuelle

- Buena adaptación a las superficies con desniveles
 - Efecto de elevación al aspirar
 - Compensación de diferencias de altura
 - Aspiración cuidadosa de piezas delicadas
- **Determinación del diámetro de la ventosa**

La fuerza de retención de una ventosa depende de su diámetro efectivo. Además, las propiedades de la pieza y el número de ventosas son factores decisivos para la fuerza de retención que puede alcanzar un sistema de vacío. (SCHMALZ, 2016)

El diámetro requerido de una ventosa puede calcularse según las fórmulas siguientes:

Para la aspiración horizontal:

$$d = 1,12 \times \sqrt{\frac{(m \times S)}{(P_U \times n)}} \quad (\text{Ec. 3})$$

m = Masa de la pieza en kg

P_U = Depresión en bar

n = Número de ventosas

S = Factor de seguridad

La información presentada en este documento es de exclusiva responsabilidad de los autores y no compromete a la EIA.

- **Factor de seguridad S:**

Dependiendo de las propiedades de la superficie de la pieza, se debe ajustar el factor de seguridad S. Como factor de seguridad se toma un valor mínimo de 1,5 para piezas lisas y no porosas. Para piezas críticas, no homogéneas, porosas, rugosas o aceitadas se debe tomar un factor de seguridad de 2,0 o superior. (SCHMALZ, 2016)

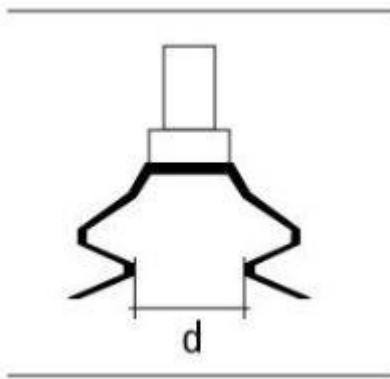


Figura 23 SCHMALZ (2016). Diámetro de la ventosa [Figura]. Recuperado de <https://www.schmalz.com/es/saber-de-vacio/el-sistema-de-vacio-y-sus-componentes/ventosas-de-vacio/dimensionamiento-de-las-ventosas/>

$$d = 1,12 \times \sqrt{\frac{(0,5 \times 3)}{(0,7 \times 1)}} = 1.64 \text{ cm} \quad (\text{Ec. 4})$$

El valor de la masa de las piezas fue un parámetro máximo asignado por el diseñador, para el cual se colocaron 500 g como el máximo a cargar, con un factor de seguridad de 3, debido a la superficie de las piezas que se quieren mover. El valor de la depresión en la fórmula anterior fue suministrados por laboratoristas.

La información presentada en este documento es de exclusiva responsabilidad de los autores y no compromete a la EIA.

La ventosa de tamaño más aproximado que se logró obtener, se puede ver en la figura 24 es una ventosa de 1.8 cm marca Festo, esta será la que se usará para el desarrollo del trabajo.



Figura 24 Ventosa de fuelle Festo

- **Mecanismo para la generación de vacío y activación de electroválvulas**

Para la generación del vacío del sistema se hizo uso de un generador de vacío marca SNS modelo SCV-15, con el fin de instalar este dispositivo en el brazo del Baxter se usó un dispositivo impreso en 3D diseñado por un estudiante de la Universidad EIA el cual permitía un montaje rápido del generador de vacío en el manipulador.



Figura 25 Ventosa y generador de vacío adaptado al manipulador izquierdo del Baxter

La información presentada en este documento es de exclusiva responsabilidad de los autores y no compromete a la EIA.

Una vez instalado el generador de vacío en el Baxter con su respectiva tubería neumática, el paso a seguir sería el sistema que controlaría la activación y desactivación de las ventosas, Baxter provee la generación de señales digitales para el control de electroválvulas actuadas a 24V, haciendo uso de este sistema y de electroválvulas 5/2 monoestables, se logró controlar el flujo de aire para la generación de vacío en la ventosa instalada en el manipulador izquierdo del Baxter.



Figura 26 Robot Baxter con las tuberías neumáticas montadas

La información presentada en este documento es de exclusiva responsabilidad de los autores y no compromete a la EIA.

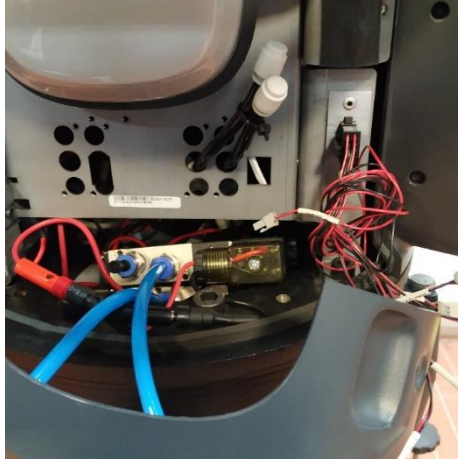


Figura 27 Conexiones eléctricas y neumáticas para la electroválvula 5/2 monoestable

2.3.2 Diseño de software

El diseño de software se dividirá en tres secciones, en la primera se registrará la programación de la robótica asociada al Baxter, en la segunda se tratará todo el sistema de visión artificial y en la última sección será donde se unifiquen la primera y segunda sección.

Antes de comenzar con las secciones del diseño de software se describirán las configuraciones iniciales del sistema para poder trabajar correctamente con ROS y el sistema operativo Ubuntu, esta descripción se hará tomando como base la información contenida en la página web del exalumno de Ingeniería Mecatrónica Daniel Cuartas, además de la página oficial de documentación para ROS. (Cuartas, 2017)

- **Paso 1 Instalación de Ubuntu 14.04 LTS:** Para la instalación de ROS y los componentes necesarios para la programación del robot Baxter, se requiere contar con el sistema operativo

La información presentada en este documento es de exclusiva responsabilidad de los autores y no compromete a la EIA.

Ubuntu 14.04 LTS, debido a los múltiples métodos de instalación posibles, el programador será el encargado de decidir los parámetros de instalación que seguirá.

- **Paso 2 Instalación de Python y OpenCV:**

Para la instalación de Python y sus respectivos paquetes el usuario deberá tener previamente instalado el gestor de paquetes Pip, con el cual se deberán instalar la versión de Python 2.7.6 y OpenCV 2.4.8.

- **Paso 3 Instalación de ROS Indigo:**

Todos los pasos deben ejecutarse de manera secuencial en la terminal de Ubuntu

Listado de recursos

```
$ sudo sh -c 'echo "deb http://packages.ros.org/ros/ubuntu trusty main" > /etc/apt/sources.list.d/ros-latest.list'
```

Keys

```
$ wget http://packages.ros.org/ros.key -O - | sudo apt-key add -
```

Update

```
$ sudo apt-get update
```

Instalar ROS Indigo Desktop Full

```
$ sudo apt-get install ros-indigo-desktop-full
```

Inicializar rosdep

```
$ sudo rosdep init  
$ rosdep update
```

Instalar rosininstall

```
$ sudo apt-get install python-roinstall
```

- **Paso 4 Crear Workspace de ROS y SDK de Baxter**

Crear Workspace de ROS

```
$ mkdir -p ~/ros_ws/src
```

Configurar fuente ROS

```
$ source /opt/ros/indigo/setup.bash
```

Crear e instalar catkin

```
$ cd ~/ros_ws  
$ catkin_make  
$ catkin_make install
```

Instalar dependencias del SDK de Baxter

```
$ sudo apt-get update  
$ sudo apt-get install git-core python-argparse python-wstool python-vcstools  
python-rosdep ros-indigo-control-msgs ros-indigo-joystick-drivers
```

Instalar SDK

```
$ cd ~/ros_ws/src  
$ wstool init .  
$ wstool merge  
https://raw.githubusercontent.com/RethinkRobotics/baxter/master/baxter\_sdk.rosinstall  
$ wstool update
```

Source ROS Setup

```
$ source /opt/ros/indigo/setup.bash
```

Construir e instalar

```
$ cd ~/ros_ws  
$ catkin_make  
$ catkin_make install
```

- **Paso 5 Configuración de parámetros de comunicación**

La información presentada en este documento es de exclusiva responsabilidad de los autores y no compromete a la EIA.

La comunicación con Baxter se hace por medio de cable ethernet, para establecer correctamente la comunicación es necesario que la conexión cableada sea configurada como solo de enlace local en IPv4. A continuación se continuara con la configuración a través de la terminal de comandos.

Para configurar la comunicación se usa el script baxter.sh el cual permite a los usuarios obtener su configuración de entorno ROS rápidamente.

Descargar baxter.sh

```
$ wget https://github.com/RethinkRobotics/baxter/raw/master/baxter.sh
$ chmod u+x baxter.sh
```

Modificación del script

```
$ cd ~/ros_ws
$ gedit baxter.sh
```

En el archivo baxter.sh se deben modificar las siguientes tres líneas y tienen que quedar de la siguiente forma.

Modificación del script

```
baxter_hostname="baxter.local"
your_ip="192.168.201.80"
ros_version="indigo"
```

La ip de conexión con Baxter se puede obtener usando el comando \$ifconfig. Una vez editado el archivo se debe guardar y cerrar para poder proceder con la inicialización del entorno SDK.

Inicie el entorno SDK

```
$ cd ~/ros_ws
$ . baxter.sh
```

Si todos los pasos anteriores se realizaron correctamente debería poder hacerse ping al Baxter y obtener una respuesta, para hacer ping al Baxter se utiliza el comando \$ ping baxter.local.

La información presentada en este documento es de exclusiva responsabilidad de los autores y no compromete a la EIA.

Si desea poder hacer simulaciones con el robot en el entorno virtual de Gazebo puede dirigirse a la página web <http://baxter.danielcuartas.com/> donde se realiza una explicación para la configuración del entorno virtual de trabajo en Gazebo.

2.3.2.1 Robótica asociada al Baxter

Todo el planteamiento robótico que se desarrollara a continuación está basado en la publicación realizada por la revista de ingeniería mecánica de la Universidad de Ohio en el año 2017, por esta razón se expondrán solo los aspectos considerados más importantes a la hora de desarrollar el programa.

Baxter es un robot humanoide que posee dos brazos con 7 grados de libertad cada uno, sin embargo, todo el desarrollo del presente trabajo se limitó al control de solo uno de estos manipuladores (Izquierdo) ya que, debido al relativamente pequeño espacio de trabajo que se podía obtener con la cámara RGB usada, usar los dos brazos se convertiría en un despropósito.

La elaboración de una solución analítica para resolver el problema cinemático del Baxter es un reto, ya que este manipulador robótico presenta dos inconvenientes importantes que hacen que en un caso estricto, la cinemática del robot sea imposible de solucionar, el primer y más sobresaliente es el hecho de que el robot es redundante, debido a su séptimo grado de libertad, esto hace que sea imposible encontrar una solución analítica para el sistema, dejando solo paso a la posibilidad de resolver la cinemática de manera numérica, ahora la segunda limitación viene de analizar el criterio de Pieper, el cual afirma que si en un robot de 6 grados de libertad, 3 de sus articulaciones se intersectan en un punto, se puede garantizar que hay una solución analítica para

La información presentada en este documento es de exclusiva responsabilidad de los autores y no compromete a la EIA.

el sistema, analizando este criterio los autores del artículo proponen una solución para la primera limitación, la cual consistiría en analizar el sistema como si solo tuviera 6 grados de libertad limitando la articulación $\theta_3 = 0$, sin embargo esto no resuelve el problema de la segunda limitación, y es que en los manipuladores del Baxter no existen 3 articulaciones consecutivas que se intersequen en un punto, debido a un offset de 10 mm en L5 (Ver figura 28). La solución que proponen los autores es hacer $L_5 = 0$ debido al hecho de que, en comparación con el resto de medidas del manipulador, estos 10 mm son insignificantes, así entonces se resuelven las restricciones del manipulador y se procede a solucionar la cinemática del mismo.

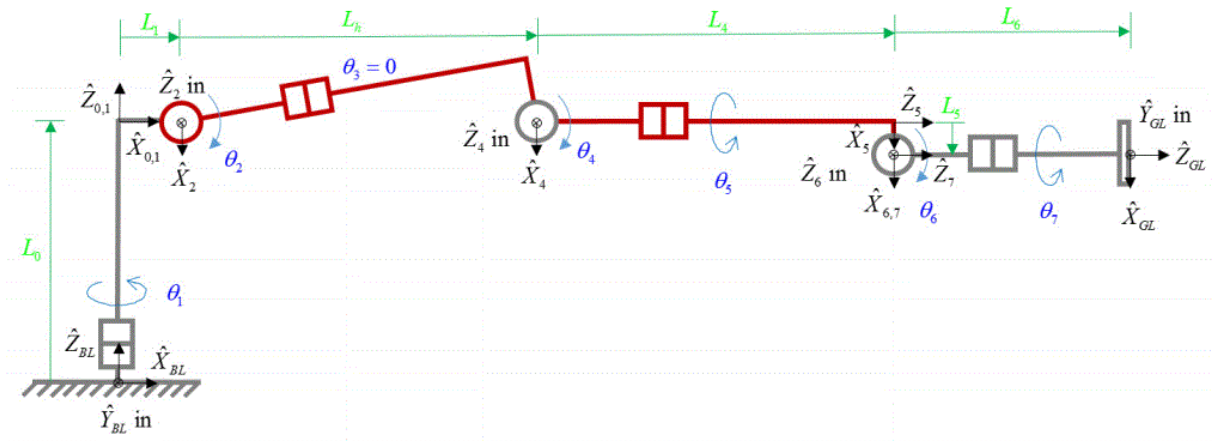


Figura 28 (II, 2017) . Diagrama de cinemática para el manipulador izquierdo con 6 grados de libertad y $\theta_3 = 0$ [Figura]. Recuperado de <https://www.ohio.edu/mechanical-faculty/williams/html/pdf/BaxterKinematics.pdf/>

El primer paso para el cálculo de la TCI del manipulador es calcular la TCD, para esto se plantea la tabla de parámetros Denavit-Hartenberg basados en las figuras 28, 29 y 30, la tabla de parámetros con todas las transformaciones para los sistemas coordenados se puede ver en la tabla 10.

La información presentada en este documento es de exclusiva responsabilidad de los autores y no compromete a la EIA.

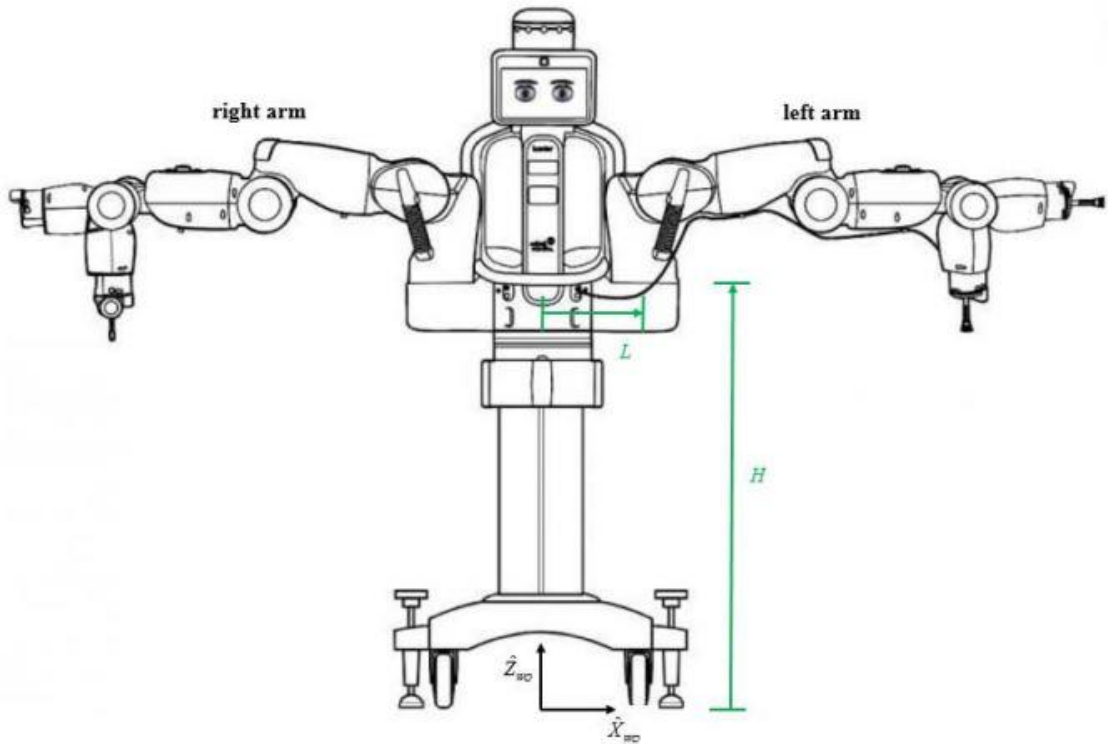


Figura 29 (II, 2017) . Vista frontal de Baxter [Figura]. Recuperado de <https://www.ohio.edu/mechanical-faculty/williams/html/pdf/BaxterKinematics.pdf/>

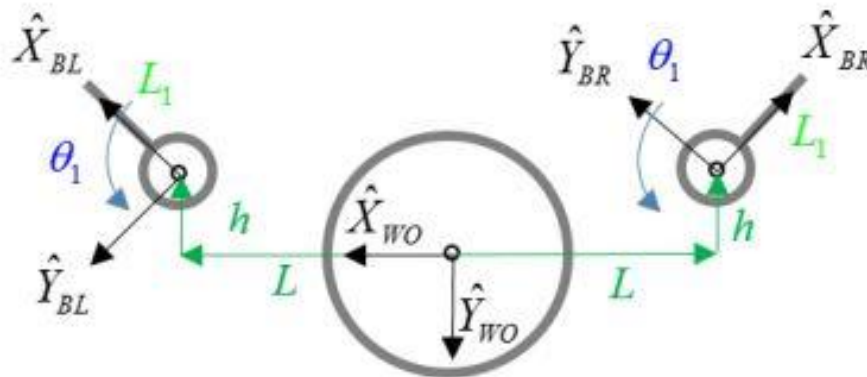


Figura 30 (II, 2017) . Detalle de la vista superior de Baxter [Figura]. Recuperado de <https://www.ohio.edu/mechanical-faculty/williams/html/pdf/BaxterKinematics.pdf/>

La información presentada en este documento es de exclusiva responsabilidad de los autores y no compromete a la EIA.

Tabla 10

Tabla de parámetros Denavit-Hartenberg para Baxter

Transformación	α_{i-1}	a_{i-1}	d_i	θ_i
TWL	0	L	HW0	0
TLh	90°	0	h	0
ThB	-90°	0	0	-45°
TB0	0	0	L0	0
T01	0	0	0	θ_1
T12	-90°	L1	0	θ_2
T23	0	Lh	0	θ_4+90°
T34	90°	0	L4	θ_5
T45	-90°	L5	0	θ_6
T56	90°	0	0	θ_7
T6GL	0	0	L6	0

Tabla 11

Tabla de parámetros necesarios

Parámetro	Valor (mm)
L	278
h	64
HW0	1104
L0	270.35
L1	69
L2	364.35
L3	69
L4	374.29
L5	10
L6	368.3
Lh	$\sqrt{L_2^2 + L_3^2}$

A partir de la información anterior es posible obtener todas las ecuaciones y transformaciones necesarias para la solución de la cinemática inversa del Baxter, la primera matriz que se debe obtener para el cálculo de la TCI será la matriz de transformación total del sistema TWG, para esto primero se debe determinar la orientación que se desea para el efector final, en este caso se desea

La información presentada en este documento es de exclusiva responsabilidad de los autores y no compromete a la EIA.

que el efector final siempre llegue a los objetos con una orientación de 90° sobre el plano XY, para garantizar esto se realiza la TCD haciendo que todas las articulaciones tengan 0°, excepto, la articulación θ_6 que en este caso será 90°, el resultado será la siguiente matriz de transformación.

$$TWG = \begin{pmatrix} -0.7071 & 0.70701 & 0 & X \\ 0.7071 & 0.70701 & 0 & Y \\ 0 & 0 & -1 & Z \\ 0 & 0 & 0 & 1 \end{pmatrix} \quad (\text{Ec. 5})$$

Como se puede observar de la matriz obtenida al realizar todas las transformaciones del sistema, aquella información que interesa es la matriz de rotación correspondiente, ya que X, Y y Z serán la posición a la que deseo que el efector final llegue.

Se necesitan adicionalmente las siguientes transformaciones.

$$TWB = \begin{pmatrix} 0.7071 & 0.70701 & 0 & L \\ -0.7071 & 0.70701 & 0 & -h \\ 0 & 0 & 1 & HW0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \quad (\text{Ec. 6})$$

$$TB0 = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & L0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \quad (\text{Ec. 7})$$

$$T6GL = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & L6 \\ 0 & 0 & 0 & 1 \end{pmatrix} \quad (\text{Ec. 8})$$

Donde TWB es el resultado del realizar el producto entre las transformaciones $TWL * TLh * ThB$.

Se calcula T06 como:

La información presentada en este documento es de exclusiva responsabilidad de los autores y no compromete a la EIA.

$$T06 = TB0^{-1} * TWB^{-1} * TWG * T6GL^{-1} \quad (Ec. 9)$$

Como forma de ilustración se tiene que las posiciones en todas las matrices dadas serán las siguientes:

$$T = \begin{pmatrix} (1,1) & (1,2) & (1,3) & (1,4) \\ (2,1) & (2,2) & (2,3) & (2,4) \\ (3,1) & (3,2) & (3,3) & (3,4) \\ (4,1) & (4,2) & (4,3) & (4,4) \end{pmatrix} \quad (Ec. 10)$$

Teniendo en cuenta lo anterior se tiene que:

$$\theta_1 = atan2(T06(2,4), T06(1,4)) \quad (Ec. 11)$$

Calcularemos algunas expresiones auxiliares:

$$E = 2 * Lh * \left(L1 - \left(\frac{T06(1,4)}{\cos(\theta_1)} \right) \right) \quad (Ec. 12)$$

$$F = 2 * Lh * T06(3,4) \quad (Ec. 13)$$

$$G = \frac{T06(1,4)^2}{\cos(\theta_1)^2} + L1^2 + Lh^2 - L4^2 + T06(3,4)^2 - \frac{2*L1*T06(1,4)}{\cos(\theta_1)} \quad (Ec. 14)$$

$$t12 = \frac{-F \pm \sqrt{E^2 + F^2 - G^2}}{G - E} \quad (Ec. 15)$$

La ecuación t12 define si las posiciones serán alcanzadas codo arriba o codo abajo.

$$\theta_2 = 2 * atan(t12) \quad (Ec. 16)$$

$$\theta_4 = atan2\left(-T06(3,4) - Lh * \sin(\theta_2), \frac{T06(1,4)}{\cos(\theta_1) - L1 - Lh * \cos(\theta_2)}\right) - \theta_2 \quad (Ec. 17)$$

La información presentada en este documento es de exclusiva responsabilidad de los autores y no compromete a la EIA.

Se obtiene la matriz de rotación R03 a partir de la transformación T03 y adicionalmente se obtiene la matriz de rotación R06

$$T03 = T01 * T12 * T23 \quad (Ec. 18)$$

$$R03 = T03(1:3,1:3) \quad (Ec. 19)$$

$$R06 = T06(1:3,1:3) \quad (Ec. 20)$$

Se obtiene la matriz de rotación R36 como:

$$R36 = R03^{-1} * R06 \quad (Ec. 21)$$

$$\theta_5 = atan2(R36(3,3), R36(1,3)) \quad (Ec. 22)$$

$$\theta_7 = atan2(-R36(2,2), R36(2,1)) \quad (Ec. 23)$$

$$\theta_6 = atan2\left(\frac{R36(2,1)}{\cos(\theta_7)}, -R36(2,3)\right) \quad (Ec. 24)$$

Se obtiene así la cinemática inversa del manipulador izquierdo del robot Baxter.

Con el fin de comprobar que el programa se realizó correctamente se simuló en el software Matlab el comportamiento que tendría el manipulador.

En la figura 31 se puede observar el manipulador en la posición donde todos los actuadores del sistema están en 0°, para esto se realizó de nuevo la matriz de TWG, asegurando que la orientación del efector final fuera 0°. Se puede observar a su vez que la simulación se realizó simplificando el vínculo entre la articulación θ_2 y θ_3 a una medida fija Lh.

La información presentada en este documento es de exclusiva responsabilidad de los autores y no compromete a la EIA.

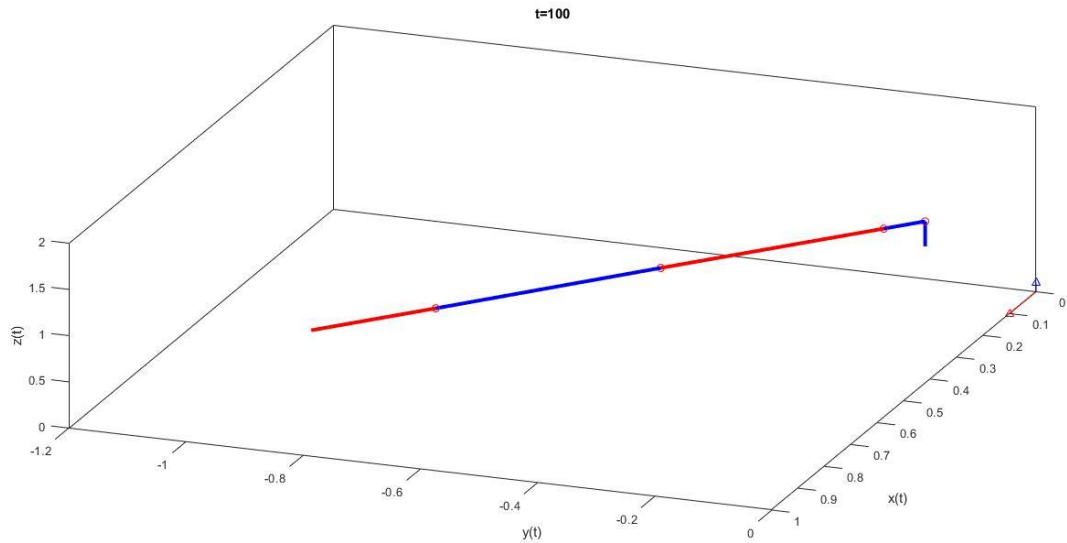


Figura 31 Simulación en Matlab para la posición de 0° en todas las articulaciones del manipulador izquierdo de Baxter

En la figura 32 se puede observar la simulación realizada haciendo que el efector final tenga una orientación de 90° sobre el plano XY, la línea punteada hace referencia a la trayectoria que describirá el efector final desde la posición en la que encontraba inicialmente

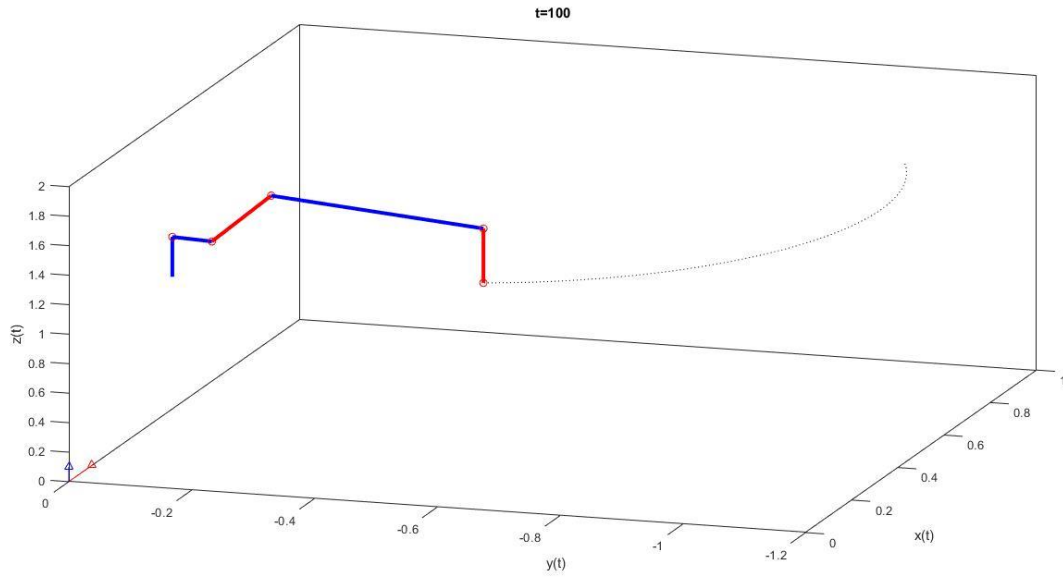


Figura 32 Simulación en Matlab para una posición aleatoria en el espacio con orientación del efector final a 90°

2.3.2.2 Programa de visión artificial implementado

Para cumplir los objetivos del trabajo era necesario suplir las siguientes necesidades:

- Elaboración de un método que permita asignar una característica común a un grupo de objetos, con el fin de poder llevar a cabo una clasificación de objetos
- Determinar la altura respecto a la cámara a la que se encuentra un objeto
- Determinar la posición XY de un objeto respecto a la cámara

Para solucionar la primera necesidad se define que la característica que se identificara en los objetos y dará la pauta para clasificación será el color, cada pieza que ingrese en la escena tendrá asignada una etiqueta de color, la cual deberá ser reconocida y servirá como parámetro de clasificación.

La información presentada en este documento es de exclusiva responsabilidad de los autores y no compromete a la EIA.

Lo siguiente a tratar será el método usado para obtener la altura de una pieza en una escena, este método fue tomado y adaptado del artículo que lleva por nombre “Estimación de la distancia a un objeto con visión computacional”, en donde haciendo uso del algoritmo SURF se calcula la altura con respecto a la cámara a la que se encuentra un objeto, sin embargo, para aplicar este método es necesario tener al menos una imagen de referencia del objeto a reconocer y conocer adicionalmente la distancia a la que se toma dicha imagen, teniendo esto en cuenta si todos los objetos a identificar en la escena son distintos entre sí, implicaría la necesidad de poseer un base de datos en la que se almacenen la información de referencia de cada nuevo objeto que se introdujera en la escena. (Magaña Z., Atoche E., Molina C., & Pérez C., 2017)

Por ultimo para calcular la distancia XY de los objetos en la escena se calculó la correspondencia entre los pixeles capturados por la cámara RGB y el mundo real, en otras palabras, cuántos pixeles de la imagen capturada corresponde a 1 cm en el mundo físico, sin embargo, para aplicar correctamente este método es necesario garantizar que la posición del centro de cada objeto en la escena sea una posición alcanzable por el actuador final del manipulador.

Planteado lo anterior se diseña una etiqueta que permitirá solucionar las tres necesidades planteadas, esta etiqueta tiene tres características, la primera es un borde exterior de un color específico, se eligieron 3 colores distintos los cuales permitirán clasificar los objetos en la escena en 3 grupos diferentes. La segunda característica es un grueso marco de color negro el cual desempeña un papel fundamental a la hora de identificar la etiqueta en la escena ya que genera un contorno rectangular cerrado alrededor de cada etiqueta, creando adicionalmente la posibilidad de obtener un centroide bien definido para cada objeto, por último el logo central de la etiqueta será

La información presentada en este documento es de exclusiva responsabilidad de los autores y no compromete a la EIA.

a partir del cual el algoritmo SURF obtendrá los descriptores de características invariantes en la imagen y serán a partir de estos que se calcule la distancia Z a los objetos.



Figura 33 Etiquetas diseñadas para el algoritmo de visión artificial

Una vez definido el objeto de interés que se buscará en la escena, se inicia la elaboración del código que la interpretará, el primer paso será reconocer todas las etiquetas visibles en el frame que se está analizando en un momento determinado, para esto se aplica inicialmente un filtro de desenfoque gaussiano, posteriormente se binariza toda la escena con la técnica denominada umbral adaptativo gaussiano, al final del procedimiento se obtiene lo que se puede observar en la figura 34.

La información presentada en este documento es de exclusiva responsabilidad de los autores y no compromete a la EIA.



Figura 34 Umbral adaptativo gaussiano aplicado a la escena capturada por la cámara RGB

Como se puede observar las imágenes presentan gran cantidad de ruido producido por el entorno, sin embargo, es posible identificar el contorno rectangular definido por el marco negro de cada etiqueta, con el fin de seleccionar solo esta información de interés se implementa un algoritmo para la identificación de contornos, dicho algoritmo solo considerara como un contorno valido a toda figura geométrica cerrada cuya área de contorno sea mayor a 1000. Además, haciendo uso de los momentos de Hue se calculará el centroide del contorno encontrado.



Figura 35 Resultado del cálculo de centroide e identificación de etiquetas en la escena

La información presentada en este documento es de exclusiva responsabilidad de los autores y no compromete a la EIA.

En la figura 35 se puede observar dos resultados, el primero es el contorno hallado por el algoritmo, dibujado en color verde sobre la imagen original y el segundo es un punto central de color rosado el cual corresponde al centroide del contorno hallado, el cual se usará más adelante para determinar la posición XY del objeto en la escena.

Identificadas las etiquetas en la imagen, se obtendrán y almacenaran dos imágenes de la escena, en la figura 36 se puede observar una imagen en escala de grises la cual corresponde a la imagen del logo que será utilizada por el algoritmo SURF.



Figura 36 Recorte del logo de la etiqueta que será usado en la implementación de SURF

La segunda imagen se puede observar en la figura 37, esta imagen es el resultado de ampliar el área de recorte con el fin de obtener información del borde de color de la etiqueta que se está analizando, eliminado a su vez gran cantidad de ruido que pueda perjudicar la determinación de color.



Figura 37 Recorte de la etiqueta encontrada para realizar la identificación del color

La información presentada en este documento es de exclusiva responsabilidad de los autores y no compromete a la EIA.

A la segunda imagen (Figura 37) se aplican dos operaciones adicionales, la primera consiste en cambiar el espacio de color de la imagen a HSV, obteniendo el resultado de la figura 38



Figura 38 Cambio del espacio de color a HSV aplicado a la figura 37

Luego usando el umbral que define la tonalidad de amarillo usada en la etiqueta se realiza un binarizado de umbral simple, el cual genera el resultado de la figura 39, así que, usando la función para encontrar contornos en una imagen y conociendo el umbral definido para un color, es posible determinar si un color específico puede ser encontrado en una imagen.



Figura 39 Binarización de imagen aplicada a la figura 38

La información presentada en este documento es de exclusiva responsabilidad de los autores y no compromete a la EIA.

El procedimiento realizado con la figura 37 se realiza de nuevo usando los umbrales que describen el color rosado y verde de las etiquetas, debido a que no se encontraran contornos con umbrales de color distinto al correcto se descartan posibles errores en la identificación del color.

Una vez determinado el color de la etiqueta, se continúa trabajando con la figura 36, con el objetivo de estimar la posición en Z del objeto en la escena, esta sección es más automática ya que OpenCV cuenta con funciones para facilitar la implementación del algoritmo SURF y la realización del keypoint matching.

Para la realización del Keypoint matching se requieren de dos imágenes, la primera será la imagen capturada del entorno y corresponde a la figura 36 y la segunda imagen corresponde a la imagen de referencia, esta imagen no cambiara y se usará siempre que se requiera calcular la posición en Z, se puede observar en la figura 40



Figura 40 Imagen de referencia para algoritmo SURF

Definidas estas imágenes se obtienen y almacenan todos los descriptores y Keypoints encontrado para cada imagen y se procede con el emparejamiento, el cual es realizado de manera automática por una función que hace parte de OpenCV denominada BFMatcher, para determinar si un emparejamiento realizado por BFMatcher es correcto se usó el criterio de matching establecido por David G. Lowe, el cual propone, que si se descartan todos los emparejamientos

La información presentada en este documento es de exclusiva responsabilidad de los autores y no compromete a la EIA.

cuya relación de distancia es mayor de 0.8 se descartarán el 90% de los emparejamientos erróneas mientras solo el 5% de los emparejamientos correctos serán descartados. (Lowe, Distinctive Image Features, 2004)

Al finalizar el match se obtiene como resultado la figura 41, en donde se pueden observar las líneas de color que indican los puntos que fueron emparejados al comparar la imagen de la etiqueta obtenida de la escena, con la imagen que se tiene de referencia para dicha etiqueta.

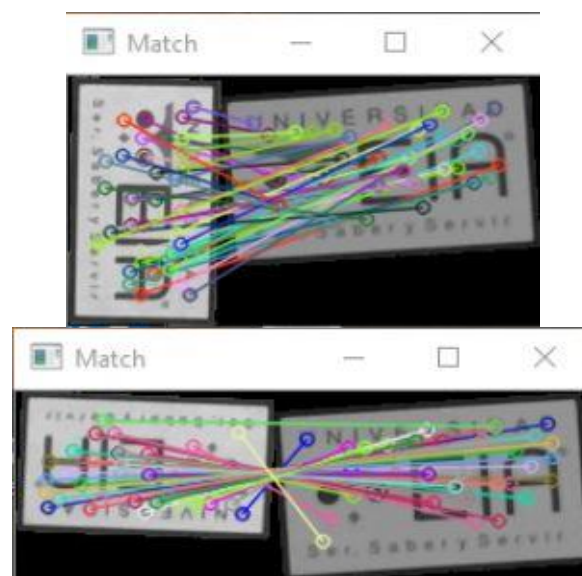


Figura 41 Resultado del emparejamiento de dos etiquetas identificadas en la escena con la imagen de referencia común

Una vez se tienen emparejados los descriptores de la imagen de referencia con los descriptores de la imagen del entorno, se calculará la distancia euclidiana entre cada par de puntos posibles para la imagen de referencia y la imagen del entorno, haciendo uso de estas distancias se aplicará la siguiente ecuación con la cual se estimará el valor de la distancia en Z para cada par de puntos emparejados.

La información presentada en este documento es de exclusiva responsabilidad de los autores y no compromete a la EIA.

$$Dest = \frac{Dref * DecRef}{DecEnt} \quad (Ec. 25)$$

- Dest: Distancia estimada
- Dref: Distancia de referencia correspondiente con la distancia a la que se tomó la imagen de referencia
- DecRef: Distancia euclidiana calculada para cada par de puntos de la imagen de referencia
- DecEnt: Distancia euclidiana calculada para cada par de puntos de la imagen del entorno

Los valores de distancia estimados se almacenarán en un vector, el cual luego se ordenará de menor a mayor para posteriormente obtener la mediana del mismo, la cual corresponderá al valor final estimado para la distancia Z.

De esta manera finaliza el cálculo de la distancia Z a la que se encuentra un objeto en la escena, para el cálculo de la posición XY se realiza una regla de tres simple como sigue.

$$Xreal = \frac{Xesc * Xpix}{XpixTot} \quad (Ec. 26)$$

Xreal: Distancia estimada en cm para una posición X deseada

Xesc: Cuantos cm mide el ancho de la escena capturada por la cámara

Xpix: Posición X en pixeles que se requiere estimar

XpixTot: Ancho total en pixeles de la escena

$$Yreal = \frac{Yesc * Ypix}{YpixTot} \quad (Ec. 27)$$

La información presentada en este documento es de exclusiva responsabilidad de los autores y no compromete a la EIA.

Yreal: Distancia estimada en cm para una posición Y deseada

Yesc: Cuantos cm mide el largo de la escena capturada por la cámara

Ypix: Posición Y en pixeles que se requiere estimar

YpixTot: Largo total en pixeles de la escena

En donde Xpix y Ypix son los valores de posición del centroide calculado (figura 35) para cada etiqueta presente en la escena.

2.3.2.3 Unificación de los algoritmos de visión artificial y robótica para el control de movimiento del Baxter

Debido a un problema relacionado con la versión de OpenCV, que ocasionaba que la lectura de los frames obtenidos con la cámara RGB no ocurriera en tiempo real, se volvió necesario implementar un hilo o proceso paralelo que se encargara de la adquisición en tiempo real de la escena. Una vez adquirido el frame de la escena se aplican todos los procesos y operaciones descritas en la sección de visión artificial, al final del análisis se obtendrá una lista con cuatro posiciones, la cual entrega, posición Z, X, Y y el color detectado en la etiqueta, como se ve a continuación.

$$Vector = [39.85634336157208, 19, 8.933333333333334, 'Amarillo']$$

Los datos que provee esta lista debían ser pos procesados debido a que la información de posición era dada respecto a la ubicación fija de la cámara y no respecto al sistema de coordenadas universal

La información presentada en este documento es de exclusiva responsabilidad de los autores y no compromete a la EIA.

asignado en la base del robot Baxter, así que para realizar la correspondencia cartesiana entre el espacio de trabajo y el Baxter, se realizaron una serie de operación las cuales serán descritas más adelante.



Figura 42 Sistemas coordenados asignado al Baxter y al espacio de trabajo

Como se puede observar en la figura 42, existen 2 sistemas coordenados, el sistema coordenado W_0 corresponde directamente con el sistema coordenado establecido en la figura 29, el cual sería

La información presentada en este documento es de exclusiva responsabilidad de los autores y no compromete a la EIA.

el sistema de coordenadas universal asignado al robot Baxter y respecto el cual Baxter calcularía la TCI y el sistema de coordenadas ESC el cual está dividido en dos partes por facilidad, la primera parte corresponde a el plano XY y señala la ubicación del pixel (0,0) y la segunda parte corresponde a la coordenada Z la cual tiene como cero la posición de la cámara RGB.

Para lograr que los sistemas coordenados se correspondan se realizan las siguientes operaciones

$$Z_{w0} = \frac{(Z_{esc} - Z_{est})}{100.0} \quad (Ec. 28)$$

$$Y_{w0} = \frac{(Y_{esc} - Y_{est})}{100.0} \quad (Ec. 29)$$

$$X_{w0} = \frac{(X_{esc} + X_{est})}{100.0} \quad (Ec. 30)$$

Donde:

Z_{w0} , Y_{w0} , X_{w0} : Distancias medidas desde el sistema coordenado universal del robot Baxter.

Z_{esc} : Altura medida desde el sistema de coordenadas $w0$ a la cámara RGB.

Z_{est} : Distancia estimada al objeto usando SURF.

Y_{esc} , X_{esc} : Y real y X real calculados en la sección de visión artificial.

Como se puede observar las ecuaciones están divididas sobre 100, debido a que las distancia serán ingresadas a la TCI en metros.

La información presentada en este documento es de exclusiva responsabilidad de los autores y no compromete a la EIA.

Los valores de posición anteriormente calculados son ingresados en la matriz TWG de la que se habló en la sección de robótica, a partir de la cual se calculará la cantidad de movimiento que deberá hacer cada articulación del manipulador para alcanzar la posición cartesiana deseada (TCI). Sin embargo, la posición final será alcanzada realizando dos movimientos, el primero posicionará el efector final a 5 cm sobre la pieza que se va a extraer, como se ve en la figura 43.

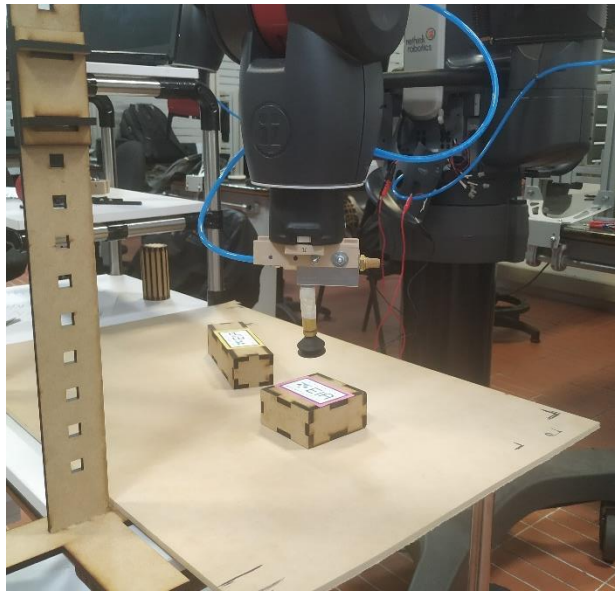


Figura 43 Posición inicial de extracción

Una vez alcanzada la posición inicial se realiza de nuevo la TCI, pero esta vez el efector final bajara los 5 cm restantes para alcanzar la pieza, como se ve en la figura 44.

Este procedimiento debió ejecutarse de esta manera ya que, si se completaba la TCI en un solo movimiento, el actuador golpeaba los objetos en la escena lo cual era indeseado.

La información presentada en este documento es de exclusiva responsabilidad de los autores y no compromete a la EIA.

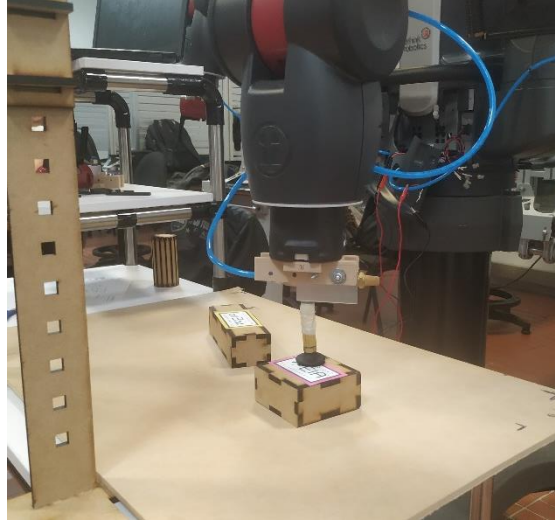


Figura 44 Posición final de sujeción

Cabe aclarar que la pieza que será seleccionado para extracción, será la pieza más alta en la escena, esto evitará que haya colisiones con las mismas al tratar de sacar una pieza de menor tamaño.

Como procedimiento final la pieza seleccionada será llevada a una posición fija, previamente definida, dependiendo del color de su etiqueta.

2.3.2.4 Interfaz de usuario

La interfaz de usuario diseñada se puede observar en la figura 45, en esta se programaron dos botones de función con los cuales se podrá tanto inicializar el programa, como detenerlo, además en ella se pueden visualizar la cantidad de objetos clasificados para cada color, y se le permitirá al usuario ver el frame que se está analizando en el momento y las etiquetas encontradas en la escena.

La información presentada en este documento es de exclusiva responsabilidad de los autores y no compromete a la EIA.

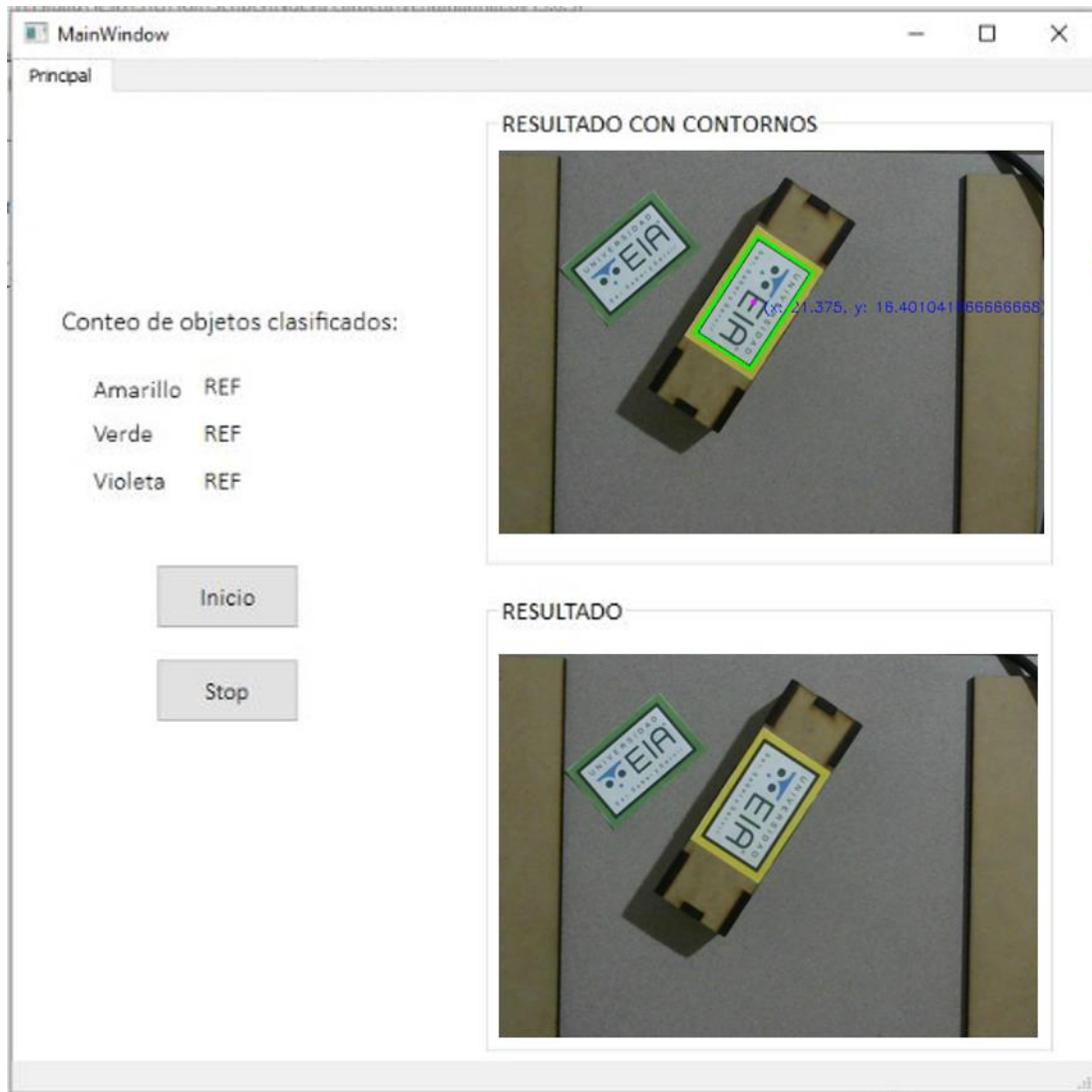


Figura 45 Interfaz de usuario

La información presentada en este documento es de exclusiva responsabilidad de los autores y no compromete a la EIA.

3. PRESENTACIÓN Y DISCUSIÓN DE RESULTADOS

Una vez se finalizada la implementación del programa se iniciaron las pruebas de selección y clasificación de objetos con el fin de cuantificar el nivel de confiabilidad del sistema implementado

- **Primera prueba: cuantificación del error en estimación de la altura**

La primera variable que se cuantifico fue la exactitud que se obtenía a la hora de estimar la distancia haciendo uso del algoritmo SURF. Como se definió en el apartado de selección de cámara con el fin de observar un área de trabajo de 60X50 cm, esta se situaría a 84 cm del espacio de trabajo, sin embargo, cuando comenzaron las pruebas se notó impresión a la hora de estimar la posición en Z, registrándose errores de estimación de hasta 11 cm con la cámara situada en dicha posición. Esto llevo a estudiar si la altura a la que se encontraba la cámara sobre el área de trabajo influía en la precisión con la que el algoritmo estimaba.

Se registraron entonces los valores estimados con la cámara instalada a 4 diferentes alturas y se calculó el error, los resultados obtenidos fueron los siguientes

- **Altura del objeto a la cámara 24.6 cm**

Se registraron 62 datos de los cuales solo fue posible usar 42, debido a que, a la hora de trabajar sobre esta altura, el programa tenía problemas identificando la etiqueta en la escena en otras palabras solo en el 67.7% de los intentos se puedo estimar la altura de los objetos en la escena.

La información presentada en este documento es de exclusiva responsabilidad de los autores y no compromete a la EIA.

Haciendo uso de los datos obtenidos para la altura se calculó el error porcentual respecto a la medida real, se obtuvo que el error de estimación en la medida era del 1.74%, lo que sería equivalente a un error de 0.43 cm en la estimación de la distancia.

- **Altura del objeto a la cámara 39.7 cm**

Se registraron 90 datos con los cuales se calculó el error porcentual en la estimación de la altura, el resultado fue un error del 0.97%, lo que sería equivalente a 0.38 cm de error en la estimación de la altura.

- **Altura del objeto a la cámara 60 cm**

Se registraron 90 datos con los cuales se calculó el error porcentual en la estimación de la altura, el resultado fue un error del 0.46%, lo que sería equivalente a 0.27 cm de error en la estimación de la altura.

- **Altura del objeto a la cámara 72 cm**

Se registraron 90 datos con los cuales se calculó el error porcentual en la estimación de la altura, el resultado fue un error del 3%, lo que sería equivalente a 2.17 cm de error en la estimación de la altura.

Es necesario tener en cuenta que el error calculado viene de promediar los datos obtenidos en las mediciones hechas, lo que disminuiría el error general percibido, por ejemplo, en el último caso donde la distancia al objeto era de 72 cm, una de las mediciones hechas obtuvo un error de estimación de más de 10 cm, error que no podría ser corregido usando los instrumentos disponibles, este tipo de error solo se percibió en el último caso donde la altura de la cámara superaba los 60 cm.

La información presentada en este documento es de exclusiva responsabilidad de los autores y no compromete a la EIA.

- **Segunda prueba: cuantificación del nivel de confiabilidad del sistema construido.**

La prueba diseñada consistía en extraer dos piezas de prueba las cuales se situarían aleatoriamente en el área de trabajo, si la pieza se extraía satisfactoriamente del área de trabajo contaría como una selección correcta, luego de extraer las dos piezas estas se posicionarían nuevamente en el área de trabajo, los resultados obtenidos se pueden ver a continuación.

Tabla 12

Resultados prueba de selección

Intento	Resultado de la selección	Pieza extraída
1	Correcta	1
2	Correcta	2
3	Correcta	2
4	Correcta	1
5	Correcta	2
6	Correcta	1
7	Correcta	2
8	Correcta	1
9	Correcta	2
10	Correcta	2
11	Correcta	1
12	Correcta	1
13	Correcta	1
14	Correcta	2
15	Incorrecta	1
16	Incorrecta	2
17	Correcta	2
18	Correcta	1
19	Correcta	2
20	Correcta	1
21	Correcta	2
22	Correcta	1
23	Correcta	1
24	Incorrecta	1
25	Correcta	1
26	Correcta	2

La información presentada en este documento es de exclusiva responsabilidad de los autores y no compromete a la EIA.

27	Correcta	2
28	Correcta	1
29	Correcta	1
30	Correcta	1
31	Correcta	2
32	Correcta	2
33	Correcta	1
34	Correcta	1
35	Correcta	2
36	Incorrecta	2
37	Correcta	2
38	Correcta	2
39	Incorrecta	1
40	Correcta	1
41	Correcta	1
42	Correcta	1
43	Correcta	2
44	Correcta	2
45	Correcta	1
46	Correcta	2
47	Correcta	1
48	Correcta	2
49	Incorrecta	2
50	Correcta	1
51	Correcta	1
52	Correcta	1
53	Correcta	2
54	Correcta	2
55	Correcta	1
56	Incorrecta	2
57	Correcta	1
58	Correcta	2
59	Correcta	2
60	Correcta	1

Se puede observar que de las 60 pruebas de selección hechas solo se cometieron errores de selección en 7 ocasiones, con esto se concluye que el sistema presenta una confiabilidad de selección del 88.33%.

La información presentada en este documento es de exclusiva responsabilidad de los autores y no compromete a la EIA.

Por último, para la prueba de clasificación de color, se tiene que si el sistema está correctamente calibrado y la selección de los umbrales de color para las etiquetas se hizo correctamente en el entorno de iluminación en el que trabajara el sistema, el 100% de las veces el color se determinara correctamente.

La información presentada en este documento es de exclusiva responsabilidad de los autores y no compromete a la EIA.

4. CONCLUSIONES

Se concluye a través de las pruebas realizadas, que la distancia a la que se encuentra la cámara del área trabajo afecta el resultado de la estimación de la posición, por esta razón y teniendo en cuenta la mayor altura a la que se puede colocar la cámara sin afectar la estimación realizada por el algoritmo SURF, se define como nueva altura 60 cm sobre la base del área de trabajo, con la desventaja de que ahora el espacio de trabajo se reduce a 25x50 cm, ya que a pesar de que la cámara capturaba un área mayor, debido a la posición de la misma se tuvo que alejar el espacio de trabajo del robot para evitar colisiones constantes y aquellos puntos que estaban más allá de 25 cm no eran alcanzado por el manipulador.

Se pude observar que cuando los objetos en la escena se alejan del área central de la imagen capturada por la cámara, el error en la estimación de la distancia XY empeora, sin embargo, no es tan grave como para impedir la selección del objeto en la escena, esto se puede observar ya que el actuador final no se posiciona exactamente sobre el centro de la etiqueta, este efecto probablemente sea ocasionado por la curvatura del lente de la cámara.

El diseño de sistema de succión debe ser mejorado debido a que en ocasiones luego de haber extraído piezas de la escena, la manguera que une el generador de vacío y la ventosa se tuerce impidiendo que la pieza seleccionada pueda ser extraída.

Es necesario estudiar, que otros parámetros afectan a la estimación que hace el algoritmo SURF, con el fin de seleccionar una cámara que brinde mejores características y elimine las limitaciones que se presentaron en el sistema montado.

La información presentada en este documento es de exclusiva responsabilidad de los autores y no compromete a la EIA.

5. CONSIDERACIONES FINALES

Todos los objetivos planteados para el desarrollo fueron cumplidos, logrando generar una solución a la necesidad de automatizar procesos repetitivos de gran importancia como lo son la selección y clasificación de objetos.

A pesar de que el sistema cumple su objetivo, el mismo tiene grandes oportunidades de mejora, ya que los dispositivos usados para el desarrollo del proyecto, como el sistema de succión, la cámara RGB o el soporte diseñado, son soluciones económicas que puede llegar a limitar considerablemente los resultados obtenidos.

Debido a que con este tipo sistema no es posible lograr una confiabilidad del 100%, la necesidad de que hallan operarios en el área de trabajo no se elimina, por esta razón Baxter es una excelente elección como manipulador robótico para llevar a cabo este tipo de actividades en donde la presencia de personas en el área de trabajo es importante.

A pesar de las restricciones que posee el algoritmo SURF, ya que el mismo se encuentra patentado y el uso que se le puede dar es únicamente educativo, aún existe otro tipo de soluciones que usan técnicas similares con las cuales se pueden alcanzar buenos resultados.

6. BIBLIOGRAFÍA

- Autodesk. (2 de abril de 2019). *Autocad*. Obtenido de <https://knowledge.autodesk.com/es/support/autocad/learn-explore/caas/CloudHelp/cloudhelp/2017/ESP/AutoCAD-Core/files/GUID-C0C610D0-9784-4E87-A857-F17F1F7FEEBE-htm.html>
- Bay, H., Tuytelaars, T., & Gool, L. V. (2006). SURF: Speeded Up Robust Features. *ETH Zurich*.
- Craig, J. J. (2006). *Robótica*. México: Pearson.
- Cuartas, D. (04 de 09 de 2017). *Proyecto de Aula Robotica Industrial*. Obtenido de <http://baxter.danielcuartas.com/>
- D.G, L. (1999). Object recognition from local scale-invariant features. *IEEE*.
- Hernández, J. M., Martinsanz, G. P., & Mata-García, M. G. (2011). Técnicas de procesamiento de imágenes estereoscópicas. *Dialnet*.
- II, R. W. (04 de 2017). *Baxter Humanoid Robot Kinematics*. Obtenido de <https://www.ohio.edu/mechanical-faculty/williams/html/pdf/BaxterKinematics.pdf>
- J.Cecil, Powell, D., & Vasquez, D. (2007). Assembly and manipulation of micro devices—A state of the art survey. *Science Direct*.
- Jovanovic, B., Mijanovic, Z., Stojanovic, R., & Lekic, N. (2015). The Simple System for Objects Classification and Counting.
- K, A. (20 de 09 de 2013). *OpenCV-Python Tutorials*. Obtenido de OpenCV-Python Tutorials: https://opencv-python-tutroals.readthedocs.io/en/latest/py_tutorials/py_feature2d/py_surf_intro/py_surf_intro.html#surf-in-opencv
- Kim, K., Kang, S., Kim, J., Lee, J., & Kim, J. (2015). Bin Picking Method Using Multiple Local Features . *12th International Conference on Ubiquitous Robots and Ambient Intelligence (URAI)* (págs. 148 - 150). IEEE.
- Logitech. (2018). *Camara Web C270*. Obtenido de <https://www.logitech.com/es-roam/product/hd-webcam-c270>
- Lowe, D. G. (1999). Object Recognition from Local Scale-Invariant Features. *IEEE*.
- Lowe, D. G. (2004). Distinctive Image Features. *International Journal of Computer Vision*.

La información presentada en este documento es de exclusiva responsabilidad de los autores y no compromete a la EIA.

- Magaña Z., J. B., Atoche E., J. R., Molina C., J. C., & Pérez C., E. (2017). Estimación de la Distancia a un Objeto con Visión. *Ingeniería Revista academica Universidad Autonoma de Yucatán*.
- Magro, R. (2013). Binarización de imágenes digitales y su algoritmia como herramineta aplicada a la ilustración entomológica. *Boletín de la Sociedad Entomológica Aragonesa*.
- MathWorks. (16 de 09 de 2019). *MathWorks*. Obtenido de MathWorks: <https://la.mathworks.com/help/images/convert-from-hsv-to-rgb-color-space.html>
- Ministerio de trabajo e inmigración. (2003). *Manipulación manual de cargas*. Madrid.
- Miyata, S., & Yashiki, Y. (2014). Evaluation of Kinect Vision Sensor for Bin-Picking Applications . *IEEE*.
- Movimed. (2018). *Movimed*. Obtenido de <http://www.movimed.com/knowledgebase/what-is-laser-triangulation/>
- Nieuwenhuisen, M., Droschel, D., Holz, D., Stuckler, J., Berner, A., Li, J., . . . Behnke, S. (2013). Mobile Bin Picking with an Anthropomorphic Service Robot. *IEEE*.
- OpenCV, D. (2013). *OpenCV*. Obtenido de OpenCV: https://opencv-python-tutroals.readthedocs.io/en/latest/py_tutorials/py_imgproc/py_thresholding/py_thresholding.html
- Robotics, R. (2015). *SDK Rethink Robotics*. Obtenido de http://sdk.rethinkrobotics.com/wiki/Hardware_Specifications
- ROS. (12 de 2013). <https://www.ros.org/about-ros/>.
- Rowland, J. J. (1995). *Intelligent assembly systems*. World Scientific.
- Sansoni, G., Bellandi, P., Leoni, F., & Docchio, f. (2014). Optoranger: A 3D pattern matching method for bin picking application. *Science Direct*.
- Sarmiento, J. M. (2012). *Procesos de segmentación de nubes de puntos*. BarcelonaTech.
- SCHMALZ. (18 de 07 de 2016). *SCHMALZ*. Obtenido de SCHMALZ: <https://www.schmalz.com/es/saber-de-vacio/el-sistema-de-vacio-y-sus-componentes/ventosas-de-vacio/dimensionamiento-de-las-ventosas/>
- SCHMALZ. (08 de 07 de 2016). *SCHMALZ*. Obtenido de SCHMALZ: <https://www.schmalz.com/es/saber-de-vacio/el-sistema-de-vacio-y-sus-componentes/ventosas-de-vacio/formas-de-las-ventosas/>

- Suliga, P. (2016). A features analysis of a laser triangulation stand used to acquire a 3D screw thread image. *International Carpathian Control Conference*, (págs. 702-705).
- Vonasek, V., Vick, A., & Saska1, M. (2017). Motion Planning with Motion Primitives for Industrial Bin Picking. *IEEE*.
- Wu, C.-H., Jiang, S.-Y., & Song, K.-T. (2015). CAD-Based Pose Estimation for Random Bin-Picking of Multiple Objects. *2015 15th International Conference on Control, Automation and Systems (ICCAS 2015) Oct. 13-16,2015 in BEXCO*. Busan: IEEE.
- Xu, J., Pu, S., Zeng, G., & Zha, H. (2012). 3D Pose Estimation for Bin-picking Task Using Convex Hull . *IEEE*.