

Visión por Computador aplicada al control de calidad en procesos de manufactura: seguimiento en tiempo real de refrigeradores

✉ CARLOS ALBERTO LÓPEZ CASTAÑO¹

CARLOS DIEGO FERRIN²

LUIS FERNANDO CASTILLO³

Resumen

En este artículo se presenta una propuesta de visión por computador para el problema de detección y seguimiento de accesorios internos en unidades de refrigeración sobre una *línea de producción*. La propuesta de visión por computador contempla una etapa de adquisición de imágenes multimodales (visible, infrarroja cercana y profundidad) provenientes de cuatro sensores, seguida de una etapa de segmentación para la extracción de contornos tanto de puertas como gabinete; estos contornos son seguidos mediante con ayuda de una etapa de seguimiento que permite que el sistema centre su atención en una sola nevera para pruebas individuales; una etapa final permite estimar las regiones de interés de los accesorios, en la nevera bajo prueba mediante una técnica de mapeo homográfico y la ayuda de un conjunto de accesorios registrados para cada modelo de unidad de refrigeración en una base de datos. El sistema de visión por computador fue evaluado sobre una arquitectura distribuida de alto desempeño con tres modelos diferentes de neveras en condiciones reales de producción. El desempeño y la efectividad de los algoritmos desarrollados demostraron ser adecuados para su aplicación en entornos reales de producción industrial.

Palabras Claves: Control de Calidad, Unidades de Refrigeración, Visión por Computador, Sistemas Distribuidos, Computación de Alto Desempeño.

¹ Universidad Nacional de Colombia, Manizales, Colombia.

² Universidad del Valle, Colombia.

³ Universidad Nacional de Colombia. Universidad de Caldas, Manizales, Colombia.



Autor de correspondencia: López Castaño, C.A. (Carlos Alberto): Dirección Correspondencia: Calle 67 28B-36 Apto 1002A. Teléfono: 313 640 9658
Correo electrónico: caralopezcas@unal.edu.co

Historia del artículo:

Artículo recibido: 30-VIII-2017 / Aprobado: 16-V-2018
Disponible online: 3 de agosto 2018
Discusión abierta hasta abril de 2020

Computer Vision applied to quality control in manufacturing processes: real-time tracking of refrigerators

Abstract

In this paper a computer vision based system is conceived and implemented for quality assurance of inner accessories in fridge-freezer during assembly process. The computer vision system includes a stage of acquisition of multimodal images (visible, near infrared and depth) from four sensors, followed by a segmentation stage for the extraction of contours both doors and cabinet; these contours are tracked by means of a tracking technique that allows the system to focus its attention on a single refrigerator for individual tests; a final stage allows to estimate the regions of interest of the accessories in the refrigerator under evaluation by means of a homographic mapping technique and the aid of a set of previously registered accessories for each refrigerator model in a database. The computer vision system was tested on a distributed and high performance computing architecture for three different models in a real scenario. Performance and effectiveness of the developed system demonstrated suitability for real time applications.

Keywords: Quality Assurance, Fridge-Freezer, Computer Vision, Distributed System, High Performance Computing.

Visão computacional aplicada ao controle de qualidade em processos de fabricação: monitoramento em tempo real de refrigeradores

Resumo

Este artigo apresenta uma proposta de visão computacional para o problema de detecção e monitoramento de acessórios internos em unidades de refrigeração em uma linha de produção. A proposta de visão por computador contempla um estágio de aquisição de imagens multimodais (visível, infravermelho próximo e profundo) provenientes de quatro sensores, seguido de um estágio de segmentação para a extração de contornos e também de portas como gabinete; esses conteúdos foram seguidos com a ajuda de um estágio de acompanhamento que permite que o sistema concentre sua atenção em um único refrigerador para testes individuais; uma etapa final permite estimar as regiões de interesse dos acessórios, no frigorífico em teste por meio de uma técnica de mapeamento homográfico e a ajuda de um conjunto de acessórios para cada modelo da taxa de resfriamento em um banco de dados. O sistema de visão computacional foi avaliado em uma arquitetura distribuída de alto desempenho, nunca com modelos em condições reais de produção. O processo e a eficácia dos algoritmos mostraram-se adequados para aplicação em ambientes reais de produção industrial.

Palavras-chave: Controle de Qualidade, Unidades de Refrigeração, Visão Computacional, Sistemas Distribuídos, Computação de Alto Desempenho.

1. Introducción

El sector global de electrodomésticos se puede definir como moderadamente dinámico, cuya tecnología de producción es madura, las compañías han perfeccionado técnicas de manufactura flexible y asistida por computador, con el fin de reducir los costos de producción; en este sentido, con la llegada de los avances tecnológicos de la cuarta revolución industrial la visión con computador se muestra muy prometedora. En línea con lo anterior, las pruebas funcionales y estéticas se realizan durante el proceso de fabricación de una unidad de refrigeración antes de ser embaladas para su posterior distribución; de esta manera, las pruebas funcionales verifican, entre otras cosas, que el consumo de potencia, al igual que el control de temperatura cumpla con las especificaciones de diseño, por su parte, las pruebas estéticas se aseguran de dos cosas: primero que no haya ningún tipo de daño en los diferentes accesorios tanto internos como externos en la unidad de refrigeración; y segundo, que para un modelo en particular de unidad de refrigeración los accesorios estén presentes, todos en su totalidad.

Las pruebas funcionales se llevan a cabo con dispositivos electrónicos que informan de manera objetiva y en tiempo real (aproximadamente 30 segundos) si la unidad de refrigeración cumplió o no con la prueba; sin embargo, las pruebas estéticas requieren de la capacidad visual de un experto (llamado también Inspector Visual de Calidad) quien debe conocer los accesorios de cada modelo en forma exacta. Por ejemplo, durante un proceso de producción diario, el experto debe discernir entre 30 o 60 modelos diferentes e igualmente hacer el dictamen si la unidad de refrigeración pasó o no la prueba estética en alrededor de 45 segundos. Para el caso de accesorios internos (en compartimento y puerta del refrigerador), un tipo de error son los accesorios faltantes o equivocados; en este sentido, concebir un sistema capaz de emular la capacidad visual del inspector es un gran reto, teniendo en cuenta además que el sistema debe ser no invasivo y ser capaz

de analizar el interior del refrigerador a medida que este se mueve en la línea de producción.

Para comprender en detalle la problemática planteada anteriormente, se debe describir en detalle el proceso de evaluación estética en la línea de producción: durante la evaluación estética un experto debe concentrar su atención en la unidad de refrigeración que está analizando, revisar cada uno de los accesorios para determinar si falta o está equivocado.

Desde la concepción de un sistema de visión por computador, el reto de identificar accesorios faltantes/equivocados en unidades de refrigeración puede ser abordado dividiendo el reto en dos grandes problemas: 1) detección y seguimiento del accesorio; y 2) clasificación del accesorio en correcto, faltante o equivocado. Ver **Figura 1**. En este trabajo, se abordará una propuesta de solución al primer problema: detección y seguimiento de accesorios en unidades de refrigeración para su posterior análisis (i.e., clasificación).

2. Materiales y métodos

Para desarrollar la presente investigación se plantea una metodología para la detección y seguimiento de accesorios, la cual se puede observar en la **Figura 2**. Dicha metodología inicia con la etapa de adquisición de imágenes, que permite obtener el conjunto de datos iniciales en forma de imagen de la escena donde se encuentra el objeto de análisis, p.e., la unidad de refrigeración con la puerta abierta. Se continúa con un bloque de segmentación que permite obtener imágenes binarias de las unidades de refrigeración. Este bloque es diferente para gabinete y puerta debido a que para el primer caso el fondo se tiene controlado y en el otro solo parcialmente. A partir de estas imágenes binarias se implementa una técnica de seguimiento de un solo objeto y se finaliza con el bloque de obtención de ROI (Región de Interés, por sus siglas en inglés) de los accesorios que permite identificar la ubicación de cada uno de estos dentro de la unidad de refrigeración que se está analizando dentro del campo de visión. Para

este último bloque se utiliza información del ROI del gabinete y de la puerta y de los accesorios registrados previamente y que se encuentran en una base de datos para cada modelo. A continuación, se describe cada uno de los bloques en más detalle.

Adquisición de imágenes

Un análisis detallado de la escena y las restricciones propias de un escenario real de producción determina que un conjunto de cuatro Kinect V2 (Kim *et al.*, 2015; Hong, Saavedra and Martinez-Corral, 2016) (dos para gabinete y dos para puerta) son necesarios para poder capturar imágenes adecuadas para la detección, seguimiento y análisis de los accesorios. Es decir, el bloque de adquisición responde a necesidades no solo de la problemática de detección y seguimiento, sino también la de clasificación de accesorios (trabajos futuros de esta investigación). En la **Figura 3** se visualiza la ubicación y orientación de los cuatro sensores. La elección de los Kinect se debe que estos permiten disponer de un conjunto de tres tipos de imágenes (imágenes multimodales) por sensor, con las siguientes características: Imagen RGB de 1920x1080 (24 bit), Imagen PROFUNDIDAD de 512x424 (11 bit) y una Imagen NIR de 512x424 (11 bit). La velocidad máxima de captura de cada sensor es de 30 Hz (Hong, Saa-

vedra and Martinez-Corral, 2016). Esta velocidad de captura bajará cuando se realice la interface entre el sensor y una estación de cómputo.

Los Kinect superiores (Sensores 1 y 3) se encuentran a 1.6 m de la base de la estructura metálica y los Kinect inferiores (Sensores 2 y 4) a una altura de 0,6 m. Además, nótese en la **Figura 3** que los Kinect se han girado 90° para aprovechar la apertura angular horizontal de los sensores la cual resulta mayor que la vertical (Hong, Saavedra and Martinez-Corral, 2016). Los soportes de los brazos de los Kinect están separados 1.0 m. La configuración anterior obedece a los requerimientos de una línea de producción que trabaja modelos de neveras pequeñas (1.5 m de altura) y grandes (2 m de altura) y una separación vertical de la banda a los Kinect de 1.7 m. Detrás de las neveras se ha ubicado un fondo de material opaco (Szeliski, 2011; Demant, Streicher-Abel and Garnica, 2013) para evitar fuentes parásitas infrarrojas para los sensores de gabinete, sin embargo, este fondo no ayuda a controlar la presencia de fuentes parásitas para los sensores de la puerta. Con ayuda de lámparas LED blancas se controla la iluminación en el espectro visible, con el fin de evitar que el sensor realice balance de blancos de forma automática en la imagen RGB.

Figura 1. Ilustración del reto en la construcción de un sistema de visión por computador para la identificación de accesorios faltantes/equivocados en unidades de refrigeración.



Figura 2. Metodología para la detección y seguimiento de accesorios en unidades de refrigeración en línea de producción

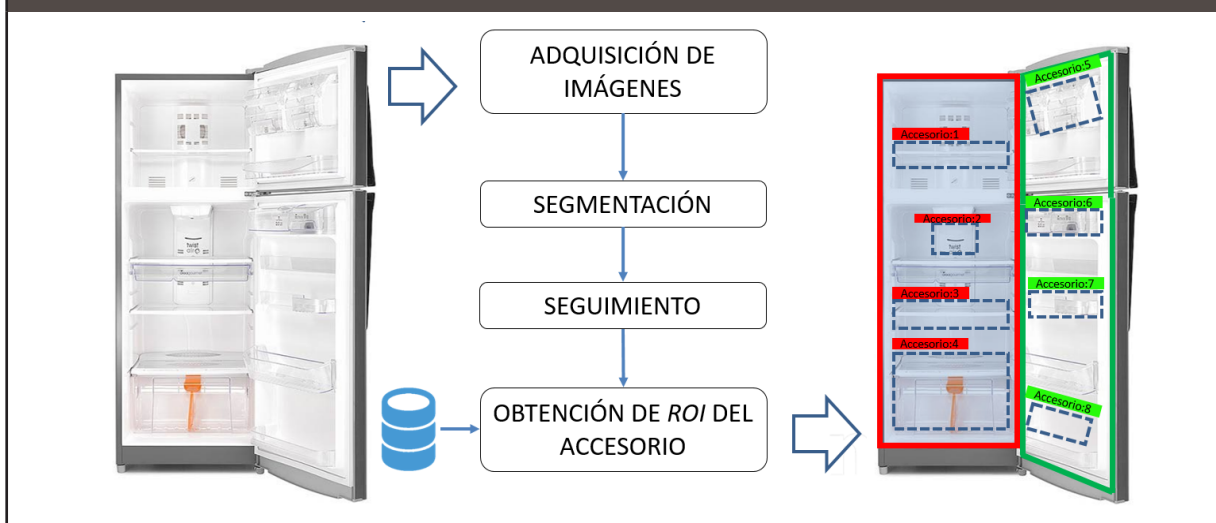
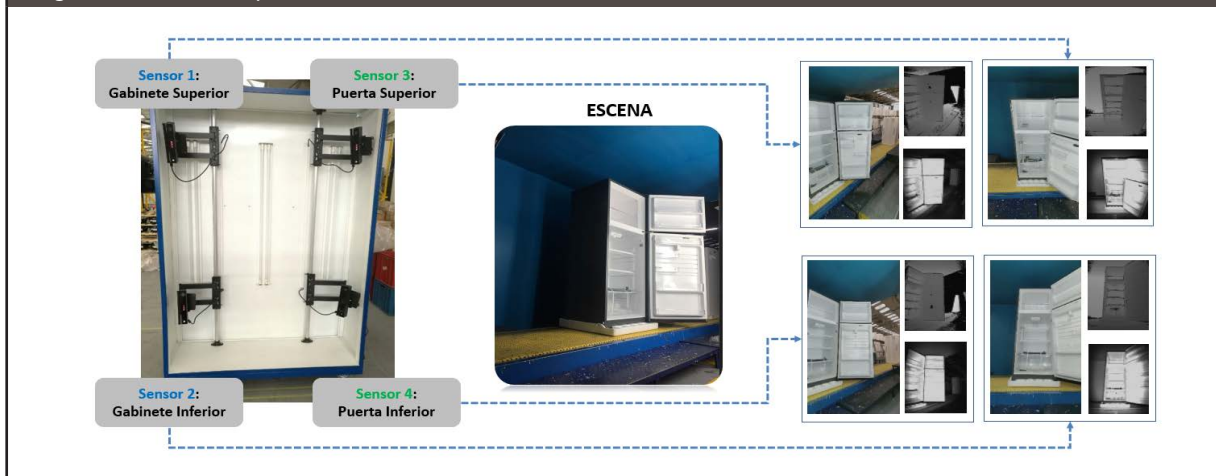


Figura 3. Sistema de adquisición de 4 sensores Kinect V2 (Izq.) capturando imágenes de la escena (cent.) genera un total 12 imágenes en una sola captura



Dado que a la fecha el SDK de Kinect V2 para Windows (Microsoft, 2017) no soporta la captura de simultánea de dos sensores en una misma estación de cómputo, se gestiona entonces la captura de imágenes de estos sensores con ayuda de la biblioteca multiplataforma Libfreenect2 (Xiang *et al.*, 2016) en el lenguaje de programación C++. Esta biblioteca captura la trama de datos enviadas por el sensor a través del protocolo USB 3.0 (utilizando LibUSB 1.0 (Organization, 2016)) hacia el computador, y con ayuda de tarjetas gráficas de NVIDIA (CUDA 8.0

(Nickolls *et al.*, 2008)) se decodifica finalmente las imágenes RGB, PROFUNDIDAD y NIR.

Segmentación

Tal como se mencionó en la sección de Adquisición de Imágenes, se utiliza un fondo mate para controlar la presencia de fuentes parásitas en los sensores de gabinete. Este fondo puede aprovecharse para implementar una técnica de sustracción de fondo (Szeliski, 2011; Elgammal *et al.*, 2014; Klette, 2014) sobre la imagen de profundidad (**Ecuación 1**), utilizando un modelo de media y

desviación estándar de la escena sin la presencia de ninguna nevera. Gracias a esto, en la **Figura 4** se propone un conjunto de técnicas para segmentar el gabinete tanto superior o inferior utilizando la ya mencionada técnica de sustracción de fondo.

$$BG(x, y) = \begin{cases} 1, & \text{si } |D(x, y) - MD(x, y)| \geq SD(x, y) \\ 0, & \text{en caso contrario} \end{cases} \quad (1)$$

Donde $BG(x, y)$ es la imagen binaria de salida de gabinete, $D(x, y)$ es la imagen de profundidad, $MD(x, y)$ y $SD(x, y)$ son modelo promedio y desviación estándar del fondo obtenido a partir de 100 imágenes $D(x, y)$. La **Ecuación 1** se aplica para gabinete superior e inferior. Los niveles de gris en la imagen de profundidad son normalizados con respecto a su valor máximo posible.

Desafortunadamente no puede utilizarse la misma técnica de sustracción de fondo para puertas, ya que en este caso no es posible ubicar un fondo sin obstaculizar el paso de las neveras a lo largo de banda. Por tal razón, se utiliza el siguiente criterio de Umbralización fija (Szeliski, 2011; Howse *et al.*, 2015) tanto en la imagen NIR como en la de profundidad: un pixel es clasificado como perteneciente a la nevera si se encuentra cercano en la imagen de profundidad y es brillante en la imagen infrarroja. Dado que el criterio involucra dos tipos de imágenes, la técnica de Umbralización así implementada se denomina Umbralización multimodal (**Ecuación 2**). En la **Figura 5** se observa el conjunto de técnicas que complementan el método de segmentación de puertas.

$$BP(x, y) = \begin{cases} 1, & \text{si } D1 \leq D(x, y) \leq D2 \wedge N1 \leq NIR(x, y) \leq N2 \\ 0, & \text{en caso contrario} \end{cases} \quad (2)$$

Donde $BP(x, y)$ es la imagen binaria de salida del gabinete, $D(x, y)$ es la imagen de profundidad, $NIR(x, y)$ es la imagen infrarroja. $D1$, $D2$, $N1$ y $N2$ son valores fijos en el intervalo $[0, 1]$. La **Ecuación 2** se aplica para puerta superior e inferior.

Obsérvese de las **Figura 4** y **Figura 5** que los métodos de segmentación comparten un mismo conjunto de técnicas después de: 1) realizar la sustracción de fondo para gabinete y 2) la Umbraliza-

ción multimodal para puertas. En primer lugar, la detección de contornos y aproximación medial obtener una primera aproximación del ROI que contiene gabinete o puerta, utilizando un algoritmo de etiqueta con conectividad 8 (Szeliski, 2011; Demant, Streicher-Abel and Garnica, 2013) y una aproximación medial del ROI. El etiquetado mediante componentes conexos con conectividad 8 tiene como propósito obtener las etiquetas, L , de los n contornos detectados y el conjunto, C , de $m(L)$ puntos 2D, (x_i^L, y_i^L) , por cada etiqueta L . Ver **Figura 6**. El ROI por cada etiqueta L , $ROI(L)$, se define mediante un conjunto de 4 puntos 2D obtenidos a través la **Ecuación 3**.

$$ROI(L) = \left\{ \left([x^*]_k^L, [y^*]_k^L \right) \mid k = 1, 2, 3, 4 \right\}$$

Donde:

$$\begin{aligned} \left([x^*]_1^L, [y^*]_1^L \right) &= \left(\min \{ x_i^L \}, \min \{ y_i^L \} \right) \\ \left([x^*]_2^L, [y^*]_2^L \right) &= \left(\max \{ x_i^L \}, \min \{ y_i^L \} \right) \\ \left([x^*]_3^L, [y^*]_3^L \right) &= \left(\max \{ x_i^L \}, \max \{ y_i^L \} \right) \\ \left([x^*]_4^L, [y^*]_4^L \right) &= \left(\min \{ x_i^L \}, \max \{ y_i^L \} \right) \end{aligned} \quad (3)$$

Dado que esta primera aproximación del ROI contiene en el caso de gabinete pixeles de pertenecientes a la puerta y en el caso de puerta pixeles de gabinete, se procede de manera heurística a redefinir el ROI mediante una aproximación medial, la cual consiste en tomar la mitad del ROI obtenido por cada etiqueta L .

$$ROI_G(L) = \left\{ \left([x_m^*]_k^L, [y_m^*]_k^L \right) \mid k = 1, 2, 3, 4 \right\}$$

Donde:

$$\begin{aligned} \left([x_m^*]_1^L, [y_m^*]_1^L \right) &= \left([x^*]_1^L, [y^*]_1^L \right) \\ \left([x_m^*]_2^L, [y_m^*]_2^L \right) &= \left(\frac{[x^*]_1^L + [x^*]_3^L}{2}, [y^*]_1^L \right) \\ \left([x_m^*]_3^L, [y_m^*]_3^L \right) &= \left(\frac{[x^*]_1^L + [x^*]_3^L}{2}, [y^*]_3^L \right) \\ \left([x_m^*]_4^L, [y_m^*]_4^L \right) &= \left([x^*]_4^L, [y^*]_4^L \right) \end{aligned} \quad (4)$$

Figura 4. Flujo secuencial de técnicas utilizadas para la segmentación de Compartimento (Gabinete)

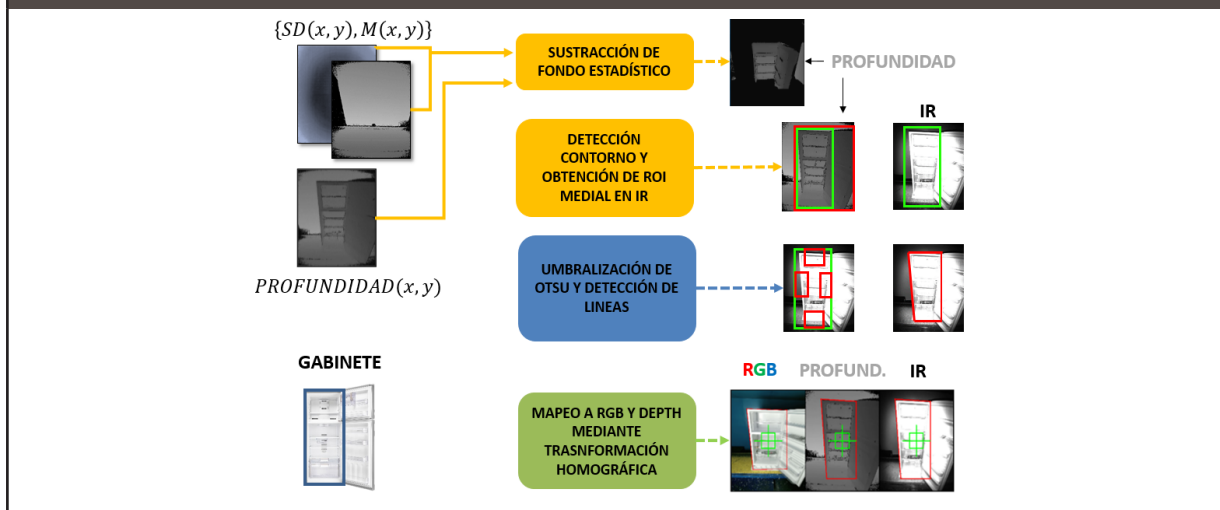
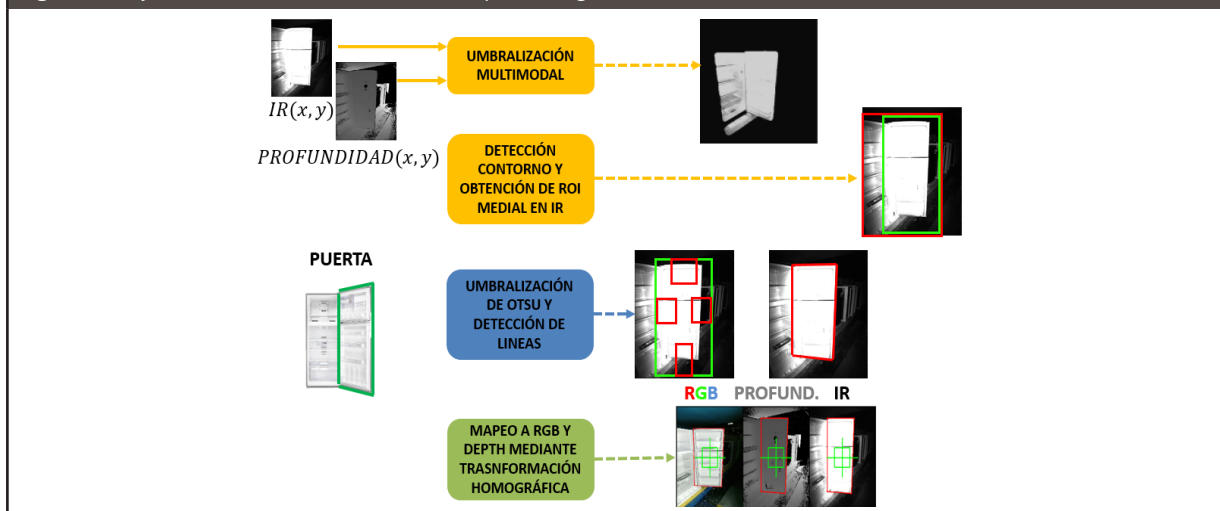


Figura 5. Flujo secuencial de técnicas utilizadas para la segmentación de Puerta



$$ROI_P(L) = \left\{ \left([x_m^*]_k^L, [y_m^*]_k^L \right) \mid k = 1, 2, 3, 4 \right\}$$

Donde:

$$\begin{aligned} ([x_m^*]_1^L, [y_m^*]_1^L) &= \left(\frac{[x^*]_1^L + [x^*]_3^L}{2}, [y^*]_1^L \right) \\ ([x_m^*]_2^L, [y_m^*]_2^L) &= ([x^*]_2^L, [y^*]_2^L) \\ ([x_m^*]_3^L, [y_m^*]_3^L) &= ([x^*]_3^L, [y^*]_3^L) \\ ([x_m^*]_4^L, [y_m^*]_4^L) &= \left(\frac{[x^*]_1^L + [x^*]_3^L}{2}, [y^*]_3^L \right) \end{aligned} \quad (5)$$

Este ROI se toma hacia la izquierda en gabinete (**Ecuación 4**) y hacia la derecha en puerta (**Ecuación 5**) si la unidad de refrigeración abre la puerta hacia la derecha en la imagen. Obviamente debe tomarse en sentido contrario en caso que la nevera abra la puerta hacia la izquierda, en la imagen.

Se busca ahora dentro de este ROI medial, obtener un ROI no necesariamente rectangular y definido por cuatro puntos coordenadas 2D en la imagen. Específicamente, se intenta encontrar las ecuaciones de línea que delimitan el contorno de la nevera por la parte superior, inferior, izquierda y

derecha tanto del gabinete como de la puerta. Estas líneas se obtiene utilizando ajuste lineal mediante mínimos cuadrados (Bertsekas, 2016) sobre el conjunto de puntos frontera, que resulta del borde obtenido mediante un *kernel* de derivada direccional sobre la imagen umbralizada mediante la técnica Otsu (Liu and Yu, 2009; Szeliski, 2011). Las subimágenes son extraídas de la imagen infrarroja y se definen de forma heurística a partir del ROI medial con ayuda de la técnica mostrada en la **Figura 7**.

Una vez encontrada estas líneas, los cuatro puntos que definen el ROI se obtienen utilizando ecuaciones de punto de intersección entre líneas (Jänich, 1994): líneas superior e izquierda para la esquina superior-izquierda del ROI, líneas superior y derecha para el punto la esquina superior-derecha del ROI, líneas derecha e inferior para el punto inferior-derecho del ROI, y entre las líneas inferior e izquierda para el punto de la esquina inferior-izquierda del ROI.

Figura 6. Resultado aplicación algoritmo de etiquetado (Szeliski, 2011; Demant, Streicher-Abel and Garnica, 2013) sobre una imagen binaria. La salida de este algoritmo son las etiquetas, L , de los n contornos detectados y el conjunto, C , de $m(L)$ puntos 2D, (x_i^L, y_i^L) , por cada etiqueta L .

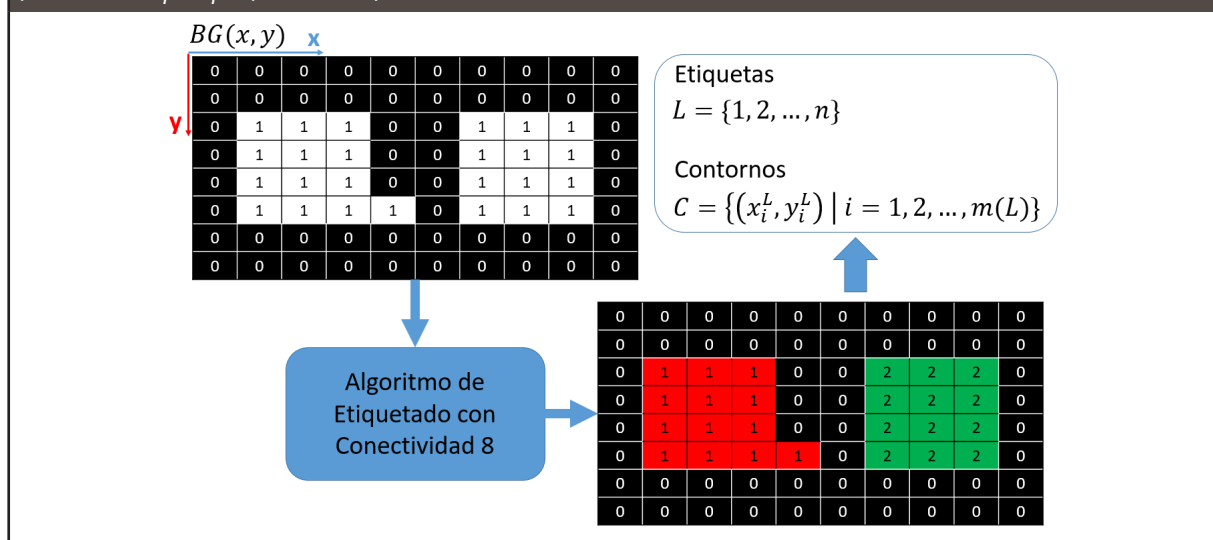
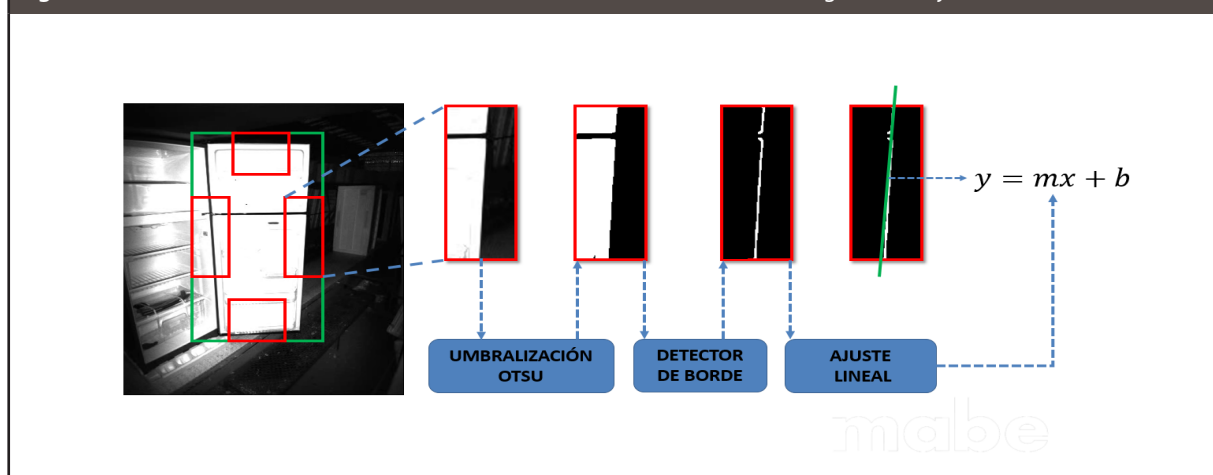


Figura 7. Técnica de obtención de ecuación de línea sobre borde de nevera en imagen infrarroja.



Para el caso de gabinete no es posible obtener una ecuación de línea ubicada al costado derecho con la técnica mencionada anteriormente (que por cierto separa gabinete de puerta en la imagen). En este caso se toma como línea derecha, aquella obtenida en el ROI medial. Se procede de igual forma para la línea izquierda en puerta. Finalmente, las coordenadas de los puntos obtenidos en la imagen NIR se toman como los mismos en la imagen de PROFUNDIDAD (se considera despreciables las distorsiones en las imágenes del Kinect V2), y mediante técnicas homográficas (Hartley and Zisserman, 2004) se mapean estos puntos a la imagen RGB. La relación homográfica entre puntos coplanares desde dos vistas (este caso NIR y RGB) viene dado por la **Ecuación 6** utilizando coordenadas homogéneas asociadas a los puntos 2D. La matriz H es la homografía y el problema consiste en hallar precisamente esta matriz. Dado que H contiene 9 incógnitas, se propone el sistema de ecuaciones en 7 para cuatro pares de puntos correspondientes y se resuelve el sistema por descomposición de valores singulares (Jänich, 1994; Hartley and Zisserman, 2004) de la matriz A.

$$x_h = Hx'_h \rightarrow \begin{bmatrix} x \\ y \\ 1 \end{bmatrix} = \begin{bmatrix} h_{11} & h_{12} & h_{13} \\ h_{21} & h_{22} & h_{23} \\ h_{31} & h_{32} & h_{33} \end{bmatrix} \begin{bmatrix} x' \\ y' \\ 1 \end{bmatrix} \quad (6)$$

Seguimiento

La técnica de segmentación generará tantos ROI's como gabinetes y puertas sean segmentadas

mediante la sustracción de fondo o la Umbralización multimodal respectivamente. Con ayuda de los ROI's, se construyen contornos los cuales son etiquetados nuevamente mediante componentes con conectividad 8 (Szeliski, 2011). Ver **Figura 6**. Se elige como posición óptima el centro de la imagen y entonces se elige el contorno de máxima área cuyo centroide está más cercano al centro de la imagen por el lado izquierdo. En caso que las neveras entren por el lado derecho, debe aplicarse el criterio anterior pero ahora por este nuevo lado. Las etapas claves de este bloque se muestran en la **Figura 8**.

Obtención de la Región de Interés (ROI) del Accesorio

Hasta ahora, los bloques de adquisición, segmentación y seguimiento se han elaborado independiente del modelo de unidad de refrigeración que se está analizando en el campo de visión de los sensores. Dado que el número de accesorios de un modelo a otro varía y que las ubicaciones y tipos de accesorios es una variable no controlada, se requiere entonces, para la obtención de los ROI's de los accesorios, conocer con anterioridad el modelo de unidad de refrigeración para la prueba. Esto se logra almacenando información los modelos y accesorios en una base de datos a través de un proceso de registro del modelo. Por cada modelo se requiere el ROI que delimita el gabinete y la puerta, y los ROI's y etiquetas de cada uno de los accesorios que se deseen analizar.

$$\begin{aligned} x' &= \frac{kh_{11}x + kh_{12}y + kh_{13}}{kh_{31}x + kh_{32}y + kh_{33}} \\ y' &= \frac{kh_{21}x + kh_{22}y + kh_{23}}{kh_{31}x + kh_{32}y + kh_{33}} \end{aligned} \rightarrow \begin{bmatrix} x_1 & y_1 & 1 & 0 & 0 & 0 & -x_1x'_1 & -y_1y'_1 & -x'_1 \\ 0 & 0 & 0 & x_1 & y_1 & 1 & -x_1y'_1 & -y_1x'_1 & -y'_1 \\ x_2 & y_2 & 1 & 0 & 0 & 0 & -x_2x'_2 & -y_2y'_2 & -x'_2 \\ 0 & 0 & 0 & x_2 & y_2 & 1 & -x_2y'_2 & -y_2x'_2 & -y'_2 \\ x_3 & y_3 & 1 & 0 & 0 & 0 & -x_3x'_3 & -y_3y'_3 & -x'_3 \\ 0 & 0 & 0 & x_3 & y_3 & 1 & -x_3y'_3 & -y_3x'_3 & -y'_3 \\ x_4 & y_4 & 1 & 0 & 0 & 0 & -x_4x'_4 & -y_4y'_4 & -x'_4 \\ 0 & 0 & 0 & x_4 & y_4 & 1 & -x_4y'_4 & -y_4x'_4 & -y'_4 \end{bmatrix} \begin{bmatrix} h_{11} \\ h_{12} \\ h_{13} \\ h_{21} \\ h_{22} \\ h_{23} \\ h_{31} \\ h_{32} \\ h_{33} \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix} \quad (7)$$

Figura 8. Etapas claves para la selección del ROI de interés en el bloque de seguimiento.

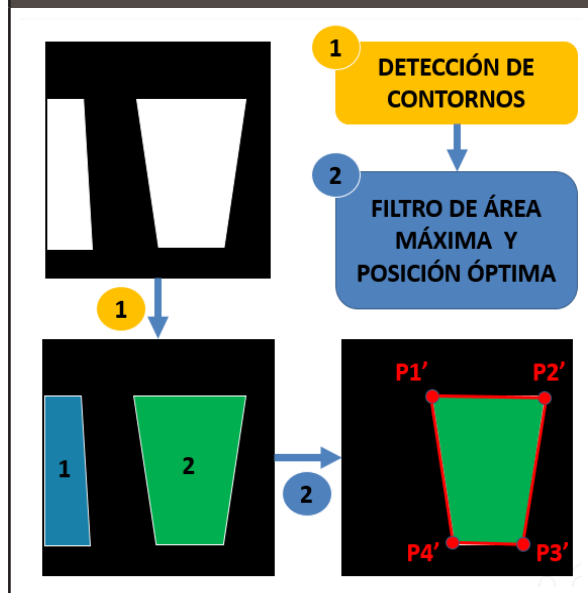
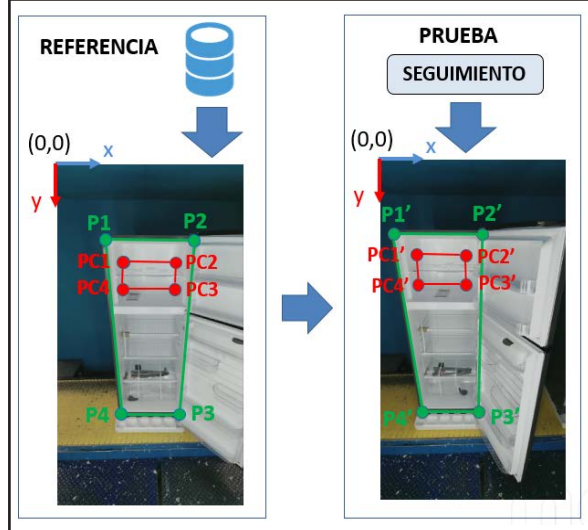


Figura 9. Los puntos $\{P_i\}$, $\{P'_i\}$ y $\{PC_i\}$ son conjuntos de puntos conocidos: $\{P_i\}$ y $\{PC_i\}$ provienen de la base de datos de la referencia registrada, $\{P'_i\}$ proviene del resultado de la etapa de seguimiento de la nevera que bajo prueba. $\{PC'_i\}$ es el conjunto de puntos desconocidos del accesorio en la nevera de prueba. ($i = 1, 2, 3$ y 4).



En la **Figura 9** se muestra las variables que se dispone para realizar la obtención de los ROI's de los accesorios para la prueba final. Suponiendo N accesorios, la estimación de los ROI's de los accesorios en la unidad de refrigeración en cuestión

se obtiene mediante el siguiente algoritmo basado en transformación homográfica (Hartley and Zisserman, 2004; Szeliski, 2011) el cual se planteó en las **Ecuaciones 6 y 7**.

Algoritmo: Estimación Posición ROI's de los Accesorios

Entrada: $\{P_i\}$, $\{PC_i\}$, $\{P'_i\}$;

$i = 1, \dots, 4; j = 1, \dots, N$.

Salida: $\{PC'_i\}^j$.

1. $N = \text{longitudArreglo}(\{PC_i\}^j)$;
2. $H = \text{homografía}(\{P_i\}, \{P'_i\})$;
3. $\{PC'_i\}^j = \text{inicializa}(4, N)$;
4. **Para** $j = 1$ hasta N
5. **Para** $i = 1$ hasta 4
6. $X = [\{PC_i\}^j.x, \{PC_i\}^j.y, 1]^T$;
7. $X_{out} = HX$;
8. $\{PC'_i\}^j.x = X_{out}.x$;
9. $\{PC'_i\}^j.y = X_{out}.y$;
10. **Fin Para**
11. **Fin Para**
12. **Retorna** $\{PC'_i\}^j$;

3. Resultados y discusión

En primer lugar, mediante pruebas realizadas durante 3 horas (en una línea de 35 segundos por nevera) se evalúa el desempeño de los algoritmos desarrollados anteriormente. Para esto, se construye una arquitectura distribuida (ver **Figura 10**) que permite la conexión de dos Kinect V2 por Workstation. La configuración y ejecución de los algoritmos se realiza con ayuda de una interfaz gráfica de usuario (ver **Figura 11**) la cual debe ejecutarse para cada sensor. Cada aplicativo con cada sensor configura un cliente-servidor tipo VRPN (*Virtual Reality Peripheral Network*) (Taylor et al., 2001) para la comunicación con la interfaz de control y visualización. Igualmente, se configura un servidor de pruebas con ayuda de la plataforma FIS (*Floor Information System*) de la compañía Mabe, y se registran 3 modelos: MCRMU300WANB (nevera pequeña), MCRMC320WACX (nevera mediana), MCR-

MC390WACX (nevera grande). La descarga de los modelos a las Workstation se realiza una sola vez, siempre y cuando el modelo no exista en la base local de la estación de cómputo. A continuación, se recogen datos de velocidad de ejecución, memoria RAM ocupada por los aplicativos y el nivel de ocupación de CPU y GPU, de los algoritmos desarrollados para la metodología aquí propuesta (ver **Figura 2**). En la **Tabla 1** se muestran los valores promedios obtenidos durante las 3 horas de prueba. Se observa que la arquitectura A2 ejecuta de forma más rápida los algoritmos que la arquitectura A1; sin embargo, el costo computacional CPU es un 10% más elevado. El coste computacional en GPU de A2 es menor que A1 en un 5%. En ambos casos el coste computacional en GPU se debe básicamente al bloque de adquisición, el cual se logra a través de la biblioteca *libfreenect2* (Xiang *et al.*, 2016). Esto indica que en ambas arquitecturas es posible ocupar todavía gran parte de la GPU en la ejecución de otros algoritmos, por ejemplo, segmentación y seguimiento. Por último, pero no menos importante, se puede ver que las demandas de memoria RAM de los aplicativos no es alta, pudiendo ejecutar los mismos, por ejemplo, en estaciones de cómputo con al menos 2 GB RAM.

Por otro lado, se realizaron pruebas con cada uno de los modelos para evaluar la efectividad del sistema. Para el modelo MCRMU300WANB se registran 6 accesorios y 7 para los otros dos modelos.

En la **Figura 12**, **Figura 13** y **Figura 14** se muestra el resultado de la obtención de los ROI's de los accesorios cuando las neveras pasan por la posición óptima (posición central de la imagen). En general, se observó, primero, que cuando la segmentación es correcta, la estimación de la ubicación de los ROI's también lo es. Segundo, los algoritmos de segmentación para gabinete generan resultados más estables que para puerta, es decir la segmentación más consistente a lo largo del movimiento de la nevera en el campo de visión. Esto se debe a que para el caso de gabinete se tiene un fondo controlado. Además, se encontró que la estimación de la línea inferior en puerta (en el bloque de segmentación de puerta) es inestable debido a brillos indeseados en NIR cerca de la banda. Tercero, el algoritmo de seguimiento responde favorablemente en pruebas realizadas con neveras (abiertas) al lado izquierdo y derecho de la nevera bajo prueba, siempre y cuando se respete la distancia mínima de separación de 70 cm entre neveras consecutivas a lo largo de la banda. Finalmente, se realizaron pruebas durante el día y la noche obteniendo los mismos resultados en la obtención de los ROI's de los accesorios. Como resultado final, estas pruebas exploratorias permitieron identificar y seguir ROI's con un 95% de efectividad a lo largo de la línea, siendo las posiciones extremas en la imagen donde el sistema realiza una detección incorrecta de los accesorios.

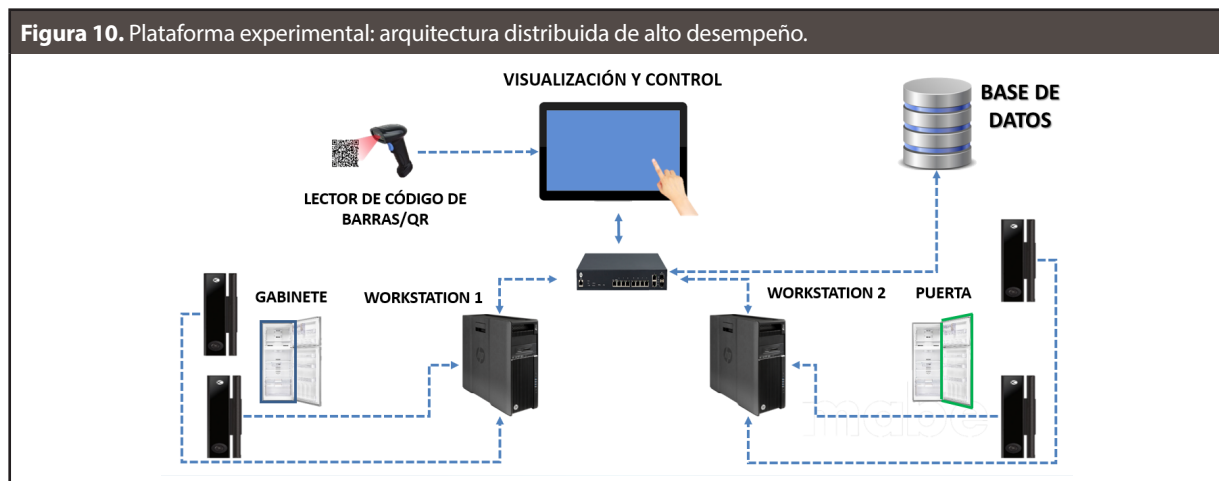


Figura 11. Interfaz Gráfica de Usuario para captura, segmentación y seguimiento de accesorios en unidades de refrigeración en tiempo real

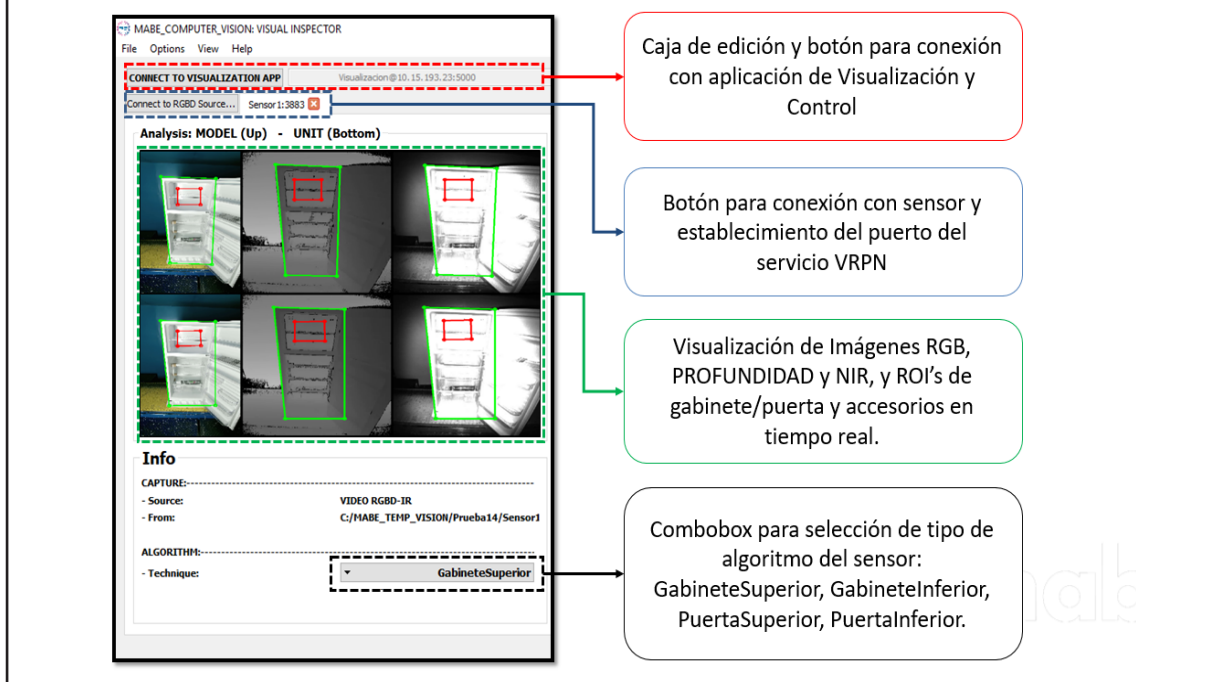


Figura 12. Resultado de obtención de ROI's de accesorios en posición central de la imagen. Prueba realizada con una nevera perteneciente al modelo MCRMU300WANB

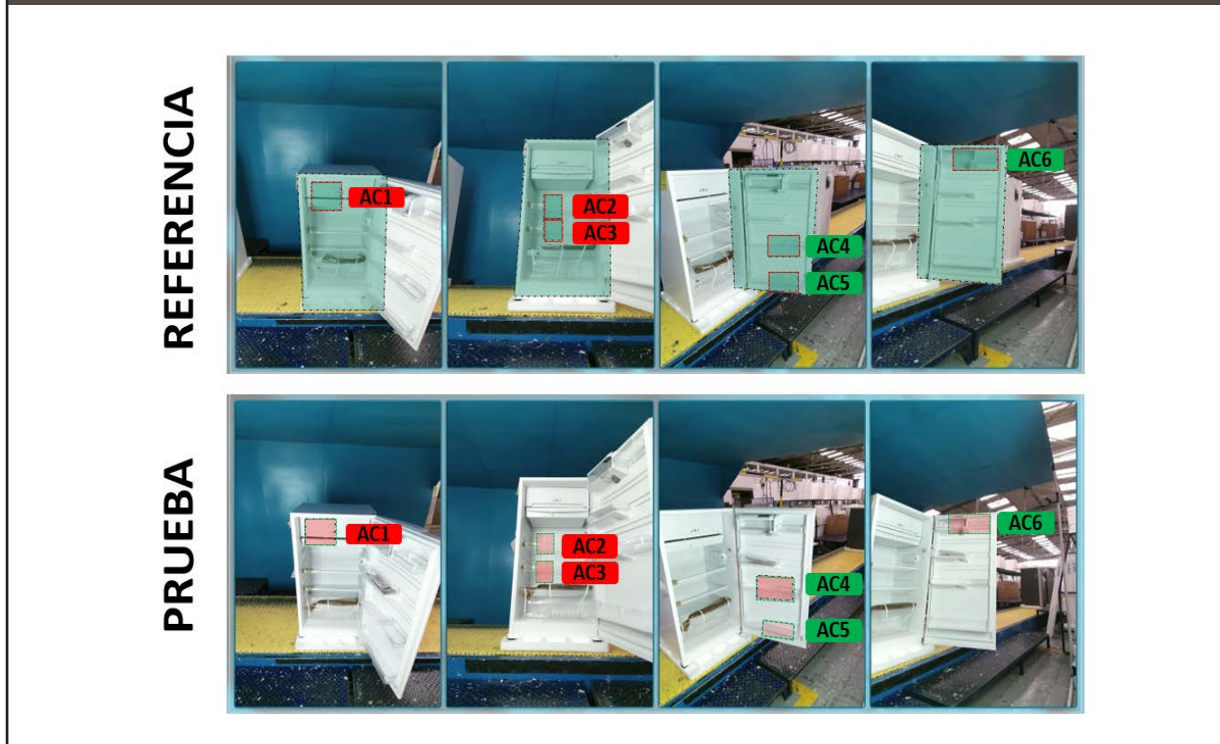


Figura 13. Resultado de obtención de ROI's de accesorios en posición central de la imagen. Prueba realizada con una nevera perteneciente al modelo MCRMC320WACX

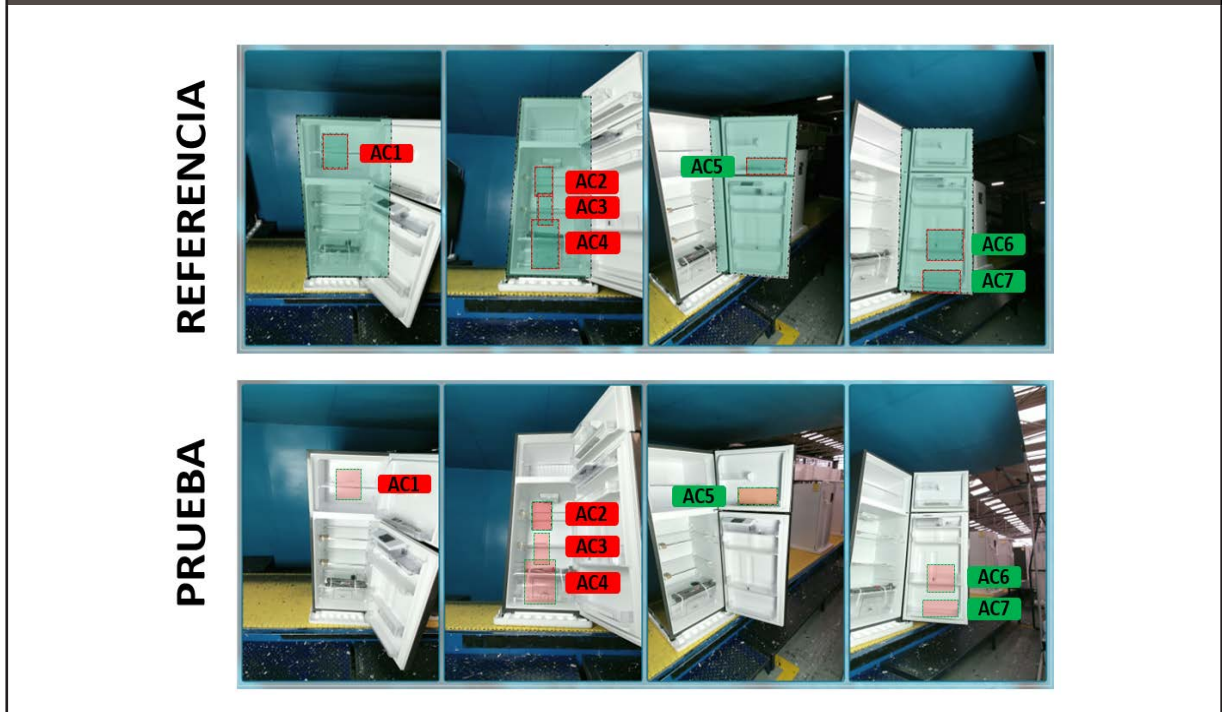


Figura 14. Resultado de obtención de ROI's de accesorios en posición central de la imagen. Prueba realizada con una nevera perteneciente al modelo MCRMC390WACX

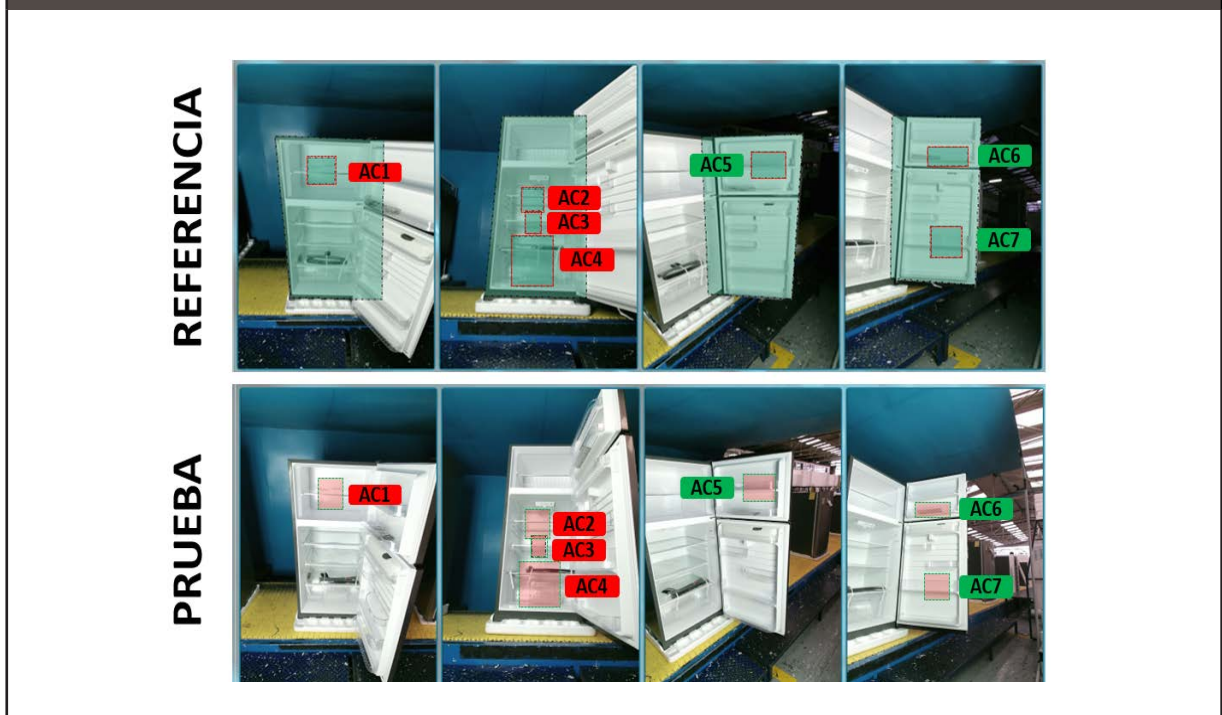


TABLA 1. VALORES PROMEDIOS DE DESEMPEÑO SOBRE DOS TIPOS DE ARQUITECTURAS COMPUTACIONALES PARA LA EJECUCIÓN DE LOS ALGORITMOS DESARROLLADOS

ARQUITECTURA COMPUTACIONAL	Velocidad (Hz)	RAM (GB)	Ocupación CPU (%)	Ocupación GPU (%)
A1: Intel(R) Xeon, 12 Núcleos, 2.40 GHz, 16 GB RAM, Nvidia Quadro K 2200.	14	1.1	60	15
A2: Intel(R) Core I7, 8 Núcleos, 3.40, GHz, 32 GB RAM, Nvidia Quadro M5000	20	1.2	70	10

4. Conclusiones

Se diseñó e implementó una metodología de visión por computador para la detección y seguimiento de accesorios internos en unidades de refrigeración en tiempo real. La rapidez de la ejecución de los algoritmos en las arquitecturas computacionales fue exitosa teniéndose 15 Hz como mínimo. La estimación de la posición de los ROI's de los accesorios en la posición central de la imagen es muy similar a la de los mismos accesorios en la referencia. Esta estimación es incorrecta cuando la segmentación también lo es. En conclusión, los algoritmos muestran un gran potencial para ser utilizados en un entorno de producción industrial.

Agradecimientos

A Mabe Colombia, por el apoyo financiero, administrativo y técnico brindado en la realización de este proyecto de investigación.

Referencias

- Bertsekas, D. (2016) *Nonlinear Programming*. 3rd edn. Massachusetts: Athena Scientific. Available at: <http://www.athenasc.com/nonlinbook.html>.
- Demant, C., Streicher-Abel, B. and Garnica, C. (2013) *Industrial Image Processing*. Berlin, Heidelberg: Springer Berlin Heidelberg. doi: 10.1007/978-3-642-33905-9.
- Elgammal, A., Tiefenbacher, P., Hofman, M., Rigoll, G., Barnich, O., López-Rubio, E., Luque-Baena, R., Huang, J., Chen, C., Cui, X., Chiranjeevi, P. and Sengupta, S.

(2014) 'Traditional and Recent Models', in Bownmans, T. (ed.) *Background Modeling and Foreground Detection for Video Surveillance*. Chapman and Hall/CRC, p. II-1-II-1. doi: 10.1201/b17223-7.

- Hartley, R. and Zisserman, A. (2004) *Multiple View Geometry in Computer Vision, Climate Change 2013 - The Physical Science Basis*. Edited by Intergovernmental Panel on Climate Change. Cambridge: Cambridge University Press. doi: 10.1017/CBO9780511811685.
- Hong, S., Saavedra, G. and Martinez-Corral, M. (2016) 'Full parallax three-dimensional display from Kinect v1 and v2', *Optical Engineering*, 56(4), p. 41305. doi: 10.1117/1.OE.56.4.041305.
- Howse, J., Puttemans, S., Hua, Q. and Sinha, U. (2015) *OpenCV 3 Blueprints*, PACKT Publishing Ltd. Available at: <http://arxiv.org/abs/1011.1669>.
- Jänich, K. (1994) *Linear Algebra*. New York, NY: Springer New York (Undergraduate Texts in Mathematics). doi: 10.1007/978-1-4612-4298-7.
- Kim, C., Yun, S., Jung, S.-W. and Sun, C. (2015) *Color and Depth Image Correspondence for Kinect V2, Lecture Notes in Electrical Engineering*. Edited by J. J. Park, H.-C. Chao, H. Arabnia, and N. Y. Yen. Berlin, Heidelberg: Springer Berlin Heidelberg (Lecture Notes in Electrical Engineering). doi: 10.1007/978-3-662-47487-7.
- Klette, R. (2014) *Concise Computer Vision*. London: Springer London (Undergraduate Topics in Computer Science). doi: 10.1007/978-1-4471-6320-6.
- Liu, D. and Yu, J. (2009) 'Otsu Method and K-means', in *2009 Ninth International Conference on Hybrid Intelligent Systems*. IEEE, pp. 344-349. doi: 10.1109/HIS.2009.74.
- Microsoft (2017) *Kinect for Windows SDK 2.0*. Available at:

- <https://www.microsoft.com/en-us/download/details.aspx?id=44561> (Accessed: 30 June 2016).
- Nickolls, J., Buck, I., Garland, M. and Skadron, K. (2008) 'Scalable parallel programming with CUDA', *Queue*, 6(2), p. 40. doi: 10.1145/1365490.1365500.
- Organization, L. (2016) 'LibUSB 1.0'. SourceForge, p. 1. Available at: <http://www.libusb.org/wiki/libusb-1.0>.
- Szeliski, R. (2011) *Computer Vision, Media*. London: Springer London (Texts in Computer Science). doi: 10.1007/978-1-84882-935-0.
- Taylor, R. M., Hudson, T. C., Seeger, A., Weber, H., Juliano, J. and Helser, A. T. (2001) 'VRPN', in *Proceedings of the ACM symposium on Virtual reality software and technology - VRST '01*. New York, New York, USA: ACM Press, p. 55. doi: 10.1145/505008.505019.
- Xiang, L., Ehtler, F., Kerl, C., Wiedemeyer, T., Gordon, R. and Facioni, F. (2016) 'libfreenect2: Release 0.2'. Zenodo, p. 1. doi: 10.5281/zenodo.594510.

**PARA CITAR ESTE ARTÍCULO /
TO REFERENCE THIS ARTICLE /
PARA CITAR ESTE ARTIGO /**

López Castaño, C.A.; Ferrin, C.D.; Castillo, L.F. (2018). Visión por Computador aplicada al control de calidad en procesos de manufactura: seguimiento en tiempo real de refrigeradores. *Revista EIA*, 15(30), julio-diciembre, pp. 57-71. [Online]. Disponible en: <https://doi.org/10.24050/reia.v15i30.1150>