



UNIVERSIDAD NACIONAL AUTÓNOMA DE MÉXICO

FACULTAD DE INGENIERÍA

**Mercado de agua para
identificación de productos
comerciales**

TESIS

**Que para obtener el título de
Ingeniero Eléctrico Electrónico**

P R E S E N T A:

Carlos Agustín Muñetón Ramírez

DIRECTORA DE TESIS:

Dra. Jimena Olveres Montiel



Ciudad Universitaria, Cd. Mx., 2019

Índice

1) Introducción.....	1
1.1 Definición del problema.....	1
1.2 Objetivo.....	3
1.3 Alcance.....	4
1.4 Estado del arte.....	4
1.5 Organización de la tesis.....	5
2) Marco Teórico.....	6
2.1 Marca de agua digital.....	6
2.2 Fingerprinting.....	11
2.3 Esteganografía.....	12
2.4 Identificación de productos.....	14
2.5 Código de barras.....	14
2.6 Código QR.....	16
3) Desarrollo.....	17
3.1 El modelo YIQ.....	17
3.2 Explicación del código de inserción.....	18
3.3 Explicación del código de extracción.....	20
4) Experimentos y resultados.....	21
4.1 Descripción de pruebas de eficiencia.....	21
4.2 Resultados de las pruebas.....	24
4.3 Ventajas y desventajas.....	42
5) Conclusiones y trabajo a futuro.....	44
Apéndice	
Referencias	

CAPÍTULO 1

INTRODUCCIÓN

Debido al uso de Internet, ha sucedido que millones de usuarios usaran material bajo *copyright* o derechos de autor, lo cual está protegido por la ley. [5] Los productores y publicadores de contenido están constantemente buscando formas más viables para controlar el acceso a sus medios, pero al mismo tiempo, los piratas casuales y profesionales se están volviendo más ingeniosos, por lo que es necesario proteger el contenido.

Cuando nos encontramos en este caso surgen dos posibilidades para protegerlo: la protección *a priori* y la protección *a posteriori*. La primera consiste en impedir que el cliente (comprador) pueda realizar una copia del material, mientras que la segunda consiste en detectar dichas copias. El clásico DRM (*digital rights management*), o derechos de autor, que intenta autorizar un usuario específico a consumir el contenido en un dispositivo específico, es protección *a priori*, y puede ser muy eficaz en ciertas situaciones, sin embargo, en muchas otras puede no ser apropiado o efectivo. Debido a que la protección *a priori*, a la larga, es vulnerable ya que se puede dar con el algoritmo de protección y por tanto anularlo, en los últimos años se ha empezado a pensar en la protección *a posteriori* como una herramienta eficiente para combatir la piratería, la **marca de agua digital** y la **huella digital**. Estas tecnologías prometen facilitar funciones críticas de negocios, como la identificación de contenido o el control de derechos de autor.

Estas tecnologías generalmente son usadas para la protección de derechos de autor, pero es posible aplicar sus características de identificación y reconocimiento para otros propósitos. En este trabajo se presentará una propuesta para utilizarlas como reemplazo del código de barras tradicional en la identificación de productos.

1.1 DEFINICIÓN DEL PROBLEMA

Existe la necesidad en el mercado actual de métodos más eficaces para adquirir códigos de productos dentro de las cadenas de autoservicios. La auditoría en los centros comerciales es vital para el eficiente reabastecimiento de los productos que éste ofrece. Un buen supermercado debe poder llevar un control de inventario adecuado en su administración de manera tal que le permita discernir si necesita reabastecer sus anaqueles, adquirir más cantidad de algún producto, o bien, verificar que aún cuenta con suficiente producto almacenado en bodega para rellenar sus estantes y poder seguir ofreciéndolo a su clientela. Esto se hace optimizando el planograma. Un **planograma** (ver figura 1) es la representación gráfica del acomodo de mercancías o productos en un área específica de un establecimiento comercial que

puede ser una góndola, un expositor o un anaquel. Tienen como característica principal el poder hacer rápida y cómodamente diversas versiones y pruebas de cómo colocar determinados artículos en un esquema. Así, se puede saber de antemano la ubicación, las cantidades, la posición exacta de los artículos, el material técnico utilizado etc., de esa implantación.

El presente trabajo ofrece una alternativa a realizar este proceso más rápido y fácilmente.



FIGURA 1.- PLANOGRAMA DE UN ANAQUEL DE SUPERMERCADO.

De acuerdo al artículo *Smart Shelfie* [22], 31% de los clientes en un supermercado terminan comprando en otro lugar si no encuentran un producto disponible y 26% compran una marca distinta del producto que buscan si no lo encuentran a primera vista. Lo anterior significa que, para un centro comercial, una mala auditoría de sus anaqueles implica pérdidas tanto para el supermercado como para las marcas que lo abastecen.

Actualmente, la auditoría de los anaqueles se realiza a mano por personal del supermercado mediante la lectura de los códigos de barras de cada uno de los productos de un anaquel.[21]

El código de barras (ver figura 5) generalmente es colocado a simple vista en la etiqueta de los productos, contrastando con el resto de la etiqueta. Luego, en las tiendas, el producto es desplazado del lado del código y el lector lo identifica. Este proceso puede ser acelerado haciendo que el lector no tenga que ver a un lado del producto en específico y que lo identifique con el simple hecho de pasarlo y, también evitar lesiones por movimientos repetitivos [16].



FIGURA 2.-DETECTOR DE CÓDIGOS DE BARRAS.

En este trabajo se presenta una alternativa al código de barras tradicional, por medio de las marcas de agua y códigos QR. La idea es colocar códigos QR como marcas de agua, en las etiquetas de los productos. Estas estarían escondidas, lo suficiente para que no sean fácilmente identificables a simple vista, pero no tanto para que una máquina no pueda reconocerlas.

Un código QR puede contener más información, y éste puede ser leído con un simple teléfono celular, lo cual lo vuelve más práctico que el código de barras tradicional, y puede servir para identificar al producto no sólo en una tienda, sino en toda la cadena de suministro, desde que el producto es fabricado hasta que llegue a su destino final.

1.2 OBJETIVO

Para propósito de este proyecto, se plantea la siguiente hipótesis: si un código QR es extraíble de la foto de una impresión de una etiqueta de algún producto, estando el código QR lo más escondido que se pueda, y se cataloga como exitoso el caso en el que el código QR extraído sea reconocido por un lector QR, entonces es posible crear una máquina o aplicación que elabore este proceso de manera automática. En este

proyecto, ese proceso se simulara en Matlab, primero insertando el código QR en la etiqueta, imprimiendo la imagen, fotografiando la imagen y, luego extrayendo el código QR de la foto.

El objetivo entonces es implementar un código en Matlab que inserte un código QR en varias imágenes, modificándolas lo menos posible y, luego leer ese código con un lector QR.

De este objetivo general se infieren los objetivos secundarios. Primero, se debe crear el código en Matlab que inserte el código en una etiqueta. Después, crear otro código en Matlab que extraiga el código de la fotografía. Esto es para simular la etiqueta que se pega a los productos de anaquel, puesto que no se tiene aún implementada la solución en los productos

Y por último, que el código extraído sea reconocible por un lector QR de un celular.

1.3 ALCANCES

Este trabajo consistirá en la elaboración de un sistema con dos códigos importantes en MATLAB: uno que inserte el código QR en forma de marca de agua en imágenes y, otro que lo pueda extraer.

El primer código toma una imagen y la procesa, de manera que el código quede escondido dentro de ella, aunque no la oculta por completo. Parte de este proyecto consiste en determinar la intensidad del marcado y la manera de insertarlo para obtener mejores resultados. Una vez hecho eso, para simular el escaneo de una etiqueta de un producto, se procedió a imprimir varias imágenes para luego tomarles fotografías, y sobre estas fotografías se aplica el segundo código.

El segundo código toma una imagen y aplica un procesamiento inverso al hecho por el primero, extrayendo el código QR de ella. Después de extraerlo, se escaneó con un lector de un celular convencional, y se considera un éxito si el lector QR lo reconoce.

1.4 ESTADO DEL ARTE

La compañía norteamericana Digimarc hace algo similar a lo planteado en esta tesis. En un video demostrativo [23] muestran cómo su sistema es más rápido que el código de barras tradicional, ya que en todas las oportunidades, el producto fue identificado al primer intento.

Ellos proponen un sistema en el que el espacio de color, el valor de cada pixel de la imagen, el espacio de color e incluso el tipo de impresora son importantes, así como que la alteración a la imagen sea lo menos visible posible. Ellos incluso tienen una aplicación de teléfono para detectar su marca.

El problema, es que ellos cobran mucho por la implementación de su servicio, y actualmente sólo lo utilizan en Estados Unidos.



FIGURA 3.-DETECCIÓN DE LOS CÓDIGOS INSERTADOS EN UNA ETIQUETA DE DIGIMARC.

1.5 ORGANIZACIÓN DE LA TESIS

La tesis consiste en 5 capítulos, siendo el primero la introducción, donde se aborda el problema, los objetivos y el estado del arte.

El segundo capítulo consiste en un marco teórico con todos los conceptos necesarios para la elaboración de esta tesis,

El tercer capítulo consiste en la descripción de los algoritmos utilizados en esta tesis. Los códigos de dichos algoritmos se encuentran en la apéndice, al final de esta tesis.

El cuarto capítulo son las pruebas y resultados de los algoritmos, así como la descripción de éstas, y también las ventajas y desventajas de su aplicación.

El quinto capítulo son las conclusiones de esta tesis, y el trabajo a futuro. Al final se anexa una apéndice con los códigos de Matlab usados, y las referencias usadas.

CAPÍTULO 2

MARCO TEÓRICO

A continuación se presentan algunos conceptos que se consideran necesarios para entender el presente trabajo, y que son tomados de otras técnicas similares a la de *watermarking* que se desarrolla.

2.1 MARCA DE AGUA DIGITAL

La marca de agua digital o *watermarking*, es una técnica esteganográfica de ocultación de información. Su objetivo principal es poner de manifiesto el uso ilícito de cierto servicio digital por parte de un usuario no autorizado. El término marca de agua procede de una analogía con las imágenes que se graban en los billetes para garantizar su autenticidad, muy difíciles y costosas de reproducir y, que generalmente no son perceptibles a simple vista, pudiéndose observar al trasluz. En inglés se utiliza la palabra *watermark*, cuya equivalencia en español es filigrana, sin embargo, lo habitual es usar la traducción literal: *water* (agua) – *mark* (marca).

Esta técnica consiste en insertar un mensaje (oculto o no) en el interior de un objeto digital. El mensaje es un grupo de bits que contiene información sobre el autor o propietario intelectual del objeto digital.

Existen dos tipos de marcaje conocidos como marcaje frágil y marcaje robusto.

Marcaje frágil.- es una inserción débil de la marca, pensada para desaparecer tan pronto como modificamos demasiado el objeto a marcar. Este tipo de marcaje se emplea, por ejemplo, para la autenticación de datos.

Marcaje robusto.- este tipo de marcaje es imposible de eliminar sin degradar el objeto a niveles por debajo de los tolerables. Se emplea, por ejemplo, para la protección de *copyright*. [1]



FIGURA 4.- IMAGEN MARCADA (IZQUIERDA), IMAGEN ORIGINAL (DERECHA).

2.1.1 PROPIEDADES

En imágenes digitales, existen algunos factores importantes a la hora de introducir marca de agua. Estas son:

-Robustez.- Esta característica se refiere a la habilidad de la marca para permanecer en la imagen independientemente de que la calidad haya sido degradada.

-Invisibilidad.- El algoritmo de *watermarking* debe introducir la marca de agua en la imagen de tal modo que ésta no afecte a la calidad de los datos que se encuentran debajo de ella.



FIGURA 5.- IMAGEN MARCADA (IZQUIERDA), IMAGEN ORIGINAL (DERECHA). LA MARCA ES PRÁCTICAMENTE INVISIBLE.

-Cantidad de información.- Para lograr introducir la máxima información, ésta se reparte por toda la imagen. En algunas aplicaciones puede ser un factor determinante.

-Seguridad.- En las técnicas de *watermarking* la marca a introducir debe de ser secreta y sólo accesible para las partes autorizadas, también debe de ser resistente a posibles manipulaciones de usuarios no autorizados.

-Recuperabilidad.- En sistemas de detección donde no disponemos de la imagen original, la marca introducida debe tener unas propiedades concretas que permitan su posterior recuperación por los usuarios permitidos.

-Rapidez de extracción.- que el proceso de detección sea rápido para toda persona autorizada. Sobre todo, en los sistemas que trabajan en tiempo real, es un factor determinante.

-Exclusión de la ambigüedad.- la marca introducida debe de identificar con total seguridad a la persona o empresa propietaria legal de los derechos de la propiedad intelectual de la imagen o al comprador.

2.1.2 APLICACIONES

El *watermarking* tiene múltiples utilidades entre las que se pueden encontrar:

- **Protección de Copyright (identificación del propietario).**- Para la protección de la propiedad intelectual, el propietario de los datos puede introducir una marca de agua representando la información de *copyright* de esos datos. Esta marca de agua puede probar su propiedad cuando alguien ha infringido sus derechos de autor.. Actualmente la industria discográfica realiza grandes esfuerzos en introducir derechos de autor fiables en grabaciones en formato digital.

- **'Fingerprinting' (toma de impresiones digitales).**- El propietario puede emplear una técnica de *'fingerprinting'* para rastrear la procedencia de copias ilegales. En este caso, el propietario de los datos digitales puede introducir diferentes marcas de agua en las copias de los datos que son suministrados a diversos clientes. Esta técnica es comparable a introducir un número de serie en los datos, que está relacionado con la identidad del cliente. Esto permite al dueño de la propiedad intelectual identificar clientes que han roto su contrato (licencia) suministrando estos datos a terceras partes. Se hablará del fingerprinting más adelante.

- **Protección a la copia.**- La información almacenada en una marca de agua puede controlar directamente dispositivos de grabación digitales con el fin de evitar copias ilegales. En este caso, la marca de agua representa un bit de prohibición de copias y los detectores de marcas en las grabadoras determinan si esos datos pueden ser almacenados o no. Actualmente la tecnología OVO incluye técnicas de marcado en su sistema de protección de copias que contiene posibles mensajes como: "copia no permitida", "copiar una sola vez", "agotado el número de copias", entre otros.

-Vigilancia de la transmisión.- Por ejemplo, para introducir marcas de agua en anuncios comerciales, un sistema de vigilancia automatizado puede verificar si los anuncios son transmitidos como han sido contratados.

-Autenticación de datos.- Las marcas de agua frágiles pueden ser empleadas para comprobar la autenticidad de los datos. Este tipo de marcas indican si los datos han sido alterados y suministra información de localización como, por ejemplo, donde han sido alterados estos datos. Las técnicas de marcado de agua se pueden emplear en otras aplicaciones además de como protección. Entre estas otras aplicaciones destacan:

-Seguridad médica.- Puede ser una medida de seguridad útil incrustar la fecha y los datos del paciente en las imágenes médicas.

-Esteganografía (ocultación de los datos).- Las técnicas de marcado de agua pueden ser empleadas para la transmisión de mensajes privados y secretos. Dentro de las técnicas comentadas previamente, vamos a enfocar este escrito al caso del empleo de técnicas de marcado de agua como sistema de *copyright*. Las marcas empleadas deben ser robustas a cualquier alteración que se produzca en su contenido e incluir información suficiente para autenticar al propietario de estos datos. De este modo, se evita que personas no autorizadas se adjudiquen la propiedad de los datos a tratar o que manipulen esta información, eliminándola o haciéndola ilegible. Las marcas que se introducen para autenticar al propietario legal de los datos a marcar, son los denominados marcados robustos, que han de ser invisibles y soportar cualquier manipulación que se produzca en su contenido. [1]

2.1.3 CLASIFICACIÓN

Las marcas de agua pueden ser clasificadas en varias categorías de acuerdo de acuerdo a sus características de las siguientes maneras. [2]

-Perceptibles e imperceptibles.- Como su nombre lo indica, las marcas de agua perceptibles son aquellas que se pueden detectar a simple vista, mientras que las imperceptibles no lo son. Una marca de agua perceptible tiene las siguientes propiedades deseables:

1. es visible pero no obstructiva.
2. es difícil de remover, y
3. se ajusta automáticamente para coincidir con diferentes contenidos de la imagen original, por ejemplo, varía la intensidad de la marca para comparar las texturas locales. [3]

-Robustas, frágiles y semifrágiles.- Las marcas de agua robustas están diseñadas para sobrevivir uso diario legítimo, o ataques intencionales o accidentales, mientras

que las frágiles son vulnerables a estos factores [3]. Por lo general, se desea que una marca de agua sea robusta, pero eso depende de la aplicación.

Las marcas de agua semi-frágiles son vulnerables a distorsiones legítimas, pero destruidas por distorsiones ilegítimas. Estas son más que nada usadas en propósitos de autenticación. [4]

2.1.4 TÉCNICAS DE INCRUSTACIÓN DE MARCAS DE AGUA

-Marcado de agua en el dominio espacial.- Consiste en incrustar la marca de agua en la componente del dominio espacial, o el bit menos significativo de la imagen original, es el método más directo para el marcado de agua digital. Sus ventajas son la baja complejidad y una implementación sencilla, pero tienden a ser menos robustas a ataques.

-Marcado de agua por la transformada.- Este tipo de marcado consiste en transformar los datos de la imagen en frecuencias utilizando la transformada de coseno discreta (DCT del inglés *Discrete Cosine Transform*), la transformada de ondícula discreta (DWT del inglés *Discrete Wavelet Transform*). Estas frecuencias son modificadas de acuerdo a los bits de marcado para que el objetivo de esconder datos sea alcanzado. Luego, la transformada inversa es ejecutada y la imagen marcada es formada.

Entre estas dos técnicas, la de dominio especial tiene las ventajas de una implementación sencilla, mejor calidad visual, y tiempo de codificación más corto. Sin embargo, tiene la desventaja de baja robustez a comparación de la técnica del dominio de la transformada, que tienen mejor robustez y buena calidad visual, pero consumen más tiempo en el proceso de las transformaciones. [15]

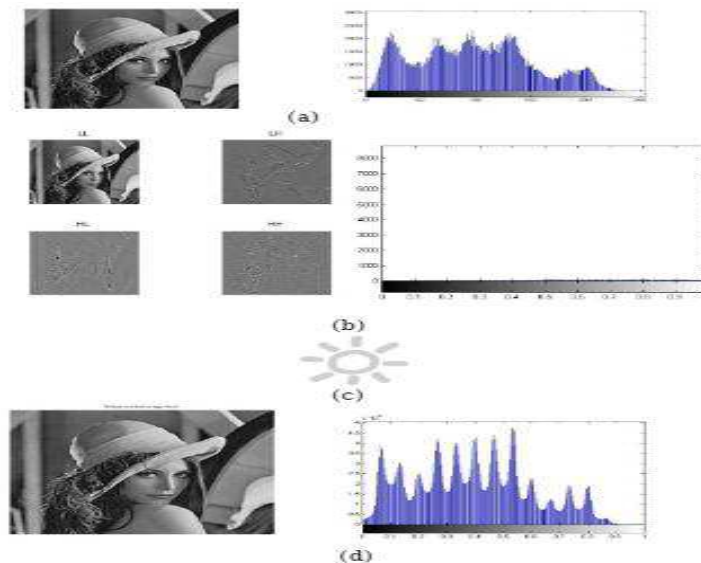


FIGURA 6.- TRANSFORMADA DWT. A) IMAGEN ORIGINAL. B) DWT DE LA IMAGEN. C) MARCA DE AGUA A INSERTAR. D) IMAGEN MARCADA

2.2 FINGERPRINTING

El ***fingerprinting*** es una técnica que identifica, extrae y luego comprime componentes característicos de un archivo, generalmente videos, permitiendo que ese video sea identificado por este archivo llamado *fingerprint*, o huella digital.

El *fingerprinting* no necesariamente se vale de una adición al archivo, no puede ser “removido” ya que no es “agregado”. [6]

Un sistema de *fingerprinting* necesita crear una base de datos que sea usada para identificar una pieza de contenido en específico, en caso de que este sistema encuentre esta pieza otra vez. Una forma de hacer esto sería almacenar toda la pieza de contenido entera en la base de datos y luego comparar cada pieza nueva encontrada con ella, pero a pesar de que funcionaría, sería muy lento y la base de datos requeriría más espacio.

Entonces, en lugar de almacenar la pieza entera, sólo una muestra estadística es almacenada. Por ejemplo, cuatro muestras pueden ser tomadas cada décima de segundo en una señal de audio, como se muestra en la figura 3. [6]

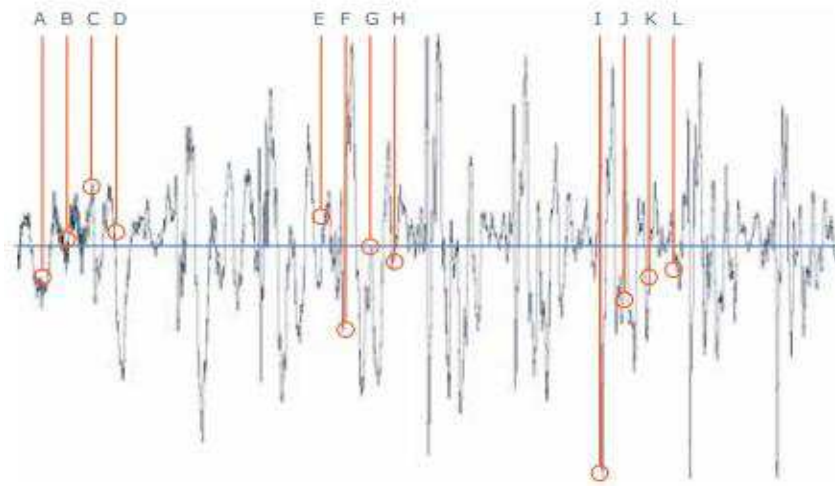


FIGURA 7.- EJEMPLO DE FINGERPRINTING CON UN MUESTREO ESTADÍSTICO.

El ejemplo de arriba es sólo un ejemplo simplificado de cómo podría funcionar un *fingerprinting* de audio. Algoritmos modernos de *fingerprinting* en realidad muestrean una gran variedad de características de audio y/o video para producir la huella digital. Para evitar la piratería, técnicas sofisticadas son usadas para ocultar la naturaleza exacta de la huella digital. Por ejemplo, muestreos temporales y espaciales aleatorios son usados para esconder la información de qué propiedades son usadas para producir la huella digital. [6]

Un buen algoritmo de *fingerprinting* debe satisfacer las siguientes propiedades y, son características a tomar en cuenta para el presente proyecto, aunque no se esté realizando el proceso de *fingerprinting*:

-Discriminación. - los algoritmos de *fingerprinting* deben generar distintas huellas para diferentes contenidos de videos, para que estas huellas sean distinguibles entre sí. Es decir, que sean inconfundibles.

-Robustez. - Los *fingerprints* de un video deben permanecer invariantes, incluso si el video sufre distorsiones perceptuales severas como rotación de marcos o cambio de brillo

-Eficiencia. - Los *fingerprints* deben tener un bajo costo computacional y del almacenamiento, para garantizar una recuperación rápida. La eficiencia es muy valiosa en aplicaciones industriales.

En general, los algoritmos de *fingerprinting* pueden ser clasificados en dos clases. En la primera, la huella es extraída de las características locales. Por ejemplo, las huellas son construidas usando puntos de Harris [7] en la literatura [8], o formadas usando las trayectorias de los puntos de interés [9]. Una de las principales desventajas de estos algoritmos es que el número y dimensión de las características extraídas son muy grandes, lo cual representa un gran costo computacional y de almacenamiento. La segunda clase, las huellas son obtenidas según las características globales del video. En la literatura [10], el gradiente de cada pixel es obtenido calculando las diferencias en luminancia en las direcciones tanto vertical como horizontal y las huellas son generadas desde el centroide de orientaciones de gradiente en luminancia (*centroid of gradient orientations*, por sus siglas en inglés CGO). Aunque el costo computacional de estos algoritmos es bajo comparado con los algoritmos de la primera clase, son vulnerables a la inserción de logotipos y algunas transformadas geométricas como la rotación de marcos.

2.3 ESTEGANOGRAFÍA

La **esteganografía** trata el estudio y aplicación de técnicas que permiten ocultar mensajes u objetos, dentro de otros, llamados portadores, de modo que no se perciba su existencia. Es decir, procura ocultar mensajes dentro de otros objetos y de esta forma establecer un canal encubierto de comunicación, de modo que el propio acto de la comunicación pase inadvertido para observadores que tienen acceso a ese canal.

Una forma de diferenciar la esteganografía con la **criptografía** común es que la criptografía solo cifra los archivos manteniendo el archivo original visible, pero al abrirlo mostrará una secuencia de caracteres que no permitirá su lectura y para ver su contenido original es necesario conocer la clave. En la esteganografía, puede verse un archivo con un formato diferente y, para conocer su contenido original será necesario conocer la clave y el software con el que se ocultó. En la criptografía, el objetivo es

asegurar la confidencialidad de la información ante los ojos de un interceptor que es capaz de ver el criptograma, aún cuando éste conoce el algoritmo que lo genera. En cambio, la esteganografía busca ocultar la presencia del mensaje en sí; ya que si se llega a identificar la posición del mensaje se conoce directamente la comunicación (conocido el algoritmo de ocultación), lo que no ocurre en el caso del criptograma.

Se pueden observar distintos actores implicados en el campo de la esteganografía [11]:

-Objeto contenedor.- se trata de la entidad que se emplea para portar el mensaje oculto. Acudiendo al ejemplo de los mensajes sobre el cuero cabelludo, el objeto contenedor es el esclavo en sí.

-Estego-objeto.- se trata del objeto contenedor más el mensaje encubierto. Siguiendo con el ejemplo, se trata del esclavo una vez se ha escrito en su cuero cabelludo el mensaje y se le ha dejado crecer el pelo.

-Adversario.- son todos aquellos entes a los que se trata de ocultar la información encubierta. En el ejemplo de la prisión, se trata del guardia que entrega los mensajes a uno y otro prisionero. Este adversario puede ser pasivo o activo. Un adversario pasivo sospecha que se puede estar produciendo una comunicación encubierta y trata de descubrir el algoritmo que se extrae del estego-objeto, pero no trata de modificar dicho objeto. Un adversario activo, además de tratar de hallar el algoritmo de comunicación encubierta, modifica el estego-objeto con el fin de corromper cualquier intento de mensajería subliminal.

-Estegoanálisis.- ciencia que estudia la detección (ataques pasivos) y/o anulación (ataques activos) de información oculta en distintas tapaderas, así como la posibilidad de localizar la información útil dentro de la misma (existencia y tamaño).[11]

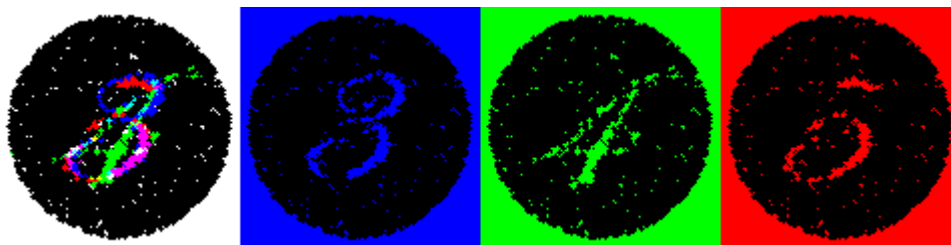


FIGURA 8.- EJEMPLO DE ESTEGANOGRAFÍA.LA MISMA IMAGEN VISTA CON LUCES AZULES, VERDES O ROJAS MUESTRA UN NÚMERO DIFERENTE.

2.4 IDENTIFICACIÓN DE PRODUCTOS

Con el fin de garantizar la frescura de los alimentos y bebidas en el mercado formal, legislaciones internacionales coincidieron en la necesidad de un proceso de identificación de productos que debe indicar su fecha de consumo preferente, y aunque por algunos años, la necesidad de agregar dicha información en los empaques se percibía como un costo adicional de manufactura, hoy las empresas de todos tamaños deben entender los beneficios.

Algunas amenazas que afectan ciertos productos como carne fresca, procesada, pescados, mariscos, alimentos para bebé, ingredientes e incluso agua para beber, colocaron la trazabilidad en el centro de atención.

Los detallistas comenzaron a exigir mayor exactitud en la codificación del producto y a aplicar multas o recargos en los casos de errores de marcaje de fechas en el empaque por parte de sus proveedores, consecuencia de ellos los fabricantes comenzaron a darse cuenta de que un proceso efectivo de identificación de producto y trazabilidad ofrecía una medida de protección a sus marcas.

La sofisticación que actualmente ha alcanzado la tecnología de identificación de producto se refleja en mayores niveles de desempeño y confiabilidad, permitiendo a los fabricantes maximizar su inversión integrando la codificación a maquinarias de empaque de alta velocidad y alcanzando tiempos de funcionamiento mayores.

Hoy día, el proceso de identificación de producto juega un rol fundamental en el control de costos de manufactura, mano de obra, distribución, energía, materiales de empaque, inventario y retiro de producto. Un bajo desempeño en la codificación y el etiquetado sólo incrementa los costos, pero un sistema sólido y eficiente puede generar ahorros en todas las áreas. [12]

2.5 CODIGO DE BARRAS

Un **código de barras** es una representación de datos óptica y legible con máquinas. Los datos usualmente representan información relacionada al objeto en el que el código es colocado.

El código de barras tradicional es un código basado en la representación de un conjunto de líneas paralelas de distinto grosor y espaciado que en su conjunto contienen una determinada información, es decir, las barras y espacios del código representan pequeñas cadenas de caracteres. De este modo, el código de barras permite reconocer rápidamente un artículo de forma única, global y no ambigua en un punto de la cadena logística y así poder realizar inventario o consultar sus características asociadas.

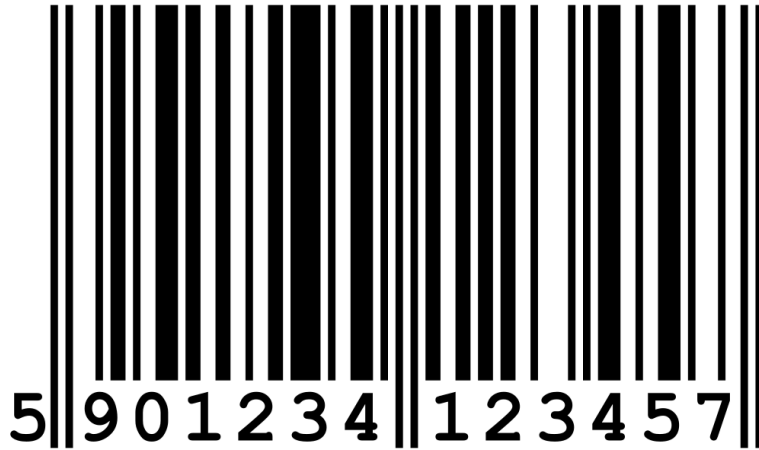


FIGURA 9.-CÓDIGO DE BARRAS EAN 13.

La correspondencia o mapeo entre la información y el código que la representa se denomina simbología. Estas simbologías pueden ser clasificadas en grupos atendiendo a dos criterios diferentes [13]:

-Continua o discreta.- una simbología continua es aquella en la que los caracteres individuales no se pueden interpretar por sí mismos. Esto es debido al hecho de que los caracteres comienzan con una barra y finalizan con un espacio que termina en la barra inicial del siguiente carácter. Un carácter no puede ser interpretado individualmente, ya que no hay manera de conocer el ancho del último espacio sin saber dónde empieza el siguiente carácter. Las simbologías continuas normalmente implementan algún tipo de barra de terminación especial o una secuencia de terminación, tal que el último espacio del último carácter de datos acaba mediante la barra de terminación. Por el contrario, una simbología discreta es aquella en la que todos y cada uno de los caracteres codificados en el símbolo pueden ser interpretados individualmente sin tener en cuenta al resto del código de barras. Estas simbologías tienen caracteres que comienzan y terminan con una barra y, están separados por cierta cantidad de espacio en blanco. El espaciado entre caracteres no lleva ninguna información, su único cometido es el de separar los mismos.

-Bidimensional o multidimensional.- en las simbologías bidimensionales las barras pueden ser anchas o estrechas. Sin embargo, las barras en las simbologías multidimensionales son múltiplos de una anchura determinada (X). De esta forma, se emplean barras con anchura X , $2X$, $3X$, y $4X$.

Ejemplos de códigos lineales son *European Article Number (EAN)*, y el Código Universal de Producto (**UPC**), que son usados para identificar productos. Ejemplos de códigos bidimensionales son el código *Aztec*, y el código *QR*, el cual ha ganado mucha popularidad en los últimos tiempos. [14]

2.6 CÓDIGO QR

Un **código QR** (del inglés *Quick Response code*, "código de respuesta rápida") es un módulo para almacenar información en una matriz de puntos o en un código de barras bidimensional. El código QR provee de un mecanismo de seguridad avanzado con sus algoritmos de corrección de errores.

Un código QR consiste en cuadros negros dentro de una cuadrícula cuadrada en un fondo blanco, que puede ser leído en dispositivo como una cámara y, procesado utilizando la corrección de errores Reed-Solomon hasta que la imagen pueda ser interpretada correctamente. Los datos requeridos luego son extraídos de patrones presentes dentro de las componentes tanto verticales como horizontales de la imagen.

Los códigos QR pueden ser generados en tamaños variados, desde módulos de 21*21 hasta módulos de 177*177, considerando niveles específicos de corrección de errores (Bajo, Medio, Cuartil, y Alto). El código consiste en un patrón de posición, el área de datos, patrón de alineación, casillas de datos, zona de amortiguamiento, y separadores. El patrón de posición consiste en los tres cuadrado en la esquina superior derecha, la esquina superior izquierda, y la izquierda inferior derecha e indican la posición del código. El patrón de alineación se usa para manejar cambios pequeños mientras se lee el código. Las casillas de datos son el área central del código y contienen toda la información codificada. Los separadores se usan para separar los patrones de posición del área de datos.

Ya que la identificación del código QR se puede hacer con cualquier teléfono inteligente y, ya que tiene puede resistir distorsiones hasta cierto punto y aún ser reconocido, se utilizó el código QR para la elaboración del presente proyecto.



FIGURA 10.- ESTRUCTURA DEL CÓDIGO QR.

CAPÍTULO 3

DESARROLLO

El presente desarrollo pretende lograr un sistema más barato y de implementación más sencilla que el propuesto por Digimarc. Éste consiste en un marcado de agua robusto e imperceptible en el dominio espacial. Después de haber marcado la imagen, esta se imprimirá, para luego tomarle una foto a la impresión, e intentar extraer al menos una de las marcas que se introdujeron al principio del proceso. Si bien la robustez no es vital para esta aplicación del marcado de agua debido a que no se esperan ataques intencionales, siempre es preferible que la marca de agua sea robusta para reducir posibles inconvenientes como alteraciones a la etiqueta del producto, cambios en la iluminación etc. La imperceptibilidad se obtendrá trabajando en el espacio de color **YIQ**.

3.1 El modelo YIQ

El modelo **YIQ** define un espacio de color, usado por el estándar de televisión NTSC. **I** significa en fase (*in-phase*, en inglés), mientras que **Q** significa cuadratura (*quadrature*, en inglés) y se refieren a los componentes usados en la modulación de amplitud en cuadratura. La componente **Y** representa la información de luminancia y es el único componente utilizado por los televisores de blanco y negro. **I** y **Q** representan la información de crominancia.

Para pasar de **RGB** (el modelo de color más utilizado) a **YIQ** se utilizan las siguientes ecuaciones:

$$\begin{bmatrix} Y \\ I \\ Q \end{bmatrix} = \begin{bmatrix} 0,299 & 0,587 & 0,114 \\ 0,595716 & -0,274453 & -0,321263 \\ 0,211456 & -0,522591 & 0,311135 \end{bmatrix} \begin{bmatrix} R \\ G \\ B \end{bmatrix} \quad \text{Ec. 3.1}$$

Para pasar de **YIQ** a **RGB**:

$$\begin{bmatrix} R \\ G \\ B \end{bmatrix} = \begin{bmatrix} 1 & 0,9563 & 0,6210 \\ 1 & -0,2721 & -0,6474 \\ 1 & -1,1070 & 1,7046 \end{bmatrix} \begin{bmatrix} Y \\ I \\ Q \end{bmatrix} \quad \text{Ec. 3.2}$$

El sistema **YIQ** pretende aprovechar las características de la respuesta humana al color. El ojo humano es más sensible a los cambios en el rango de naranja a azul (**I**) que en el rango morado a verde (**Q**).

Las transformadas de color no correlacionadas, como la YIQ o YUV, son más adecuadas para el marcado de agua, la detección se dificulta y las marcas pueden ser más débiles lo cual implica mayor imperceptibilidad [17]. También resultan tener más robustez que el espacio RGB ante ataques como compresión, escalado, rotación, recortes o ruido. [18] Esto, sumado a la resistencia inherente del código QR a distorsiones ofrecerá un algoritmo más efectivo y robusto para la correcta identificación de los productos.

3.2 EXPLICACIÓN DEL MÉTODO DEL CÓDIGO DE INSERCIÓN

La primera parte de este algoritmo consiste en la inserción de las marcas de agua, en este caso un mismo código QR, en toda la imagen para después imprimirla. Todo el procesamiento de imágenes se llevará a cabo en MATLAB y en Photoshop,

El primer paso es convertir la imagen del espacio de color **RGB**, al espacio **YIQ**, esto debido a que las componentes **I** y **Q** son menos sensibles a los cambios para el ojo humano. En este caso, se trabajará con la componente **Q**. Una vez aislada la componente **Q** de las otras, la imagen se marcará con el **método de los píxeles circundantes** propuesto por *Feng-Hsing, Lakhmi C. Jain, y Jeng-Shyang Pan* [19]. Este método provee de ventajas como buena imperceptibilidad, mejor robustez contra compresión JPEG, fácil implementación y un parámetro para controlar el balance entre imperceptibilidad y robustez. La mejor robustez contra compresión JPEG implica que el método es resistente a distorsiones causadas de manera no deliberada, como las causadas al imprimir una imagen y luego tomarle una fotografía a la impresión, lo cual es adecuado para esta aplicación.

Primero, se calculará el valor medio de los píxeles circundantes de cada píxel de la imagen. Esto formará una nueva matriz de valores medio, la cual se llamará μ , esto es:

$$\mu = \sum_{j=x_i-1}^{x_i+1} \sum_{k=y_i-1}^{y_i+1} \frac{q(j, k)}{8} \quad \text{Ec 3.3}$$

dónde $q(j, k)$ es el píxel circundante de cada píxel de la imagen $p_i(x_i, y_i)$.

Hecho esto, a los píxeles elegidos para almacenar la marca se les insertará la marca w (que sólo contiene valores binarios) mediante la siguiente ecuación:

$$p_i'(x_i, y_i) = \begin{cases} \mu - \delta & \text{si } w_i = 0 \\ \mu + \delta & \text{si } w_i = 1 \end{cases} \quad \text{Ec. 3.4}$$

En esta ecuación, δ es una variable controlada por el usuario, tal que una δ pequeña mantendrá una mejor calidad de la imagen, pero la marca perderá robustez, en tanto

que una δ mayor mejorará la robustez, pero la imagen perderá calidad, y la marca se hará más visible.

Después de que se complete la inserción, simplemente se colocan los píxeles marcados p_i' en sus posiciones originales en la imagen p_i . Todo esto fue hecho en la componente **Q** de la imagen, por lo que lo único que queda es reemplazar la componente **Q** original con la componente **Q** que se acaba de marcar, para después volver a transformar la imagen al formato **RGB**, para completar la inserción.

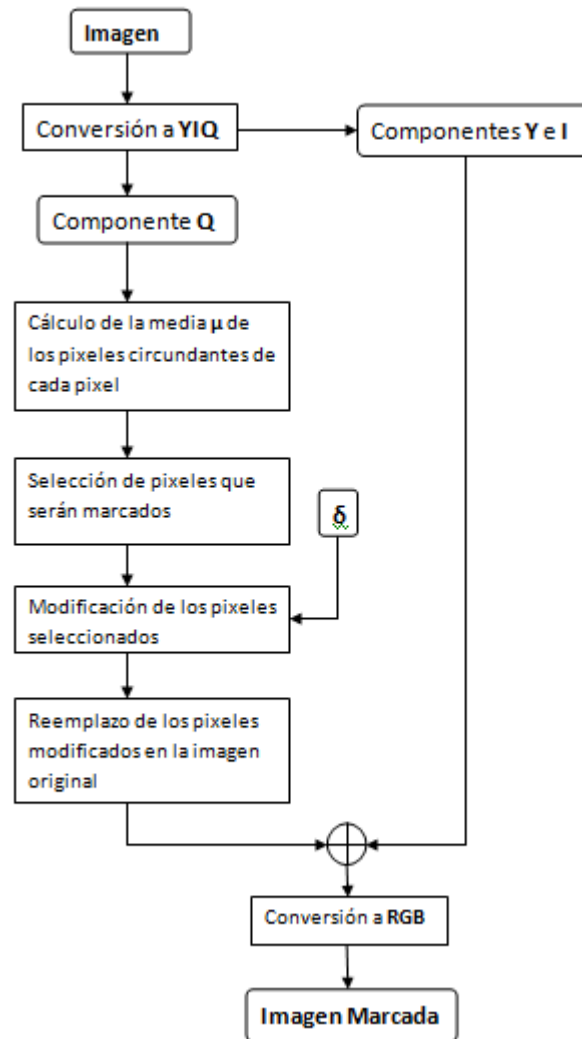


FIGURA 11.- DIAGRAMA DE FLUJO DEL PROCESO DE INSERCIÓN

Cabe señalar, que tanto δ , como la parte de la imagen en la que se colocará la marca, son decisión del usuario. En el capítulo siguiente se determinarán estos parámetros en base a los resultados experimentales que se obtengan.

3.3 EXPLICACIÓN DEL MÉTODO PARA EL CÓDIGO DE EXTRACCIÓN

Para la extracción de la marca escondida, la imagen original no es requerida. De la misma forma que en el proceso de inserción, el primer paso es transformar la imagen de **RGB** a **YIQ**, luego aislar la componente **Q**, luego se calculará el valor medio de los pixeles circundantes de cada pixel de la imagen, de la misma forma que se hizo para la inserción. A esta nueva matriz se le llamará μ' , y con esta nueva matriz, se determinarán los bits correspondientes a la marca de agua mediante la siguiente ecuación:

$$w_i'(x_i, y_i) = \begin{cases} 0 & \text{si } p_i'(x_i, y_i) < \mu' \\ 1 & \text{si } p_i'(x_i, y_i) \geq \mu' \end{cases}$$

Ec. 3.5

Donde $p_i'(x_i, y_i)$ es la imagen de la cual se obtuvo μ' , es decir, la componente **Q** de la imagen marcada.

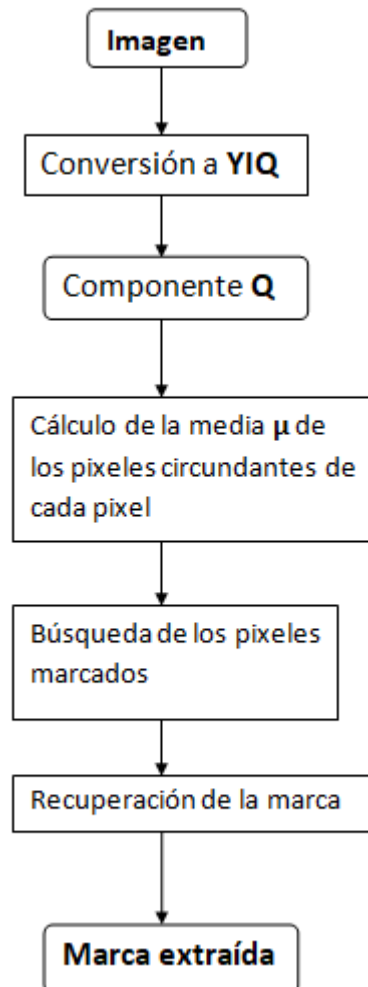


FIGURA 12.- DIAGRAMA DE FLUJO DEL PROCESO DE EXTRACCIÓN

CAPÍTULO 4

EXPERIMENTOS Y RESULTADOS

Para evaluar el desempeño del algoritmo se realizarán tres pruebas de eficiencia en una serie de imágenes. La primera es obtener el **PSNR** (del inglés *Peak Signal-to-Noise Ratio*), la segunda es obtener el **SSIM** (del inglés *Structural Similarity Index*), y la tercera es simplemente tratar de leer el código QR extraído con un celular, de la imagen desplegada en la pantalla de un ordenador, y del escaneo de una impresión.

4.1 DESCRIPCIÓN DE LAS PRUEBAS DE EFICIENCIA

PSNR

La **Proporción Máxima de Señal a Ruido** o **PSNR** es un término definido como la relación entre la máxima energía posible de una señal y el ruido que afecta a su representación fidedigna. Debido a que muchas señales tienen un gran rango dinámico, el PSNR se expresa generalmente en escala logarítmica, utilizando como unidad el decibelio. Para definir el PSNR, primero se formula el error cuadrático medio **MSE** (del inglés *Mean Square Error*):

$$\text{MSE}(O, O') \equiv \frac{1}{M \times N} \sum_{i=0}^{M-1} \sum_{j=0}^{N-1} (p(i, j) - p'(i, j))^2$$

Ec. 4.1

Donde $p(i, j)$ es cada pixel de la imagen original y $p'(i, j)$ cada pixel de la imagen marcada. Después, PSNR queda definido como:

$$\text{PSNR}(O, O') \equiv 10 \times \log \left(\frac{255^2}{\text{MSE}(O, O')} \right)$$

Ec. 4.2

Donde O es la imagen original y O' la imagen marcada. 255 es el valor máximo que puede tener un pixel en ambas imágenes. Entre mayor sea la similitud entre ambas imágenes, mayor será el PSNR, y si éste se calcula con dos imágenes iguales, MSE será igual a 0, lo que hace que PSNR tienda a infinito.

Por lo general, un PSNR de **30 dB o mayor** se toma como aceptable. [19] La mayor limitante del PSNR es que es una cifra meramente matemática, y no toma en cuenta cómo funciona el sistema de visión humana. Esto se ve mejor en la **figura 5**, en la que

dos imágenes prácticamente iguales generan un PSNR menor a 30 dB. A la imagen alterada simplemente se le sumó 10 a cada pixel. Es por eso que se introdujo el **SSIM**.

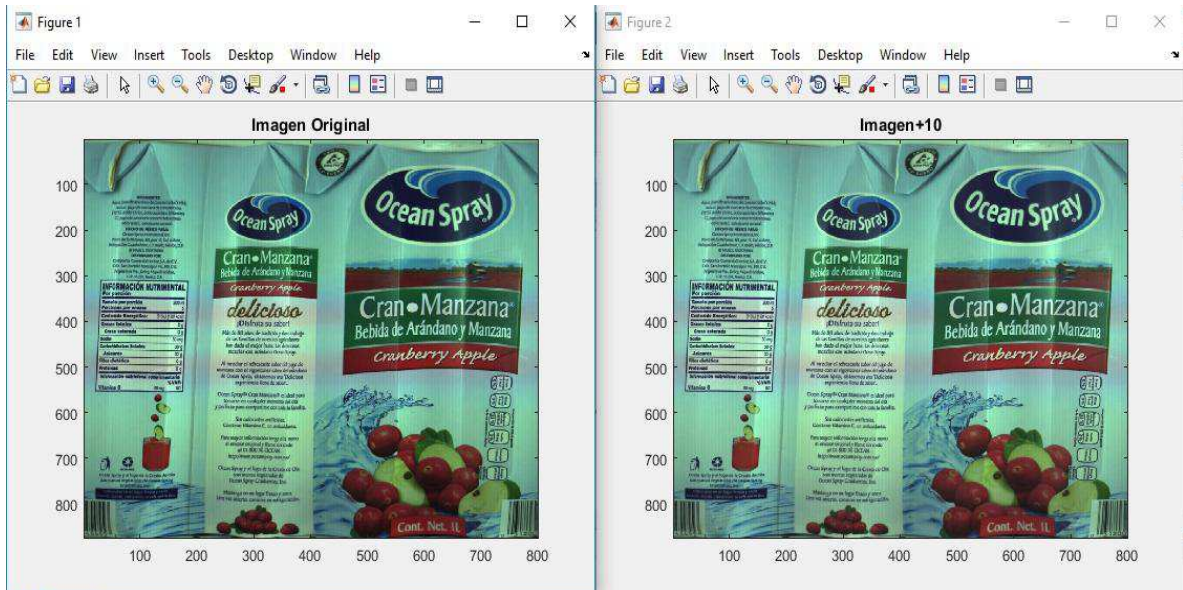


FIGURA 13.- COMPARATIVA DE UNA IMAGEN ORIGINAL, Y LA MISMA IMAGEN DESPUÉS DE QUE SE LE SUMÓ 10 A CADA PIXEL. EL PSNR ES DE 28.13 dB.

SSIM

El **índice de similitud estructural** es un método para predecir la calidad percibida de imágenes y videos digitales. La diferencia respecto a otras técnicas como el error medio cuadrático o el PSNR, es que estas estiman errores absolutos, mientras que el SSIM es un modelo basado en la percepción que considera la degradación como un cambio percibido de información estructural, mientras que agrega fenómenos perceptuales importantes, incluida máscaras de luminosidad y máscaras de contraste. La información estructural es la idea que los pixeles tienen fuertes dependencias entre sí, especialmente cuando están cerca en el espacio. Estas dependencias llevan información importante sobre la estructura de los objetos en la escena visual. Las mascaradas de luminosidad son un fenómeno en el cual las distorsiones en la imagen son menos visibles en regiones brillantes, mientras que en las mascaradas de contraste las distorsiones son menos visibles en regiones donde hay más texturas.

El índice de similitud estructural consiste en la multiplicación de tres términos: luminosidad, contraste y estructura:

$$SSIM(x, y) = [l(x, y)^\alpha \cdot c(x, y)^\beta \cdot s(x, y)^\gamma]$$

Ec. 4.3

Donde, sus componentes luminancia l , contraste c , y estructura s equivalen a:

$$l(x, y) = \frac{2\mu_x\mu_y + c_1}{\mu_x^2 + \mu_y^2 + c_1}$$

$$c(x, y) = \frac{2\sigma_x\sigma_y + c_2}{\sigma_x^2 + \sigma_y^2 + c_2}$$

$$s(x, y) = \frac{\sigma_{xy} + c_3}{\sigma_x\sigma_y + c_3}$$

Ecs. 4.4

donde μ_x , μ_y , σ_x , σ_y , and σ_{xy} son las medias locales, desviaciones estándar y covarianzas cruzadas de x , y respectivamente. Con $C_1 = (K_1L)^2$ y $C_2 = (K_2L)^2$ donde L es el rango dinámico de cada pixel, **255** en este caso y $K_1 = 0.001$ y $K_2 = 0.003$.

Si $\alpha = \beta = \gamma = 1$ (por default), y $C_3 = C_2/2$ (selección default de C_3) el índice se simplifica a:

$$SSIM(x, y) = \frac{(2\mu_x\mu_y + c_1)(2\sigma_{xy} + c_2)}{(\mu_x^2 + \mu_y^2 + c_1)(\sigma_x^2 + \sigma_y^2 + c_2)}$$

Ec. 4.5

El SSIM resultante es un valor entre 0 y 1, y 1 únicamente es alcanzable en dos imágenes completamente idénticas, mientras que 0 son dos imágenes completamente diferentes. Generalmente se calcula en una ventana de 8x8.

4.2 RESULTADOS DE LAS PRUEBAS

Se trabajó sobre 8 imágenes, que son etiquetas de productos. Estas fueron encontradas en internet, y son las siguientes:

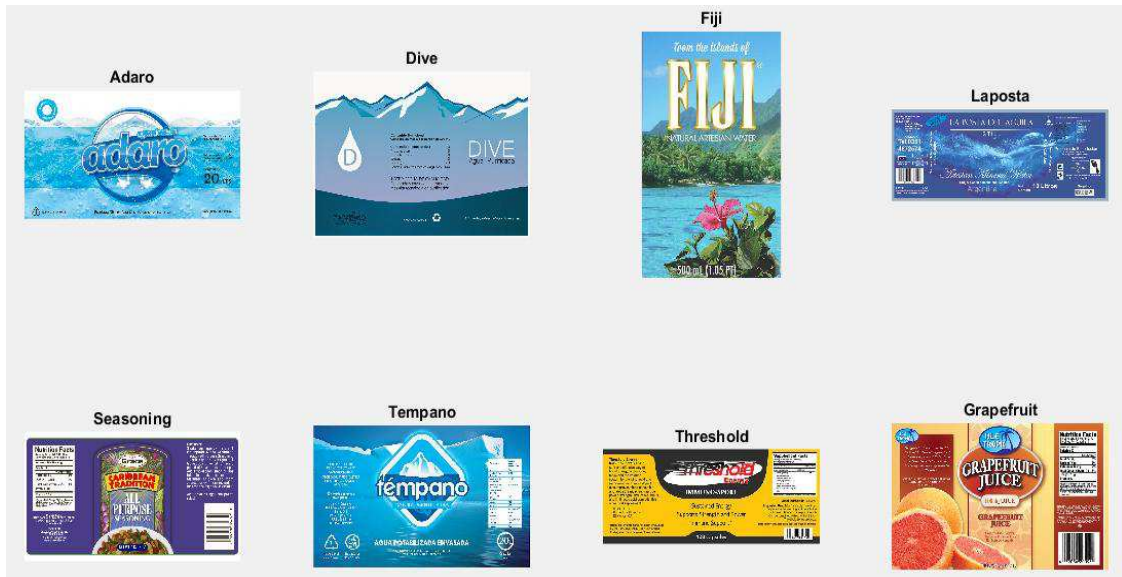


FIGURA 14.- IMÁGENES SOBRE LAS QUE SE TRABAJÓ. A)ADARO, B)DIVE, C)FIJI, D)LAPOSTA, E)SEASONING, F)TEMPANO, G)THRESHOLD, H)GRAPEFRUIT.

La primera prueba consistió en insertar exactamente 9 códigos QR en cada imagen, y luego calcular el **PSNR** y el **SSIM** de la imagen resultante respecto a la imagen original. Todo esto se hizo en Matlab.

El proceso de inserción consistió en dividir la imagen en 9 segmentos de igual área, y luego a través de un generador de números aleatorios, determinar en qué parte de ese segmento se le insertará su código QR respectivo. La segmentación de la imagen en 9 partes iguales tiene el inconveniente de que se necesita que cada una de las dimensiones m y n de la imagen sean divisibles por 3, es por eso que antes de llevar a cabo cualquier procesamiento, se escala la imagen una dimensión que cumpla con este requisito con el comando de matlab **imresize**. El cambio, sin embargo, es mínimo y sólo puede agregar a lo mucho 2 píxeles por cada dimensión.

El código QR de cada imagen fue generado con un generador en línea de códigos QR, y éste es sólo una cadena de caracteres que describe cada imagen, p.e. el código de *grapefruit* sólo es la palabra *grapefruit*. Esto es meramente ilustrativo, por lo que en la práctica el código puede ser el número de identificación del producto, de la empresa que lo fabrica o distribuye, o incluso una URL, pero en este caso se eligió el nombre de la imagen para cada código. El código QR fue reescalado con el comando **imresize** de Matlab, a un tamaño igual la dimensión mínima de la imagen entre **3.5**, es decir, si la

imagen es de 640 x 1030, se toma el 640 como referencia, se divide entre 3.5, y el resultado es el tamaño del nuevo código QR.

Se eligió 3.5 para que el código quepa en cada uno de las 9 partes en las que se dividió la imagen. Una vez hecho todo esto, el código QR se insertó, con una $\delta = 10$ en la tercera componente de la imagen transformada de RGB a YIQ, que es la componente Q. La nueva imagen resultante se guardó con el comando de Matlab `imsave` en formato PNG, un formato sin pérdidas para luego extraer los códigos.

Es en este punto antes de la extracción, en la que se evaluó el PSNR y el SSIM de las imágenes marcadas respecto a las originales:



FIGURA 15.- IMAGEN “LAPOSTA” A) SIN MARCAS, B) MARCADA.

Imagen	SSIM ($\delta=10$)	SSIM ($\delta=20$)	PSNR($\delta=10$)	PSNR($\delta=20$)
Adaro	0.9452	0.8218	30.5299	28.4528
Dive	0.9288	0.7752	30.0923	27.7492
Fiji	0.8975	0.8394	31.7229	28.1982
Grapefruit	0.984	0.9324	29.8192	27.9623
Laposta	0.9485	0.9100	31.7977	28.3830
Seasoning	0.9536	0.8317	31.0644	27.9410
Tempano	0.9758	0.9261	30.2024	28.1433
Threshold	0.8340	0.8237	31.6007	27.6854

FIGURA 16.- TABLA DE PSNR Y SSIM PARA CADA IMAGEN MARCADA, CON $\delta=10$ y $\delta=20$.

Para la extracción, se utilizaran 3 métodos diferentes, esto porque el método introducido anteriormente no siempre funciona, como se ve en la siguiente imagen:

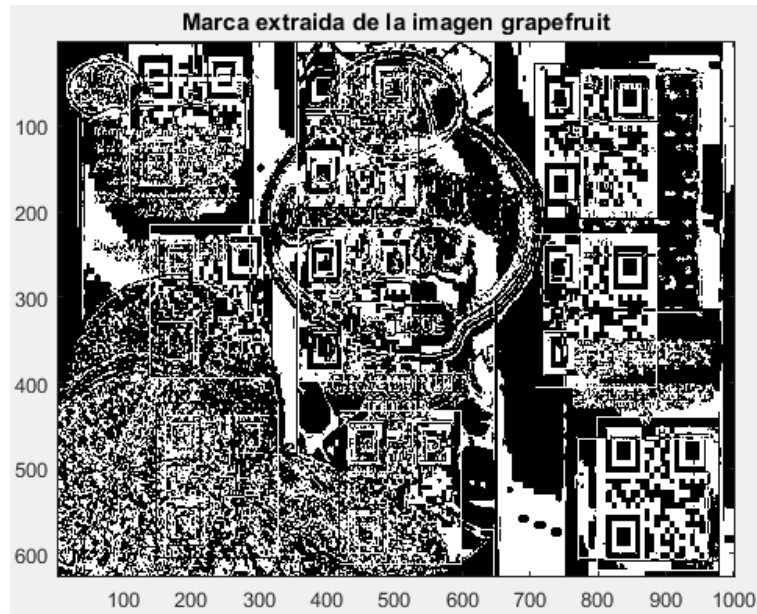


FIGURA 17.- MARCA EXTRAÍDA DELA IMAGEN GRAPEFRUIT.

Debido a esto, las marcas se extraerán de 3 maneras diferentes: 1) de la forma como se habló en el capítulo anterior, pero aplicando un filtro de mediana de 50x50 a la matriz μ antes de extraer la marca, 2) utilizando la imagen original como referencia para extraer la marca, y 3) binarizando la matriz μ en lugar de la extracción utilizada anteriormente. Las extracciones lucen como sigue:

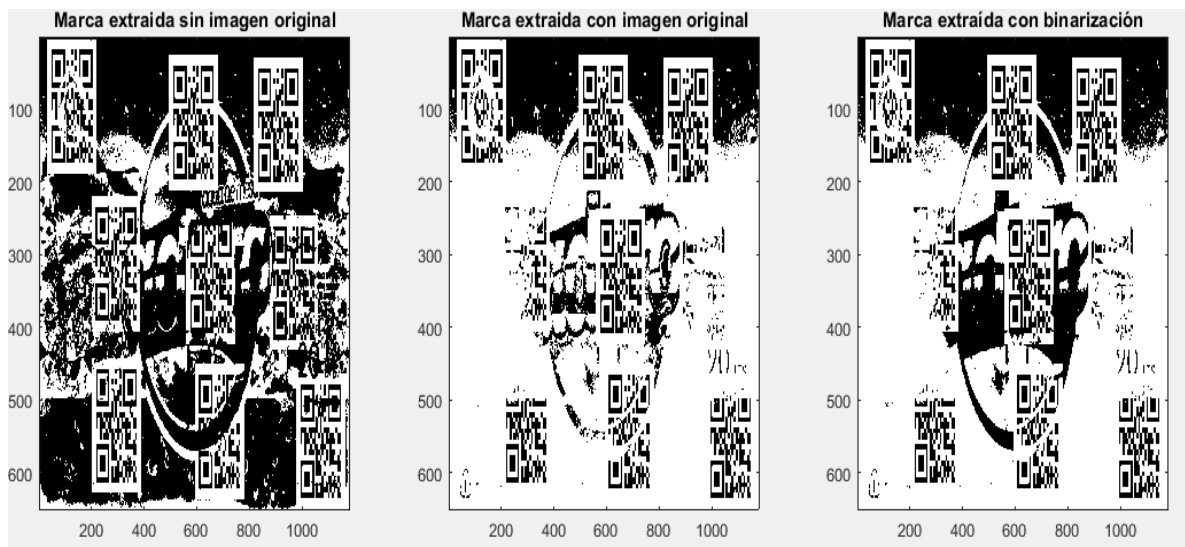


FIGURA 18.- MARCAS EXTRAÍDAS DE LA IMAGEN "ADARO". A) SIN USAR LA IMAGEN ORIGINAL, B) USANDO IMAGEN ORIGINAL, C) EXTRAÍDA CON BINARIZACIÓN.

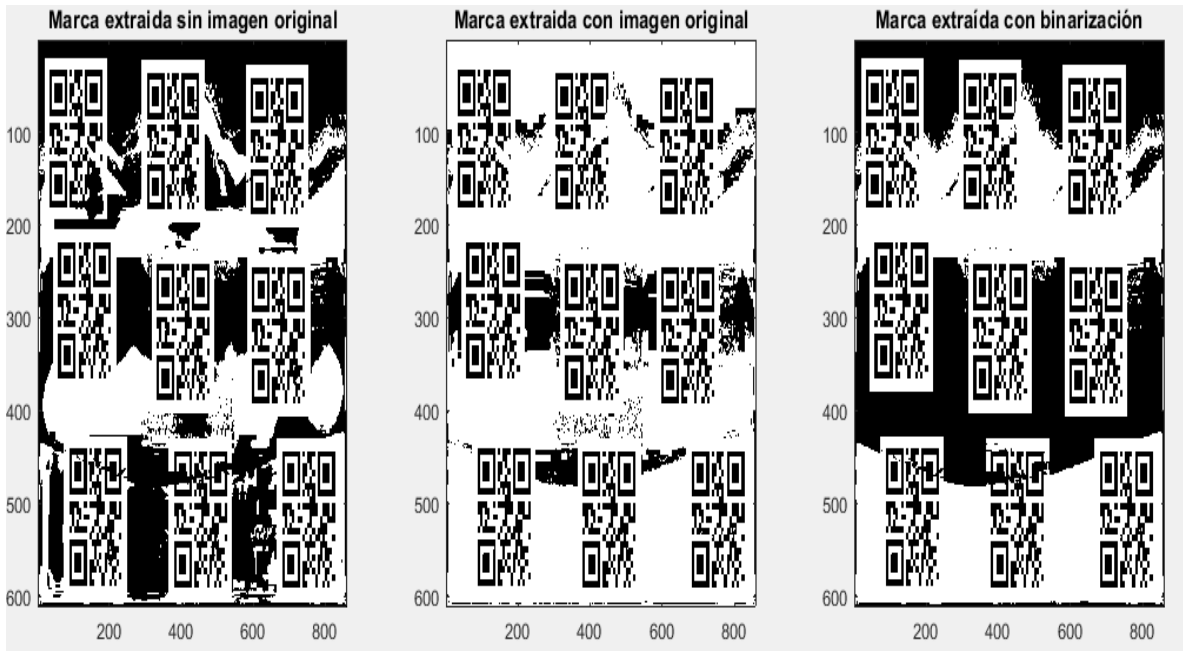


FIGURA 19.- MARCAS EXTRAÍDAS DE LA IMAGEN “DIVE”. A) SIN USAR LA IMAGEN ORIGINAL, B) USANDO IMAGEN ORIGINAL, C) EXTRAÍDA CON BINARIZACIÓN.

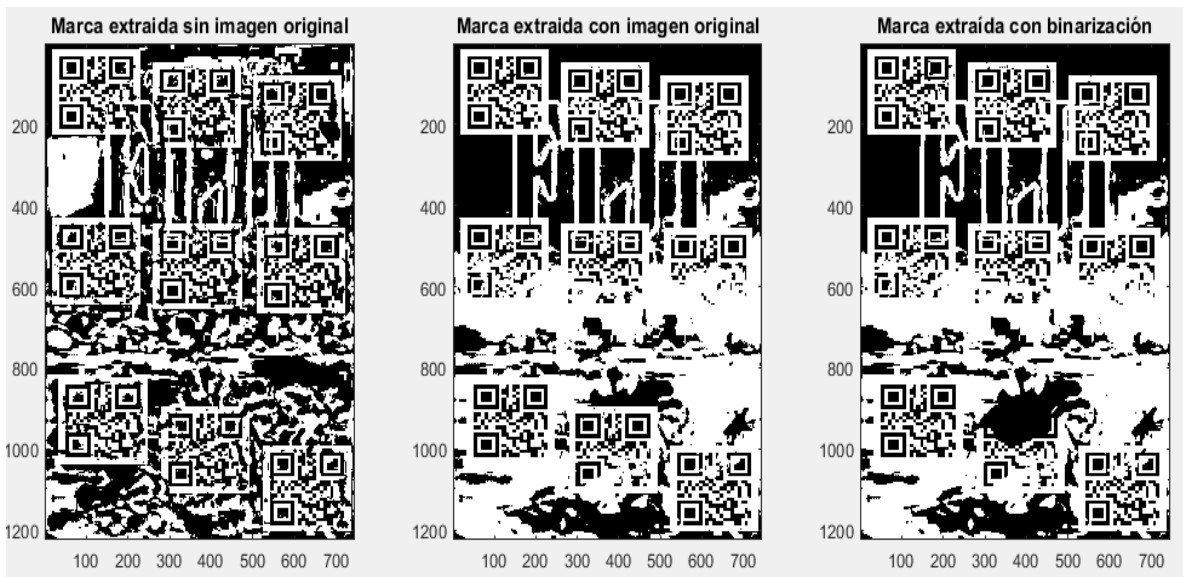


FIGURA 20.- MARCAS EXTRAÍDAS DE LA IMAGEN “FIJI”. A) SIN USAR LA IMAGEN ORIGINAL, B) USANDO IMAGEN ORIGINAL, C) EXTRAÍDA CON BINARIZACIÓN.

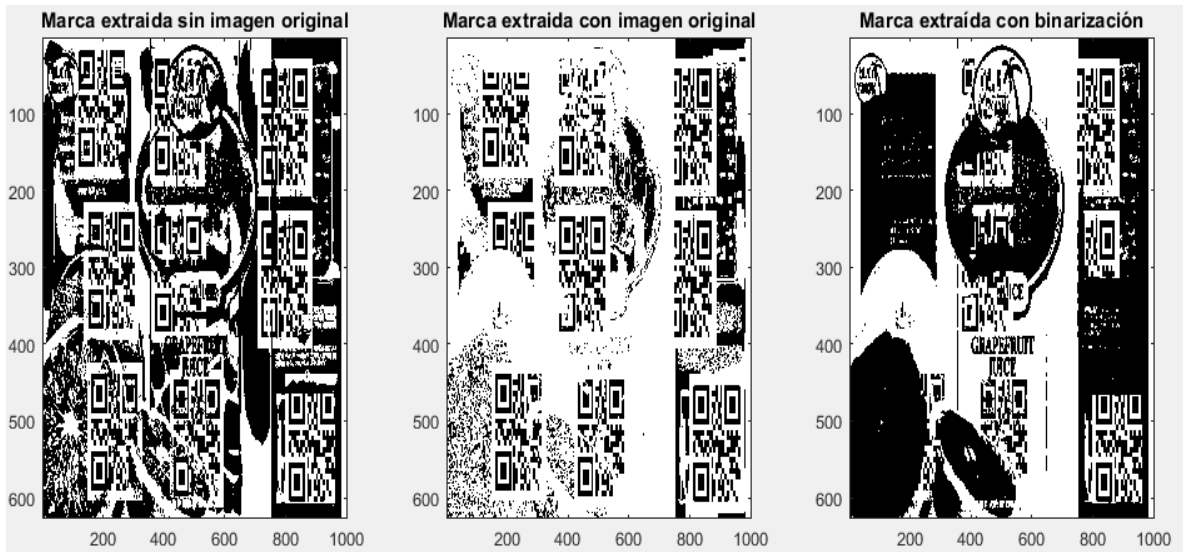


FIGURA 21.- MARCAS EXTRAÍDAS DE LA IMAGEN “GRAPEFRUIT”. A) SIN USAR LA IMAGEN ORIGINAL, B) USANDO IMAGEN ORIGINAL, C) EXTRAÍDA CON BINARIZACIÓN.

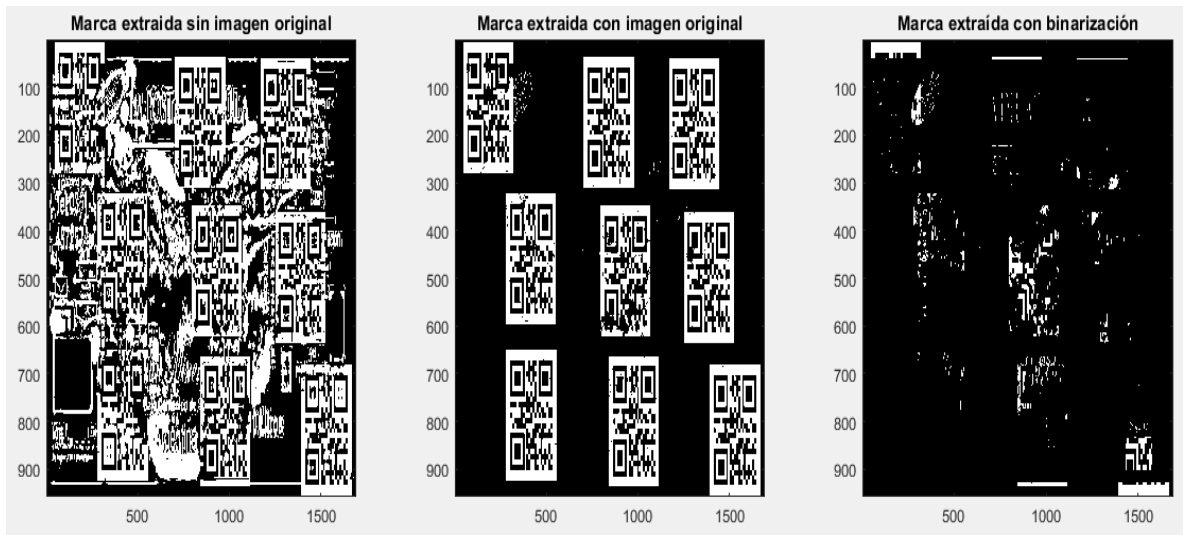


FIGURA 22.- MARCAS EXTRAÍDAS DE LA IMAGEN “LAPOSTA”. A) SIN USAR LA IMAGEN ORIGINAL, B) USANDO IMAGEN ORIGINAL, C) EXTRAÍDA CON BINARIZACIÓN.

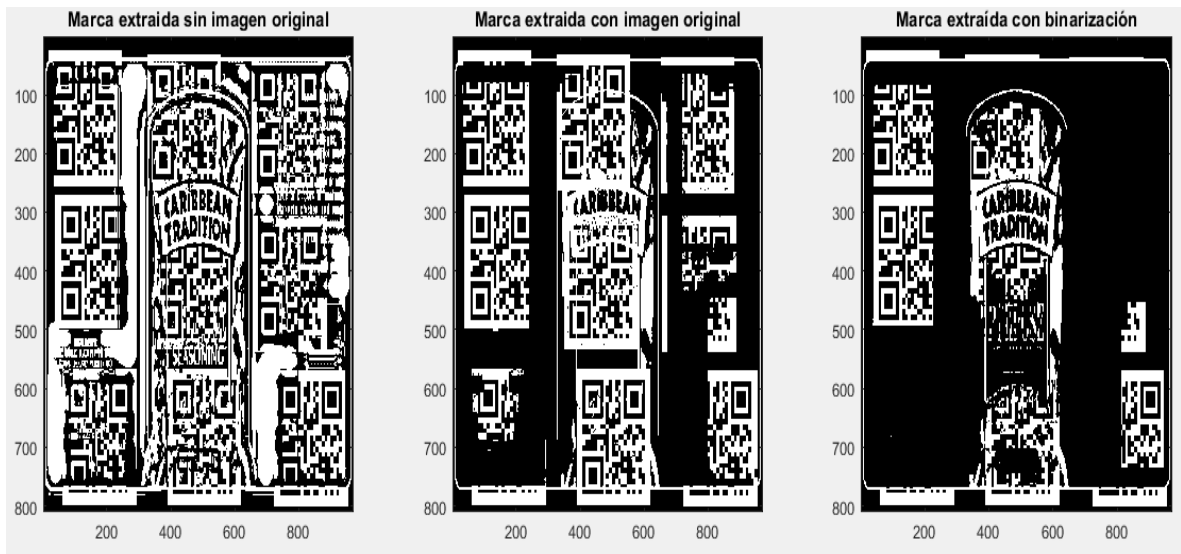


FIGURA 23.- MARCAS EXTRAÍDAS DE LA IMAGEN “SEASONING”. A) SIN USAR LA IMAGEN ORIGINAL, B) USANDO IMAGEN ORIGINAL, C) EXTRAÍDA CON BINARIZACIÓN.

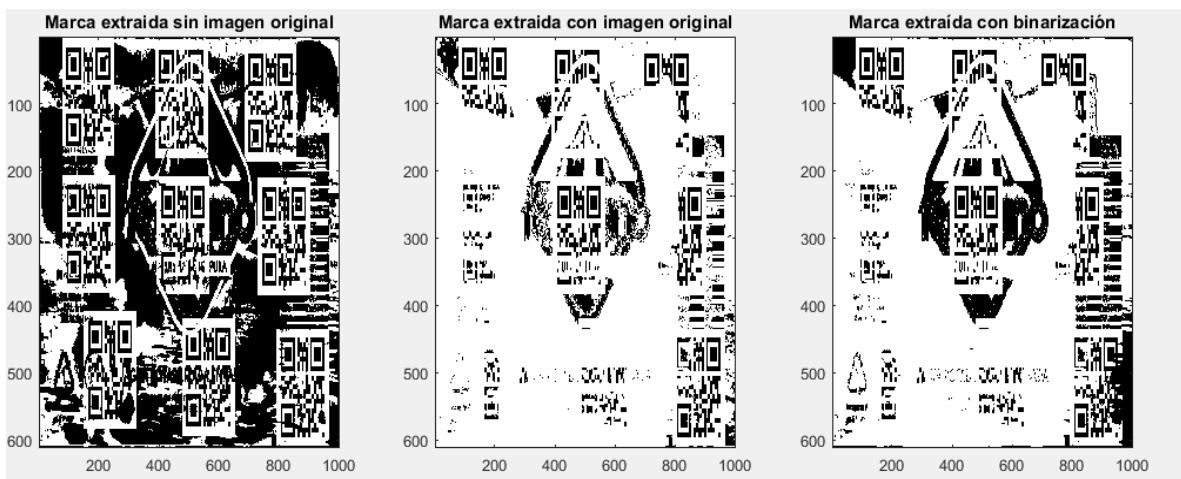


FIGURA 24.- MARCAS EXTRAÍDAS DE LA IMAGEN “TEMPANO”. A) SIN USAR LA IMAGEN ORIGINAL, B) USANDO IMAGEN ORIGINAL, C) EXTRAÍDA CON BINARIZACIÓN.

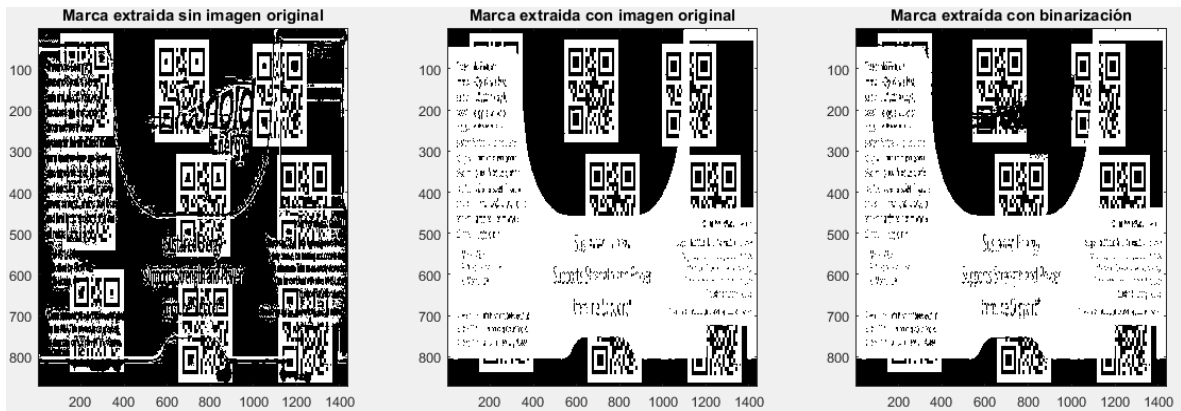


FIGURA 25.- MARCAS EXTRAÍDAS DE LA IMAGEN “THRESHOLD”. A) SIN USAR LA IMAGEN ORIGINAL, B) USANDO IMAGEN ORIGINAL, C) EXTRAÍDA CON BINARIZACIÓN.

Imagen	Marcas detectables sin imagen original	Marcas detectables con imagen original	Marcas detectables con binarización
Adaro	4 de 9	2 de 9	2 de 9
Dive	3 de 9	9 de 9	8 de 9
Fiji	0 de 9	0 de 9	0 de 9
Grapefruit	1 de 9	1 de 9	0 de 9
Laposta	0 de 9	7 de 9	0 de 9
Seasoning	1 de 9	0 de 9	0 de 9
Tempano	0 de 9	0 de 9	0 de 9
Threshold	0 de 9	1 de 9	0 de 9

FIGURA 26.- TABLA DE CÓDIGOS QR DETECTABLES DE CADA IMAGEN CON CADA MÉTODO DE EXTRACCIÓN UTILIZADO.

En la tabla anterior se puede apreciar que, incluso con la imagen original las marcas no siempre son extraíbles. Muchos factores están implicados en este hecho, como el posicionamiento de las marcas respecto a bordes, cambios severos de color en donde las marcas están colocadas, la distribución de colores en la parte en la que la marca fue insertada, o que la marca haya sido insertada en un área de la imagen que tenga un alto brillo o uno muy bajo.

De esta forma se plantea que si se cambia la posición, tamaño y/o intensidad de las marcas estas se pueden volver detectables. La prioridad es que se vuelvan detectables las marcas extraídas sin la imagen original, porque en la práctica ese es el método más práctico y económico para identificación de productos, ya que no se necesita una base de datos con las imágenes originales.

En la práctica, basta que una sola marca sea leída para la identificación del producto.

La posibilidad de que el producto sea detectado simplemente con desplazar el producto a través de un detector sin necesidad de apuntarlo o pasarlo por una parte en particular se incrementa si varias marcas son colocadas y todas son detectables. Un producto que viene en una caja, por ejemplo, sólo necesitaría una marca en cada una de las caras de la caja para que este proceso sea optimizado. Esto se vuelve más

complejo con un producto como una botella de agua cilíndrica, ya que la distribución de marcas no es tan simple. Sin embargo, se puede asumir que en un producto cilíndrico bastaría con 4 marcas distribuidas de manera equitativa a lo largo de la etiqueta para que esta sea detectada al ser pasada por un detector, sin necesidad de apuntarla. Por estos motivos, se busca que al menos 4 de 9 marcas sean detectables.

Los resultados de los cambios realizados son los siguientes:

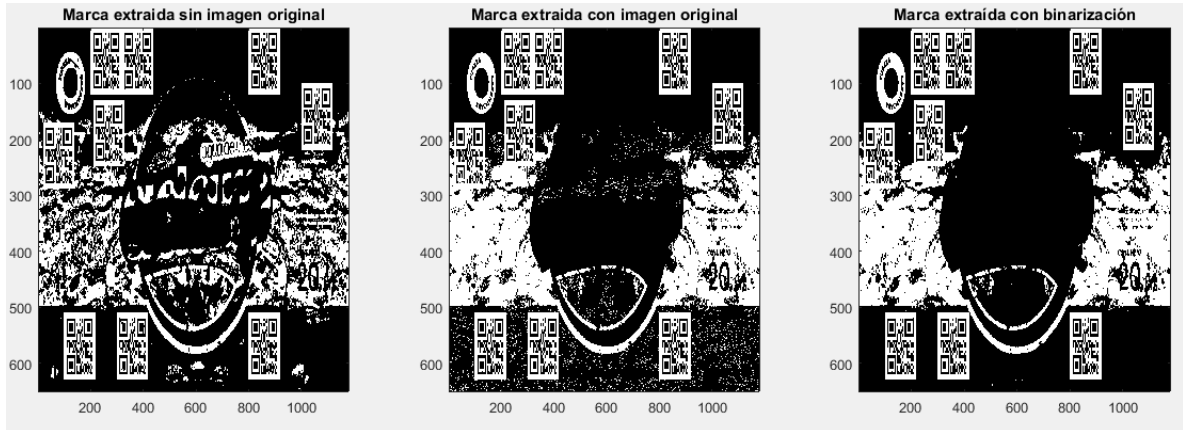


FIGURA 27.- MARCAS EXTRAÍDAS DE LA IMAGEN “ADARO” DESPUES DE MODIFICAR LAS MARCAS INSERTADAS. A) SIN USAR LA IMAGEN ORIGINAL, B) USANDO IMAGEN ORIGINAL, C) EXTRAÍDA CON BINARIZACIÓN.

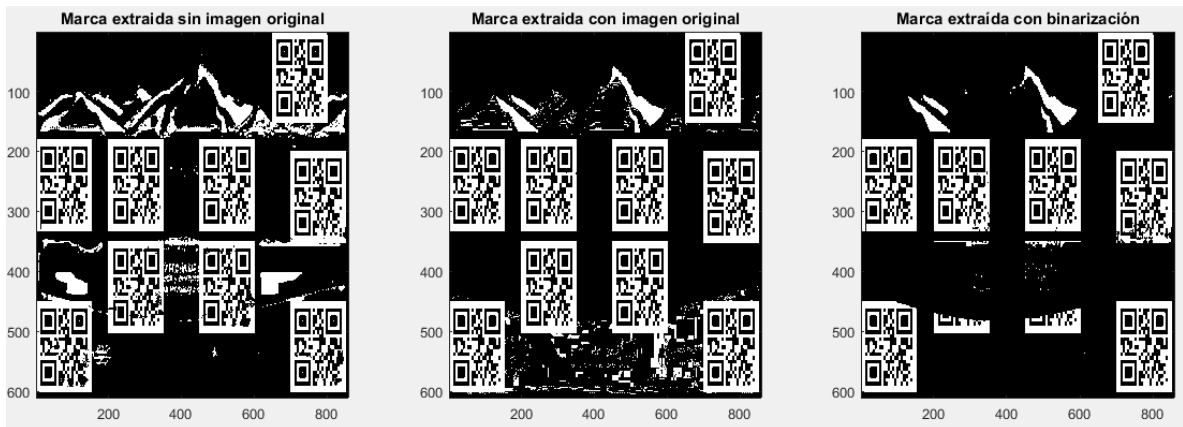


FIGURA 28.- MARCAS EXTRAÍDAS DE LA IMAGEN “DIVE” DESPUES DE MODIFICAR LAS MARCAS INSERTADAS. A) SIN USAR LA IMAGEN ORIGINAL, B) USANDO IMAGEN ORIGINAL, C) EXTRAÍDA CON BINARIZACIÓN.

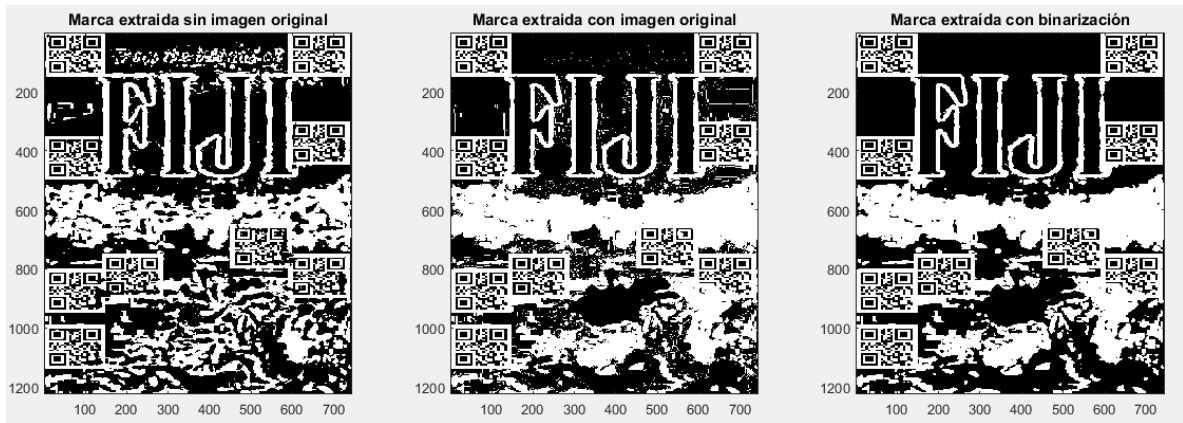


FIGURA 29.- MARCAS EXTRAÍDAS DE LA IMAGEN “FIJI” DESPUES DE MODIFICAR LAS MARCAS INSERTADAS. A) SIN USAR LA IMAGEN ORIGINAL, B) USANDO IMAGEN ORIGINAL, C) EXTRAÍDA CON BINARIZACIÓN.

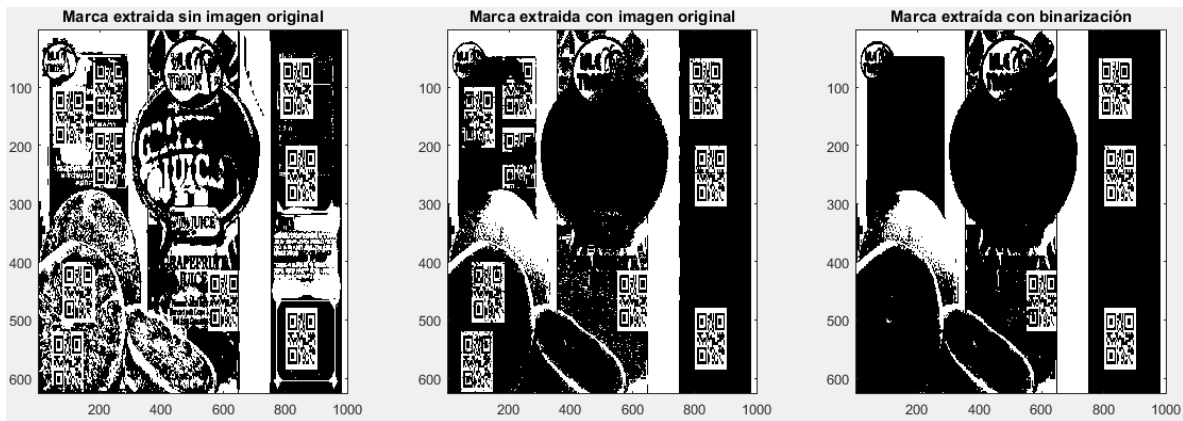


FIGURA 30.- MARCAS EXTRAÍDAS DE LA IMAGEN “GRAPEFRUIT” DESPUES DE MODIFICAR LAS MARCAS INSERTADAS. A) SIN USAR LA IMAGEN ORIGINAL, B) USANDO IMAGEN ORIGINAL, C) EXTRAÍDA CON BINARIZACIÓN.

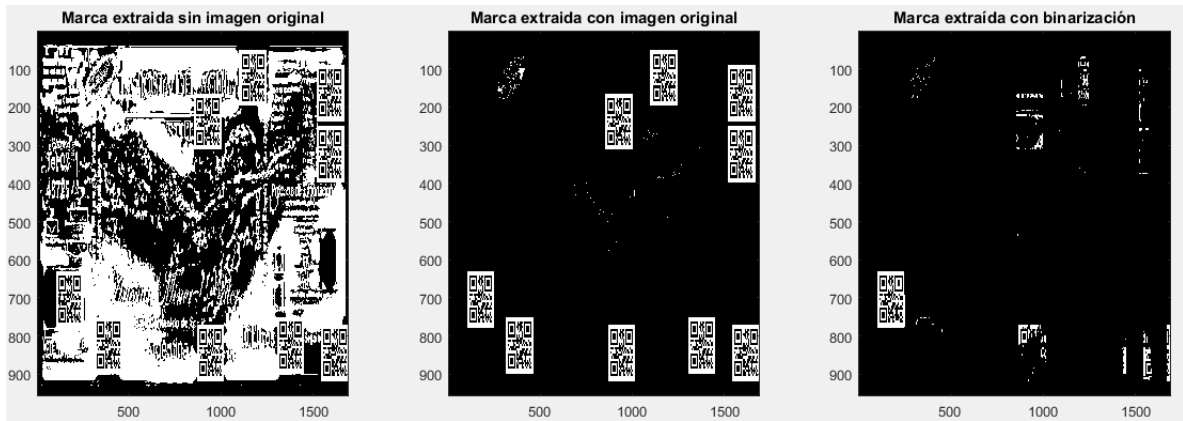


FIGURA 31.- MARCAS EXTRAÍDAS DE LA IMAGEN “LAPOSTA” DESPUES DE MODIFICAR LAS MARCAS INSERTADAS. A) SIN USAR LA IMAGEN ORIGINAL, B) USANDO IMAGEN ORIGINAL, C) EXTRAÍDA CON BINARIZACIÓN.

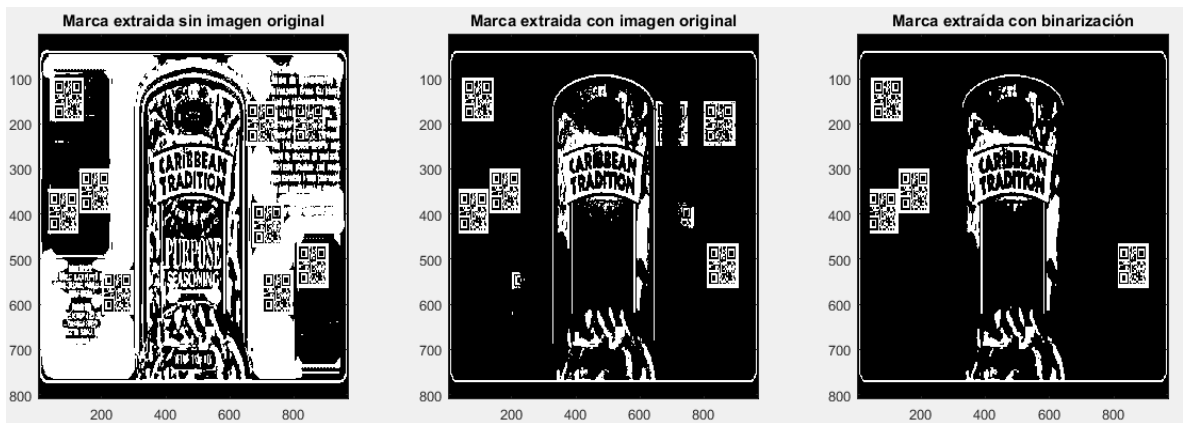


FIGURA 32.- MARCAS EXTRAÍDAS DE LA IMAGEN “SEASONING” DESPUES DE MODIFICAR LAS MARCAS INSERTADAS. A) SIN USAR LA IMAGEN ORIGINAL, B) USANDO IMAGEN ORIGINAL, C) EXTRAÍDA CON BINARIZACIÓN.

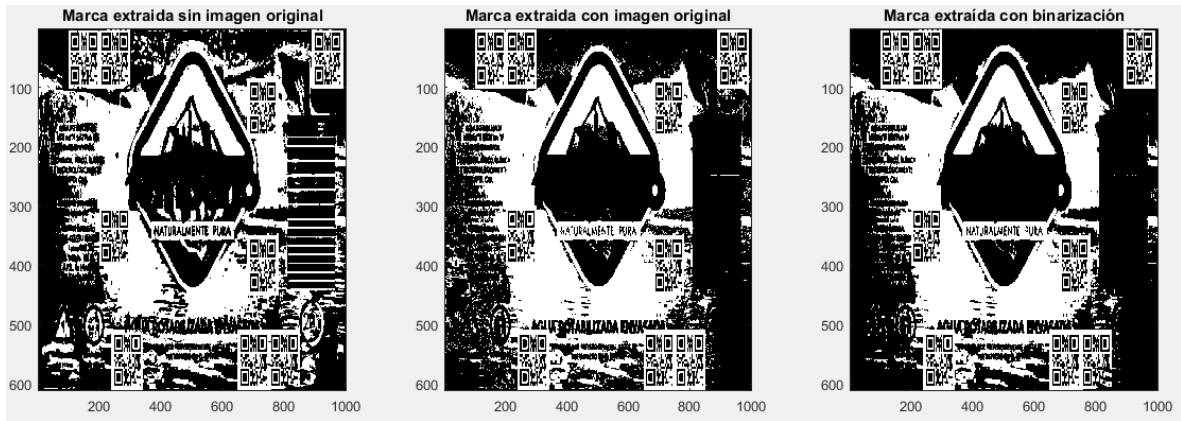


FIGURA 33.- MARCAS EXTRAÍDAS DE LA IMAGEN “TEMPANO” DESPUES DE MODIFICAR LAS MARCAS INSERTADAS. A) SIN USAR LA IMAGEN ORIGINAL, B) USANDO IMAGEN ORIGINAL, C) EXTRAÍDA CON BINARIZACIÓN.

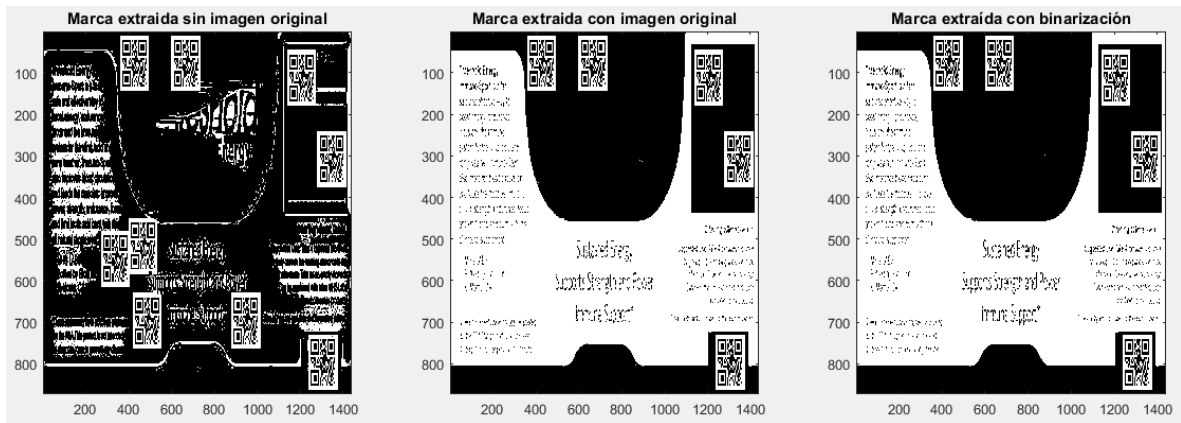


FIGURA 34.- MARCAS EXTRAÍDAS DE LA IMAGEN “THRESHOLD” DESPUES DE MODIFICAR LAS MARCAS INSERTADAS. A) SIN USAR LA IMAGEN ORIGINAL, B) USANDO IMAGEN ORIGINAL, C) EXTRAÍDA CON BINARIZACIÓN.

Imagen	Marcas detectables sin imagen original	Marcas detectables con imagen original	Marcas detectables con binarización
Adaro	8 de 9	8 de 9	8 de 9
Dive	7 de 9	9 de 9	7 de 9
Fiji	7 de 9	7 de 9	7 de 9
Grapefruit	8 de 9	5 de 9	3 de 9
Laposta	9 de 9	9 de 9	1 de 9
Seasoning	8 de 9	4 de 9	4 de 9
Tempano	8 de 9	9 de 9	8 de 9
Threshold	8 de 9	5 de 9	5 de 9

FIGURA 35.- TABLA DE CÓDIGOS QR DETECTABLES DE CADA IMAGEN CON CADA MÉTODO DE EXTRACCIÓN UTILIZADO, DESPUES DE MODIFICAR LAS MARCAS.

De los resultados obtenidos se pueden concluir las siguientes cosas:
 -Las marcas se conservan mejor cuando no hay un borde (un cambio drástico en la distribución de colores) atravesándolas o muy cerca.

-Aunque parezca contradictorio, las marcas se conservan mejor cuando no se usa la imagen original como referencia.

-Las marcas son más visibles en colores lisos que en zonas con muchos colores.

Hay que notar que de todas las imágenes, sólo a la imagen “laposta” se le incremento la delta de 10 a 20.

La segunda prueba es imprimir las imágenes modificadas en la prueba anterior, para después escanearlas e intentar extraer los códigos de esas fotos. La impresión se lleva a cabo en una impresora laser a color. Los resultados de las extracciones fueron los siguientes:

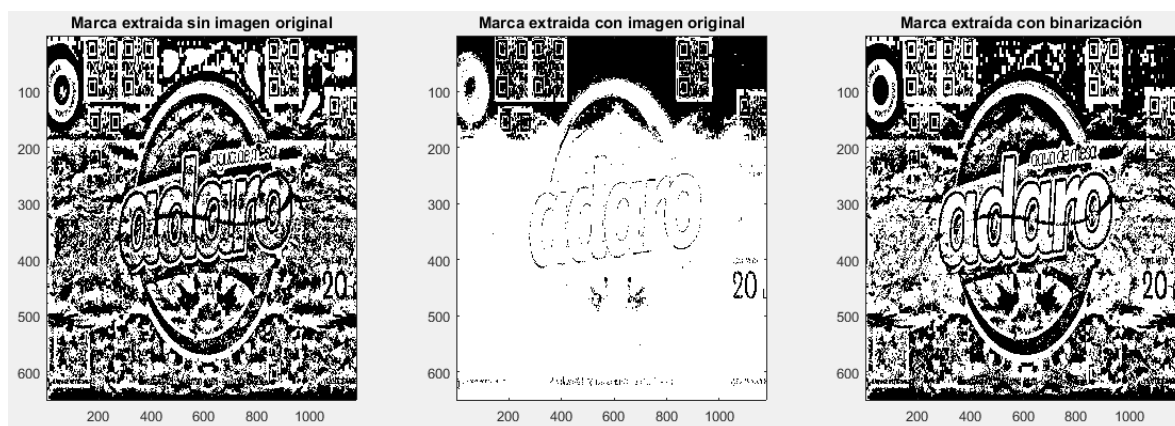


FIGURA 36.- MARCAS EXTRAÍDAS DE EL ESCANEADO DE LA IMAGEN “ADARO”. A) SIN USAR LA IMAGEN ORIGINAL, B) USANDO IMAGEN ORIGINAL, C) EXTRAÍDA CON BINARIZACIÓN.

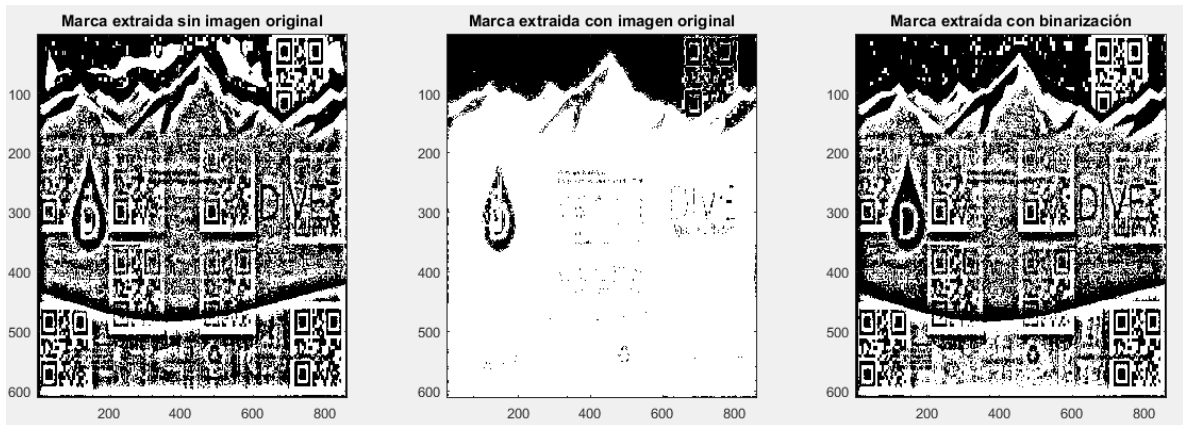


FIGURA 37.- MARCAS EXTRAÍDAS DE EL ESCANEO DE LA IMAGEN "DIVE". A) SIN USAR LA IMAGEN ORIGINAL, B) USANDO IMAGEN ORIGINAL, C) EXTRAÍDA CON BINARIZACIÓN.

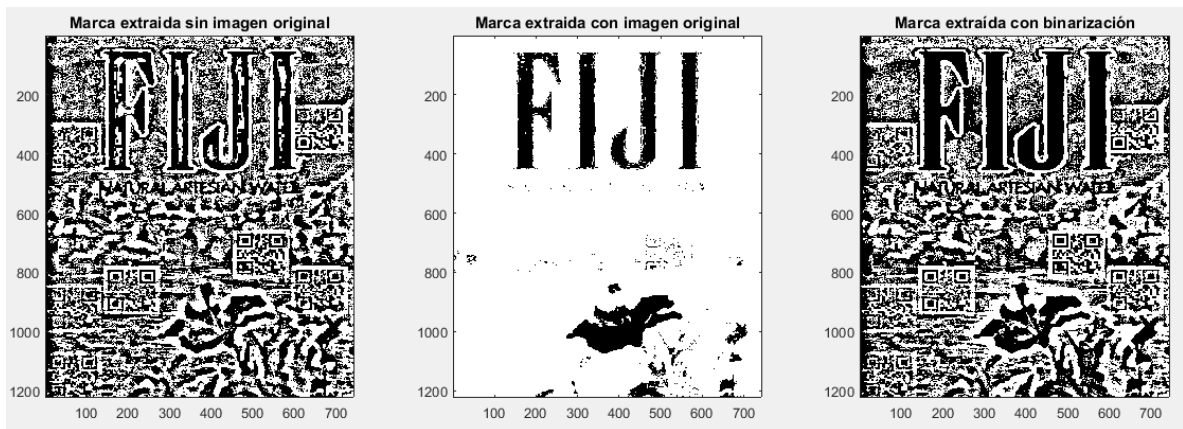


FIGURA 38.- MARCAS EXTRAÍDAS DE EL ESCANEO DE LA IMAGEN "FIJI". A) SIN USAR LA IMAGEN ORIGINAL, B) USANDO IMAGEN ORIGINAL, C) EXTRAÍDA CON BINARIZACIÓN.

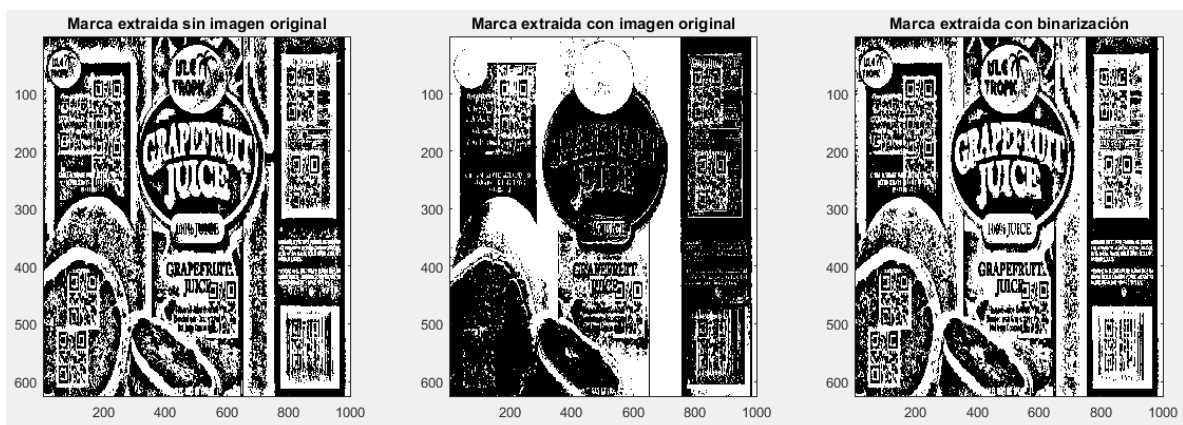


FIGURA 39.- MARCAS EXTRAÍDAS DE EL ESCANEO DE LA IMAGEN "GRAPEFRUIT". A) SIN USAR LA IMAGEN ORIGINAL, B) USANDO IMAGEN ORIGINAL, C) EXTRAÍDA CON BINARIZACIÓN.

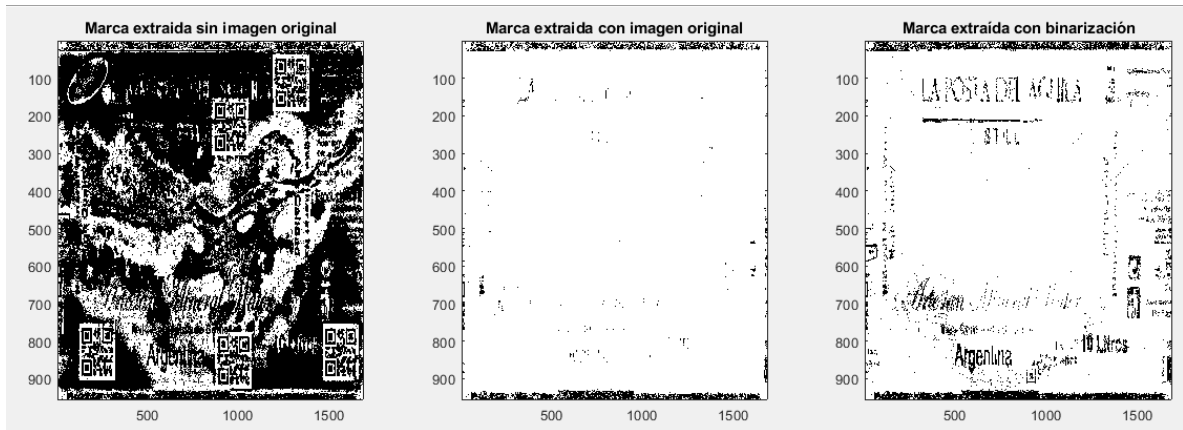


FIGURA 40.- MARCAS EXTRAÍDAS DE EL ESCANEADO DE LA IMAGEN “LAPOSTA”. A) SIN USAR LA IMAGEN ORIGINAL, B) USANDO IMAGEN ORIGINAL, C) EXTRAÍDA CON BINARIZACIÓN.

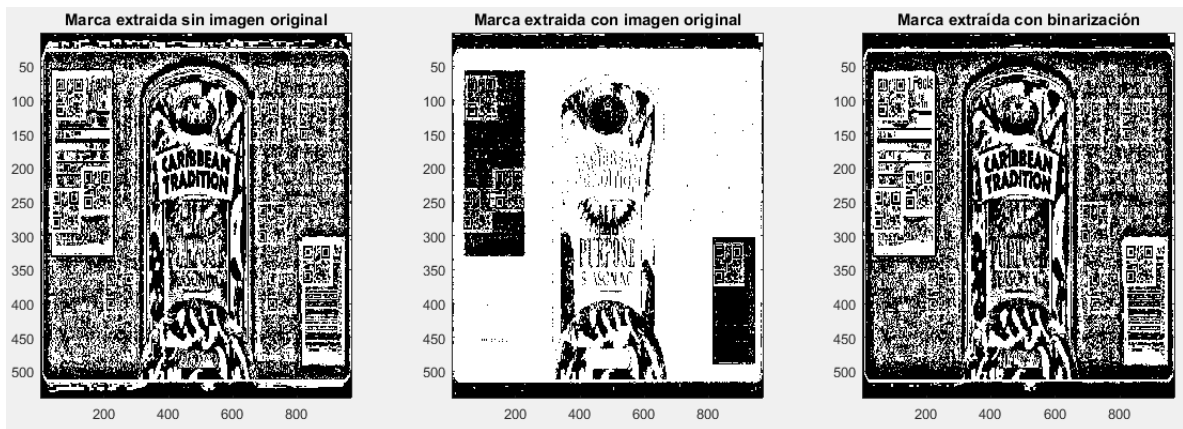


FIGURA 41.- MARCAS EXTRAÍDAS DE EL ESCANEADO DE LA IMAGEN “SEASONING”. A) SIN USAR LA IMAGEN ORIGINAL, B) USANDO IMAGEN ORIGINAL, C) EXTRAÍDA CON BINARIZACIÓN.

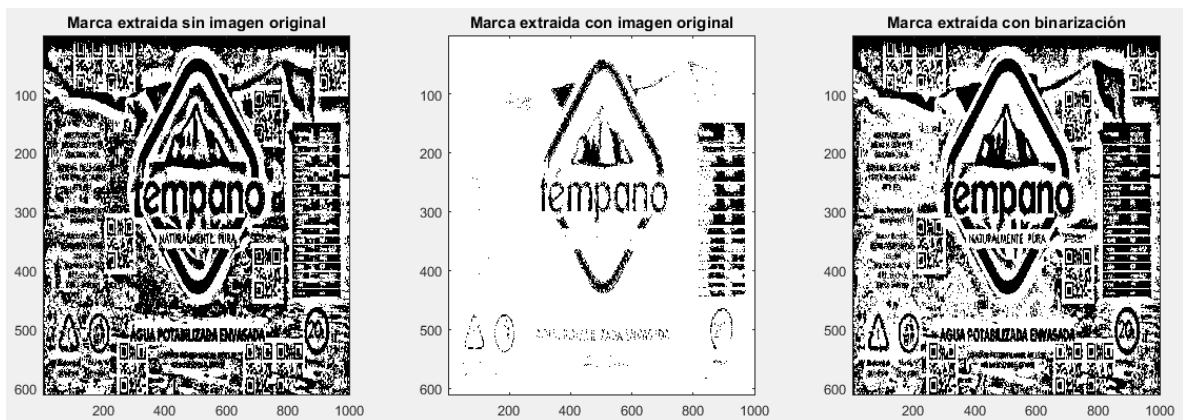


FIGURA 42.- MARCAS EXTRAÍDAS DE EL ESCANEADO DE LA IMAGEN “TEMPANO”. A) SIN USAR LA IMAGEN ORIGINAL, B) USANDO IMAGEN ORIGINAL, C) EXTRAÍDA CON BINARIZACIÓN.

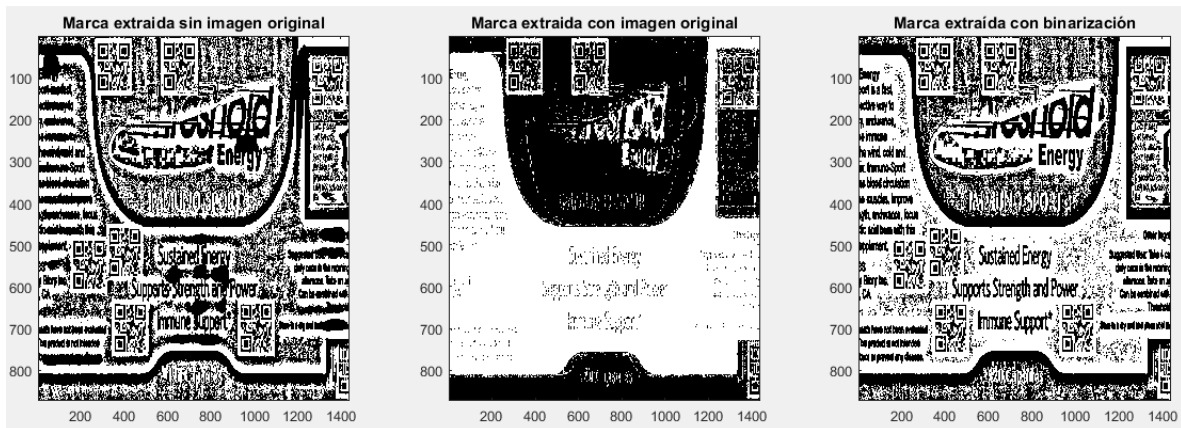


FIGURA 43.- MARCAS EXTRAÍDAS DE EL ESCANEADO DE LA IMAGEN “THRESHOLD”. A) SIN USAR LA IMAGEN ORIGINAL, B) USANDO IMAGEN ORIGINAL, C) EXTRAÍDA CON BINARIZACIÓN.

Imagen	Marcas detectables sin imagen original	Marcas detectables con imagen original	Marcas detectables con binarización
Adaro	3 de 9	0 de 9	3 de 9
Dive	1 de 9	0 de 9	1 de 9
Fiji	1 de 9	0 de 9	1 de 9
Grapefruit	1 de 9	0 de 9	2 de 9
Laposta	5 de 9	0 de 9	5 de 9
Seasoning	2 de 9	0 de 9	1 de 9
Tempano	3 de 9	0 de 9	4 de 9
Threshold	5 de 9	2 de 9	5 de 9

FIGURA 44.- TABLA DE CÓDIGOS QR DETECTABLES DE CADA IMAGEN CON CADA MÉTODO DE EXTRACCIÓN UTILIZADO, DESPUES DE ESCANEAR LAS MARCAS.

Para la extracción de estas marcas, se tuvo que agregar un filtro de agudización con el comando `imsharpen` de Matlab, el cuál incrementa el contraste de la imagen, facilitando así la extracción de las marcas. Sin éste, la extracción de las marcas hubiera sido muy inferior a los resultados obtenidos. La cantidad de marcas extraídas en todas las imágenes es inferior a la cantidad de la prueba anterior. Esto era de esperar, ya que los resultados de esta prueba dependen de la calidad de la impresora y de la calidad del escaneo. En un escenario ideal, la cantidad de marcas extraídas debería de ser igual, pero en este caso, se modificaron las imágenes una vez más, utilizando los resultados de esta prueba como referencia, como por ejemplo ver qué áreas de la imagen son menos vulnerables teniendo en cuenta las limitaciones de la

impresora, o en qué áreas las marcas se conservaron mejor. En este punto se buscó no utilizar el comando imsharp para extraer las imágenes. Por este motivo, se incrementó la delta de 10 a 20 en todas las imágenes. La imagen laposta ya tenía una delta = 20, y fue en la que se obtuvieron los mejores resultados, por lo que esa se quedó así.

Los resultados obtenidos fueron los siguientes:

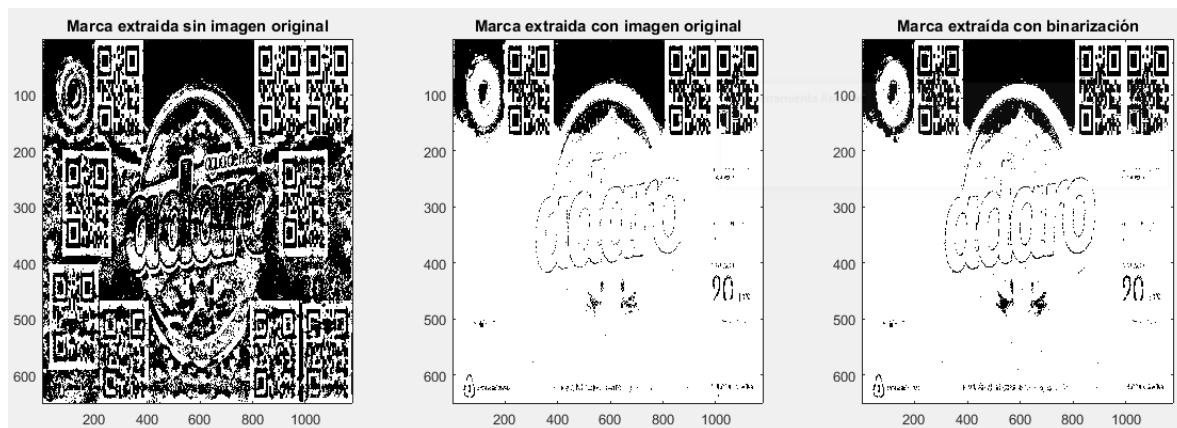


FIGURA 45.- MARCAS EXTRAÍDAS DEL SEGUNDO ESCANEO DE LA IMAGEN “ADARO”. A) SIN USAR LA IMAGEN ORIGINAL, B) USANDO IMAGEN ORIGINAL, C) EXTRAÍDA CON BINARIZACIÓN.

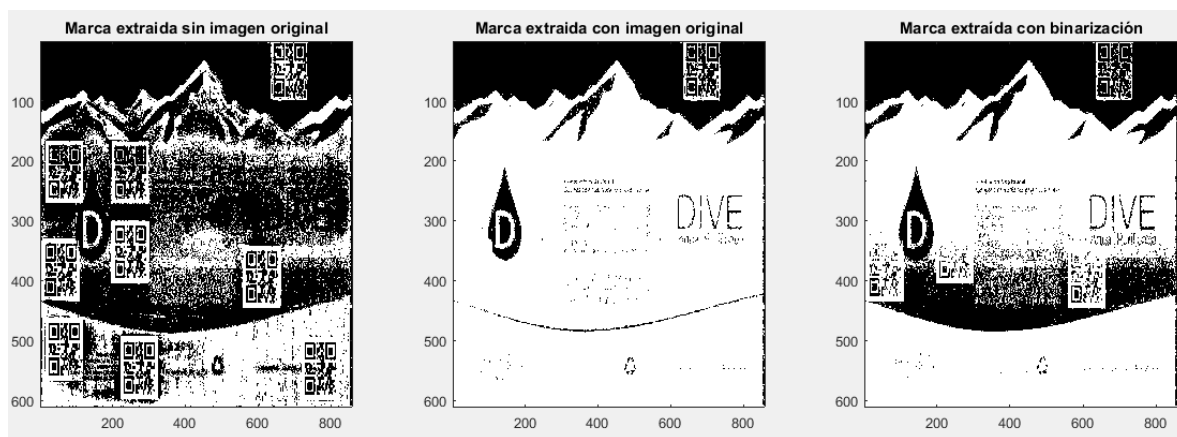


FIGURA 46.- MARCAS EXTRAÍDAS DEL SEGUNDO ESCANEO DE LA IMAGEN “DIVE”. A) SIN USAR LA IMAGEN ORIGINAL, B) USANDO IMAGEN ORIGINAL, C) EXTRAÍDA CON BINARIZACIÓN.

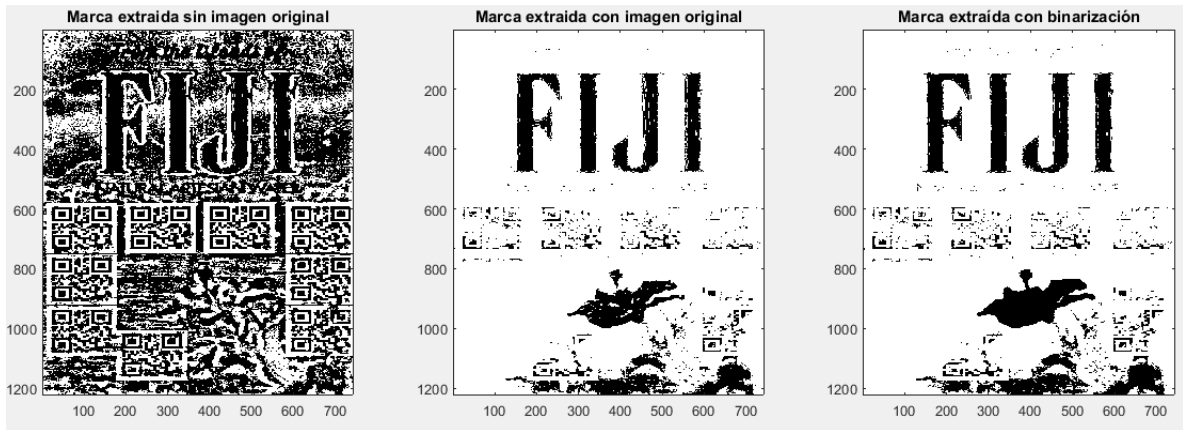


FIGURA 47.- MARCAS EXTRAÍDAS DEL SEGUNDO ESCANEADO DE LA IMAGEN "FIJI". A) SIN USAR LA IMAGEN ORIGINAL, B) USANDO IMAGEN ORIGINAL, C) EXTRAÍDA CON BINARIZACIÓN.

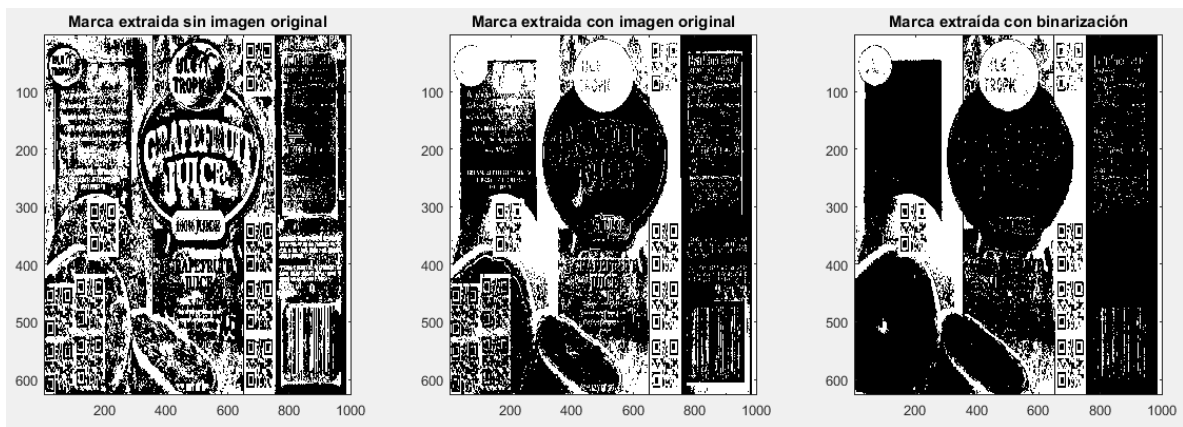


FIGURA 48.- MARCAS EXTRAÍDAS DEL SEGUNDO ESCANEADO DE LA IMAGEN "GRAPEFRUIT". A) SIN USAR LA IMAGEN ORIGINAL, B) USANDO IMAGEN ORIGINAL, C) EXTRAÍDA CON BINARIZACIÓN.

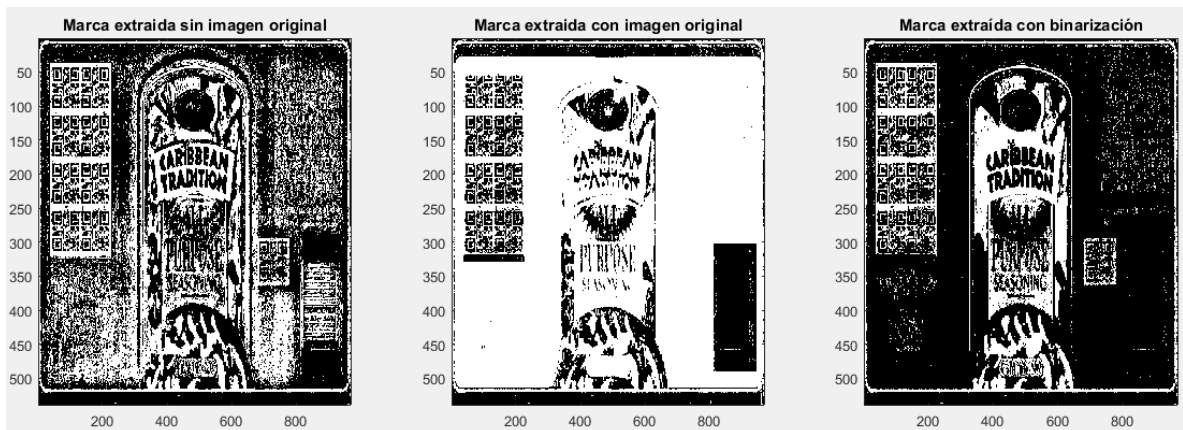


FIGURA 49.- MARCAS EXTRAÍDAS DEL SEGUNDO ESCANEADO DE LA IMAGEN "SEASONING". A) SIN USAR LA IMAGEN ORIGINAL, B) USANDO IMAGEN ORIGINAL, C) EXTRAÍDA CON BINARIZACIÓN.

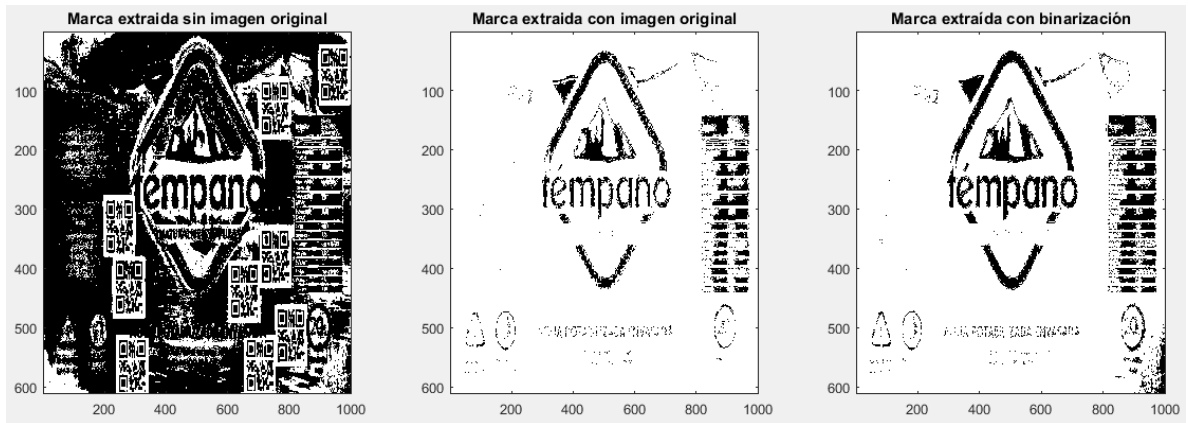


FIGURA 50.- MARCAS EXTRAÍDAS DEL SEGUNDO ESCANEADO DE LA IMAGEN “TEMPANO”. A) SIN USAR LA IMAGEN ORIGINAL, B) USANDO IMAGEN ORIGINAL, C) EXTRAÍDA CON BINARIZACIÓN.

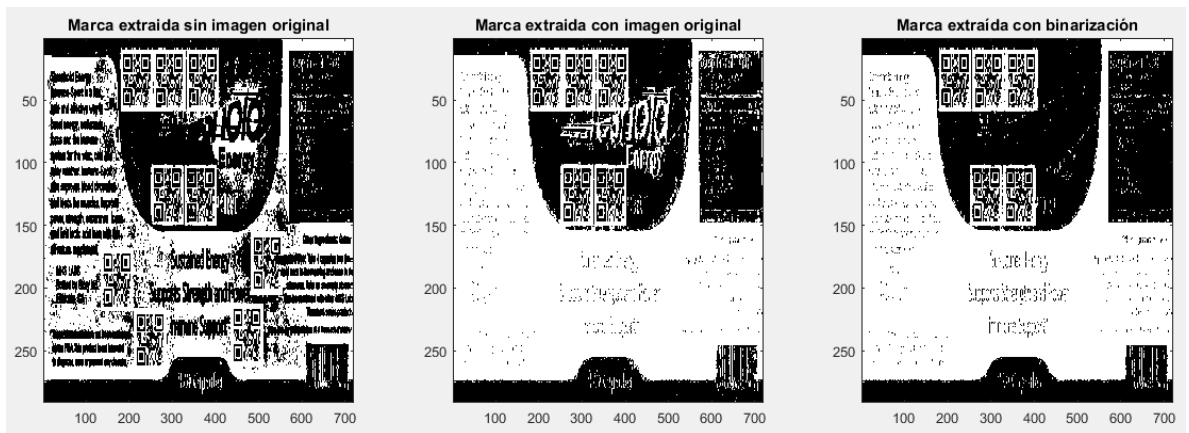


FIGURA 51.- MARCAS EXTRAÍDAS DEL SEGUNDO ESCANEADO DE LA IMAGEN “THRESHOLD”. A) SIN USAR LA IMAGEN ORIGINAL, B) USANDO IMAGEN ORIGINAL, C) EXTRAÍDA CON BINARIZACIÓN.

Imagen	Marcas detectables sin imagen original	Marcas detectables con imagen original	Marcas detectables con binarización
Adaro	4 de 9	2 de 9	3 de 9
Dive	5 de 9	1 de 9	0 de 9
Fiji	7 de 9	0 de 9	0 de 9
Grapefruit	6 de 9	4 de 9	4 de 9
Seasoning	7 de 9	4 de 9	2 de 9
Tempano	9 de 9	0 de 9	0 de 9
Threshold	9 de 9	5 de 9	5 de 9

FIGURA 52.- TABLA DE CÓDIGOS QR DETECTABLES DE CADA IMAGEN CON CADA MÉTODO DE EXTRACCIÓN UTILIZADO, DESPUES DE CAMBIAR DELTA DE 10 A 20, Y ESCANEARLAS POR SEGUNDA OCASIÓN.

Tal como se estableció anteriormente, en esta ocasión no se utilizó el comando `imsharp` para la extracción, y aún así los resultados fueron mejores que en la prueba anterior.

De todas las pruebas se puede concluir que el mejor método de extracción es **sin la imagen original**.

4.3 VENTAJAS Y DESVENTAJAS

A diferencia del código de barras tradicional, éste es un método de identificación de productos que no coloca una marca visible en las etiquetas de los productos, por lo que esta puede ser colocada en varias partes de la etiqueta, haciendo así más rápida su detección y agilizando procesos como el de cobro en un cajero de súper mercado.

Por ejemplo, si el producto en cuestión es una caja con forma cuadrada, al estar las marcas distribuidas en toda la imagen de la etiqueta, simplemente con pasar el producto una vez a través del detector bastaría, ahorrando el tiempo que se perdería con la persona de la caja buscando el código de barras y apuntando éste por el detector.

Este método tiene la desventaja de que la intensidad de las marcas en ciertas imágenes puede ser incómoda de ver para algunas personas debido a que se vuelven fácilmente distinguibles (figura 51), y para el equipo de marketing de una empresa esto podría disuadirlos de utilizar este método.

No obstante, esto puede ser corregido. El entorno en el que se trabajó en esta tesis fue con una impresora laser casera, y un escáner, también casero, es decir, la calidad del equipo utilizado no es con el que se trabaja en estos ámbitos. Una vez conociendo a detalle esta información se puede reducir la intensidad de la marca delta, el tamaño, y la posición en cada imagen según la necesidad, como se ve con la imagen “fiji” en la figura 51.

También existe la opción de utilizar la otra componente de crominancia **I**, que contiene información de los cambios en el rango de naranja a azul, a diferencia de **Q** que contiene información de los cambios en el rango morado a verde. Estos cambios varían con cada imagen, por lo que en unas imágenes esto funcionaría mejor que en otras.

Otra opción es cambiar el espacio de color con el que se trabaja. Se podría trabajar en el espacio **YCbCr**, que es similar a **YIQ**, con la diferencia de que **Cr** es la diferencia del color rojo menos **Y**, y **Cb** indica la diferencia del color azul menos **Y**.

También existe el espacio **CIE*Lab**, donde **a** cuyo rango va de verde a morado, y **b** cuyo rango de azul a amarillo.

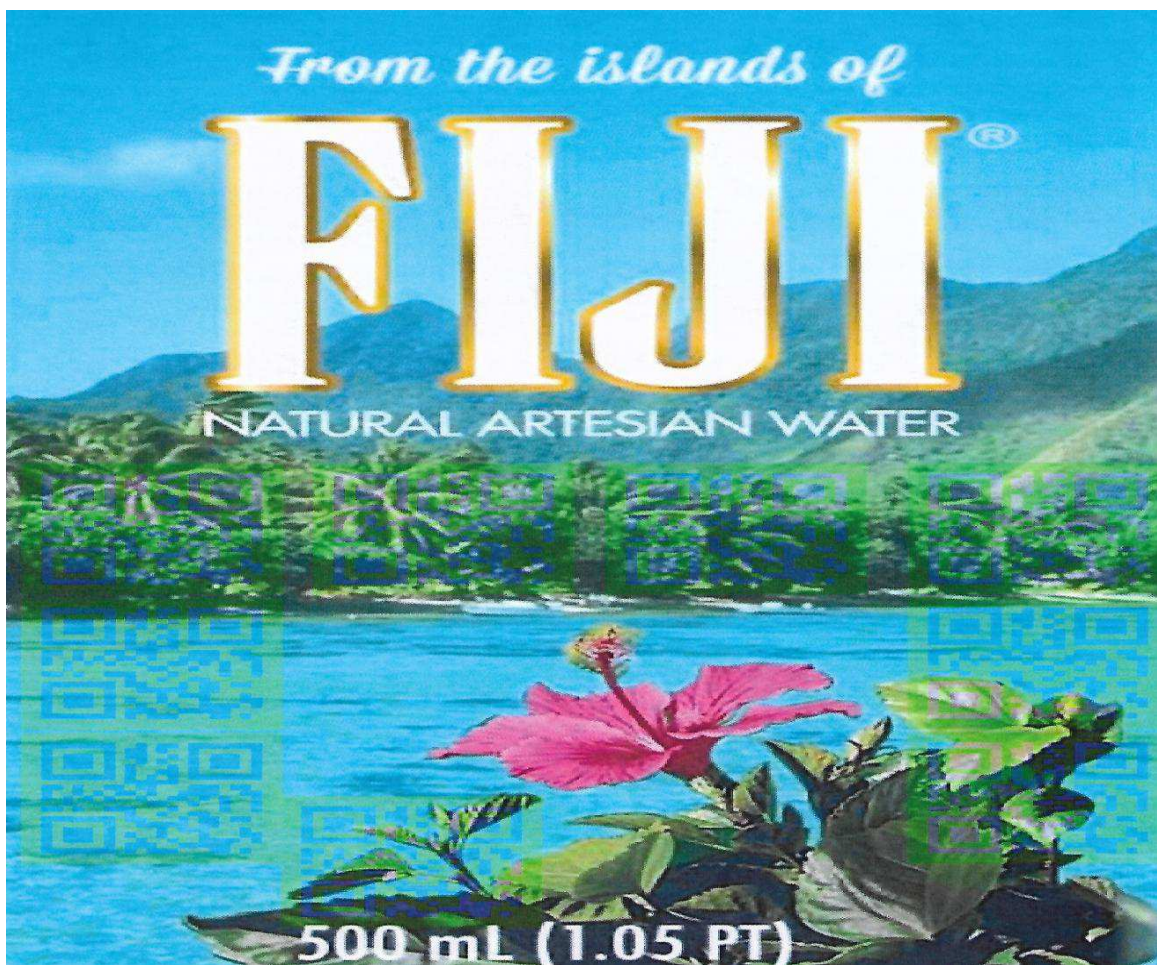


FIGURA 53.- ESCANEOS DE LAS IMÁGENES "DIVE" (ARRIBA) Y "FIJI" (ABAJO) CON DELTA=20.

CAÍTULO 5

CONCLUSIONES Y TRABAJO A FUTURO

Con base en los objetivos establecidos, se puede concluir que el desarrollo del proyecto se cumplió satisfactoriamente, ya que se logró el objetivo principal de simular el proceso en Matlab, y extraer los códigos QR de las impresiones. Esto directamente implica que se cumplieron los objetivos secundarios, que son: crear el código en Matlab que inserte el código en una etiqueta, y después, crear otro código en Matlab que extraiga el código de la fotografía.

En la figura 16 se puede ver que en las imágenes menos lisas, la alteración que implica insertar una marca con más intensidad afecta menos la calidad de la imagen, ya que tanto el SSIM como el PSNR no se reducen tan drásticamente como en las imágenes más lisas.

El método utilizado tuvo resultados aceptables en todas las imágenes, pero en lo que se refiere a la visibilidad de las marcas, estos fueron más variados. Cabe notar que, incluso si las marcas eran bastante evidentes en las imágenes en su formato digital, en las impresiones están eran menos visibles. Esto no imposibilitó que fueran detectables, pero se tuvo que incrementar la intensidad (δ) de las marcas.

La ubicación de las marcas dentro de la imagen también tuvo un papel importante, ya que, si bien en áreas menos lisas de cada imagen la marca era menos notoria a simple vista, esto también dificulta su detección, aunque en el caso de la imagen "laposta", a pesar de que las marcas apenas y son visibles a simple vista, estas fueron detectadas satisfactoriamente.

El tamaño también fue muy importante en la detección. Debido a las limitaciones tanto de la impresora como del escáner, las marcas más pequeñas eran más difíciles de detectar, pero si la marca era muy grande, se incrementa la posibilidad de que un cambio drástico en los colores en esa parte de la imagen la atravesara, la altere y la vuelva indetectable, por lo que se ubicó en lugares lisos, o bien, texturas que no tengan cambios drásticos de color.

A futuro, este método efectivamente puede ser usado como un reemplazo al código de barras tradicional, pero antes debe ser pulido utilizando como referencia a las características de la industria, recordando que en este trabajo se utilizó una impresora casera, un escáner casero, e imágenes de dimensiones que pudieran no ser del tamaño o proporción de dimensiones de la etiqueta de un producto (por ejemplo, la etiqueta de una botella sería muy larga y no muy alta). Se tendría que usar una impresora comúnmente utilizada en la elaboración de etiquetas, y se tendría que plantear, o incluso elaborar, un detector especializado para este método. Ya que la detección se realizó de manera exitosa en esta tesis, para efectos de su utilización

como reemplazo del código de barras, se tendría que considerar la velocidad de procesamiento, de tal forma que el simple hecho de pasar el producto por el detector, éste sea identificado, y esto de manera más rápida y eficiente que el código de barras tradicional. Estos factores pudieran corregir los problemas que surgieron durante la elaboración de esta tesis, como el tamaño e intensidad de las marcas, pero a su vez podrían surgir otros que deberán ser corregidos.

Una vez tomando en consideración los factores previamente mencionados, es posible que se tengan que hacer cambios más importantes al método, como por ejemplo utilizar otro espacio de color, o incluso múltiples espacios de color, o esconder las marcas en varias componentes en un mismo espacio de color.

REFERENCIAS BIBLIOGRÁFICAS

- [1].- E. Martínez Villacampa, E. Sayrol Clos, “*Introducción al Watermarking*”
- [2].- JS. Pan, HC. Huang, L. C.Jain, “*Intelligent Watermarking Techniques*” p.15
- [3].- JS. Pan, HC. Huang, L. C.Jain, “*Intelligent Watermarking Techniques*” p.16
- [4].- JS. Pan, HC. Huang, , L. C.Jain, “*Intelligent Watermarking Techniques*” p.20
- [5].- Ley de propiedad intelectual, artículo 31.
- [6].- D. Milano, “*Content Control: Digital Watermarking and Fingerprinting*,” White Paper, Rhozet, a business unit of Harmonic Inc.,
<http://www.rhozet.com/whitepapers/Fingerprinting—Watermarking.pdf>,
Último Acceso 18 ene. 2019.
- [7].- C. Harris, M. Stevens, “*A combined corner and edge detector*,” in Proc. of 4th Alvey Vision Conference, pp. 153-158, 1988. Article (CrossRef Link).
- [8].- A. Joly, O. Buisson, C. Frelicot, “*Content-based copy retrieval using distortion-based probabilistic similarity search*” IEEE Transactions on Multimedia, vol. 9 (2), pp.293–306, 2007. Article (CrossRef Link).
- [9].- J. Law-To, O. Buisson, V. Gouet-Brunet, N. Boujemaa, “*Robust voting algorithm based on labels of behavior for video copy detection*,” in Proc. of ACM Multimedia, pp. 835-844, 2006. Article (CrossRef Link).
- [10].- S. Lee and C. Yoo, “*Robust video fingerprinting for content-based video identification*,” IEEE Transactions on Circuits and Systems for Video Technology, vol. 18, no. 7, pp. 983–988, July, 2008. Article (CrossRef Link).
- [11].- Cuaderno de notas del observatorio, “*Esteganografía el arte de ocultar información*”, Instituto Nacional de Tecnologías de la Comunicación (INTECO) , España
<http://www.egov.ufsc.br/portal/sites/default/files/esteganografia1.pdf>
- [12].- Pymempresario.com. “*La importancia de la Identificación de Productos*”.
Última modificación 15 ago. 2011
<https://www.pymempresario.com/2011/08/la-importancia-de-la-identificacion-de-productos/>
- [13].- ACPservicios.com. “*Barcode*”
Última modificación 18 ene. 2019
<https://www.acp-servicios.es/index.php/barcode>
- [14].- TheEconomist.com. “*Why QR Codes are on the rise*”
Última modificación 2 nov. 2017
<https://www.economist.com/the-economist-explains/2017/11/02/why-qr-codes-are-on-the-rise>

- [15].- FH. Wang, JS. Pan, L.C. Jain, (2009) "*Digital Watermarking Techniques. In: Innovations in Digital Watermarking Techniques. Studies in Computational Intelligence*", vol 232. Springer, Berlin, Heidelberg p.14-15
- [16].- Digimarc.com. "*Easy checkout*"
Última modificación 18 ene. 2019
<https://www.digimarc.com/solutions/easy-checkout>
- [17].- S.A.M.Gilani , I.Kostopoulos and A.N.Skodras, "*COLOR IMAGE-ADAPTIVE WATERMARKING*", Electronics Laboratory, University of Patras, Greece
- [18].- B. Gunjal, S. Mali, "*Comparative Performance Analysis of DWT-SVD Based Color Image Watermarking Technique in YUV, RGB and YIQ Color Spaces*", International Journal of Computer Theory and Engineering, Vol. 3, No. 6, December 2011
- [19].- H. Jindal, D. Mandal, "*Performance Analysis of Image Compression Using Discrete Cosine Transform with Various Discrete Wavelet Transform*", Electronics and Communication Department, Institute of Engineering & Technology Bhaddal, Ropar, India
- [20].- Universidad de Haifa. "*Information on the YIQ color space*"
Última modificación 21 mar. 2004
<http://cs.haifa.ac.il/hagit/courses/ist/Lectures/Demos/ColorApplet/me/infoyiq.html>
- [21].- P. Pablo Becerril Calderón, "*Sistema de detección de productos cilíndricos con descriptores de imágenes basados en puntos característicos*", UNIVERSIDAD NACIONAL AUTÓNOMA DE MÉXICO
- [22].- P. Kaulgud, S. Prahlad, R. Satapathy, (2015). IEEE Region 10 "*Symposium. Smart Shelfie - Internet of Shelves: For Higher On-Shelf Availability*".
- [23].-Digimarc canal de Youtube. "*Digimarc® Barcode Demonstration*"
Última modificación 1 may. 2014
https://www.youtube.com/watch?time_continue=1&v=XoJFXEUI-s8&ab_channel=Digimarc

REFERENCIAS DE LAS IMÁGENES

- Figura 3 <http://www.rhozet.com/whitepapers/Fingerprinting—Watermarking.pdf>
Figura 6 <https://etiqueta10.blogspot.com/2013/07/codigos-qr.html>
Figura 7 <http://www.curtis-computing.com/sobek/security.php?ID=17>
Figura 8 <https://pt.slideshare.net/vidhi00/vidhis-ppt?related=1>

APÉNDICE

Código de inserción

Código de inserción utilizado para insertar la marca en la componente Q de la imagen A.

```
A1= rgb2ntsc(A);
A2 = A1(:,:,3);

for i=2:(m-1)
    for j=2:(n-1)

mu(i,j) = (A2(i-1,j-1)/8) + (A2(i,j-1)/8) + (A2(i+1,j-1)/8) + (A2(i-1,j)/8) + (A2(i+1,j)/8) + (A2(i-1,j+1)/8) + (A2(i,j+1)/8) + (A2(i+1,j+1)/8);

        end
    end

for k=1:o
    for l=1:p

        if marca(k,l) == 1
            Amarcada(k+x-1,l+y-1) = mu1(k+x-1,l+y-1) - delta;
        else
            Amarcada(k+x-1,l+y-1) = mu1(k+x-1,l+y-1) + delta;
        end

        end
    end

for i=1:m
    for j=1:n

        if (i>=x & i<(x+o) & j>=y & j<(y+p))
            AmarcadaFinal(i,j) = Amarcada(i,j);
        else
            AmarcadaFinal(i,j) = A2(i,j);
        end

        end
    end
end
```

Código de extracción

Código de extracción utilizado para extraer la marca de la componente Q de la imagen A.

```
for i=2:(m-1)
    for j=2:(n-1)

        mu(i,j) = ((Amarcada(i-1,j-1)/8) + (Amarcada(i,j-1)/8) + (Amarcada(i+1,j-1)/8) + (Amarcada(i-1,j)/8) + (Amarcada(i+1,j)/8) + (Amarcada(i-1,j+1)/8) + (Amarcada(i,j+1)/8) + (Amarcada(i+1,j+1)/8));

    end
end

for i=1:m
    for j=1:n

        if A2(i,j) < mu(i,j)
            extraida(i,j) = 1;
        else
            extraida(i,j) = 0;
        end
    end
end
```