



Universidad de las Ciencias Informáticas  
AIRI, Artificial Intelligence: Research and Innovation, Facultad 3

## **Cálculo de saturación de grasas en imágenes de jamón de la línea productiva de la empresa Bravo SA**

Trabajo de diploma para optar por el título de Ingeniero en Ciencias Informáticas

**Autores:**

Adolfo Díaz Quiros  
Celso Raúl Sánchez Viera

**Tutores:**

Ing. Michel Álvarez Cancio  
Ing. Damarys Martínez Águila

La Habana. Junio, 2017

## DECLARACIÓN DE AUTORÍA

Declaramos ser autores de la presente tesis y reconocemos a la Universidad de las Ciencias Informáticas los derechos patrimoniales sobre esta, con carácter exclusivo.

Para que así conste firmamos la presente a los \_\_\_\_\_ días del mes de \_\_\_\_\_ del año \_\_\_\_\_.

---

Adolfo Díaz Quiros

---

Celso Raúl Sánchez Viera

---

Ing. Michel Álvarez Cancio

---

Ing. Damarys Martínez Águila

## **Dedicatoria**

Todo mi trabajo y esfuerzo se lo dedico a mi hermano Alfredo, quien más que nada desearía hoy que estuviese a mi lado, y quien estoy seguro que me mira orgulloso desde allá arriba de lo que he sido capaz de lograr.

Celso Raúl Sánchez Viera

Quiero dedicar la presente investigación a mi hermana Adanay de La Rúa, quien se encuentra cursando estudios en la enseñanza media. Sé que todo mi esfuerzo le va a servir de ejemplo para superarse en su vida profesional. También quiero dedicarle este trabajo a mi padre Adolfo Díaz, quien no se encuentra físicamente, pero llevo en mi corazón y estoy seguro que estaría muy orgulloso de mis resultados.

Adolfo Díaz Quiros

## **Agradecimientos**

Agradecer antes que nada a mi compañero de tesis Adolfo y a mis tutores que han luchado día a día junto conmigo para que esto hoy fuese posible, a mi pareja Rachel por apoyarme y darme su amor incondicional, a mis hermanas Talía y Clara por ser siempre el objeto de mi felicidad, a mi madre por ser siempre mi todo por su ejemplo y su amor, a mis padres Alfredo y Martin que han sido fuente de invaluable consejos y guía, a mis abuelos que han dedicado todo de sí para que pueda ser quien soy hoy, a mi tío Lester que ha marcado las pautas por las que me guían para ser un buen hombre, agradecer a mi hermano Alfredo por haber sido el hombre más importante de mi vida y a mi hermano Sergio que siempre ha estado ahí para mí y espero tener la dicha de que siempre sea así, y a mis más queridos amigos Roger, Daymel, Rafael, David, Marcelo, Rubén, Alejandro el flaco, Félix, Jose Carlos, Luis Javier ,Raúl, en general a todos, todos los que he conocido en estos 5 años que han hecho de esto una experiencia inolvidable llena de alegrías. Así mismo a todos los profesores que me han enseñado y me han aguantado durante la carrera. A todos, muchas gracias.

Celso Raúl Sánchez Viera

Quiero comenzar agradeciendo a mi nakama Celso y a mi tutor Michel, por el esfuerzo y dedicación que han tenido para juntos lograr el desarrollo de esta investigación. A mi novia Claudia Guerra por apoyarme, aconsejarme minuto a minuto y darme su amor, por ser la persona más especial que he conocido en el mundo mundial, por ser cariñosa y atenta. A mi familia por estar presente en todo momento, por complacerme en todo antojo, por inculcarme valores y educarme. Hablo de: mi abuela Felicia por confiar, creo que no alcanzaría esta tesis para agradecerle y decirle cuanto siento por ella, mi madre Ana Luisa por ser también mi padre, a mis tías Idalmis, Reglita, Danay, a mis primas Yenisleydis, Yennly, Yaritza, Indira, Indiana, Yenisey, a mi tío Daniel. Quiero agradecer a Lachy Papi por tener un corazón tan grande, sin él pudiese ser que no me estuviese graduando, tengo una deuda eterna con él, gracias por hacer cómoda mi estancia en la UCI. A mi padrastro Ariel por haber sido mi motor impulsor, por inculcarme informática desde edades tempranas, por enseñarme que el principal obstáculo en la vida es nuestra propia mente. A el resto de mis nakamas del edificio 58, Rafael, Marcelo, Alejandro, Luis Javier, David, Rubén, Félix y Jose Carlos. A mis amigos inseparables Héctor, Luis Yasiel, Heriberto, Fernando, entre otros. Al colectivo de profesores de la facultad 3, con los que he compartido mis mejores y peores experiencias. A la Universidad de las Ciencias Informáticas y a la Revolución por brindarme la posibilidad de obtener el título de Ingeniero. A todos muchas salud y bendiciones, GRACIAS.

Adolfo Díaz Quiros

## **Resumen**

En la actualidad el jamón es uno de los alimentos más consumidos por la población en nuestro país, debido a su aporte en calorías y proteínas con alto valor biológico. Es una buena opción para aquellas personas que desean llevar una dieta saludable. Sin embargo esta carne ha tenido ciertos patrones que le dan una mala imagen asociado a enfermedades cardiovasculares y parásitos. Para eliminar estos defectos de calidad, la empresa Bravo S.A. se encuentra perfeccionando sus procesos productivos. Principalmente mejorando la calidad de la materia prima utilizada para la confección de sus productos fundamentales. Para ello necesitan una forma de estimar la saturación de grasas durante la elaboración. Por tal motivo la presente investigación se enmarca en cálculo de saturación de grasas en imágenes de jamón de la línea productiva de la empresa Bravo SA. En las mismas se han instalado un cumulo de cámaras digitales, las cuales adquieren imágenes de la materia prima. Luego estas serían analizadas por el departamento de calidad de la empresa mediante el uso de la solución propuesta, a fin de mejorar la calidad de su producto final. La solución se basa en el preprocesamiento de las imágenes digitales, para luego aplicarles un método de segmentación apropiado que logre detectar la saturación de grasas sin desechar la saturación de cárnicos. Posteriormente se da paso a la cuantificación de los resultados posibilitando a la empresa una mejor estimación.

### **Palabras claves:**

Saturación de grasas, segmentación, cuantificación, preprocesamiento, imágenes digitales.

## **Abstract**

At present ham is one of the foods most consumed by the population in the country, due to its contribution in calories and proteins with biological value. It is a good choice for those people wanting to eat a healthy diet. However this meat has had certain patterns that give a bad image associated with cardiovascular diseases and parasites. To eliminate these quality defects Bravo SA company is perfecting its production processes, mainly by enhancing the quality of the raw material used to make its fundamental products. In order to do it is needed a way to estimate the fat saturation during the making process. Therefore the current research is framed in calculating the ham fat saturation in Bravo S.A production lines. In which were installed a set of digital cameras, which acquire images of the raw material. , for later to be analyzed by the quality department of the company by using the proposed solution in order to improve the final product quality. The solution is based in the digital image processing, which is after applied a proper segmentation method which achieves to detect the fat saturation without ignoring the meat saturation. Later the results are quantified making possible for the business a better estimation.

**Keywords:** Fat saturation, segmentation, quantification, preprocessing.

# 1 Índice

Introducción .....	1
1 Capítulo 1: Fundamentación Teórica .....	6
1.1 Imagen digital .....	6
1.1.1 Imágenes a Color .....	6
1.1.2 Imágenes de jamón de la empresa Bravo SA. ....	6
1.2 Procesamiento de Imágenes .....	7
1.3 Preprocesamiento de imágenes .....	8
1.3.1 Espacios de color .....	8
1.3.2 Mejora del contraste .....	9
1.3.3 Detección de bordes .....	10
1.4 Segmentación.....	11
1.4.1 El histograma de una imagen .....	12
1.4.2 Umbralización.....	13
1.4.3 Métodos adaptativos.....	14
1.4.3.1 Método de Wellner 1993.....	14
1.4.3.2 Método de Phansalkar 2010 .....	15
1.4.3.3 Método de Sauvola 2000.....	16
1.4.3.4 Método White's 1983 .....	17
1.4.4 Métodos Globales.....	17
1.4.4.1 Metodo de Otsu 1979 .....	18
1.4.4.2 Método de Huang (Shannon) 1995.....	19
1.4.4.3 Método de Tsai 1995 .....	20
1.4.4.4 Método de Khashman 2007 .....	21
1.4.4.5 Método de Tsallis 1988.....	23
1.5 Métricas utilizadas para la validación.....	24
1.5.1 Medidas del error.....	25

1.5.1.1	Medidas estadísticas de error.....	25
1.5.1.2	Medidas de localización.....	28
1.5.1.3	Métrica Hausdorff .....	29
1.6	Herramientas utilizadas durante el desarrollo .....	30
1.6.1	Entorno de desarrollo .....	30
1.6.2	Lenguaje de programación .....	30
1.6.3	Herramienta para el modelado.....	30
1.6.4	Metodología de desarrollo .....	31
1.7	Conclusiones del capítulo .....	32
2	Capítulo 2: Propuesta de solución .....	33
2.1	Descripción de la solución .....	33
2.1.1	Pseudocódigo de la clase gestora principal .....	35
2.2	Selección de los métodos de preprocesado.....	36
2.3	Selección de los métodos de umbralización .....	39
2.4	Fase de Planeación .....	39
2.4.1	Requisitos del sistema.....	40
2.4.1.1	Requisitos no funcionales.....	40
2.4.2	Historias de Usuarios.....	40
2.4.3	Planificación de la Entrega .....	42
2.4.4	Plan de iteraciones .....	43
2.4.5	Plan de entrega .....	44
2.5	Diseño .....	44
2.5.1	Tarjetas CRC.....	44
2.5.2	Estándares de codificación .....	45
2.6	Conclusiones del capítulo .....	45
3	Capítulo 3: Implementación y Pruebas.....	47
3.1	Fase de implementación.....	47

3.1.1	Iteración 1.....	47
3.1.2	Iteración 2.....	49
3.1.3	Iteración 3.....	51
3.2	Interfaz de usuario .....	53
3.3	Resultados de aplicar el algoritmo .....	54
3.4	Validación del sistema .....	55
3.5	Pruebas .....	56
3.5.1	Pruebas Unitarias .....	57
3.5.2	Pruebas de caja blanca .....	57
3.5.3	Pruebas de aceptación .....	59
3.6	Conclusiones del capítulo .....	63
	Conclusiones .....	64
	Recomendaciones .....	65
	Referencias bibliográficas .....	66
	Anexos.....	70
	Anexo #1 Historias de usuarios .....	70
	Anexo #2 Tarjetas CRC.....	76

## Introducción

El valor nutritivo del jamón lo señala como uno de los alimentos más completos y de gran importancia en la dieta humana, debido en gran parte a su aporte en proteínas de alto valor biológico. Sin embargo, durante muchos años esta carne ha tenido ciertos patrones que le dan una mala imagen (contenido graso, con alto contenido en calorías y colesterol, asociado a enfermedades cardiovasculares y parásitos). Ante esta situación, los productores de carne y la industria cárnica en los últimos años han buscado la manera de obtener productos que minimicen los riesgos para el consumidor. En principio mediante la modificación de los procesos tecnológicos, para mejorar las características nutritivas de los productos cárnicos. A fin de mejorar su imagen y adaptarse a las necesidades del mercado (mayor contenido de proteínas y menos contenido graso). Por todo ello, la detección rápida de este tipo de defectos es fundamental para determinar el control del proceso para definir la calidad del producto (1).

En la Universidad de Extremadura en España, los miembros del área de ``Tecnología de los Alimentos`` llevan varios años estudiando factores que influyen en la elaboración de productos derivados del cerdo y en la calidad final de éstos. Una de las inquietudes de estos estudios es la de buscar metodologías alternativas y no destructivas que pudiesen caracterizar dichos productos (2).

A raíz de los problemas antes planteados, realizó un software comercial que permitía calcular el nivel de grasa intramuscular de carne de jamones de cerdo Ibérico. Utilizando un programa comercial llamado Q Win de Leica para el análisis de imágenes planas de filetes de carne. El cual busca satisfacer todas las necesidades para la medición de características interactivas con anotación de imágenes con la posibilidad de almacenar y revisar imágenes a través de su propia galería. Adicionalmente otorga la ventaja de procesamiento de imágenes a color, escala de grises y binarias, incluyendo la medición automática multi-parámetros. El procedimiento era semiautomático puesto que un experto humano marcaba sobre la imagen de jamón las zonas que corresponden a grasa y posteriormente el programa calculaba el porcentaje de grasa basándose en las zonas previamente marcadas (2). Más adelante en la tesis doctoral de Hernández Cázares en el 2010 (1) propone un método para la identificación automática de estas estructuras en imágenes de jamón de cerdo Ibérico, usando un sensor enzimático basado en la enzima comercial diamino oxidasa.



*Figura 1: Imagen de una loncha comercial de jamón Ibérico*

La empresa Bravo S.A., de la provincia de la Habana, está perfeccionando sus procesos productivos en busca de un aumento de la calidad en sus productos finales. Esto trajo consigo reorganizaciones en las plantas de producción y la incorporación de un cúmulo de cámaras digitales para el control del proceso de confección de sus productos fundamentales (jamón y chorizo). Específicamente el jamón es confeccionado con carnes provenientes del cerdo y la res.

Existen varios estudios que asocian parámetros de calidad a elementos visuales del producto, demostrándose que se pueden identificar parámetros como:

1. -Saturación de grasas.
2. -Saturación de cárnicos varios.

La empresa no cuenta con modernos equipamientos de resonancia magnética ni sensores enzimáticos, aunque sí tienen a su disposición un dispositivo de adquisición de imágenes a color RGB con las siguientes características (cámara):

- Resolución de 13,0 MP (4160 X 3120)
- Lente focal a 3,97 mm.
- Flash automático
- Auto foco
- Zoom de hardware y software
- Zoom suave
- Balance automático de blancos y de exposición.

Con este dispositivo instalado en la línea de producción se obtienen imágenes como la representada en la figura 2. A partir de las imágenes obtenidas, la empresa necesita estimar el porcentaje de saturación de grasas con el que quedaría el producto final y a su vez determinar la calidad de la materia prima.



*Figura 2: Imagen tomada en la línea de producción de la empresa Bravo SA*

Actualmente la empresa no tiene a su disposición una herramienta que permita calcular el porcentaje de saturación de grasas, en la imagen de muestra de jamón (figura 2). Con el objetivo de ser analizada detalladamente por el departamento de calidad.

La situación antes descrita genera incertidumbre en el resultado obtenido por los especialistas al evaluar el porcentaje de saturación de grasas. De igual forma dificulta la selección de las materias primas idóneas para la producción del jamón por lo que se identifica el siguiente **problema a resolver**: ¿Cómo identificar grasas a partir de imágenes de jamón de la línea productiva de la empresa Bravo S.A.?

**Objeto de estudio:** Procesamiento de imágenes digitales.

**Campo de acción:** Segmentación de imágenes de jamón de la empresa Bravo S.A.

**Objetivo general:** Desarrollar un algoritmo que permita el cálculo de saturación de grasas de imágenes de jamón provenientes de la línea productiva de la empresa Bravo S.A.

**Objetivos específicos:**

- Elaborar el marco teórico de la investigación mediante el estudio de las técnicas actuales de procesamiento digital de imágenes, con énfasis en el campo de la segmentación.
- Identificar los requisitos funcionales y no funcionales para el desarrollo de la propuesta de solución.
- Diseñar un algoritmo para el cálculo automático del porcentaje de grasa identificada en una imagen de jamón de la línea productiva de la empresa Bravo SA.
- Implementar los elementos definidos en la etapa del diseño.
- Validar los resultados de la segmentación de las grasas y de la implementación del sistema.

**Idea a defender:** Con el desarrollo de un algoritmo para el procesamiento de las imágenes de jamones, la empresa Bravo S.A. contará con una herramienta para la estimación del porcentaje de grasas en la conformación del producto final.

**Tareas a cumplir:**

- 1- Identificación de las diferentes técnicas de procesamiento de imágenes digitales.
- 2- Revisión de las técnicas de filtrado y mejora en imágenes digitales.
- 3- Estudio del uso de algoritmos de segmentación por umbralización en imágenes.
- 4- Descripción de los pasos a seguir para el uso del algoritmo propuesto y definir los requisitos de software.
- 5- Diseñar la solución propuesta en función de los requisitos especificados.
- 6- Implementación de las técnicas de filtrado y mejora de la imagen de jamón.
- 7- Implementación del algoritmo de segmentación por umbralización propuesto.
- 8- Validar los resultados de la segmentación mediante medidas estadísticas de error, de localización y las métricas de Hausdorff.

- 9- Validación de la implementación del sistema a partir de la aplicación de pruebas unitarias y funcionales.

La lógica de la investigación se desarrolla a partir de **indagaciones teóricas y empíricas** en las cuales rige el **enfoque general dialéctico materialista del conocimiento**.

En el **nivel teórico** del conocimiento se utilizaron los métodos siguientes:

**Análisis documental:** Permitió realizar el estudio bibliográfico como aspecto esencial en la conformación del marco teórico referencial de la investigación. Constituye además el sustento de las valoraciones realizadas, así como el estudio de las principales técnicas de preprocesamiento y segmentación de imágenes digitales.

**Análisis histórico-lógico:** Posibilitó la aproximación a la evolución de los referentes teóricos del tema, analizar diferentes criterios relacionados con la trayectoria y evolución de software de cálculo de porcentaje de grasas, para poder comprender el nivel de desarrollo alcanzado por este tema en el mundo, basándose en las experiencias existentes se logrará un mayor entendimiento y comprensión de este tipo de sistemas,

**Método Analítico-Sintético:** Con el objetivo de analizar las teorías, documentos e información, permitiendo la extracción de los elementos más importantes que se relacionan con la segmentación en imágenes de jamón.

**Sistematización de experiencia:** Contribuyó a identificar ideas, aportaciones teóricas y prácticas de los estudios realizados.

Dentro de los métodos del **nivel empírico** del conocimiento se destacan los siguientes:

**Observación:** Se observaron los resultados obtenidos en la caracterización e identificación de los principales algoritmos utilizados, para decidir cuáles arrojaron resultados adecuados.

**Experimento:** Método empírico mediante el cual, se realizarán experimentos y pruebas de los principales algoritmos de segmentación de imágenes de muestra de jamón, para examinar los resultados en busca de escoger los más adecuados.

La presente investigación consta de tres capítulos, definidos de la siguiente forma:

### **Capítulo 1: Fundamentación Teórica.**

Se realizó un estudio de los algoritmos existentes para el procesamiento de imágenes digitales. Se presenta un conjunto de conceptos relacionados con el objeto y el campo. Además, se describe la metodología, las herramientas y el lenguaje de programación utilizado en el desarrollo de la investigación.

### **Capítulo 2: Análisis y diseño del algoritmo.**

Se describen las principales características del algoritmo propuesto, guiando todo el proceso por las fases definidas por la metodología XP (Programación Extrema). Se analizan elementos necesarios para garantizar el éxito en el proceso de desarrollo, tales como: las historias de usuario, patrones de diseño, tarjetas CRC (Clase, Responsabilidad y Colaborador) y los estándares de codificación.

### **Capítulo 3: Implementación y pruebas al algoritmo.**

Se da continuidad a las fases de la metodología XP, con la realización de pruebas unitarias, caja blanca y aceptación. Además, se describen los resultados de la aplicación de las métricas para validar la segmentación. Mostrándose finalmente la interfaz de usuario del sistema realizado. Las pruebas realizadas arrojaron resultados satisfactorios.

# 1 Capítulo 1: Fundamentación Teórica

En el presente capítulo se realizará una disquisición desde el punto de vista teórico se analiza en detalle el problema general en que se enmarca la investigación. Se abordan los principales conceptos asociados al dominio del problema. Se realiza un estudio del arte del procesamiento de imágenes, haciendo énfasis en las principales técnicas de preprocesamiento y los métodos de segmentación existentes en imágenes de jamón. Luego se identifican algoritmos para detectar la saturación de grasas en imágenes de muestra de jamón de la línea productiva de la empresa Bravo S.A.

## 1.1 Imagen digital

Una imagen digital (monocromática) puede considerarse como una matriz cuyos índices de filas y columnas identifican un punto de la imagen y el valor del correspondiente elemento de la matriz indica el nivel de gris en ese punto. Esto es lo que se conoce como píxel, abreviatura del término en inglés “picture elements”. Por el contrario en una imagen a colores, la información de ese punto se codificaría según el espacio de color a utilizar, creando una imagen multicanal. Estos espacios de color proporcionan un método para representar, ordenar y manipular los colores. El principio en el que se basan la mayoría de estos modelos consiste en que a partir de la combinación de tres colores primarios se puede obtener cualquier color de la naturaleza (3) (4).

### 1.1.1 Imágenes a Color

Las imágenes digitales a color están gobernadas por los mismos conceptos de muestreo, cuantificación y resolución que las imágenes en escala de grises. Sin embargo, en lugar de un único valor de intensidad que expresa el nivel de gris, los píxeles de las imágenes a color están cuantificados usando tres componentes. Que tendrán distinto significado según el modelo o espacio de color utilizado.

El modelo de color es una representación matemática del conjunto de colores que permite la especificación de un color de manera estándar. Los más conocidos son:

- RGB (sigla en inglés de red, green, blue, en español «rojo, verde y azul»), usado en los gráficos por computador; CMY (siglas de cian, magenta y amarillo).
- CMYK (siglas de cyan, magenta, yellow y negro), usados en sistemas de impresión.
- HSV (del inglés Hue, Saturation, Value – Matiz, Saturación, Valor).

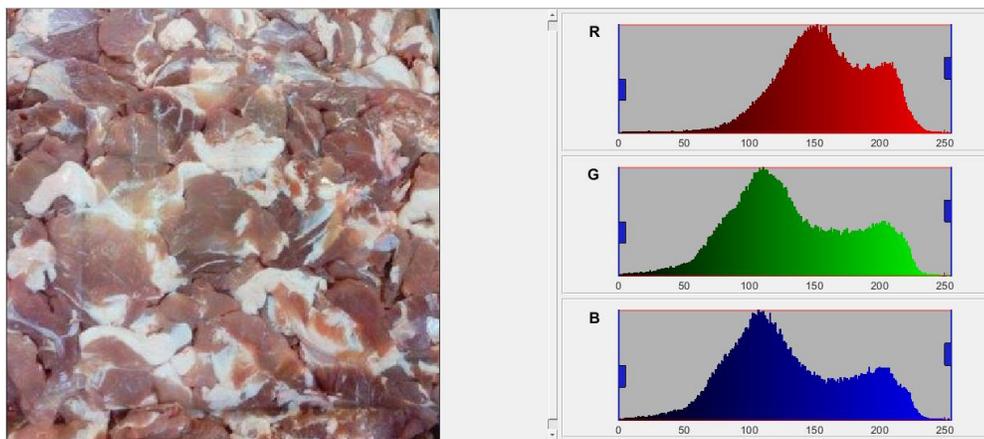
### 1.1.2 Imágenes de jamón de la empresa Bravo SA.

El estudio realizado a las imágenes obtenidas del dispositivo de captura instalado en la línea de producción de la empresa Bravo S.A., arrojó que la sección de interés de las mismas presentan las siguientes características (imagen):

1. Dimensiones: 345x360x3
2. Profundidad de bits: 8bits

3. Niveles de color: 256
4. Espacio nativo de color: RGB
5. Número de Bytes: 372600
6. Formato nativo de imagen: jpg (Menor calidad pues este formato usa compresión con pérdida)

Realizando un análisis al histograma de la imagen es apreciable que la oscuridad relativa presente es baja, mostrando mayor cantidad de píxeles en zonas de intensidad media y existiendo un bajo contraste. Desglosando el espacio de color RGB los histogramas relacionados a cada canal independiente muestran similitud en los valores de los canales azul y verde, así como mayores valores en los niveles del canal rojo.



*Figura 3: Histogramas de los 3 canales de la imagen de jamón*

Como resultado de este análisis se puede obtener información de la imagen. Siendo necesaria para su procesamiento.

## **1.2 Procesamiento de Imágenes**

El procesamiento digital de imágenes (PDI) es el procesado, entendiendo este como el almacenamiento, transmisión y representación de información, de imágenes digitales por medio de una computadora digital. El PDI consta con 5 etapas: adquisición de la imagen, preprocesamiento de la imagen, segmentación de la imagen, representación y descripción y por último reconocimiento e interpretación. Resulta conveniente para la investigación en curso, clasificar dichos procesos en categorías. De acuerdo a cada una de las etapas mencionadas, existen diferentes algoritmos involucrados, para un mejor entendimiento de las mismas se enuncian a continuación (5):

1. Adquisición de la imagen: Es el proceso a través del cual se obtiene una imagen deseada por el usuario con el fin de analizarla.
2. Preprocesamiento: incluye técnicas como la reducción de ruido y realce de detalles. Se realiza para que el resultado sea más apropiado para una aplicación en particular. Dichas técnicas consisten en la

mejora de la nitidez o aclarado de las imágenes desenfocadas, eliminación de ruido, mejora del contraste y detención de bordes.

3. Segmentación de la imagen: Subdivide la imagen digital en varias partes (grupos de píxeles) u objetos. Tareas habituales: búsqueda y selección de formas determinadas en la imagen, máscaras de la imagen.
4. Representación y descripción: Mediante los procesos de descripción se obtienen características (tamaño, perímetro, etc.) necesarias para diferenciar un objeto de otro.
5. Reconocimiento e interpretación: El reconocimiento es el proceso que identifica los objetos, y la interpretación le asocia un significado a un conjunto de objetos reconocidos. Generalmente es necesario agrupar estas áreas de acuerdo con la complicación y el grado de detalles que es consecuencia de su implementación.

Según las necesidades de obtención de información dentro de las imágenes y las características propias de la imagen en cuestión, se analizan cuáles de los algoritmos existentes muestran un mejor resultado en cada una de las etapas planteadas.

A continuación se hará referencia a algunas técnicas, en correspondencia con la investigación, si se ajustan al tipo de imagen tratada y los resultados obtenidos.

### **1.3 Preprocesamiento de imágenes**

Al digitalizar una imagen, es común la presencia de ruido o degradación, así como otros efectos indeseados, razón por la cual es importante restaurarla antes de ser procesada. En la etapa de preprocesamiento de imágenes se intenta aumentar la calidad de la imagen, con el fin de reconocer mejor las fallas que puedan existir en ella. Algunas de las técnicas empleadas en el preprocesamiento son eliminación de ruido, mejora del contraste y restauración (6).

#### **1.3.1 Espacios de color**

Los espacios de color proporcionan un método para especificar, ordenar y manipular colores. Estas representaciones se corresponden con n-dimensional ordenaciones de las sensaciones de color (vectores de n componentes). Los colores se representan mediante puntos en estos espacios. Existen numerosos espacios de color en la actualidad. La gran mayoría de ellos se han desarrollado para aplicaciones específicas, aunque todos parten de un mismo concepto: la teoría tricromática de colores primarios rojo, verde y azul (7).

Es un sistema de interpretación del color, es decir, una organización específica de los colores en una imagen o video. Depende del modelo de color en combinación con los dispositivos físicos que permiten las representaciones reproducibles de color, por ejemplo las que se aplican en señales analógicas (televisión a color) o representaciones digitales. Un espacio de color puede ser arbitrario, con colores particulares asignados según el sistema y estructurados matemáticamente (8).

**Modelo RGB.** El primero de los modelos de color a comentar y más comúnmente empleado es el RGB, basado directamente en el modelo triestímulo y síntesis aditiva. Es un modelo de color dependiente de dispositivo. Las imágenes en el modelo de color RGB están formadas por tres planos de imágenes independientes, cada uno correspondiente a un color primario. Cuando son introducidas en un monitor RGB, las tres imágenes se combinan en la pantalla de fósforo para producir una imagen de color compuesta (5). El empleo del modelo RGB para el procesamiento de imágenes es útil cuando éstas vienen expresadas en términos de los tres planos de colores. Alternativamente, la mayoría de las cámaras en color que se usan para adquirir imágenes digitales utilizan el formato RGB, lo que hace, si cabe, más interesante este formato.

**Modelo CMY.** Los colores cyan, amarillo y magenta son los secundarios del color luz o síntesis aditiva. Alternativamente, se corresponden con los primarios de la síntesis sustractiva del color. La amplia mayoría de los dispositivos que emplean pigmentos coloreados en la formación de los colores requieren la especificación CMY o CMYK (síntesis sustractiva a la que se añade una componente de tinta negra). El uso de los modelos de color CMY y CMY-K está concentrado en los sistemas de impresión. En visión por computador raramente son usados (7).

**Modelo YCbCr.** Se trata de una codificación no lineal del espacio RGB. El parámetro Y indica la luminancia, los parámetros Cb y Cr indican el tono del color: Cb ubica el color en una escala entre el azul y el amarillo, Cr indica la ubicación del color entre el rojo y el verde. El espacio YCbCr es una versión escalada y desplazada del espacio de color YUV. El parámetro Y representa la luminancia (es decir, información en blanco y negro), mientras que U y V representan la crominancia (es decir, información con respecto al color). Cb (Cr respectivamente) es la diferencia entre la componente azul (rojo, respectivamente) y un valor de referencia (9).

**Modelo HSI.** Sus siglas corresponden a H (Hue/Tonalidad), S (Saturation/Saturación), I (Intensity/Intensidad). El modelo HSI debe su utilidad a dos hechos básicos. Primero, la componente de intensidad, I, está desacoplada de la información cromática contenida en la imagen. Segundo, las componentes de tono y saturación están íntimamente relacionadas con la forma en que los seres humanos percibimos el color. Estas características hacen que el modelo HSI sea una herramienta ideal para desarrollar algoritmos de procesamiento de imágenes basados en alguna de las propiedades de la percepción del color del sistema visual humano (7).

### **1.3.2 Mejora del contraste**

A continuación se muestran algunas de las técnicas estudiadas para mejorar el contraste en las imágenes.

**La ecualización adaptativa del histograma.** Un histograma es una función que muestra una imagen con los distintos niveles de gris que contiene. La idea de la ecualización de histograma original es asignar un rango más amplio de niveles de gris a los valores de gris más frecuentes de una imagen, induciendo a que el histograma de la imagen resultante sea más uniforme que el original (10). Lo cual implica una mayor utilización de los recursos disponibles y un aumento del contraste. Sin embargo, la ecualización de histograma tiene resultados pobres cuando la imagen tiene una iluminación no uniforme espacialmente (10).

**El Algoritmo White Patch (WP),** o Parche Blanco en Español toma en cuenta el valor más grande en cada componente de color como una representación del blanco de la imagen (11). Computacionalmente, este parche blanco es calculado al encontrar la máxima intensidad en cada canal. El algoritmo WP puede hacerse más robusto si se calcula un histograma para cada componente de color y se considera el iluminante como un alto porcentaje de la acumulación del mismo (11). Este método es usado frecuentemente para remover sombras en la imagen.

**El Algoritmo Gray World (GW),** o Mundo Gris en Español es el método más conocido de constancia de color, usado como referencia por otros algoritmos (11), el GW está basado en la suposición de que en promedio, el mundo es gris, y estima el iluminante usando el color promedio de todos los píxeles. Se asume que la información dada por el promedio de cada canal de la imagen representa el nivel gris (11).

**Imadjust,** mapea los valores de intensidad en una imagen en escala de grises a nuevos valores, tal que el 1% de los datos es saturado a bajas y altas intensidades. Mejora el contraste de la imagen resultante. En la figura 6 se muestra el resultado de aplicar dicho método.

### **1.3.3 Detección de bordes**

El objetivo de esta técnica es resaltar las fronteras del objeto a segmentar (12) (13), tiene como ventaja la simplicidad, una vez encontradas las fronteras, pero a su vez presenta dificultades en encontrar estas fronteras (borde), requiriendo operadores paso alto de detección de bordes (Sobel y Canny) mostrando alto nivel de ruido (12) (13). Esta técnica utiliza operadores sofisticados y la construcción de fronteras se hace a partir de bordes.

**Operador de Canny.** Uno de los métodos más importantes para realizar una detección global de bordes sobre una imagen es el conocido como método de Canny (14). Esta técnica, que se caracteriza por estar optimizada para la detección de bordes diferenciales, consta de tres etapas principales: filtrado, decisión inicial, e histéresis (15). La primera etapa consiste en un filtrado de convolución de la derivada primera de una función gaussiana normalizada discreta sobre la imagen, realizada en dos direcciones: horizontal y vertical. La función gaussiana posee dos parámetros fundamentales, valor medio y desviación típica. En

este caso, el valor medio es nulo, siendo la ecuación que define el filtro gaussiano de la forma como se aprecia en la ecuación 1 (16):

$$g(x) = k \exp\left(-\frac{x^2}{2\sigma^2}\right) \quad (1)$$

El operador de Canny consiste en tres grandes pasos (14):

- Obtención del gradiente: en este paso se calcula la magnitud y orientación del vector gradiente en cada píxel.
- Supresión no máxima: en este paso se logra el adelgazamiento del ancho de los bordes, obtenidos con el gradiente, hasta lograr bordes de un píxel de ancho.
- Histéresis de umbral: en este paso se aplica una función de histéresis basada en dos umbrales; con este proceso se pretende reducir la posibilidad de aparición de contornos falsos (17).

**Operador de Sobel.** Los operadores de Sobel trabajan con el peso del píxel central de las filas y columnas externas de las máscaras, lo que hace que los bordes detectados sean más definidos (18). Con los operadores de Sobel se obtienen los bordes y además proporcionan un efecto de suavizado, ya que detectan bordes principales (18). Este efecto es una ventaja considerando que las derivadas realzan el ruido. El operador de Sobel se encuentra dentro de los operadores de primer orden.

## 1.4 Segmentación

Los algoritmos de segmentación de imágenes monocromáticas se basan en alguna de las tres propiedades siguientes (19):

- a) Discontinuidad en los tonos de gris de los píxeles de un entorno, que permite detectar puntos aislados, líneas y aristas (bordes).
- b) Similitud en los tonos de gris de los píxeles de un entorno, que permite construir regiones por división y fusión, por crecimiento o por umbralización.
- c) La conectividad de los píxeles desempeña un papel importante en la segmentación de imágenes. Junto con que una región D se dice conexa o conectada si para cada par de píxeles de la región existe un camino formado por píxeles de D que los conecta. Un camino de píxeles es una secuencia de píxeles adyacentes (que pertenecen a su entorno inmediato).

Los métodos de segmentación se pueden agrupar en tres clases diferentes (20):

- a) Métodos basados en píxeles, que a su vez pueden ser:
  - a. locales (basadas en las propiedades de los píxeles y su entorno)
  - b. globales (basadas en la información global obtenida, por ejemplo, con el histograma de la imagen).
- b) Métodos basados en bordes.

- c) Métodos basados en regiones, que utilizan las nociones de homogeneidad y proximidad geométrica, como las técnicas de crecimiento, fusión o división.

### 1.4.1 El histograma de una imagen

El histograma es una representación gráfica de la distribución de los distintos tonos de una imagen. Puede ayudar a controlar la exposición en nuestras fotos, así como para corregir los colores en determinadas imágenes (5).

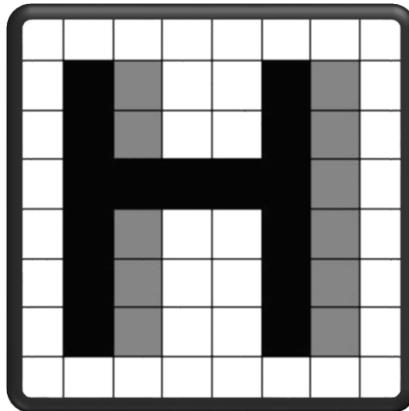


Figura 4: Imagen de 2 bit (5)

La figura 4 es una imagen de 2bit, posee 64 pixeles representados gráficamente en la figura 5. Para un análisis detallado.

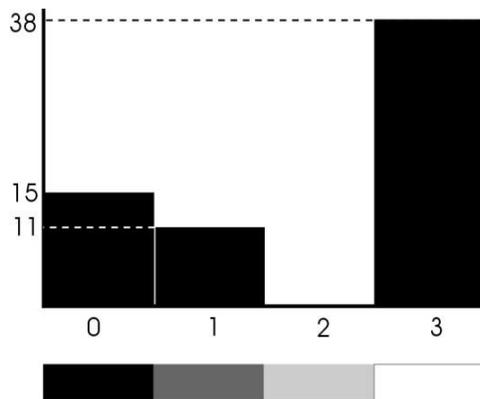


Figura 5: Histograma de la figura 4 (5)

El eje horizontal representa los diferentes tonos de gris desde el negro puro (0) al blanco puro (3). El eje vertical representa el número de pixeles que contiene la imagen para cada tono representado en el eje horizontal.

El histograma resulta una herramienta de gran interés en tareas de análisis automático de imágenes, como las que se llevan a cabo en el área de la visión por computador (21).

A continuación se muestra el histograma (figura 6) correspondiente a la imagen de muestra de jamón tomada en la línea de producción de la empresa Bravo S.A.

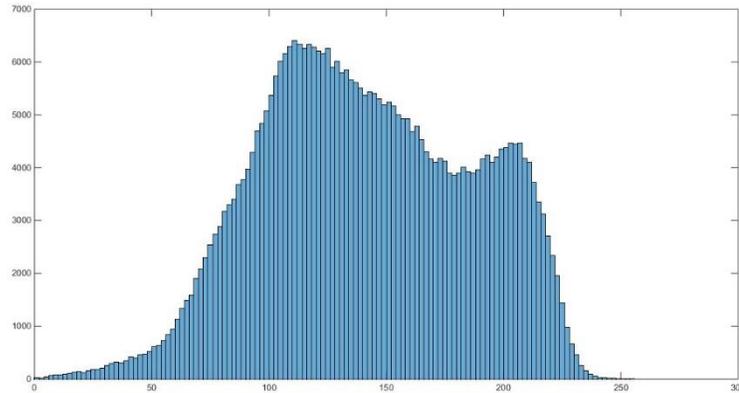


Figura 6: Histograma de la imagen de jamón (figura 2)

Muestra el histograma de niveles de intensidad de la imagen capturada por la cámara. Además, una formación de intensidades relativamente compacta, luego compuesta por el máximo global seguido de un valle y un mínimo local, en la cual predominan los tonos medios oscuros.

### 1.4.2 Umbralización

La umbralización es una técnica de segmentación ampliamente utilizada en las aplicaciones industriales. Se emplea cuando hay una clara diferencia entre los objetos a extraer respecto del fondo de la escena. Los principios que rigen son la similitud entre los píxeles pertenecientes a un objeto y sus diferencias respecto al resto. Por tanto, la escena debe caracterizarse por un fondo uniforme y por objetos parecidos.

Al aplicar un umbral,  $T$ , la imagen en escala de grises,  $f(x,y)$ , quedará binarizada; etiquetando con '1' los píxeles correspondientes al objeto y con '0' aquellos que son del fondo. Por ejemplo, si los objetos son claros respecto del fondo, se aplicará:

$$g(x,y) = \begin{cases} 1 & \leftrightarrow f(x,y) > T \\ 0 & \leftrightarrow f(x,y) \leq T \end{cases} \quad (2)$$

En el caso de que los objetos sean oscuros respecto del fondo, la asignación sería a la inversa:

$$g(x,y) = \begin{cases} 1 & \leftrightarrow f(x,y) > T \\ 0 & \leftrightarrow f(x,y) \leq T \end{cases} \quad (3)$$

El umbral puede depender de  $f(x,y)$ , de alguna propiedad local del píxel,  $p(x,y)$ , y hasta de su propia posición:

$$T = T(f(x,y), p(x,y), x, y) \quad (4)$$

Si el umbral sólo depende de  $f(x,y)$  se dice que es un umbral global; en el caso de que además dependa de  $p(x,y)$ , por ejemplo, el valor medio de los píxeles vecinos, el umbral es denominado local; y si depende también de la posición  $(x,y)$  del píxel, se denominará dinámico.

La mayoría de las técnicas de umbralización se basan en estadísticas sobre el histograma unidimensional (el visto hasta ahora) de una imagen. También se utiliza la matriz de co-ocurrencia de una imagen. Para localizar los umbrales se pueden usar procedimientos paramétricos y no paramétricos. En los paramétricos, la distribución de los niveles de gris de una clase de objeto lleva a encontrar los umbrales. En los procedimientos no paramétricos, los umbrales se obtienen de forma óptima de acuerdo a algún criterio.

### 1.4.3 Métodos adaptativos

En muchas aplicaciones, no se puede obtener un umbral global para un histograma, es decir, no se puede obtener una buena segmentación con un único umbral para toda la imagen. Esto ocurre cuando el fondo no es constante y el contraste de los objetos varía en la imagen, por lo que la umbralización daría buenos resultados en una parte de la imagen, pero para el resto de la imagen, la segmentación no sería la adecuada. Si las variaciones del fondo de la imagen pueden ser descritas a través de funciones conocidas, dependientes de la posición en la imagen, se podría intentar corregir la segmentación utilizando la corrección de niveles de gris; para que, posteriormente, la aplicación de un único umbral a toda la imagen, diera buenos resultados en la segmentación. Otra solución sería el uso de la umbralización local o adaptativa (22).

#### 1.4.3.1 Método de Wellner 1993

La mayoría de los algoritmos en la literatura son costosos computacionalmente y requieren muchas iteraciones de los mismos, tardando largos tiempos de ejecución. La idea básica es recorrer la imagen e ir calculando una media de movimiento de los últimos  $s$  píxeles analizados. Cuando el valor de un píxel analizado es significativamente menor a la media, es puesto en negro y sino puesto en blanco (23).

Solo se necesita una iteración por la imagen y el algoritmo es bastante simple.

Siendo  $p_n$  el valor del píxel en el punto  $n$  de la imagen umbralada. Por el momento se trata la imagen como si fuera una sola fila de píxeles compuestos por todas las filas en la imagen alineadas una detrás de la otra.

Siendo  $f_s(n)$  la suma de los valores de los últimos  $s$  píxeles en el punto  $n$ .

$$f_s(n) = \sum_{i=0}^{s-1} p_{n-i} \quad (5)$$

El valor de la imagen resultante  $T(n)$  es 1 (para los negros) o 0 (para los blancos), dependiendo en cuál es el  $t$  por ciento de oscuridad mayor que la media de los anteriores  $s$  píxeles (23).

$$T(n) = \begin{cases} 1 & \text{si } p_n < \left( \frac{f_s(n)}{s} \right) \left( \frac{100-t}{100} \right) \\ 0 & \text{de otra forma} \end{cases} \quad (6)$$

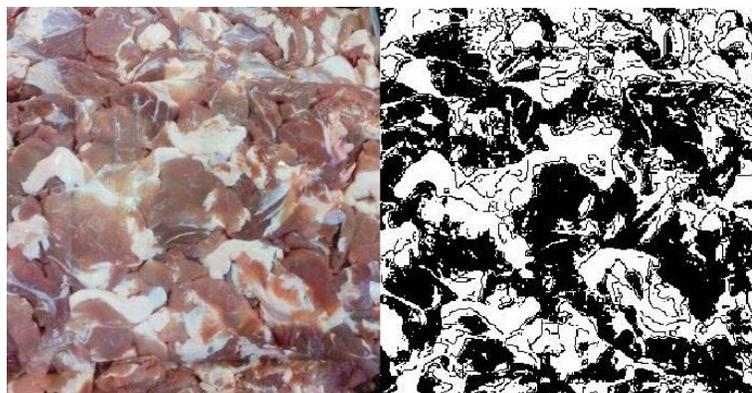


Figura 7: Resultado de aplicar el método de umbralización adaptativo Wellner 1993

La figura 7 muestra la imagen de materia prima de la empresa Bravo S.A (izquierda) junto a la imagen binaria (derecha), resultado de la segmentación por el método de adaptativo Wellner 1993. En la cual se puede apreciar una aproximación de la segmentación deseada (saturación de grasas). Se evidencia la sobre segmentación.

#### 1.4.3.2 Método de Phansalkar 2010

Basado en la técnica usada en el método propuesto por Sauvola and Pietikainen para el umbralado de imágenes de documentos se desarrolla este método para un problema específico (24).

$$T(x, y) = m(x, y) \left[ 1 + pe^{-q*m(x,y)} + k \left( \frac{s(x, y)}{R} - 1 \right) \right] \quad (7)$$

Donde T es el umbral local y es hallado de la media M, la desviación estándar S es el resultado de las intensidades de los pixeles en una ventana de W x W centrada alrededor del pixel. R es el rango dinámico de la desviación estándar (0.5 para una imagen normalizada). K es una constante que toma valores del rango [0.2, 0.5] arrojando k=0.25 los mejores resultados (24).

Las variables p y q son constantes, el valor de q es elegido por encima de la media local, la constante p decide la magnitud en la que el término exponencial afectara el umbral (24).

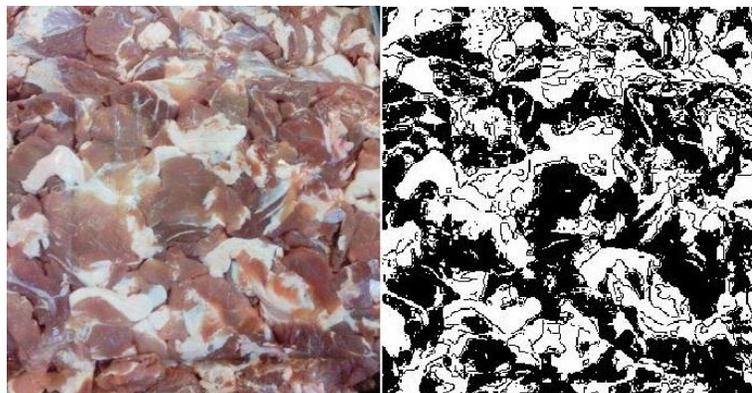


Figura 8: Resultado de aplicar el método de umbralización adaptativo Phansalkar 2010

La figura 8 muestra la imagen de materia prima de la empresa Bravo S.A (izquierda) junto a la imagen binaria (derecha), resultado de la segmentación por el método de adaptativo Phansalkar 2010. En la figura se puede apreciar una aproximación de la segmentación deseada (saturación de grasas) en la cual se evidencia la sobre segmentación.

### 1.4.3.3 Método de Sauvola 2000

Esta técnica de binarización es orientada a ser usada como base en diferentes análisis y procesamiento de documentos.

La salidas de estas técnicas representan el valor umbral propuesto por cada n pixel decidido por el usuario estos valores son usados para decidir la salida final de la binarización (25).

$$T(x, y) = m(x, y) \left[ 1 + k \left( \frac{s(x, y)}{R} - 1 \right) \right] \quad (8)$$

Donde T es el umbral local, y es hallado de la media M y la desviación estándar S de las intensidades de los pixeles en una ventana de W x W centrada alrededor del pixel, R es el rango dinámico de la desviación estándar (0.5 para una imagen normalizada), K es una constante que toma valores del rango [0.2, 0.5] arrojando k=0.25 los mejores resultados (25).

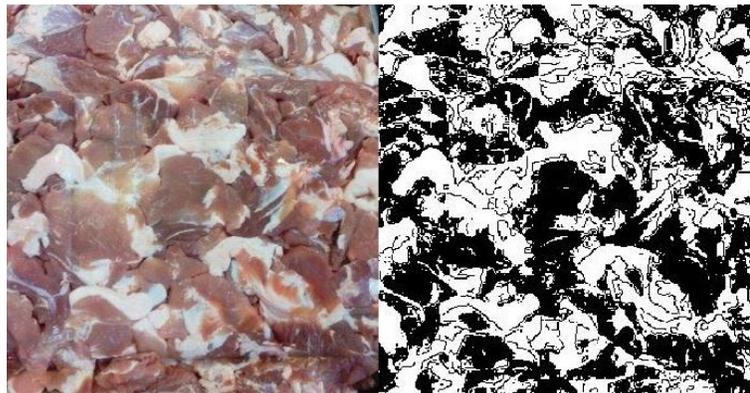


Figura 9: Resultado de aplicar el método de umbralización adaptativo Sauvola 2000

La figura 9 muestra la imagen de materia prima de la empresa Bravo S.A (izquierda) junto a la imagen binaria (derecha), resultado de aplicar el método de umbralización adaptativo Sauvola 2000. En la cual se puede apreciar una aproximación de la segmentación deseada (saturación de grasas) en la cual se aprecia la sobre segmentación.

#### 1.4.3.4 Método White's 1983

En busca de sobreponerse a las dificultades de la extracción de caracteres se diseña este algoritmo de umbralado o circuito a ser usado para obtener información a priori acerca de las imágenes de caracteres para la asignación de recursos para procesarlo según su costo (26).

Conceptualmente se compara el valor de gris de un pixel con la media de valores de gris de una vecindad adecuada para un carácter, si el pixel es significativamente más oscuro que los pixeles vecinos es asignado negro (26).

$$y(n) = y(n - 1) + f[u(n) - y(n - 1)] \quad (9)$$

La solución usa una media "corrida":  $y(n)$  proveniente de una cadena de valores de gris digitalizados y muestreados:  $u(n)$  una fracción  $f$  del valor de gris es agregado para complementar la media anterior (26).

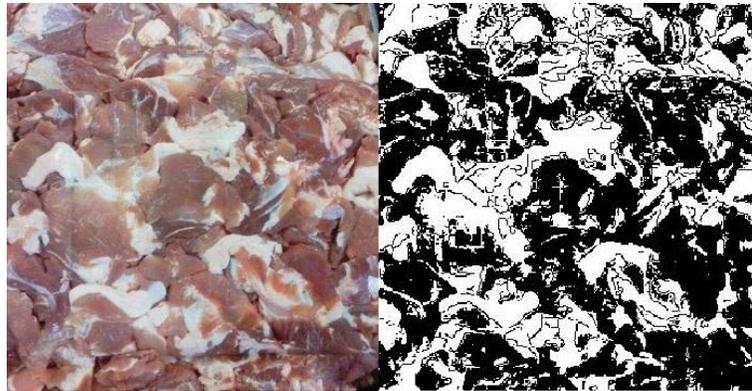


Figura 10: Resultado de aplicar el método de umbralado adaptativo White's 1983

La figura 10 muestra la imagen de materia prima de la empresa Bravo S.A (izquierda) junto a la imagen binaria (derecha), resultado de la segmentación por el método de adaptativo White's 1983. En la cual se puede apreciar una aproximación de la segmentación deseada (saturación de grasas). Se aprecia la sobre segmentación.

#### 1.4.4 Métodos Globales

En los métodos globales existen  $T_{global}$  que es el valor de intensidad del píxel segmentado,  $g$  es el valor de cada píxel y  $t$  es el valor del umbral. El punto clave en esta técnica es la elección del valor umbral adecuado. Esto se puede realizar de forma manual, pero normalmente se necesita un método que permita calcular este valor de forma automática. El histograma de la imagen será el elemento más importante para definir el valor umbral, tanto si se elige hacerlo de una forma manual como automática (27).

$$T_{global} g = \begin{cases} 0 & \text{si } g < t \\ 1 & \text{si } g \geq t \end{cases} \quad (10)$$

#### 1.4.4.1 Metodo de Otsu 1979

Una imagen es una función bidimensional de la intensidad del nivel de gris, y contiene  $N$  píxeles cuyos niveles de gris se encuentran entre  $1$  y  $L$ . El número de píxeles con nivel de gris  $i$  se denota como  $f_i$ , y la probabilidad de ocurrencia del nivel de gris  $i$  en la imagen está dada por:

$$p_i = \frac{f_i}{N} \quad (11)$$

En el caso del umbralado en dos niveles de una imagen (llamada binarización), los píxeles son divididos en dos clases:  $C_1$ , con niveles de gris  $[1, \dots, t]$ ; y  $C_2$ , con niveles de gris  $[t+1, \dots, L]$ . Entonces, la distribución de probabilidad de los niveles de gris para las dos clases es:

$$C_1: \frac{p_1}{w_1(t)}, \dots, \frac{p_t}{w_1(t)} \quad (12)$$

$$C_2: \frac{p_{t+1}}{w_2(t)}, \frac{p_{t+2}}{w_2(t)}, \dots, \frac{p_L}{w_2(t)} \quad (13)$$

Donde

$$w_1(t) = \sum_{i=1}^t p_i \quad w_2(t) = \sum_{i=t+1}^L p_i \quad (14)$$

También, la media para la clase  $C_1$  y la clase  $C_2$  es

$$\mu_1 = \sum_{i=1}^t \frac{i * p_i}{w_1(t)} \quad \mu_2 = \sum_{i=t+1}^L \frac{i * p_i}{w_2(t)} \quad (15)$$

Sea  $\mu_T$  la intensidad media de toda la imagen. Es fácil demostrar

$$w_1 * \mu_1 + w_2 * \mu_2 = \mu_T \quad w_1 + w_2 = 1 \quad (16)$$

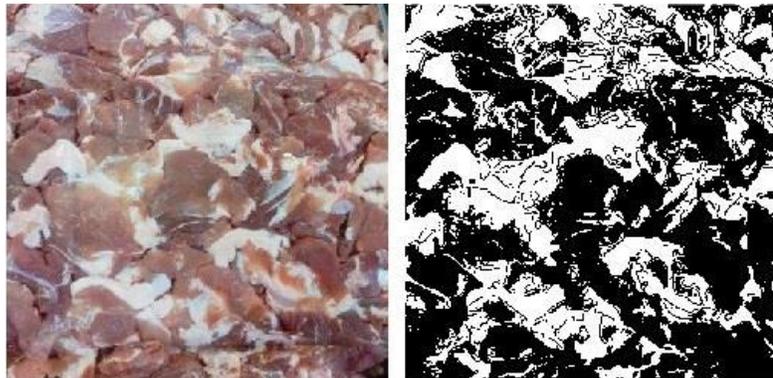


Figura 11: Resultado de aplicar el método de umbralado global Otsu 1979

La figura 11 muestra la imagen de materia prima de la empresa Bravo S.A (izquierda) junto a la imagen binaria (derecha), resultado de la segmentación utilizando el umbral de valor 0.50392 obtenido por el método global Otsu 1979. En la cual se puede apreciar una aproximación de la segmentación deseada (saturación de grasas). Se evidencia la sobre segmentación.

#### 1.4.4.2 Método de Huang (Shannon) 1995

La teoría difusa ha sido aplicada al umbralado de imágenes para particionar una imagen en sus partes significantes minimizando la medida de difusión de la imagen, la misma puede ser expresada como entropía índice de difusión e índice de no-difusión

$$X = \{(x_{mn}, \mu_x(X_{mn}))\} \quad (17)$$

Siendo  $X$  una imagen de tamaño  $M \times N$  y  $L$  niveles con  $X_{mn}$  es el nivel de gris de un pixel  $(m,n)$  en  $X$

Denotamos  $\mu_x(X_{mn})$  como el valor de membresía que representa el grado de posesión de cierta característica de dicho pixel, que es, un subespacio difuso de la imagen

Siendo  $h(g)$  el número de ocurrencias en el nivel de gris  $g$  en una imagen de entrada. Dado cierto valor umbral  $t$ , la media de niveles de gris del fondo  $\mu_0$  y el objeto  $\mu_1$  pueden ser respectivamente obtenidos de la siguiente forma

$$\mu_0 = \frac{\sum_{g=0}^t gh(g)}{\sum_{g=0}^t h(g)} \quad (18)$$

Y

$$\mu_1 = \frac{\sum_{g=t+1}^{L-1} gh(g)}{\sum_{g=t+1}^{L-1} h(g)} \quad (19)$$

La media de niveles de gris  $\mu_0$  y  $\mu_1$  puede ser considerada como los valores objetivos para el fondo y el objeto para el umbral dado  $t$ . La relación entre un pixel en  $x$  y su región correspondiente debería ser intuitiva dependiendo de su diferencia de nivel de gris con su región perteneciente, la función que evalúa la relación para un  $a(m, n)$  pixel puede ser definida como:

$$\mu_x(X_{mn}) = \frac{1}{1 + |X_{mn} - \mu_0|/C} \text{ si } X_{mn} \leq t \quad (20)$$

$$\mu_x(X_{mn}) = \frac{1}{1 + |X_{mn} - \mu_1|/C} \text{ si } X_{mn} > t \quad (21)$$

Donde  $C$  es una constante que toma valor tal que  $1/2 \leq \mu_x(X_{mn}) \leq 1$  para un umbral dado  $t$ , todo pixel en la imagen de entrada debería pertenecer o al objeto o al fondo



Figura 12: Método de umbralado global Huang (Shannon) 1995

La figura 12 muestra la imagen de materia prima de la empresa Bravo S.A (izquierda) junto a la imagen binaria (derecha), resultado de la segmentación utilizando el umbral de valor 0.47266 obtenido por el método global Huang (Shannon) 1995. En la cual se puede apreciar una aproximación de la segmentación deseada (saturación de grasas). Se evidencia la sobre segmentación.

#### 1.4.4.3 Método de Tsai 1995

Este método propone otro método de umbralado basado en el principio de conservación del momento el cual también es aplicado a una detección de bordes en subpíxeles. Específicamente antes del umbralado se computan los momentos de niveles de gris de la imagen de entrada. Los umbrales son entonces seleccionados de forma tal que los momentos de la imagen umbralada no son afectados. Este acercamiento puede ser visto como una transformación de una imagen que preserva el momento la cual recupera una imagen ideal de una versión borrosa, la misma puede automáticamente y determinísticamente seleccionar múltiples umbrales sin iteración ni búsqueda, además un valor representativo de gris puede ser obtenido por cada clase umbralada.

Dada una imagen  $f$  con  $n$  píxeles cuyo nivel de gris en el píxel  $(x,y)$  es denotada por  $f(x,y)$  se busca umbralizar  $f$  en dos clases de píxeles los comprendidos por debajo del umbral y los por encima. El  $i$ -ésimo momento  $m_i$  de  $f$  es definido como:

$$m_i = \left(\frac{1}{n}\right) \sum_j n_j(z_j)^i \quad (22)$$

$$= \sum_j p_j(z_j)^i \quad (23)$$

Donde  $n_j$  es el número total de píxeles en  $f$  con valor de gris  $z_j$  y  $p_j = n_j/n$

Y  $m_0$  se define como 1. La imagen  $f$  puede ser considerada como una versión borrosa de una imagen bi-nivel ideal la cual consiste en píxeles con solo dos valores de gris  $z_0$  y  $z_1$

Donde  $z_0 < z_1$ . El umbralado de preservación de movimiento propuesto selecciona un valor de umbral tal que si todo nivel de gris por debajo del umbral en  $f$  es remplazado por  $z_0$  y todo nivel por encima es remplazado por  $z_1$  entonces los primeros tres momentos de la imagen  $f$  son preservados en la imagen bi-nivel resultante  $g$ . La imagen  $g$  obtenida puede ser vista como una versión no-borrosa ideal de  $f$

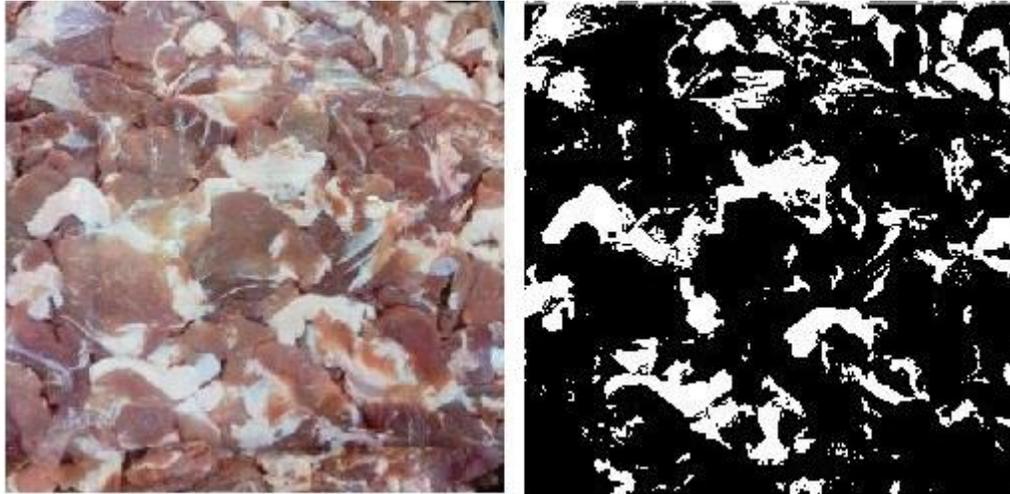


Figura 13: Método de umbralado global Tsai(1995)

La figura 13 muestra la imagen de materia prima de la empresa Bravo S.A (izquierda) junto a la imagen binaria (derecha), resultado de la segmentación utilizando el umbral de valor 0.74118 obtenido por el método global Tsai(1995). En la cual se puede apreciar una aproximación de la segmentación deseada (saturación de grasas). Se evidencia la sub-segmentación.

#### 1.4.4.4 Método de Khashman 2007

Este método usa una técnica de umbralado global de un solo escenario la cual busca el valor umbral óptimo para imágenes de documentos usando el valor de luminancia (punto de mayor luminancia) y la media de intensidades de la imagen. La relación entre los valores de color de una imagen en escala de gris provee un punto de umbral del fondo y el objeto de la imagen. El método difiere de métodos basados en histogramas al usar la luminancia en vez de valores picos en el histograma para hacer una separación eficiente para imágenes tanto bimodal como unimodal, considerando toda la información perteneciente a la imagen. El valor de luminancia puede ser considerado siempre como 255, pero en imágenes de documentos es observado que muy pocos tienen un valor de luminancia de 255, por lo tanto el valor de luminancia es considerado una variable en este método, siendo computada como:

$$G_{max} = F_{max}(g) \quad (24)$$

Donde  $G_{max}$  denota el valor de luminancia y  $F_{max}$  es una función que busca el máximo punto de gris en el  $x$  plano del histograma de luminancia representado como  $g$ .

La media de intensidades de una imagen es usada para computar el punto de separación inicial entre el fondo y el objeto de esta forma.

$$M = \left( \sum_{y=0}^{dim_y} \sum_{x=0}^{dim_x} I[x, y] \right) / (dim_y \times dim_x) \quad (25)$$

Donde  $M$  representa la media de la imagen  $dim_x$  y  $dim_y$  denotan las  $[x, y]$  dimensiones de la imagen respectivamente e  $I[x, y]$  representa la imagen en escala de grises original.

La desviación local es la diferencia entre el valor de la luminancia y la media de intensidades

Que representa la división para el umbral óptimo desde el punto de separación inicial como sigue:

$$D = G_{max} - M \quad (26)$$

Donde  $D$  es la desviación local para el valor medio de luminancia.

Por lo tanto la desviación entre el valor medio el cual es el punto de separación inicial, la desviación local, dan el punto de separación exacto para un documento:

$$\Omega = M - D \text{ por lo tanto } \Omega = |M - (G_{max} - M)| \quad (27)$$

Donde  $\Omega$  denota el umbral óptimo para la imagen, este método permite minimizar la pérdida de información y la difusión de los textos en el análisis de documentos en general la formula se puede escribir:

$$MI[x, y] = \begin{cases} 0 & \text{si } I[x, y] \leq \Omega \\ 255 & \text{si no} \end{cases} \quad (28)$$

Donde  $MI[x, y]$  representa la imagen umbralada



Figura 14: Método de umbralado global Khashman (2007)

La figura 14 muestra la imagen de materia prima de la empresa Bravo S.A (izquierda) junto a la imagen binaria (derecha), resultado de la segmentación utilizando el umbral de valor 0.90039 obtenido por el método global Khashman (2007). En la cual se puede apreciar una aproximación de la segmentación deseada (saturación de grasas). Se evidencia la sub segmentación.

#### 1.4.4.5 Método de Tsallis 1988

El método de entropía de Tsallis se basa en la técnica de Kapur(1985), pero usando la entropía de Tsallis debido a la presencia de información añadida en lagunas clases de la imagen, El método sustituye una expresión puramente aditiva encontrada en el método original de Kapur por una expresión pseudo-aditiva definida por tsallis para los sistemas independientes estadísticos

Para una imagen con k niveles de gris siendo  $p_i = p_1, p_2, p_3 \dots p_k$  siendo la distribución de probabilidad de los niveles. De esta distribución se derivan dos distribuciones de probabilidad una para el objeto (clase A) y otra para el fondo (clase B). Las distribuciones de probabilidad del objeto y fondo están dadas por:

$$P_A: \frac{P_1}{P_A}, \frac{P_2}{P_A}, \dots, \frac{P_t}{P_A}, \quad (29)$$

$$P_B: \frac{P_{t+1}}{P_B}, \frac{P_{t+2}}{P_B}, \dots, \frac{P_k}{P_B}, \quad (30)$$

Donde

$$P_A = \sum_{i=1}^t P_i \quad (31)$$

Y

$$P_B = \sum_{i=t+1}^k P_i \quad (32)$$

La entropía de Tsallis para cada distribución es definida como:

$$P_q^A(t) = \frac{1 - \sum_{i=1}^t \left(\frac{P_i}{P_A}\right)^q}{q - 1} \quad (33)$$

$$P_q^B(t) = \frac{1 - \sum_{i=t+1}^k \left(\frac{P_i}{P_B}\right)^q}{q - 1} \quad (34)$$

La entropía de Tsallis  $S_q(t)$  es paramétricamente dependiente del valor de umbral t para el fondo y el objeto. Es formulado como la suma de cada entropía, permitiendo la propiedad pseudo-aditiva para sistemas independientes estadísticamente

$$S_q(t) = \frac{1 - \sum_{i=1}^t \left(\frac{P_i}{P_A}\right)^q}{q - 1} + \frac{1 - \sum_{i=t+1}^k \left(\frac{P_i}{P_B}\right)^q}{q - 1} + (1 - q) \cdot \frac{1 - \sum_{i=1}^t (P_A)^q}{q - 1} \cdot \frac{1 - \sum_{i=t+1}^k (P_B)^q}{q - 1} \quad (35)$$

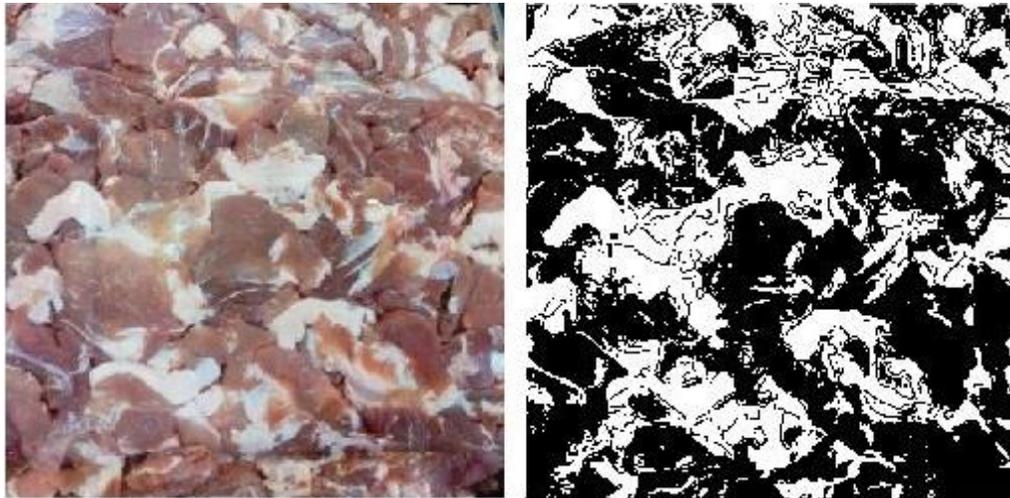


Figura 15: Método de umbralado global Tsallis 1988

La figura 15 muestra la imagen de materia prima de la empresa Bravo S.A (izquierda) junto a la imagen binaria (derecha), resultado de la segmentación utilizando el umbral de valor 0.49609 obtenido por el método global Tsallis 1988. En la cual se puede apreciar una aproximación de la segmentación deseada (saturación de grasas). Se evidencia la sobre segmentación.

### 1.5 Métricas utilizadas para la validación

Los problemas de segmentación de imágenes involucran la división de una imagen en un conjunto de regiones diferentes, de acuerdo a una medida de similitud determinada. Las métricas de desempeño para este tipo de problemas son generalmente más complejas. Existen dos versiones del problema de segmentación, segmentación genérica de bajo nivel y detección y localización de anomalías, en la segmentación genérica de bajo nivel el objetivo es producir descripciones del contenido de una imagen, mientras que en el problema de detección y localización de anomalías el objetivo es encontrar todas las regiones de una imagen que contengan formas anómalas (28).

Al comparar el resultado de un algoritmo de segmentación con una imagen segmentada de referencia (p.e. Segmentación manual de un experto) se tienen en cuenta las siguientes condiciones de comparación (28):  
**Segmentación correcta de una región:** Esto sucede cuando el solapamiento entre la región segmentada por el algoritmo propuesto y la región correspondiente en la imagen de referencia cumplen con cierto umbral mínimo de similitud.

**Sub-segmentación de una región:** La sub-segmentación sucede cuando varias regiones de la imagen de referencia corresponden solo a una región de la imagen segmentada por el algoritmo propuesto.

**Sobre-segmentación de una región:** Ocurre cuando varias regiones de la imagen segmentada por el algoritmo propuesto, corresponden a una sola región de la imagen de referencia.

**Región faltante:** La región faltante, existe cuando una región en la imagen de referencia no corresponde con ninguna región existente en la imagen segmentada por el algoritmo propuesto.

**Región producida por ruido:** Este tipo de regiones aparecen en la imagen segmentada y no corresponden a ninguna región de la imagen de referencia.

Un ejemplo de las condiciones de comparación anteriores se puede ver en la siguiente imagen.

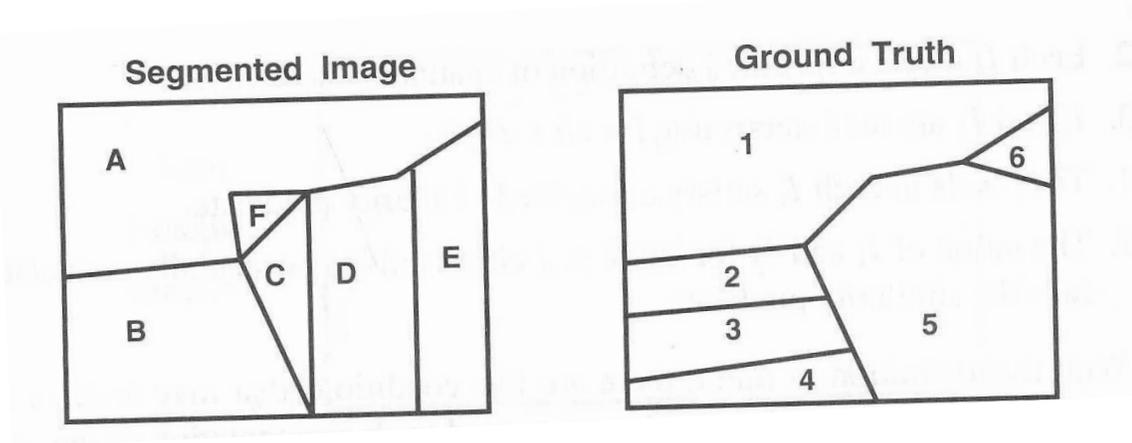


Figura 16: Ejemplo de condiciones de comparación de segmentación (29).

Donde:

- A corresponde a 1 como una instancia de una segmentación correcta.
- 5 corresponde a C, D y E como una sub\_ segmentación.
- B corresponde a 2,3 y 4 como una sobre\_ segmentación.
- 6 es una región perdida.
- F es una región producida por ruido.

### 1.5.1 Medidas del error

Las medidas de error son usadas frecuentemente a la hora de diseñar algoritmos de segmentación, clasificación y en la detección de contornos en sistemas de inteligencia artificial. Estas medidas se clasifican en tres grupos (30):

- Medidas estadísticas de error.
- Medidas de localización.
- Métrica Hausdorff

#### 1.5.1.1 Medidas estadísticas de error.

Las medidas estadísticas de error nos darán la medida de la frecuencia con la que se clasifican incorrectamente los píxeles, esta medida tiene cuatro variables de evaluación

Definidas a continuación (30) (31):

**Verdaderos Positivos (VP):** Son los números de píxeles que pertenecen a la imagen clasificados correctamente.

**Verdaderos Negativos (VN):** Son los números de píxeles pertenecientes al fondo clasificados como tales.

**Falsos Positivos (FP):** Son los números de píxeles del fondo clasificados como parte de la imagen.

**Falsos Negativos (FN):** Son los números de píxeles pertenecientes a la imagen clasificados como parte del fondo.

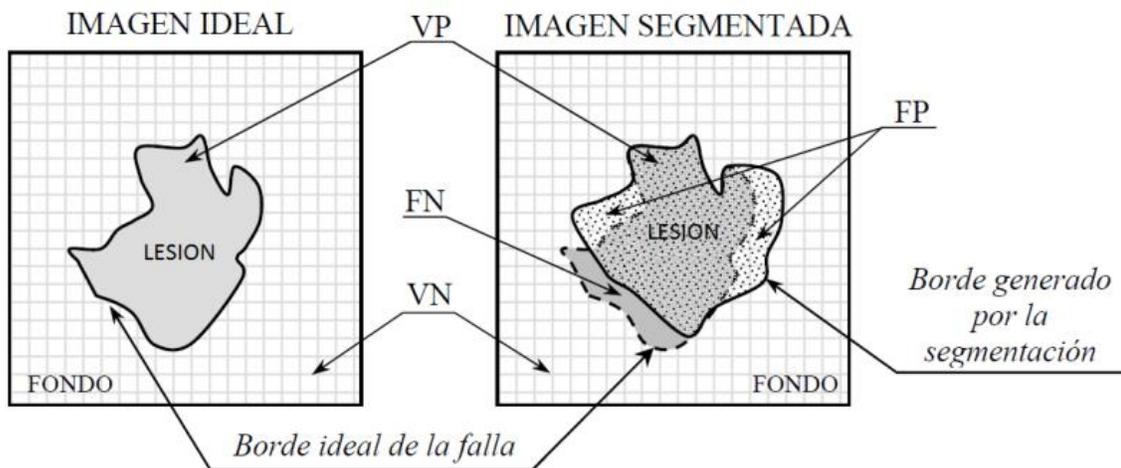


Figura 17: Ejemplo de variables de evaluación (31).

La Precisión (P), definida por la ecuación (48) nos dará una idea de la sobre-segmentación en la segmentación obtenida con el algoritmo, respecto a la segmentación verdadera, tomando valor 1, cuando no existen zonas de falsos positivos (sobre-segmentadas), y valores cercanos a 0, en caso contrario (31).

$$P = \frac{VP}{VP + FP} \quad (36)$$

La Exactitud (E), a diferencia de la precisión se diferencia en que la precisión toma su valor máximo 1 en caso de que la segmentación automática de la imagen este completamente contenida dentro de la segmentación ideal, y la exactitud toma su valor máximo 1 en caso que la segmentación realizada por el algoritmo coincida plenamente con la realizada por el experto mediante la ecuación 37 (31):

$$E = \frac{VP + VN}{VP + VN + FP + FN} \quad (37)$$

Un ejemplo de lo antes dicho se puede ver en esta imagen donde para el caso de la Precisión la segmentación automática queda contenida en la lesión verdadera por lo que la Precisión nos dará su máximo valor como una aproximación a la segmentación correcta.

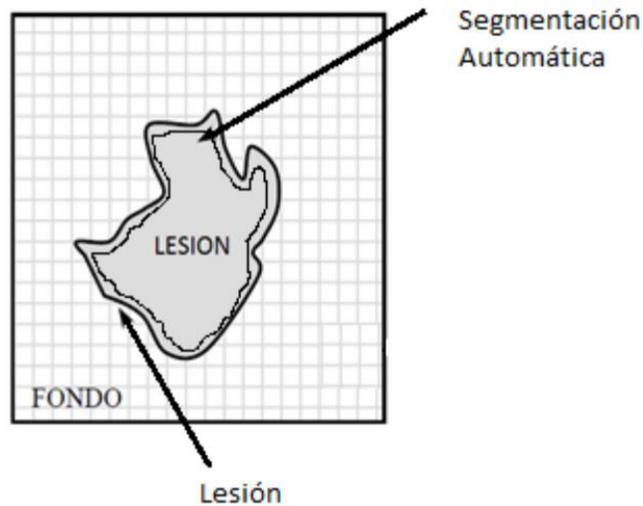


Figura 18: Ejemplo de precisión (31).

Mientras que, la Exactitud tomará su mayor valor cuando realmente coincida con el contorno de la lesión.

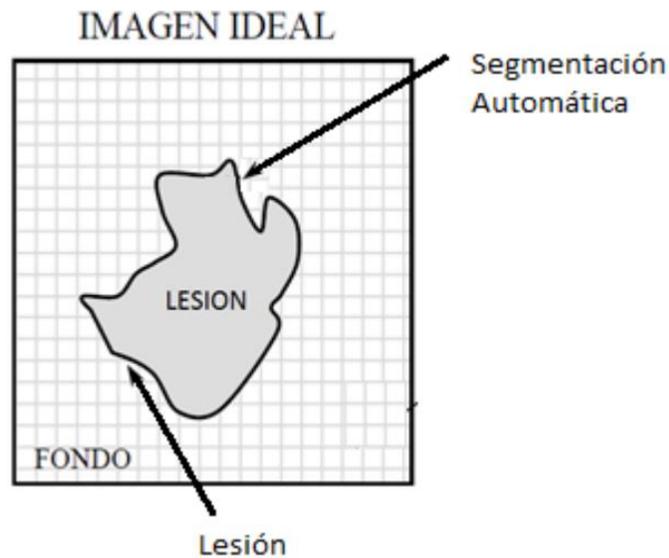


Figura 19: Ejemplo de exactitud (31).

Otra de las medidas estadísticas de error es la Revocación (R) la cual nos dará cuenta de la sub-segmentación que existe en la imagen segmentada con respecto a la segmentación verdadera. Tomando valor máximo 1 cuando no existen zonas de falsos negativos (sub-segmentadas) y valores cercanos a 0, en caso contrario. Esta medida devuelve el máximo valor en caso de que la segmentación sea perfecta, es decir, que coincida plenamente con la realizada por el experto. Puede ser descrita por la ecuación 38 (31).

$$R = \frac{VP}{VP + FN} \quad (38)$$

El Promedio Señal a Ruido (NSR) se expresa mediante la ecuación 39. Esta expresión es matemáticamente igual a la división de los Falsos Positivos con los Verdaderos Positivos ( $FP/VP$ ) (30).

$$NSR = \frac{n(B \setminus A)}{n(B \cap A)} \quad (39)$$

Donde,  $A$  representa a la segmentación ideal realizada por el especialista,  $B$  la segmentación automática, y  $n(X)$  la cantidad de pixeles de  $X$ .

La Medida FM o Medida de Primer Plano es otra de las medidas estadísticas de error y depende plenamente de la Precisión y la Revocación tomando su valor máximo 1 en caso de que la segmentación sea perfecta. Puede ser descrita por la ecuación 40 (31).

$$FM = \frac{2 * P * R}{P + R} \quad (40)$$

A diferencia de las medidas anteriores la Tasa de Error (ER) descrita por la ecuación devuelve 0 en caso de que la segmentación sea perfecta, lo que indica que no existe error en la segmentación, ya que la segmentación manual y la automática coinciden plenamente. La ecuación 41 la describe (31).

$$ER = \frac{FP}{VP + FN} \quad (41)$$

La Tasa negativa (NRM), la cual esta expresada por la ecuación (54), al igual que la Tasa de Error devolverá 0 en caso de que la segmentación sea perfecta, esta expresión se describe por la ecuación 42 (31),

$$NRM = \frac{NR_{FN} + NR_{FP}}{2} \quad (42)$$

Donde

$$NR_{FN} = \frac{FN}{VP + FN} \quad (43)$$

Y

$$NR_{FP} = \frac{FP}{VP + FN} \quad (44)$$

### 1.5.1.2 Medidas de localización

Las medidas de localización son aquellas que se basan en el cálculo de la distancia mínima entre un punto y un conjunto de puntos, es decir, cumplen con la condición siguiente:

Si  $\rho(x,y)$  es la distancia entre dos pixeles  $x$  e  $y$ , entonces la distancia más corta entre un pixel  $x \in X$  y  $A \subseteq X$  se define como (30):

$$d(x, A) = \inf\{p(x, a) : a \in A\} \quad (45)$$

Donde  $X$  es el conjunto de pixeles de la imagen y  $A$  es el conjunto de pixeles de valor 1 (binario) en la imagen. Entre las medidas de localización se encuentran las medidas de Peli y Malah, estas utilizan dos criterios estadísticos característicos de la detección de errores, los cuales son el error medio (Media) y el error cuadrático medio (Varianza) (32) (30): Estas dos medidas están definidas por las ecuaciones 46 y 47.

$$Media = \frac{1}{n(B)} * \sum_{x \in B} d(x, A) \quad (46)$$

$$Varianza = \frac{1}{n(B)} * \sum_{x \in B} d(x, A)^2 \quad (47)$$

Donde  $d(x, A)$  es la distancia entre el  $i$ -ésimo pixel de la segmentación automática  $B$  y el pixel más cercano en el contorno de prueba  $A$  y  $n(B)$  es la cantidad de pixeles de la imagen binaria  $B$ .

La Media como medida de localización nos dará una idea de la distancia media que existe entre los pixeles del contorno de prueba y los de la segmentación. Puede darse el caso en el que una segmentación tenga pixeles muy cercanos y otro grupo de pixeles muy lejanos, o sea, que exista dispersión en los pixeles y aun así obtener una buena medida. Sin embargo, esto no pasa con la Varianza, ya que penaliza los pixeles más alejados haciendo que la medida de similitud crezca mucho si hay pixeles muy dispersos. Como consecuencia de esto se puede afirmar que si una segmentación aporta un buen valor para la media y uno malo para la varianza entonces los pixeles de la misma están dispersos, por lo que no es una buena segmentación (32).

Otra de las medidas de localización es la Figura de Merito de Pratt (FOM), definida por la ecuación 48 (30):

$$FOM(A, B) = \frac{1}{\max\{n(A), n(B)\}} * \sum_{x \in B} \frac{1}{1 + \alpha * d(x, A)^2} \quad (48)$$

Donde alfa es una constante de escala con valor  $1/9$ ,  $A$  (segmentación ideal),  $B$  (prueba),  $n(A)$  y  $n(B)$  cantidad de pixeles de  $A$  y  $B$  respectivamente y  $d(x, A)$  distancia de un pixel en la imagen  $B$  al pixel más cercano en  $A$ .

FOM es una medida para errores de Falsos Positivos y no es sensible a errores de Falsos Negativos. Lo mismo ocurre con el error medio y cuadrático medio, que además son muy sensibles al ruido de fondo ya que si el error de la imagen  $B$  contiene incluso un solo pixel muy distante de  $A$ , su valor de la distancia elevará drásticamente la distancia media (30).

### 1.5.1.3 Métrica Hausdorff

Dentro de la métrica Hausdorff, la distancia de Hausdorff ( $h$ ) de un contorno segmentado ( $CS$ ) a un contorno de referencia ( $CR$ ) se define por la ecuación 49 (33) (30) (28):

$$h(CS, CR) = \max[\sup d(cs, CR), \sup d(cr, CS)] \quad (49)$$

Donde  $cs$  y  $cr$  son puntos de los contornos  $CS$  y  $CR$  respectivamente, y  $d(cs, cr)$  es cualquier métrica entre estos puntos, por su simplicidad se utiliza frecuentemente la distancia euclidiana.

La distancia de Hausdorff permite obtener una buena medida de similitud entre dos imágenes, pero en la práctica no suele usarse como medida de error, pues es muy sensible al ruido, ya que un solo pixel erróneo puede causar que el valor de  $h(CS, CR)$  se eleve hasta el máximo posible.

La distancia de Baddeley es una medida alternativa a la distancia de Hausdorff para imágenes binarias, donde, Baddeley propone reemplazar el operador Max(máximo) por un valor medio para reducir la sensibilidad al ruido. Esta distancia se describe por la ecuación 50 (34) (30):

$$\Delta^p(A, B) = \left[ \frac{1}{N} \sum_{x \in X} |d(x, A) - d(x, B)|^p \right]^{1/p} \quad (50)$$

Donde  $X$  es un dominio común de dos resultados de segmentación  $A$  y  $B$ ,  $N$ = número total de píxeles en el conjunto  $X$  y  $p > 1$ , teniendo en cuenta que cuando  $p$  se acerca al infinito la distancia de Baddeley se acerca a la distancia de Hausdorff.

## **1.6 Herramientas utilizadas durante el desarrollo**

A continuación se muestran las herramientas utilizadas para la realización de las tareas planteadas.

### **1.6.1 Entorno de desarrollo**

Para la realización de pruebas de diversos algoritmos y técnicas sobre la imagen oblicua, se utilizó la herramienta de software MATLAB (abreviatura de MATrix LABoratory, "laboratorio de matrices"). MATLAB, en su contenido básico, es un entorno integrado de trabajo que permite el análisis y la computación matemáticos interactivos de forma sencilla con más de 500 funciones matemáticas, estadísticas y técnicas implementadas, así como la visualización a través de todo tipo de gráficos de ingeniería y científicos (35). Además, el entorno básico de MATLAB se complementa con una amplia colección de cajas de herramientas que contienen funciones específicas para determinadas aplicaciones en diferentes ramas de las ciencias y la ingeniería. Se hizo énfasis específicamente en el área del procesamiento de imágenes MATLAB (35).

### **1.6.2 Lenguaje de programación**

Como lenguaje de programación en el sistema se utilizó Matlab o también conocido como lenguaje M. Matlab permite a la hora de programar una serie de elementos típicos para la modificación del flujo de una secuencia de instrucciones (36). La sintaxis es muy parecida a la de cualquier lenguaje de programación. Todos estos operadores se pueden usar en la ventana de comandos, en línea, o en un fichero .m (36). La programación se lleva a cabo mediante un lenguaje que es muy parecido a lenguajes de alto nivel como BASIC o C (36). De este modo se ahorra tiempo y esfuerzo en sucesivas sesiones pues no es necesario escribir todas las sentencias de nuevo (37).

### **1.6.3 Herramienta para el modelado**

Como herramienta para el modelado de la solución se ha seleccionado Visual Paradigm for UML (VP UML v8.0), una de las herramientas de Ingeniería de Software Asistida por Computadora (CASE, según sus siglas en inglés). VP-UML v8.0 soporta los principales estándares de la industria tales como el Lenguaje de Modelado Unificado (UML, según sus siglas en inglés), SysML, BPMN, XMI, entre otros. Ofrece un conjunto completo de herramientas, brindando a los equipos de desarrollo de software todo lo necesario para la captura de requisitos, planificación de software, planificación de controles, modelado de clases y modelado de datos (38).

#### **1.6.4 Metodología de desarrollo**

El proceso de desarrollo de software, es definido como el conjunto de actividades necesarias para transformar los requisitos de un usuario en un sistema de software, tiene como finalidad la obtención de un producto que cumpla con las expectativas del cliente (39).

Según la filosofía de desarrollo se pueden clasificar las metodologías en dos grupos. Las metodologías tradicionales, que se basan en una fuerte planificación durante todo el desarrollo, y las metodologías ágiles, en las que el desarrollo de software es incremental, cooperativo, sencillo y adaptado.

Luego de la evaluación de las metodologías de desarrollo y el estudio de sus características, etapas de desarrollo y ventajas que posibilitan. Se determina utilizar una metodología ágil, dado que la prioridad es satisfacer al cliente mediante tempranas y continuas entregas de software. El cliente es parte del equipo de desarrollo, el equipo de desarrollo consta de dos personas. Además de la dificultad para un equipo de desarrollo pequeño el optar por una metodología robusta, a causa de la cantidad de documentación generada y la alta resistencia a los cambios durante el desarrollo. Lo cual permitió identificar la Programación Extrema (XP) como una alternativa acertada, además de ser la metodología de desarrollo del grupo de investigación (40) (39).

Fue seleccionada por el grupo de investigación AIRI (Artificial Intelligence: Research and Innovation) para el desarrollo de los proyectos del mismo. Resulta apropiada para proyectos con requisitos imprecisos, cambiantes, en los que existe un alto riesgo técnico y por la facilidad de desarrollo que propone la metodología.

Ventajas de XP (39):

- Comienza en pequeño y añade funcionalidad con retroalimentación continua.
- El manejo del cambio se convierte en parte sustantiva del proceso.
- El costo del cambio no depende de la fase o etapa.
- El cliente o el usuario se convierte en parte del equipo.

XP consta de 4 fases (41):

**Planificación:** Es la fase donde los desarrolladores y clientes establecen los tiempos de implementación ideales de las historias de usuario, la prioridad con la que serán implementadas y las historias que serán implementadas en cada iteración.

**Diseño:** La metodología XP hace especial énfasis en los diseños simples y claros. Por ello XP propone implementar el diseño más simple posible que funcione. Se sugiere nunca adelantar la implementación de funcionalidades que no correspondan a la iteración en la que se esté trabajando.

**Codificación:** En la fase de codificación de desarrolla en función de cada historia de usuario, además de ser fase donde se definen las tareas de la ingeniería y los tiempos reales en que se realizaron cada una de

las funcionalidades especificadas, en la cual la implementación, debe realizarse de acuerdo los estándares de codificación.

**Pruebas:** Estas pruebas se realizan al final del ciclo en el que se desarrollan, para verificar que las iteraciones no han afectado a las anteriores. Las pruebas de aceptación que hayan fallado en el ciclo anterior son analizadas para evaluar su corrección, así como para prever que no vuelvan a ocurrir.

## **1.7 Conclusiones del capítulo**

En el presente capítulo se realizó un estudio sobre procesamiento de imágenes digitales, fundamentalmente en el campo de la umbralización de imágenes de jamón de la empresa Bravo S.A. Evidenciándose la necesidad de informatizar este proceso para aumentar la calidad del producto confeccionado. Se describen las principales etapas y conceptos del procesamiento digital de imágenes, destacando la importancia del preprocesamiento de imágenes. Además, se pudo profundizar en los principales métodos de segmentación de imágenes llegando a la conclusión que no existe un método de segmentación que alcance resultados aceptables para todo tipo de imagen de cárnicos, por lo que la selección del método apropiado puede ser muy difícil y tediosa. Durante el desarrollo del capítulo se seleccionaron y describieron las herramientas y tecnologías a ser usadas en la propuesta de solución. Se decidió que la metodología a utilizar en el desarrollo, fuese la metodología XP, definida por el grupo de investigación y por su flexibilidad para el proceso de desarrollo.

## **2 Capítulo 2: Propuesta de solución**

En el capítulo se expone la propuesta de solución para detectar y realzar las zonas pertenecientes a grasas en las imágenes de jamón provenientes de la línea productiva de la empresa Bravo S.A. Se describe la fase inicial de la metodología XP: Planificación, obteniendo los artefactos importantes de esta fases, tales como, las Historias de Usuarios, Plan de Iteraciones, Plan de Duración de Iteraciones y Plan de Entregas. También durante el desarrollo del capítulo se describe la fase de Diseño basándose fundamentalmente en el desarrollo de las tarjetas Clase-Responsabilidad-Colaboración (CRC).

### **2.1 Descripción de la solución**

Dada la extracción de los requisitos funcionales y la investigación realizada sobre las principales vertientes del procesamiento de imágenes digitales en cuanto a segmentación, así como la aplicación de numerosos métodos y evaluación de sus desempeños en contexto llevaron a la realización de la siguiente solución. Mostrándose en un diagrama de actividades el cual sigue una secuencia lógica que acompaña la imagen dada como entrada del usuario a través de los procesos necesarios para la extracción de datos requeridos.

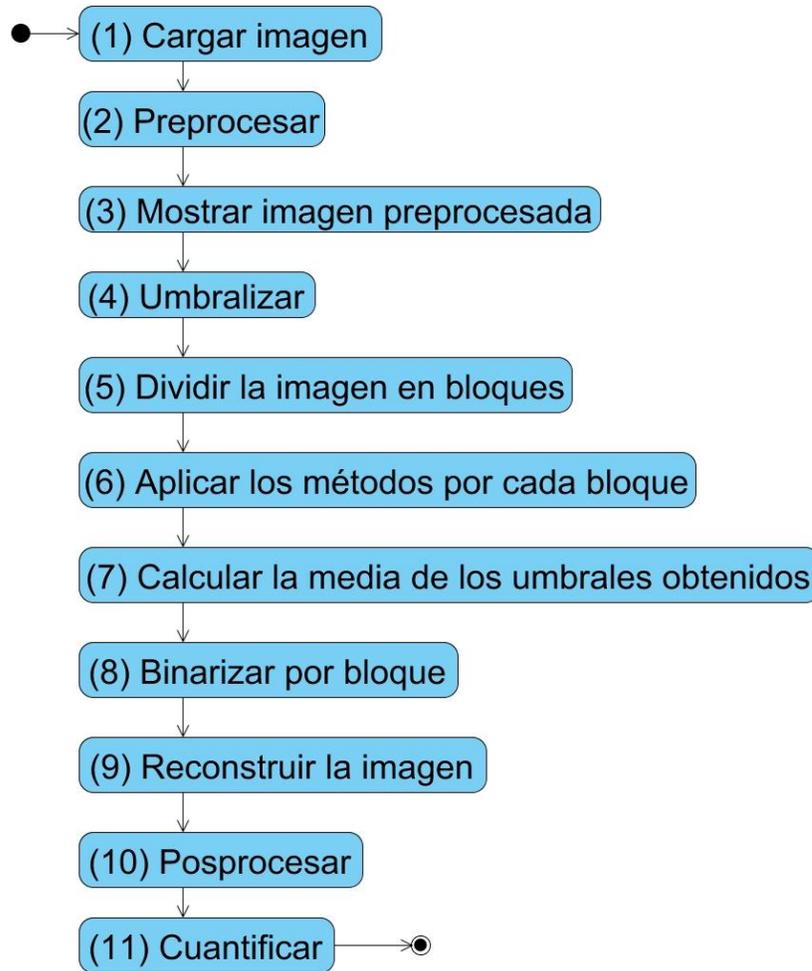


Figura 20: Propuesta de solución representada en un diagrama de actividades.

**Paso 1.** Carga la imagen tomada por el dispositivo de captura, al software

**Paso 2.** Se modifica el espacio de color RGB nativo por la interpretación YCbCr con objetivo de aislar el canal de luminosidad sobre el cual se ejecutan los procesos, luego se aplica el algoritmo Imadjust, el cual mapea los valores de intensidad en escala de grises saturando altas y bajas intensidades, realizando contraste, luego es aplicado el algoritmo de “Canny” para detectar bordes y reducción de ruido y fusionado con la imagen tratada para diferenciar claramente las regiones críticas

**Paso 3.** Es mostrada en pantalla la imagen resultante.

**Paso 4.** Se ejecuta la selección de métodos (ver paso 6) sobre la imagen procesada obteniendo un umbral base el cual se usa como referencia en la obtención de los próximos

**Paso 5.** Se secciona la imagen procesada en la cantidad de bloques definidas por el usuario con el fin de lograr un umbralado dinámico

**Paso 6.** Es aplicado a cada bloque (sub-imagen) los métodos de Kashmann, Huang y Tsai obteniendo los umbrales respectivos por cada bloque

**Paso 7.** Es calculada la media entre los umbrales arrojados siendo la media el umbral obteniendo ajustado a cada bloque

**Paso 8.** Cada bloque es binarizado usando su umbral correspondiente

**Paso 9.** Es reconstruida la imagen general uniendo las sub-imágenes obtenidas ya binarizadas

**Paso 10.** El área segmentada es realzada y superpuesta en la imagen original para una mejor apreciación por parte del usuario

**Paso 11.** El área segmentada es cuantificada con respecto a la imagen original obteniendo el porcentaje de grasas.

### **2.1.1 Pseudocódigo de la clase gestora principal**

cargar imagen

nombre\_archivo, ruta= explorador\_archivos

si explorador\_archivos es vacío

mostrar mensaje de error

sino

mostrar barra de carga

fin

variable global imagen=leer(nombre de archivo, ruta)

datos=características(imagen)

mostrar datos

graficar imagen

variable global auxiliar = imagen;

fin

Procesar

inicializo cantidad de niveles de color de la imagen

barra de carga 0%

lab = conversión de espacio de color de (imagen)b= selección de luminancia de YCbCr de(lab)

mejora=ajuste de contraste en (b)

bordes= método de borde de canny en (b)

mejora = bordes en mejora

imagen = mejora

barra de carga 20%

variable global procesada = mejora

variable global flag = 1

mostrar (procesada)

```

M, N = tamaño(imagen)
cantidad_bloques= valor
imagen_segmentada, umbral = umbralizar en (imagen)
parametro = umbral
imagen_segmentada_bloques= umbralizacion_x_bloques en (imagen usando:cantidad_bloques,
umbralizar, cantidad_de_niveles)
variable global imagen_segmentada
barra de carga 70%
binaria= imagen_segmentada_bloques > 0
mascara = lab en canal luminancia
mascara -> binaria = blanco
lab -> luminancia = mascara
imagen_rgb= (lab) de YCbCr a RGB
para i de 0 hasta M
para j de 0 hasta N
si imagen_rgb en (i,j) es blanco
imagen_rgb en (i,j) es verde
fin
fin
barra de carga 100%
mostrar_en_tabla(umbral_huang)
mostrar_en_tabla(umbral_khashmann)
mostrar_en_tabla(umbral_tsai)
mostrar_en_tabla(umbral_medio)
mostrar imagen_rgb
datos_grasa= propiedades (imagen segmentada_en_bloques )
caracteristicas= datos_grasa->obtener_area
area_grasa= suma [caracteristicas]
area_imagen= M*N
porciento= area_grasa/area_imagen*100
mostrar porciento

```

## 2.2 Selección de los métodos de preprocesado

A continuación se detalla el preprocesamiento realizado a la imagen de muestra de jamón. Descritos en el orden de la solución propuesta.

## Espacio de color YcbCr

Una vez cargada la imagen de disco, se procede con su preprocesado, comenzando con la selección del espacio de color YCbCr. Este no representa un espacio de color absoluto, dado que es una forma de codificar información RGB. Es utilizado en sistemas de video y fotografía digital. Fue seleccionado principalmente por las posibilidades que brinda el canal Y. La cual representa la componente de luminosidad y las señales Cb y Cr son los componentes de crominancia, diferencia de azul y diferencia de rojo, respectivamente.



*Figura 21: Imagen obtenida luego de seleccionar el canal Y del espacio de color YCbCr.*

La componente de luminosidad o claridad (propiedad de los colores) como se muestra en la figura 21, permite obtener de la imagen de muestra de jamón, una mayor apreciación de las zonas grasas en la imagen. En términos generales, es algo muy similar a la versión en blanco y negro de la imagen original (42).

## Algoritmo Imadjust

Como segundo paso de preprocesado se le aplica a la imagen el algoritmo de Imadjust. El cual mapea los valores de intensidad en una imagen en escala de grises a nuevos valores, tal que el 1% de los datos es saturado a bajas y altas intensidades. Mejora el contraste de la imagen resultante.



*Figura 22: Imagen obtenida luego de aplicar el algoritmo Imadjust.*

Luego de aplicarle el algoritmo *Imadjust* a la imagen en escala de grises (figura 21), se obtiene el resultado mostrado en la figura 22. Donde se evidencia un realce del contraste de la imagen, quedando realzadas en la imagen las zonas correspondientes a grasas.

### Operador de Canny

Finalmente culmina el preprocesado aplicando el algoritmo de Canny. Este posee como principal ventaja su gran adaptabilidad para poder ser aplicado a diversos tipos de imágenes. Posee múltiples etapas para detectar una amplia gama de bordes. Además de no disminuir su performance ante la presencia de ruido y encontrar la intensidad del gradiente en la imagen original (43).



Figura 23: Imagen obtenida luego de aplicar el algoritmo de Canny.

Se muestra en la figura 23 el resultado de aplicarle el algoritmo de Canny a la imagen de la figura 22. Quedando realzados los bordes de las zonas correspondientes a grasas. A fin de lograr una imagen preparada para sus posteriores procesos.

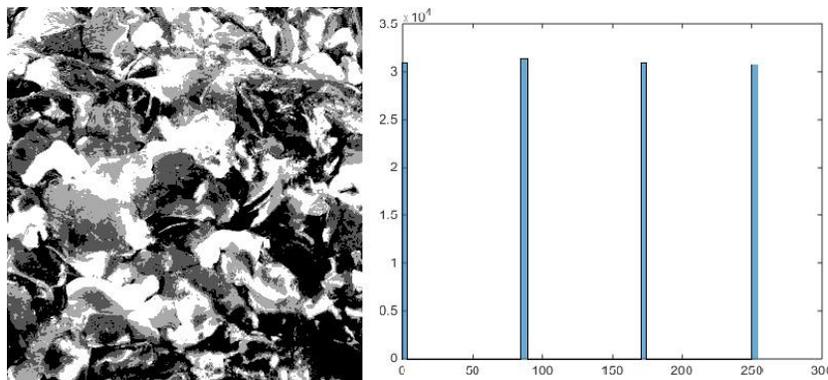


Figura 24: Resultado del preprocesamiento, mostrada a 4 niveles de grises con su histograma.

La imagen anterior muestra el desglose de los píxeles de la imagen agrupados en 4 niveles de intensidad luego de aplicados los métodos de preprocesado. Siendo apreciable la separación de las zonas cercanas a 255 o blanco, demostrando la objetividad y utilidad de dichos métodos en tributo a la solución final, habiendo mejorado la imagen en pos de una segmentación eficaz.

### 2.3 Selección de los métodos de umbralización

En la tabla siguiente se muestran los resultados de la segmentación mediante las métricas de precisión y exactitud de los diferentes métodos analizados.

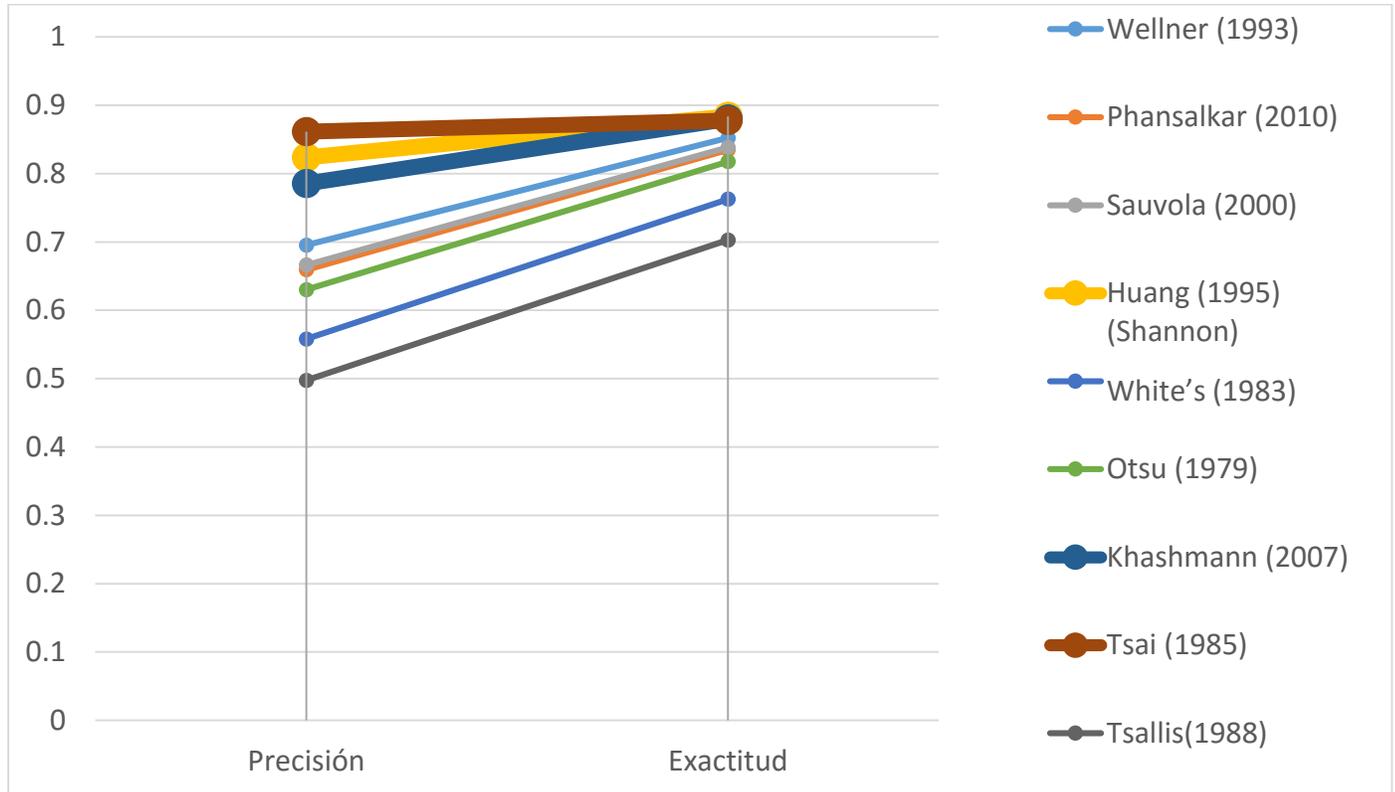


Grafico 1: Resultados de las pruebas realizadas utilizando métricas (precisión y exactitud).

Se tomaron un grupo de imágenes de muestra de jamón a las cuales se le aplicaron los métodos estudiados. Posteriormente fueron comparados los resultados con el mismo grupo de imágenes segmentadas manualmente por el departamento de calidad de la empresa Bravo S.A. Siendo visible la superioridad de los métodos Tsai, Huang y Khashmann. Al ser tres resultados relevantes y diferentes, se procede a calcular la media, mostrando esta un resultado más exacto y preciso. En el próximo capítulo se realiza la validación de la solución propuesta.

### 2.4 Fase de Planeación

La fase de planeación es la etapa inicial del desarrollo de software de la metodología XP. En este punto comienza a interactuar con el cliente para identificar cuáles son las historias de usuario (HU). Donde se definen el número y tamaño de las historias de usuario. Se plantean los ajustes necesarios a la metodología según las características del proyecto y el cliente definen el nivel de prioridad de las historias de usuario. Como el tiempo y el esfuerzo que conllevarán su desarrollo.

### **2.4.1 Requisitos del sistema**

Los requisitos se deben descubrir antes de empezar a construir un producto, y puede ser algo que el producto debe hacer o una cualidad que el producto debe tener. Un requisito existe ya sea porque el tipo de producto demanda ciertas funciones o cualidades, o porque el cliente quiere que ese requerimiento sea parte del producto final. Así que si no se tienen los requerimientos correctos, no se puede diseñar o construir el producto correcto y, consecuentemente, el producto no permitirá a los usuarios finales realizar su trabajo.

Requisitos funcionales

RF#1. Cargar imagen.

RF#2. Selección de bloques.

RF#3. Procesado.

RF#3.1.1 Cambio de espacio de color a YCbCr.

RF#3.1.2 Aplicación de filtro de realce de intensidad.

RF#3.1.3 Aplicación de filtro de realce de bordes (Canny).

RF#3.1.4 Extracción del canal de luminancia independiente.

RF#3.2 Dividir la imagen en la cantidad de bloques seleccionados.

RF#3.3 Umbralado mediante método de Khashmann.

RF#3.4 Umbralado mediante método de Tsai.

RF#3.5 Umbralado mediante método de Huang.

RF#3.6 Dividir la imagen en la cantidad de bloques seleccionados.

RF#3.7 Binarizar.

RF#3.8 Reconstruir imagen.

RF#3.9 Cuantificar.

RF#4. Mostrar resultados.

#### **2.4.1.1 Requisitos no funcionales**

##### **Requisitos de hardware:**

Las capacidades del cómputo deben ser superior a:

1. 2 GB de memoria RAM
2. Cualquier procesador Intel o AMD x86-64
3. Espacio en disco de 2 GB
4. Se recomienda la tarjeta gráfica acelerada por hardware compatible con OpenGL 3.3 con memoria GPU de 1GB.

#### **2.4.2 Historias de Usuarios**

Las historias de usuario son utilizadas como herramientas para dar a conocer los requerimientos del sistema, las mismas son pequeños textos en el que el cliente describe una actividad del sistema. Las

historias de usuario solo muestran la silueta de una tarea o requisito. Estimar el tiempo que tomara para realizar las entregas, en cada entrega se puede desarrollar una o varias historias de usuarios en dependencia de la complejidad de las mismas (44).

Tabla 1: Historia de usuario # 2

Historia de Usuario	
<b>Número:</b> 2	<b>Nombre de la Historia de Usuario:</b> Filtro de realce de bordes
<b>Modificación a la Historia de Usuario:</b> Ninguna	
<b>Usuario:</b> Empresa Bravo S.A.	<b>Iteración asignada:</b> 1
<b>Prioridad en negocio:</b> media	<b>Puntos estimados:</b> 1 semanas
<b>Riesgo en desarrollo:</b> medio	<b>Puntos reales:</b> 1 semanas
<b>Programador responsable:</b> Celso	
<b>Descripción:</b> Se aplica un filtro de realce de bordes a la imagen cargada por el usuario.	
<b>Observaciones:</b> La imagen debe estar en RGB	

Tabla 2: Historia de usuario # 3

Historia de Usuario	
<b>Número:</b> 3	<b>Nombre de la Historia de Usuario:</b> Filtro de ajuste de la intensidad del mapa de color de la imagen.
<b>Modificación a la Historia de Usuario:</b> Ninguna	
<b>Usuario:</b> Empresa Bravo S.A.	<b>Iteración asignada:</b> 1
<b>Prioridad en negocio:</b> alta	<b>Puntos estimados:</b> 1 semanas
<b>Riesgo en desarrollo:</b> alta	<b>Puntos reales:</b> 1 semanas
<b>Programador responsable:</b> Adolfo	

**Descripción:** Se aplica un filtro de ajuste de valores de intensidad del mapa de color de la imagen, obteniéndose una imagen con su histograma extendido.

**Observaciones:** Debe aplicarse sobre los valores de intensidad del mapa de color.

### 2.4.3 Planificación de la Entrega

En la fase de la planificación de entrega se definen las prioridades de cada historia de usuario, y consecuentemente se realiza una estimación del esfuerzo necesario de cada una de ellas. Se definen las entregas con el cliente donde se tiene por norma que una entrega debe adecuarse a tres meses como máximo. Para el desarrollo satisfactorio de la solución propuesta, se realizó una estimación del esfuerzo para cada una de las HU, arrojando los siguientes resultados:

*Tabla 3: Puntos de estimación por historias de usuario*

No	Historias de usuario	Puntos de estimación(semanas)
1.	Cargar imagen	1
2.	Filtro de realce de bordes	1
3.	Filtro de ajuste de la intensidad del mapa de color de la imagen.	1
4.	Detector de bordes basado en la segunda derivada (Canny).	2
5.	Cambio de espacio de color a YCbCr	2
6.	Seleccionar canal de luminancia	1
7.	Umbralado mediante método de Huang's (1995)	3
8.	Umbralado mediante método de Khashman (2007)	3
9.	Umbralado mediante método de Tsai (1995)	3
10.	Dividir la imagen en bloques	1
11.	Binarizar	1
12.	Reconstruir Imagen	1
13.	Calculo de saturación de grasas	1

14.	Visualizar resultados	1
-----	-----------------------	---

#### 2.4.4 Plan de iteraciones

Luego de definir las historias de usuarios e identificar el tiempo y el esfuerzo para el desarrollo de cada una de ellas, se propone un plan de iteraciones donde las historias de usuario están contenidas. El plan se propone, regulando el rango establecido para cada una de ellas; teniendo en cuenta todos los aspectos antes tratados, se pretende realizar el desarrollo en 3 iteraciones, referidas a continuación:

##### Iteración 1

La iteración tiene como finalidad realizar el preprocesado de la imagen mediante algunas técnicas de filtrado espaciales y de histograma. También abarca los cambios de espacio de color de la imagen para obtener el canal de la luminancia.

##### Iteración 2

Esta iteración se encarga de aplicar las técnicas de umbralado para la segmentación de la imagen.

##### Iteración 3

En esta iteración se define la media de los valores de umbralado para binarizar la imagen por secciones (sub-imágenes), se reconstruye la imagen en una máscara única para la realización del cálculo y muestra los resultados al usuario.

Tabla 4: Plan de duración de las iteraciones.

Iteraciones	Historias de usuario	Duración
Iteración 1	Cargar imagen	8
	Filtro de realce de bordes	
	Filtro de ajuste de la intensidad del mapa de color de la imagen.	
	Detector de bordes basado en la segunda derivada (Canny).	
	Cambio de espacio de color a YCbCr	
	Seleccionar canal de luminancia	
Iteración 2	Umbralado mediante método de Huang's (1995)	9
	Umbralado mediante método de Khashman (2007)	
	Umbralado mediante método de Tsai (1995)	
Iteración 3	Dividir la imagen en bloques	5
	Binarizar	
	Reconstruir Imagen	

	Calculo de saturación de grasas	
	Visualizar resultados	

### 2.4.5 Plan de entrega

El plan de entregas tiene como objetivo definir el número de liberaciones que se realizarán en el transcurso del proyecto y las iteraciones que se requieren para desarrollar cada una.

Tabla 5: Plan de entrega.

	Iteración 1	Iteración 2	Iteración 3
Cantidad de HU	8	9	5
Fecha de entrega	20/02/2016	27/04/2016	03/06/2016

## 2.5 Diseño

La metodología XP sugiere que hay que conseguir diseños simples y sencillos. Hay que procurar hacerlo todo lo menos complicado posible para conseguir un diseño entendible y fácil de implementar que a la larga costará menos tiempo y esfuerzo desarrollar.

Define que se debe utilizar una correcta especificación de los nombres de métodos y clases ayudará a comprender el diseño y facilitará sus posteriores ampliaciones y la reutilización del código (41) (45).

### 2.5.1 Tarjetas CRC

La metodología XP en lugar de utilizar diagramas para desarrollar modelos, representa las clases mediante tarjetas CRC (Clase, Responsabilidad y Colaboración). Ayudan a definir actividades durante el diseño del sistema.

Estas tarjetas representan una entidad del sistema, a la cual asignar responsabilidades y colaboraciones. El formato físico de las tarjetas CRC facilita la interacción entre los participantes del proyecto, en sesiones en las que se aplican técnicas de grupos como tormenta de ideas o juego de roles, y se ejecutan escenarios a partir de la especificación de requisitos, historias de usuarios y casos de uso. De esta forma, van surgiendo las entidades del sistema junto con sus responsabilidades y colaboraciones (46).

Estas tarjetas permiten trabajar con una metodología basada en objetos, permitiendo que el equipo de desarrollo completo contribuya en la tarea del diseño. El nombre de la clase se coloca a modo de título en la tarjeta, las responsabilidades se colocan a la izquierda y las clases que se implican en cada responsabilidad a la derecha, en la misma línea que su requerimiento correspondiente.

A continuación se muestra en la tabla 6, una de las tarjetas CRC donde están presentes sus funcionalidades y las clases que colaboran con ella.

Tabla 6: Tarjeta CRC # 1 (Clase Principal).

Tarjeta CRC	
Clase: AlgoritmoUmbralizado	
Responsabilidades	Colaboradores
<ol style="list-style-type: none"> <li>1. Carga la imagen.</li> <li>2. Aplica filtros de preprocesado.</li> <li>3. Muestra la imagen a procesar.</li> <li>4. Definir tamaño de bloque.</li> <li>5. Aplicar Threshold en toda la imagen.</li> <li>6. Aplicar BlockThreshold</li> </ol>	<ol style="list-style-type: none"> <li>1. imread.</li> <li>2. imadjust</li> <li>3. edge.</li> <li>4. imsharpen.</li> <li>5. rgb2ycbr.</li> <li>6. lab.</li> <li>7. histeq.</li> </ol>

### 2.5.2 Estándares de codificación

Un estándar de codificación completo comprende todos los aspectos de la generación de código. Un código fuente completo debe reflejar un estilo armonioso, como si un único programador hubiera escrito todo el código de una sola vez. Al comenzar un proyecto de software, se establece un estándar de codificación para asegurarse de que se trabaje de forma coordinada. Cuando el proyecto de software incorpore código fuente previo, o bien cuando realice el mantenimiento de un sistema de software creado anteriormente, el estándar de codificación debería establecer cómo operar con la base de código existente (47).

- Utilizar los comentarios para explicar qué hace el código.
- Usar nombres significativos para las clases, métodos, propiedades y variables.
- Los métodos de clases, se definen de forma que la primera letra debe ser minúscula, si son varias palabras se debe de intercalar entre minúsculas y mayúsculas.
- Se define una declaración por línea.
- Se evita asignar el mismo valor a varias variables en la misma sentencia, pues dificulta su lectura.

## 2.6 Conclusiones del capítulo

En el desarrollo de este capítulo se describió la propuesta de solución, enmarcada en el procesamiento digital de imágenes de jamón, específicamente en las etapas del preprocesamiento y la segmentación de imágenes. Se logró identificar las técnicas a utilizar dentro del preprocesamiento, desglosándose en la selección del espacio de color YCbCr, el filtro desarrollado por John F. Canny en 1986 y algoritmo Imadjust. En el caso de la segmentación, se puede observar que la combinación de 3 algoritmos aplicados a pequeños

bloques de la imagen, la cual devuelve un resultado preciso para detectar grasas en las imágenes de jamón. Se utiliza la metodología XP para realizar el análisis y diseño de la aplicación, generando como artefactos un total de 14 HU que describen los aspectos principales a tener en cuenta para el desarrollo de la solución. También se logra traducir los requisitos funcionales a implementar, usando las tarjetas CRC que propone la fase de diseño de la metodología.

### 3 Capítulo 3: Implementación y Pruebas

Durante el desarrollo del presente capítulo se estará ilustrando las fases de implementación y pruebas que propone la metodología XP. Generando como artefacto en el caso de la fase de implementación las tareas de desarrollo que dan solución a cada una de las historias de usuarios identificadas en la fase de planificación. También se realizan un conjunto de pruebas al sistema, divididas en pruebas unitarias y de aceptación como propone la metodología, finalmente se realiza la validación de los resultados obtenidos utilizando métricas de segmentación.

#### 3.1 Fase de implementación

Dentro del proceso de desarrollo de la metodología del desarrollo del software, la parte más importante es la implementación. Se especifica en esta fase, la implementación de las HU en su correspondiente iteración, obteniéndose en cada una de ellas una versión funcional del producto. Lo primero es hacer un chequeo de cada HU, en conjunto con el plan de iteraciones y se modifica en caso de ser necesario, para esto se crean tareas de desarrollo para de esta forma poder organizar la implementación. Estas tareas, al contrario de las HU escritas en el lenguaje del cliente, son escritas en un lenguaje técnico. Como parte de la planificación realizada en el capítulo anterior se detallan a continuación las iteraciones de desarrollo sobre el sistema.

##### 3.1.1 Iteración 1

En esta iteración se implementaron las funcionalidades que dan lugar a la preparación de la imagen, relacionadas con las técnicas de ajuste de contraste, selección del espacio de color y detectar bordes en la imagen digital para su procesado.

Tabla 7: Tiempo de implementación de las HU de la iteración #1

Historias de usuario	Tiempo de implementación	
	Estimación	Real
Cargar imagen	1	1
Filtro de realce de bordes	1	1
Filtro de ajuste de la intensidad del mapa de color de la imagen.	1	1
Detector de bordes basado en la segunda derivada (Canny).	2	2
Cambio de espacio de color a YCbCr	2	2
Seleccionar canal de luminancia	1	1

A continuación se muestran las tareas de ingeniería efectuadas para las funcionalidades implementadas en esta iteración:

*Tabla 8: Tarea de ingeniería de la funcionalidad Cargar imagen.*

Tarea	
Número de tarea: 1	Número de HU: 1
Nombre: Cargar imagen	
Tipo de tarea: desarrollo	Puntos de estimación: 1
Programador responsable: Adolfo Díaz Quiros	
Descripción: Se carga la imagen en disco seleccionada por el usuario	

*Tabla 9: Tarea de ingeniería de la funcionalidad Filtro de realce de bordes.*

Tarea	
Número de tarea: 2	Número de HU: 2
Nombre: Filtro de realce de bordes	
Tipo de tarea: desarrollo	Puntos de estimación: 1
Programador responsable: Celso Raúl Sánchez Viera	
Descripción: Se aplica un filtro de realce de bordes a la imagen cargada por el usuario.	

*Tabla 10: Tarea de ingeniería de la funcionalidad Filtro de ajuste de la intensidad del mapa de color de la imagen.*

Tarea	
Número de tarea: 3	Número de HU: 3
Nombre: Filtro de ajuste de la intensidad del mapa de color de la imagen.	
Tipo de tarea: desarrollo	Puntos de estimación: 1
Programador responsable: Adolfo Díaz Quiros	
Descripción: Se aplica un filtro de ajuste de valores de intensidad del mapa de color de la imagen, obteniéndose una imagen con su histograma extendido.	

Tabla 11: Tarea de ingeniería de la funcionalidad Detector de bordes basado en la segunda derivada (Canny).

Tarea	
Número de tarea: 4	Número de HU: 4
Nombre: Detector de bordes basado en la segunda derivada (Canny).	
Tipo de tarea: desarrollo	Puntos de estimación: 2
Programador responsable: Celso Raúl Sánchez Viera	
Descripción: Se aplica un detector de bordes con mascara basada en la segunda derivada.	

Tabla 12: Tarea de ingeniería de la funcionalidad Cambio de espacio de color a YCbCr

Tarea	
Número de tarea: 5	Número de HU: 5
Nombre: Cambio de espacio de color a YCbCr	
Tipo de tarea: desarrollo	Puntos de estimación: 2
Programador responsable: Adolfo Díaz Quiros	
Descripción: Se modifica el espacio de color de RGB a YCbCr para trabajar con la luminancia.	

Tabla 13: Tarea de ingeniería de la funcionalidad Seleccionar canal de luminancia.

Tarea	
Número de tarea: 6	Número de HU: 6
Nombre: Seleccionar canal de luminancia	
Tipo de tarea: desarrollo	Puntos de estimación: 1
Programador responsable: Celso Raúl Sánchez Viera	
Descripción: Se obtienen los valores de luminancia del espacio de color YCbCr.	

### 3.1.2 Iteración 2

En esta iteración se implementan los métodos de umbralado.

Tabla 14: Tiempo de implementación de las HU de la iteración #2

Historias de usuario	Tiempo de implementación	
	Estimación	Real
Umbralado mediante método de Huang's (1995)	3	3
Umbralado mediante método de Khashman (2007)	3	3
Umbralado mediante método de Tsai (1995)	3	3

Tabla 15: Tarea de ingeniería de la implementación umbralado mediante método de Huang's (1995).

Tarea	
Número de tarea: 7	Número de HU: 7
Nombre: Umbralado mediante método de Huang's (1995).	
Tipo de tarea: desarrollo	Puntos de estimación: 3
Programador responsable: Celso Raúl Sánchez Viera	
Descripción: Se obtienen un valor de umbral global de la sección de la imagen que se aplique, basándose en el valor de la entropía de Shannon.	

Tabla 16: Tarea de ingeniería de la implementación umbralado mediante método de Khashman (2007).

Tarea	
Número de tarea: 8	Número de HU: 8
Nombre: Umbralado mediante método de Khashman (2007).	
Tipo de tarea: desarrollo	Puntos de estimación: 3
Programador responsable: Adolfo Díaz Quiros	
Descripción: Se obtienen un valor de umbral global de mínima pérdida de la sección de la imagen que se aplique, basándose en el valor de la luminancia y la intensidad media.	

Tabla 17: Tarea de ingeniería de la implementación umbralado mediante método de Tsai (1995).

Tarea	
Número de tarea: 9	Número de HU: 9

Nombre: Umbralado mediante método de Tsai (1995)	
Tipo de tarea: desarrollo	Puntos de estimación: 3
Programador responsable: Celso Raúl Sánchez Viera	
Descripción: Se obtienen un valor de umbral global de la sección de la imagen que se aplique, usando una extracción de características del histograma de la imagen para definir el umbral (esto lo hace encontrado los valles entre el <i>foreground</i> y el <i>background</i> ).	

### 3.1.3 Iteración 3

En esta iteración se divide la imagen por bloques para su procesado más detallado y se realiza el pos-procesado, mostrando la cuantificación de la imagen.

Tabla 18: Tiempo de implementación de las HU de la iteración # 3.

Historias de usuario	Tiempo de implementación	
	Estimación	Real
Dividir la imagen en bloques	1	1
Binarizar	1	1
Reconstruir Imagen	1	1
Calculo de saturación de grasas	1	1
Visualizar resultados	1	1

Tabla 19: Tarea de ingeniería de la funcionalidad Dividir la imagen en bloques.

Tarea	
Número de tarea: 10	Número de HU: 10
Nombre: Dividir la imagen en bloques	
Tipo de tarea: desarrollo	Puntos de estimación: 1
Programador responsable: Adolfo Díaz Quiros	
Descripción: Se divide la imagen en sub-imágenes más pequeñas.	

Tabla 20: Tarea de ingeniería de la funcionalidad Binarizar

Tarea	
Número de tarea: 11	Número de HU: 11
Nombre: Binarizar	
Tipo de tarea: desarrollo	Puntos de estimación: 1
Programador responsable: Celso Raúl Sánchez Viera	
Descripción: Se otorgan valores de intensidad=1 a los píxeles con valores por encima de la media de los umbrales resultantes de los métodos: (Huang's, Khashman y Tsai)	

Tabla 21: Tarea de ingeniería de la funcionalidad Reconstruir imagen.

Tarea	
Número de tarea: 12	Número de HU: 12
Nombre: Reconstruir Imagen	
Tipo de tarea: desarrollo	Puntos de estimación: 1
Programador responsable: Adolfo Díaz Quiros	
Descripción: Se unifican las sub-imágenes binarizadas en una única máscara binaria, la cual se superpone al canal de luminancia.	

Tabla 22: Tarea de ingeniería de la funcionalidad cálculo de saturación de grasas.

Tarea	
Número de tarea: 13	Número de HU: 13
Nombre: Cálculo de saturación de grasas	
Tipo de tarea: desarrollo	Puntos de estimación: 1
Programador responsable: Celso Raúl Sánchez Viera	
Descripción: Se contabilizan la cantidad de píxeles segmentados contra el total.	

Tabla 23: Tarea de ingeniería de la funcionalidad Visualizar resultados.

Tarea
-------

Número de tarea: 14	Número de HU: 14
Nombre: Visualizar resultados	
Tipo de tarea: desarrollo	Puntos de estimación: 1
Programador responsable: Adolfo Díaz Quiros	
Descripción: Se muestran la imagen segmentada y el valor de la saturación de grasas.	

### 3.2 Interfaz de usuario

La imagen representada a continuación muestra la interfaz que permite visualizar los resultados de aplicar el algoritmo. La misma cuenta con un grupo de funcionalidades para que los clientes puedan aplicar mejoras de acuerdo a las características de las imágenes con las cuales se realiza el análisis.

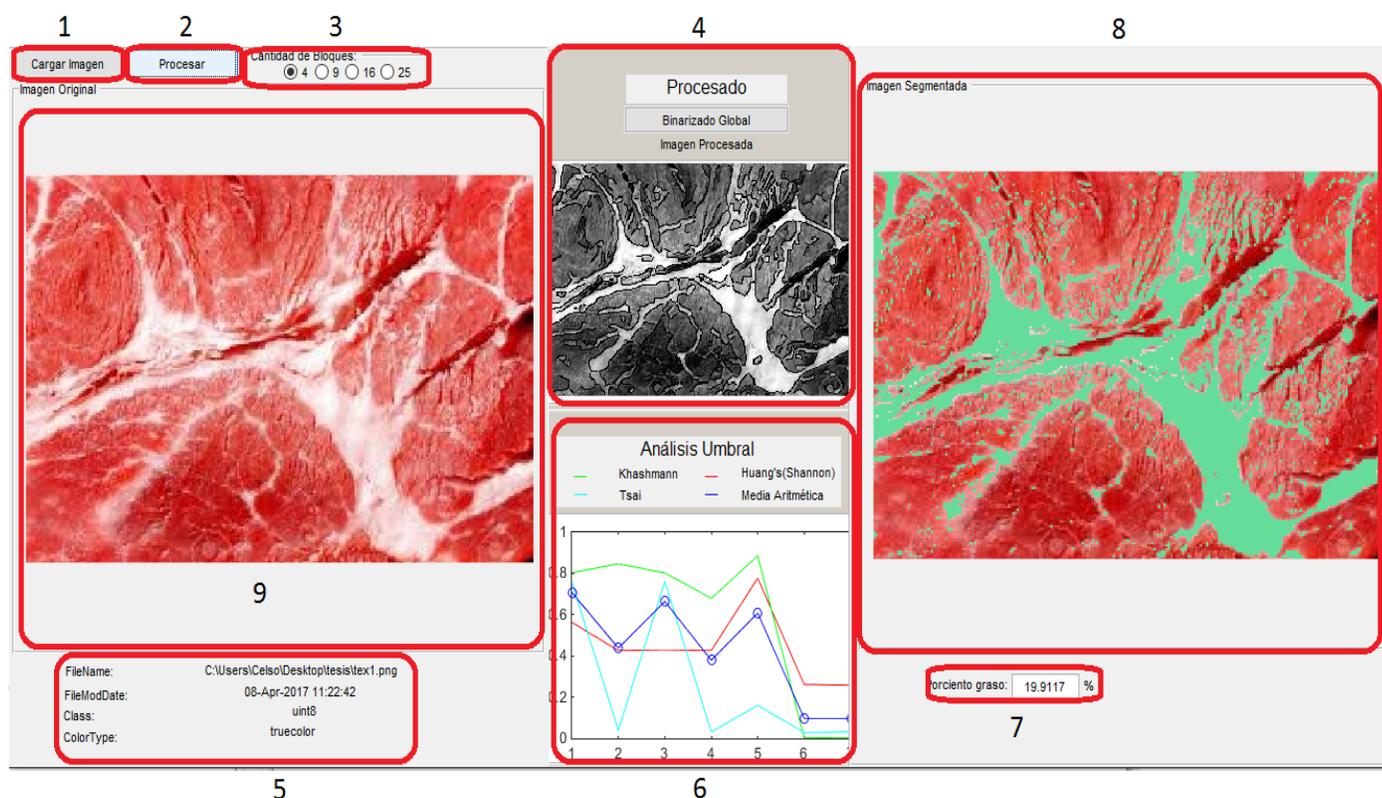


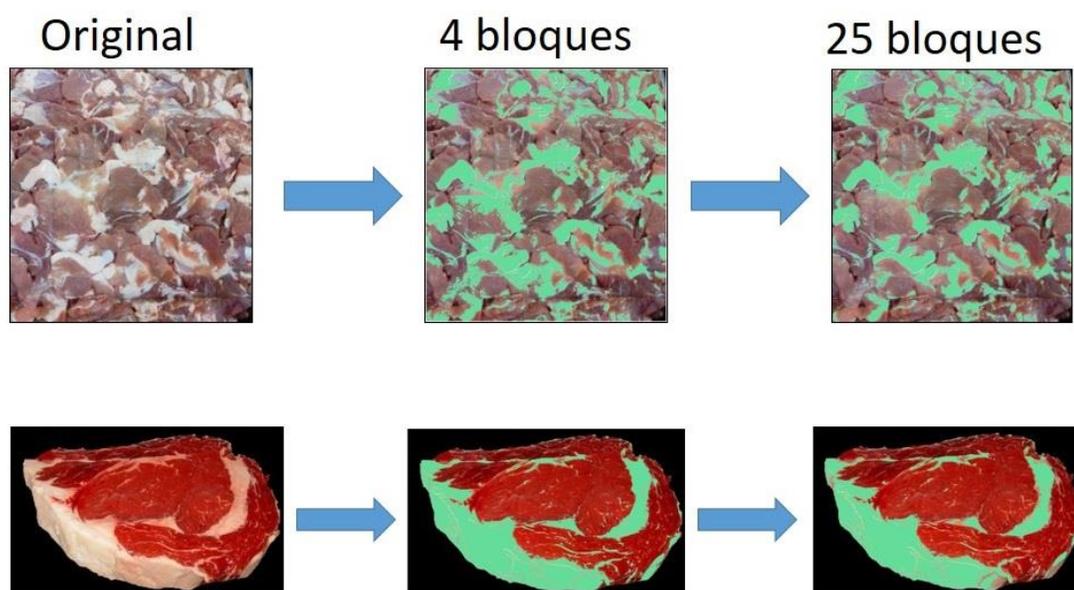
Figura 25: Interfaz de usuario.

1. Botón que permite cargar la imagen para su posterior procesado.
2. Botón que permite realizar el procesado la imagen.
3. Permite al usuario seleccionar la cantidad de bloques en que se va a dividir la imagen para ser procesada.

4. Botón que permite mostrar el resultado de cada método aplicado a la imagen.
5. Muestra los detalles de la imagen original.
6. Muestra el análisis del umbral.
7. Muestra por ciento de grasas
8. Muestra la imagen segmentada
9. Muestra la imagen original.

### 3.3 Resultados de aplicar el algoritmo

El algoritmo fue aplicado a 8 imágenes de carnes con grasa incluyendo las provenientes de la línea de producción de la empresa Bravo S.A. A continuación se muestran los resultados obtenidos de aplicar este algoritmo a dos de estas.



*Figura 26: Resultados del procesado de dos imágenes.*

En la primera fila muestra una imagen que presenta zonas grasas relativamente pequeñas y dispersas. Le sigue la segunda fila con una imagen con zonas grasas más extensas y los comportamientos de ambas bajo los mismos parámetros de bloques. Se puede apreciar en el primer caso, que la imagen tratada con 4 bloques presenta deficiencias en la segmentación siendo diferente la tratada con 25 bloques. En el segundo caso la imagen presenta deficiencia en la segmentación usando 25 bloques. Lo anterior demuestra la proporción directa entre cantidad de bloques y dispersión de zonas grasas u objeto de interés.

### 3.4 Validación del sistema

Para la validación de los resultados se obtuvieron imágenes de jamón segmentadas manualmente por el departamento de calidad de la empresa Bravo S.A. y usadas como referencia para la comparación de las diferentes métricas propuestas,

La siguiente tabla muestra la contraposición de las imágenes segmentadas manualmente por el departamento de calidad y la segmentada por la solución propuesta:



Figura 27: Comparación de la segmentación manual y la segmentación de la solución propuesta.

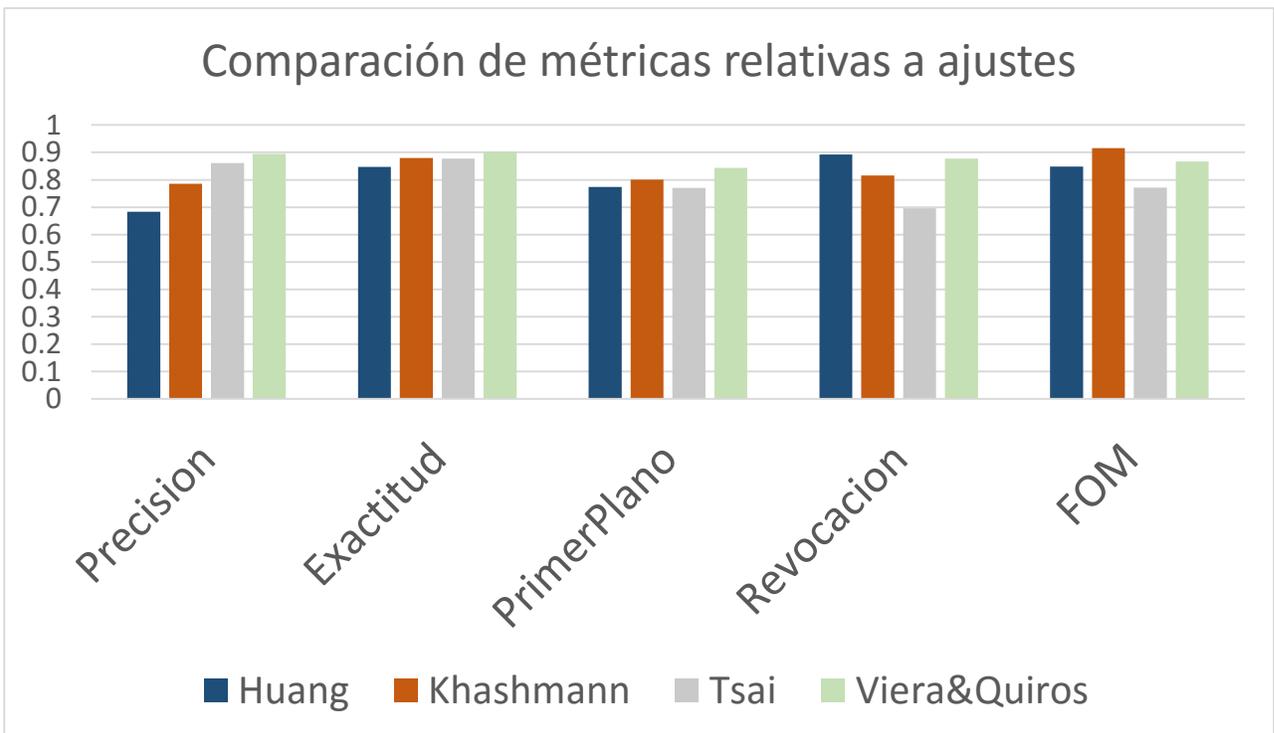


Grafico 2: Validación del algoritmo Viera&Quiros (2017)

Las métricas mostradas en la tabla anterior son relativas a la certeza de la segmentación, es decir, miden lo correcta que fue la segmentación automática con respecto a la segmentación del experto; muestra

superioridad en los resultados de Precisión, Exactitud y Primer plano y buenos resultados en Revocación y FOM. Junto a los resultados de la solución propuesta se encuentran los resultados bajo las mismas métricas de las segmentaciones arrojadas por los métodos de Huang, Khashmann y Tsai

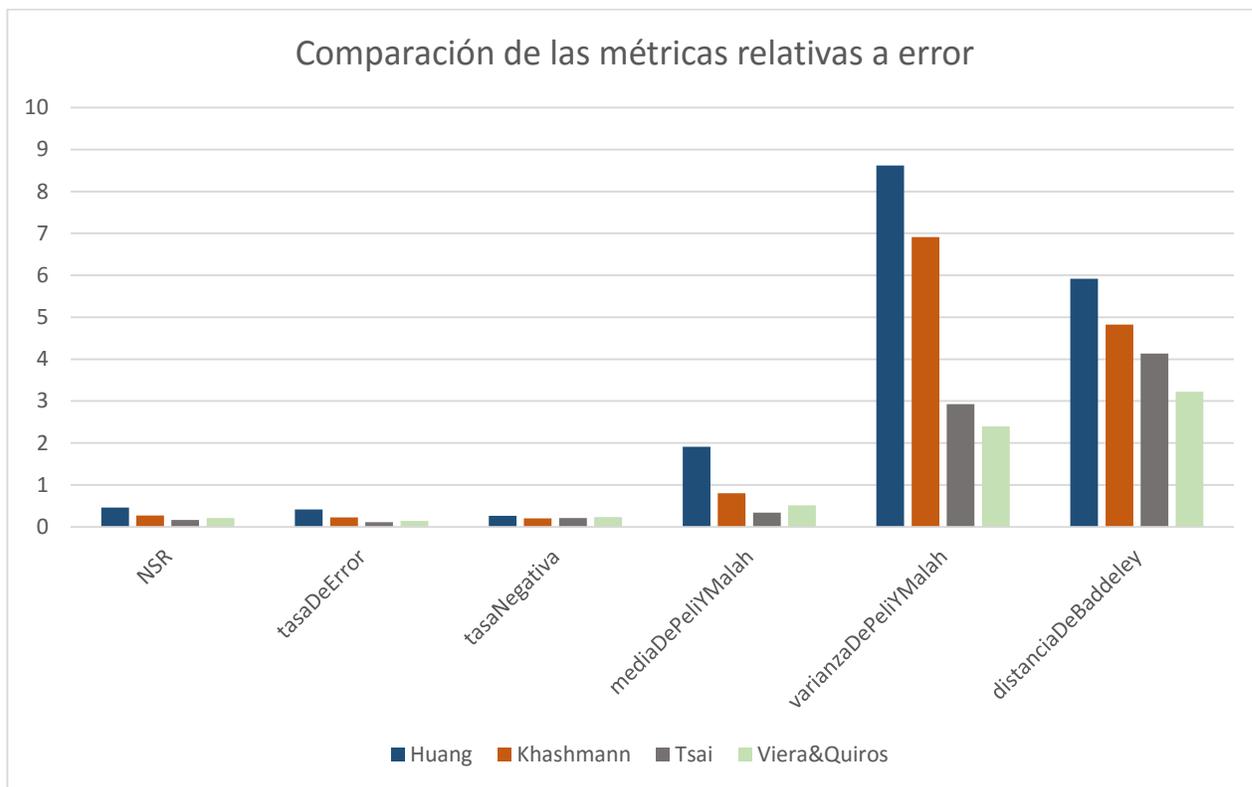


Grafico 3: Validación del algoritmo Viera&Quiros.

Las métricas mostradas en la tabla anterior son relativas al error de la segmentación, miden el error relativo a la segmentación automática con respecto a la segmentación del experto, demostrando mejores resultados de manera general. Junto a los resultados de la solución propuesta se encuentran los resultados bajo las mismas métricas de las segmentaciones arrojadas por los métodos de Huang, Khashmann y Tsai

### 3.5 Pruebas

Uno de los pilares de la Programación Extrema (XP) es el proceso de pruebas (45). XP anima a probar constantemente tanto como sea posible. Esto permite aumentar la calidad de los sistemas reduciendo el número de errores no detectados y disminuyendo el tiempo transcurrido entre la aparición de un error y su detección. También permite aumentar la seguridad de evitar efectos colaterales no deseados a la hora de realizar modificaciones y refactorizaciones. XP divide las pruebas del sistema en dos grupos (45): pruebas unitarias las cuales se concentran en los componentes individuales asegurándose que funcionen de manera apropiada como unidad, y pruebas de aceptación o pruebas funcionales destinadas a evaluar si al final de una iteración se consiguió las funcionalidades requeridas diseñadas por el cliente final.

### 3.5.1 Pruebas Unitarias

Una prueba unitaria es la verificación de un módulo (unidad de código) determinado dentro de un sistema. Los programadores realizan estas pruebas cuando: la interfaz de un método no es clara, la implementación es complicada, para testear entradas y condiciones inusuales luego de modificar algo. Para poder integrar el código realizado al ya existente, el mismo debe aprobar satisfactoriamente todos los casos de prueba definidos. En XP los programadores deben escribir las pruebas unitarias para cada módulo antes de escribir el código. No es necesario escribir casos de prueba para todos los módulos, sólo para aquellos en que exista la posibilidad de que puedan fallar.

### 3.5.2 Pruebas de caja blanca

El método prueba de caja blanca se basa en un examen cercano al detalle procedimental. Se prueban las rutas lógicas del software y la colaboración entre componentes, al proporcionar casos de pruebas que ejerciten conjuntos específicos de condiciones, bucles o ambos (48). A continuación, se describirá la realización de la técnica de camino básico, desglosándola en sus elementos fundamentales: confección del grafo de flujo, cálculo de la complejidad ciclomática, extracción de los caminos independientes, realización de los casos de pruebas y análisis de los resultados obtenidos. A continuación se muestra la prueba realizada al código perteneciente al método cargar imagen (load en inglés), el cual permite mejorar el contraste de la imagen. Es uno de los métodos a los cuales se le aplica la métrica de complejidad ciclomática debido a su relevancia para el sistema.

```
1 function Load_Callback(hObject, eventdata, handles)
2 [FileName, Path]=uigetfile({'*.jpg;*.png;*.bmp'}, 'Abrir Imagen');
3 if isequal(FileName,0)
4     errordlg('No se ha abierto la imagen ',' Error ');
5     return
6 else
7     h = waitbar(0,'Espere por favor...');
8     steps = 20;
9     for step = 1:steps
10        waitbar(step / steps)
11    end
12    close(h)
13    I=imread(strcat(Path,FileName));
14    global imagen
15    imagen=I;
16    S=whos('I');
17    info = imfinfo(strcat(Path,FileName));
18    %muestro las propiedades de la imagen en labels
19    set(handles.text2,'String',info.FileName );
20    set(handles.text3,'String',info.FileModDate);
21    set(handles.text6,'String',S.class);
22    set(handles.text8,'String',info.ColorType);
23    %muestro la imagen
24    axes(handles.axesImg0)
25    %background=imread(strcat(Path,FileName));
26    axis off;
27    imshow (I);
28    global aux
29    aux=I;
30 end
```

Figura 28: Código del método "cargar imagen".

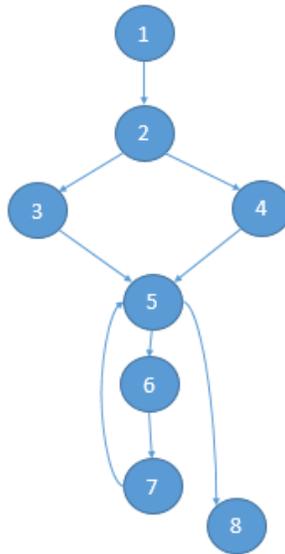


Figura 29: Grafo de flujos.

A partir del código fuente se diseñó el grafo como se muestra en la figura, luego se procede a calcular la complejidad ciclomática  $V(G)$ , la cual define el número de caminos independientes del conjunto básico de un programa, dando un límite superior para el número de pruebas que se deben realizar para asegurar que se ejecute cada sentencia al menos una vez.

$$V(G) = \text{Aristas}(A) - \text{Nodos}(N) + 2 \quad (51)$$

$$V(G) = 9 - 8 + 2$$

$$V(G) = 3$$

$$V(G) = 1 + \text{nodospredicados} \quad (52)$$

$$V(G) = 1 + 2$$

$$V(G) = 3$$

El valor  $V(G)$  expresa la cantidad de caminos linealmente independientes de la estructura de control del programa, por lo que se definen los siguientes caminos:

- Camino básico 1: 1-2-3-5-8
- Camino básico 2: 1-2-4-5-8
- Camino básico 3: 1-2-4-5-6-7-5-8

Cada camino independiente es un caso de prueba a realizar, de forma que se garantiza que durante la prueba se ejecuta por lo menos una vez cada sentencia del programa. En el caso anterior se calcularon cuatro caminos básicos, por tanto surge la necesidad de hacer igual número de casos de prueba. A continuación se muestra uno de los casos de pruebas realizados.

Camino 3:

Tabla 24: Casos de pruebas realizados al camino 4.

<b>Entrada</b>	Se exploran los archivos y se carga la imagen.
<b>Resultado esperado</b>	Se muestra la imagen en el espacio definido para ello así como la información relativa a la misma
<b>Condiciones</b>	-Se carga una imagen en el formato jpg, png o bmp. -La imagen debe ser una imagen a color (truecolor)

### 3.5.3 Pruebas de aceptación

Las pruebas de aceptación son creadas en base a las Historias de Usuarios, en cada ciclo de la iteración del desarrollo. El cliente debe especificar uno o diversos escenarios para comprobar que una historia de usuario ha sido correctamente implementada. Las pruebas de aceptación son de gran importancia, dado que miden el grado de satisfacción del cliente con el producto desarrollado (48). Por lo tanto, son los clientes los responsables de verificar que los datos de estas pruebas sean correctos. Así mismo, en caso de que fallen varias pruebas, son ellos los encargados de indicar el orden de resolución de los fallos (48). Una HU no se puede considerar terminada hasta tanto pase correctamente todas las pruebas de aceptación. Dado que la responsabilidad es grupal, es recomendable publicar los resultados de las pruebas de aceptación, de manera que todo el equipo esté al tanto de esta información. A continuación se presenta el diseño de caso de prueba de 3 de las HU a las que se le realizaron las pruebas.

Tabla 25: Caso de prueba de aceptación de la HU 1

<b>Caso de prueba de aceptación</b>	
<b>Código:</b> HU_1	<b>Historia de Usuario:</b> 1
<b>Nombre:</b> Cargar Imagen	
<b>Descripción:</b> Comprueba la funcionalidad de cargar una imagen a sistema para ser procesada	
<b>Condiciones de ejecución:</b> Existe una imagen de características necesarias	

<p><b>Entrada/Pasos de Ejecución:</b></p> <ul style="list-style-type: none"> <li>• El usuario selecciona una imagen en formato jpg, bpm o png y en formato truecolor.</li> <li>• El sistema muestra la imagen en el componente predefinido junto con la información relativa a la imagen.</li> </ul>
<p><b>Resultado Esperado:</b></p> <p>El sistema muestra la imagen en el componente predefinido junto con la información relativa a la imagen.</p>

Tabla 26: Caso de prueba de aceptación de la HU 2

<b>Caso de prueba de aceptación</b>	
<b>Código:</b> HU_2	<b>Historia de Usuario:</b> 2
<b>Nombre:</b> Filtro de realce de bordes	
<b>Descripción:</b> Comprueba la funcionalidad la aplicación del método de realce de bordes	
<b>Condiciones de ejecución:</b> Existe una imagen de características necesarias en el sistema, de un solo canal de intensidad	
<p><b>Entrada/Pasos de Ejecución:</b></p> <ul style="list-style-type: none"> <li>• El usuario selecciona una imagen en formato jpg, bpm o png y en formato truecolor.</li> <li>• El usuario oprime el botón "Procesar".</li> <li>• El sistema muestra la imagen procesada en el componente.</li> </ul>	

<p><b>Resultado Esperado:</b></p> <p>El sistema muestra la imagen procesada en el componente predefinido siendo obtenidos los bordes presentes mediante el método de "Canny".</p>
<p><b>Evaluación de la Prueba:</b></p> <p>Prueba satisfactoria.</p>

Tabla 27: Caso de prueba de aceptación de la HU 3

Caso de prueba de aceptación	
<b>Código:</b> HU_3	<b>Historia de Usuario:</b> 14
<b>Nombre:</b> Visualizar resultados	
<b>Descripción:</b> Comprueba la funcionalidad de mostrar los resultados del procesado de la imagen de entrada.	
<b>Condiciones de ejecución:</b> Existe una imagen de características necesarias	
<p><b>Entrada/Pasos de Ejecución:</b></p> <p>El usuario selecciona una imagen en formato jpg, bpm o png y en formato truecolor.</p> <p>El sistema muestra la imagen en el componente predefinido junto con la información relativa a la imagen, muestra la binarización de la imagen mediante una ejecución global de la imagen ,muestra la relación de umbrales relativos a cada bloque tratado, muestra la imagen segmentada realizada y la cuantificación de materias grasas.</p>	
<p><b>Resultado Esperado:</b></p> <p>El sistema muestra la imagen en el componente predefinido junto con la información relativa a la imagen, muestra la binarización de la imagen mediante una ejecución</p>	

global de la imagen, muestra la relación de umbrales relativos a cada bloque tratado, muestra la imagen segmentada realizada y la cuantificación de materias grasas.

### Evaluación de la Prueba:

Prueba satisfactoria

### Resultado de las pruebas de aceptación

Por cada HU se realizó una prueba de aceptación. Fueron detectadas un total de 3 no conformidades en las iteraciones realizadas, las cuales fueron solucionadas.

Como parte de estas pruebas se procedió a la creación de 6 casos de prueba para verificar los resultados del software, en los mismos se tomaron imágenes de diferentes características de productos cárnicos.

Estas pruebas arrojaron los siguientes resultados:

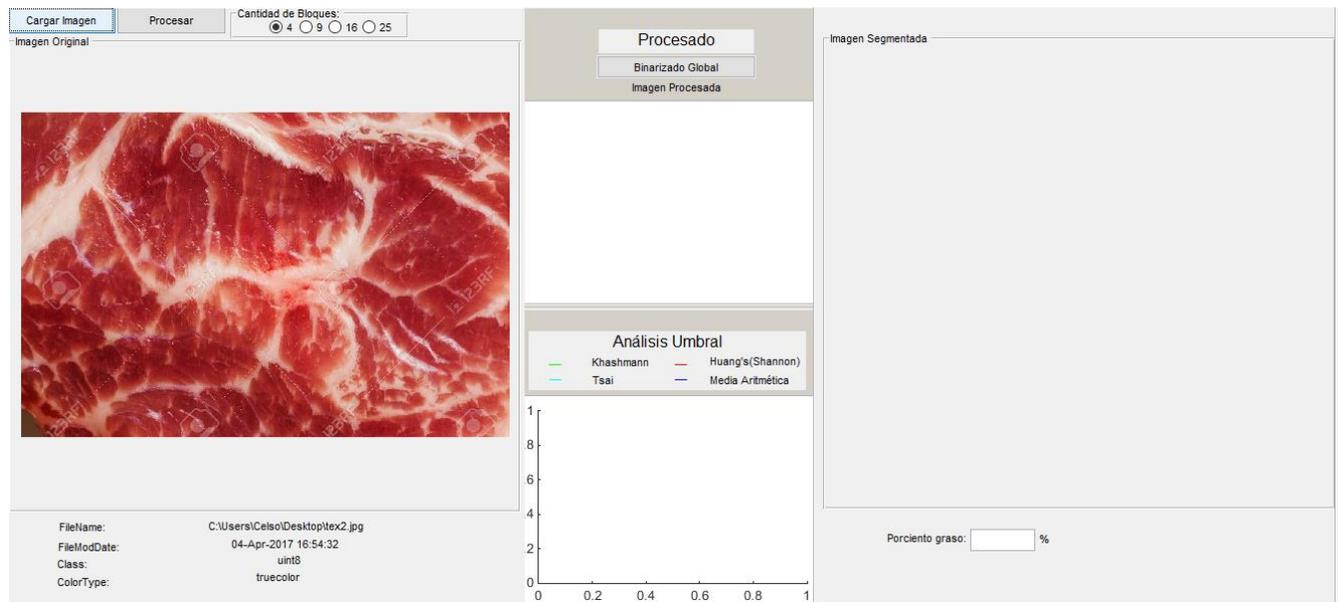


Figura 30: Imagen original cargada satisfactoriamente en el sistema.

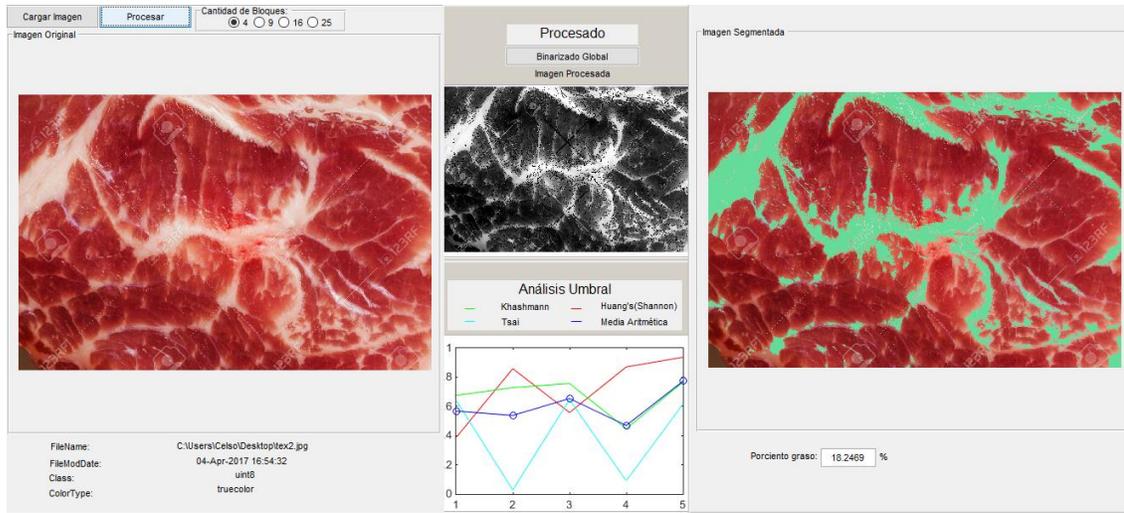


Figura 31: Resultado final de la segmentación utilizando el algoritmo Viera&Quiros(2017)

### 3.6 Conclusiones del capítulo

En el presente capítulo referente a implementación y pruebas, se describieron las tareas de desarrollo para darle solución a las HU y se llevaron a cabo las pruebas realizadas a la solución, mostrando resultados satisfactorios, lo que demuestra que el desarrollo dirigido por pruebas desempeña un papel fundamental en el proceso de desarrollo de la aplicación con una metodología ágil XP. También se realizó la validación del sistema la cual permitió demostrar que el algoritmo satisface las necesidades del cliente y cumple con el objetivo de la investigación.

## Conclusiones

Con el desarrollo del presente trabajo de diploma se cumplieron todas las tareas propuestas. De esta forma se arribó a las siguientes conclusiones:

- Los métodos globales de umbralización constituyen una forma eficiente y ágil de extraer información de una imagen y caracterizar los objetos que la constituyen.
- De la literatura consultada se analizaron a profundidad 9 métodos aplicados sobre las imágenes de la empresa arrojando resultados significativos los métodos de Tsai, Khasmann y Huang.
- El tratamiento individual de partes de la imagen (bloques) permite evitar errores producidos por diferencias de valores de intensidad en un mismo objeto en diferentes zonas de la imagen original
- El uso de bloques debe ser consecuente a las características de la imagen a tratar, a mayor dispersión de áreas de interés a seccionar, mayor debe ser la cantidad de bloques, y para objetos grandes y compactos en la imagen menor cantidad de bloques.
- La segmentación resultante de la aplicación de los métodos de Tsai, Kashman y Huang y la media entre ellos es medida mediante la comparación de un criterio experto con los resultados obtenidos, arrojando métricas las cuales mesuraron como superiores los resultados de la media.

## Recomendaciones

- Se recomienda para estudios futuros automatizar la selección de la cantidad de bloques a partir de las características de la imagen
- Se recomienda la investigación y optimización de métodos basados en bordes y basados en regiones para la segmentación en tiempo real de imágenes en pos de competir en efectividad y eficiencia con los resultados obtenidos.

## Referencias bibliográficas

1. **Hernández Cázares , Aleida Selene .** *Control de calidad y seguridad de la carne y productos cárnicos curados mediante el uso de sensores enzimáticos.* Valencia : s.n., 2010.
2. **Martín-Merás, María Luisa Durán.** *ALGORITMOS DE RECONOCIMIENTO DE PATRONES EN IMÁGENES PARA LA CARACTERIZACIÓN DEL JAMÓN DE CERDO IBÉRICO.* DEPARTAMENTO DE INFORMÁTICA, UNIVERSIDAD DE EXTREMADURA . Cáceres : s.n., 2002. TESIS PARA OPTAR POR EL GRADO DE DOCTOR EN CIENCIAS.
3. **Nuñez Peguero, Pedro Daniel.** *Realce y Restauración de imagen.* 2013.
4. **Escalante Ramírez, Boris.** *Procesamiento Digital de Imágenes.* 2006.
5. **Gonzalez, Rafael C y Woods, Richard E.** *Digital image processing.* s.l. : PRENTICE-HALL INTERNATIONAL EDITION, 2002. ISBN: 9780130946508.
6. *Suavizado mediante técnicas de filtrado morfológico en imágenes de microscopía celular.* **OLIVA, Karla Rodríguez.** 1, La Habana : s.n., 2017, Revista Cubana de Ciencias Informáticas, Vol. 11. ISSN 2227-1899.
7. **ORTIZ ZAMORA, Francisco Gabriel.** *Procesamiento morfológico de imágenes en color: aplicación a la reconstrucción geodésica.* 2002. ISBN: 84-688-0091-0.
8. *Colour spaces: perceptual, historical and applicational background.* **TKALCIC, Marko y TASIC, Jurij F.** 2003. ISBN: 0-7803-7763-X, .
9. **Jorge A. Marcial Basilio, Gualberto Aguilar Torres, Gabriel S-nchez, Karina Toscano Medina, Héctor M. Pérez Meana.** *Novedosa técnica para la detección de imágenes pornográficas empleando modelos de color HSV y YCbCr.* 2012. págs. 79-90. ISSN: 0120-6230.
10. **Zuiderveld., Karel.** *Contrast limited adaptive histogram equalization.* In *Graphics gems IV.* [ed.] Academic Press Professional. s.l. : Paul S. Heckbert (Ed.), 1994. ISBN:0-12-336155-9.
11. **CEPEDA NEGRETE, Jonathan y SANCHEZ YANEZ, Raul E.** *Algoritmos de Constancia de Color para el Mejoramiento de Imágenes Oscuras.* 2010 : V Congreso Internacional de la Ciencia de Sistemas.
12. **COTO, Ernesto.** *Métodos de segmentación de imágenes médicas.* s.l. : Universidad Central de Venezuela: Lecturas en Ciencias de la Computación., 2003,. págs. 9-15. Vols. vol. 1,. ISSN: 1316-6239.
13. *Métodos de segmentación de nubes en imágenes satelitales.* **VARGAS, Ernesto Gómez y NEIRA, Nelson Obregón y ARANGO, Diego Fernando Rocha.** 36, 2013, Tecnura, Vol. 17, págs. 96-110. ISSN: 0123-921X.
14. *DETECCIÓN DE BORDES DE UNA IMAGEN USANDO MATLAB.* **Sanchez Zorrilla, Victor Manuel, y otros.** [ed.] Pistas Educativas. 122, 2016, Vol. vol. 38. ISSN: 2448-847X.
15. **Kim, Daniel.** *Sobel Operator and Canny Edge Detector.* 2013.

16. **Rodriguez, R., y Sossa, J.** *Procesamiento y Análisis Digital de Imágenes*. Mexico : Alfaomega Grupo Editor, 2012. ISBN: 978-84-9964-077-8.
17. **Valverde, J.** *Detección de bordes mediante el algoritmo de Canny*. 2010.
18. **SALCEDO ECHES, Edith Andrea y CHIRIBOGA MALDONADO, María del Pilar.** *Implementación de algoritmos para preprocesamiento, detección de bordes y segmentación de imágenes digitales a color utilizando MATLAB*. 2003.
19. **Wilson., G. X. Ritter and J.~N.** *Handbook of Computer Vision in Image Algebra*. s.l. : CRC Press, 2000. ISBN: 0849300754, 978-0849300752.
20. **Shapiro., R. M. Haralick and L. G.** *Computer and Robot Vision*. s.l. : Addison-Wesley, 1993. ISBN: 0201569434, 9780201569438.
21. **GONZÁLEZ RODRÍGUEZ, Lázara Liset, DRC. HERNÁNDEZ MONTERO, Fidel Ernesto y GONZÁLEZ CORDÓN, Carlos Javier, MOHAMMED AMER, Mohammed Ahmad.** *Procesamiento de imágenes de fondo de ojo para el diagnóstico médico*. .
22. *Umbral adaptativo para la detección de objetos en movimiento usando visión computacional*. **M.Mazo, J.L.Lázaro, J.Ureña, J.J.Garcia**. España : s.n., 2005, Vol. 1. isbn: 84-8102-964-5.
23. **Wellner, Pierre.** *Adaptive Thresholding for the DigitalDesk*. EPC. s.l. : Rank Xerox, 1993. Technical Report EPC-1993-110.
24. **Neerad Phansalkar, Sumit More , Ashish Sabale , Dr. Madhuri Joshi.** *Adaptive Local Thresholding for Detection of Nuclei in Diversely Stained Cytology Images*. Department of Electronics and Telecommunication, College of Engineering Pune. India : IEEE, 2011. págs. 218-220. ISBN: 978-1-4244-9799-7, 978-1-4244-9798-0.
25. *Adaptive Document Image Binarization*. **Sauvola, M. Pietikainen**. 2, 2000, Vol. 33, págs. 225-236.
26. *Image thresholding for optical character recognition and other applications requiring character image extraction*. **Rohrer, J.M. White and G.D.** 4, 1983, IBM Journal of Research and Development, Vol. 27, págs. 400-411. ISSN: 0018-8646, 0018-8646.
27. **GONZÁLEZ MARTÍNEZ, VANESA.** *Técnicas de umbralización para la detección de anomalías en la pared aórtica mediante OCT*. 2013.
28. **Arámbula, Fernando.** *Análisis de imágenes médicas*. 2008.
29. *Handbook of Medical Imaging*. **M.Sonka y J.M.Fitzpatrick**. s.l. : SPIE Press, 2000, Vol. 2. ISBN 978-0-12-077790-7.
30. **López, María del Mar Pujol.** *Incorporación de características en la función de energía para segmentación de imágenes usando campos aleatorios de Markov*. 2000.
31. **Machado, Nery Javier.** *SEGMENTACIÓN DE MELANOMA BASADO EN METODOS DE BINARIZACION*. SAN LORENZO, PARAGUAY : s.n., Diciembre, 2014.

32. *EVALUACIÓN DE MÉTODOS DE SEGMENTACIÓN APLICADOS A IMÁGENES DE MUESTRAS DE SANGRE.* **Deniz, Arniel Labrada y García, Silena Herold.** La Habana, Cuba : s.n., 2016.
33. *Valmet: A new validation tool for assessing and improving 3D object segmentation.* **Gerig, Guido, Jomier, Matthieu y Chakos, Miranda.** USA : Springer, Berlin, Heidelberg, 2001, Vol. 2208. ISBN: 978-3-540-42697-4, 978-3-540-45468-7.
34. *Errors in binary images and an Lp version of the Hausdorff metric.* **Baddeley, A.J.** Netherlands : Nieuw Archief Voor Wiskunde, 1992. ISSN: 0028-9825.
35. **López, César Pérez.** *Matlab y sus aplicaciones en las Ciencias y la Ingeniería.* Madrid : s.n., 2002. ISBN: 84-205-3537-0.
36. **VILLANUEVA, M. Vargas, SORIA y C, Manuel Berenguel y CANTARERO, Teodoro Álamo.** *Tutorial de Introducción a MATLAB.* 2012.
37. **M. Vargas Villanueva, Manuel Berenguel Soria y Teodoro Álamo Cantarero.** *Tutorial de Introducción a MATLAB.* 2012.
38. **SIERRA, María.** *Trabajando con Visual Paradigm for UML.* 2008.
39. **Sánchez, M.A.M.** *Metodologías De Desarrollo De Software.* 2004.
40. **Joskowicz, José.** *Reglas y prácticas en eXtreme Programming.* 2008.
41. **Bustamante, Dayana y Rodriguez, Jean.** *Metodología Actual:Metodología XP.* 2014.
42. **POYNTON, Charles.** *Digital video and HD: Algorithms and Interfaces.* Elsevier. 2012. ISBN: 978-0-12-391926-7.
43. **REBAZA, Jorge Valverde.** *Detección de bordes mediante el algoritmo de Canny.* Escuela Académico Profesional de Informática. Perú : Universidad Nacional de Trujillo , 2007.
44. **CANÓS, José H., LETELIER, Patricio y PENADÉS, M<sup>a</sup> Carmen.** *Metodologías ágiles en el desarrollo de software. Metodologías Ágiles en el Desarrollo de Software.* 2003. págs. 1-8.
45. **Beck, Kent.** *Extreme Programming Explained.* 1999. ISBN: 201-61641-6 .
46. *Aspectos tempranos : un enfoque basado en Tarjetas CRC.* **Casas, Sandra y Reinaga, Hector .** 1, Argentina : s.n., 2009, Vol. 6. ISSN: 1909-0056, 1657-7663.
47. **Garcia, Oscar.** Nomenclatura estándar del código Java. [En línea] 22 de Agosto de 2011. <http://www.elclubdelprogramador.com/2011/08/22/java-nomenclatura-estandar-del-codigo-java/>.
48. **PRESSMAN, Roger S.** *Software engineering: a practitioner`s approach.* 2005. ISBN: 0-07-285318-2.
49. **Suarez, Maria Lorena.** *Competencias en TIC: Colección de Fasículos Digitales. Cuaderno 1: Introducción a la programación y sus lenguajes.* 2015.
50. **Haralick, Robert M., Shapiro, Linda G.** *Image Segmentation Techniques.* University of Cambridge Computer Laboratory. Cambridge : s.n., 1985. First Year Report and Project Proposal.
51. **Pratt, W.** *Digital Image Processing.* 1991.

52. *A Survey of Threshold Selection Techniques*. **Weszka, Joan S.** 2, 1978, *Computer Graphics and Image Processing* , Vol. 7, págs. 259-265. ISSN: 0146-664X.

## Anexos

### Anexo #1 Historias de usuarios

Tabla 28: Historia de usuario # 1

Historia de Usuario	
Número: 1	Nombre de la Historia de Usuario: Cargar imagen
Modificación a la Historia de Usuario: Ninguna	
Usuario: Empresa Bravo S.A.	Iteración asignada: 1
Prioridad en negocio: Alta	Puntos estimados: 1 semanas
Riesgo en desarrollo: medio	Puntos reales: 1 semanas
Programador responsable: Adolfo	
Descripción: Se carga la imagen en disco seleccionada por el usuario.	
Observaciones:	

Tabla 29: Historia de usuario # 4

Historia de Usuario	
Número: 4	Nombre de la Historia de Usuario: Detector de bordes basado en la segunda derivada (Canny).
Modificación a la Historia de Usuario: Ninguna	
Usuario: Empresa Bravo S.A.	Iteración asignada: 1
Prioridad en negocio: alta	Puntos estimados: 2 semanas
Riesgo en desarrollo: alta	Puntos reales: 2 semanas
Programador responsable: Celso	
Descripción: Se aplica un detector de bordes con mascara basada en la segunda derivada.	

**Observaciones:** Debe aplicarse sobre los valores de intensidad del mapa de color.

*Tabla 30: Historia de usuario # 5*

Historia de Usuario	
<b>Número:</b> 5	<b>Nombre de la Historia de Usuario:</b> Cambio de espacio de color a YCbCr
<b>Modificación a la Historia de Usuario:</b> Ninguna	
<b>Usuario:</b> Empresa Bravo S.A.	<b>Iteración asignada:</b> 1
<b>Prioridad en negocio:</b> alta	<b>Puntos estimados:</b> 2 semanas
<b>Riesgo en desarrollo:</b> alta	<b>Puntos reales:</b> 2 semanas
<b>Programador responsable:</b> Adolfo	
<b>Descripción:</b> Se modifica el espacio de color de RGB a YCbCr para trabajar con la luminancia.	
<b>Observaciones:</b> Debe aplicarse sobre la imagen RGB.	

*Tabla 31: Historia de usuario # 6*

Historia de Usuario	
<b>Número:</b> 6	<b>Nombre de la Historia de Usuario:</b> Seleccionar canal de luminancia
<b>Modificación a la Historia de Usuario:</b> Ninguna	
<b>Usuario:</b> Empresa Bravo S.A.	<b>Iteración asignada:</b> 1
<b>Prioridad en negocio:</b> alta	<b>Puntos estimados:</b> 1 semanas
<b>Riesgo en desarrollo:</b> alta	<b>Puntos reales:</b> 1 semanas
<b>Programador responsable:</b> Celso	
<b>Descripción:</b> Se obtienen los valores de luminancia del espacio de color YCbCr.	

**Observaciones:** Debe aplicarse sobre la imagen en YCbCr.

*Tabla 32: Historia de usuario # 7*

Historia de Usuario	
<b>Número:</b> 7	<b>Nombre de la Historia de Usuario:</b> Umbralado mediante método de Huang's (1995)
<b>Modificación a la Historia de Usuario:</b> Ninguna	
<b>Usuario:</b> Empresa Bravo S.A.	<b>Iteración asignada:</b> 2
<b>Prioridad en negocio:</b> alta	<b>Puntos estimados:</b> 3 semanas
<b>Riesgo en desarrollo:</b> alta	<b>Puntos reales:</b> 3 semanas
<b>Programador responsable:</b> Celso	
<b>Descripción:</b> Se obtienen un valor de umbral global de la sección de la imagen que se aplique, basándose en el valor de la entropía de Shannon	

*Tabla 33: Historia de usuario # 8*

Historia de Usuario	
<b>Número:</b> 8	<b>Nombre de la Historia de Usuario:</b> Umbralado mediante método de Khashman (2007)
<b>Modificación a la Historia de Usuario:</b> Ninguna	
<b>Usuario:</b> Empresa Bravo S.A.	<b>Iteración asignada:</b> 2
<b>Prioridad en negocio:</b> alta	<b>Puntos estimados:</b> 3 semanas
<b>Riesgo en desarrollo:</b> alta	<b>Puntos reales:</b> 3 semanas
<b>Programador responsable:</b> Adolfo	
<b>Descripción:</b> Se obtienen un valor de umbral global de mínima pérdida de la sección de la imagen que se aplique, basándose en el valor de la luminancia y la intensidad media.	

**Observaciones: Debe aplicarse sobre el canal de luminancia de la sección recortada de la imagen.**

*Tabla 34: Historia de usuario # 9*

Historia de Usuario	
<b>Número:</b> 9	<b>Nombre de la Historia de Usuario:</b> Umbralado mediante método de Tsai (1995)
<b>Modificación a la Historia de Usuario:</b> Ninguna	
<b>Usuario:</b> Empresa Bravo S.A.	<b>Iteración asignada:</b> 2
<b>Prioridad en negocio:</b> alta	<b>Puntos estimados:</b> 3 semanas
<b>Riesgo en desarrollo:</b> alta	<b>Puntos reales:</b> 3 semanas
<b>Programador responsable:</b> Celso	
<b>Descripción:</b> Se obtienen un valor de umbral global de la sección de la imagen que se aplique, usando una extracción de características del histograma de la imagen para definir el umbral (esto lo hace encontrado los valles entre el <i>foreground</i> y el <i>background</i> ).	
<b>Observaciones: Debe aplicarse sobre el canal de luminancia de la sección recortada de la imagen.</b>	

*Tabla 35: Historia de usuario # 10*

Historia de Usuario	
<b>Número:</b> 10	<b>Nombre de la Historia de Usuario:</b> Dividir la imagen en bloques
<b>Modificación a la Historia de Usuario:</b> Ninguna	
<b>Usuario:</b> Empresa Bravo S.A.	<b>Iteración asignada:</b> 3
<b>Prioridad en negocio:</b> alta	<b>Puntos estimados:</b> 1 semanas
<b>Riesgo en desarrollo:</b> media	<b>Puntos reales:</b> 1 semanas
<b>Programador responsable:</b> Adolfo	

<b>Descripción:</b> Se divide la imagen en sub-imágenes más pequeñas.
<b>Observaciones:</b> Debe aplicarse sobre el canal de luminancia de la imagen en YCbCr.

*Tabla 36: Historia de usuario # 11*

Historia de Usuario	
<b>Número:</b> 11	<b>Nombre de la Historia de Usuario:</b> Binarizar
<b>Modificación a la Historia de Usuario:</b> Ninguna	
<b>Usuario:</b> Empresa Bravo S.A.	<b>Iteración asignada:</b> 3
<b>Prioridad en negocio:</b> alta	<b>Puntos estimados:</b> 1 semanas
<b>Riesgo en desarrollo:</b> alto	<b>Puntos reales:</b> 1 semanas
<b>Programador responsable:</b> Celso	
<b>Descripción:</b> Se otorgan valores de intensidad=1 a los píxeles con valores por encima de la media de los umbrales resultantes de los métodos: (Huang's, Khashman y Tsai)	
<b>Observaciones:</b> Debe aplicarse sobre el canal de luminancia de la imagen en YCbCr.	

*Tabla 37: Historia de usuario # 12*

Historia de Usuario	
<b>Número:</b> 12	<b>Nombre de la Historia de Usuario:</b> Reconstruir Imagen
<b>Modificación a la Historia de Usuario:</b> Ninguna	
<b>Usuario:</b> Empresa Bravo S.A.	<b>Iteración asignada:</b> 3
<b>Prioridad en negocio:</b> alta	<b>Puntos estimados:</b> 1 semanas
<b>Riesgo en desarrollo:</b> alto	<b>Puntos reales:</b> 1 semanas
<b>Programador responsable:</b> Adolfo	
<b>Descripción:</b> Se unifican las sub-imágenes binarizadas en una única máscara binaria, la cual se superpone al canal de luminancia.	

**Observaciones: Debe aplicarse sobre el canal de luminancia de las sub-imagen en YCbCr.**

*Tabla 38: Historia de usuario # 13*

Historia de Usuario	
<b>Número:</b> 13	<b>Nombre de la Historia de Usuario:</b> Calculo de saturación de grasas
<b>Modificación a la Historia de Usuario:</b> Ninguna	
<b>Usuario:</b> Empresa Bravo S.A.	<b>Iteración asignada:</b> 3
<b>Prioridad en negocio:</b> alta	<b>Puntos estimados:</b> 1 semanas
<b>Riesgo en desarrollo:</b> alto	<b>Puntos reales:</b> 1 semanas
<b>Programador responsable:</b> Celso	
<b>Descripción:</b> Se contabilizan la cantidad de píxeles segmentados contra el total.	
<b>Observaciones:</b> Debe aplicarse sobre la máscara de segmentación única.	

*Tabla 39: Historia de usuario # 14*

Historia de Usuario	
<b>Número:</b> 14	<b>Nombre de la Historia de Usuario:</b> Visualizar resultados
<b>Modificación a la Historia de Usuario:</b> Ninguna	
<b>Usuario:</b> Empresa Bravo S.A.	<b>Iteración asignada:</b> 3
<b>Prioridad en negocio:</b> alta	<b>Puntos estimados:</b> 1 semanas
<b>Riesgo en desarrollo:</b> alto	<b>Puntos reales:</b> 1 semanas
<b>Programador responsable:</b> Adolfo	
<b>Descripción:</b> Se muestran la imagen segmentada y el valor de la saturación de grasas.	
<b>Observaciones:</b>	

## Anexo #2 Tarjetas CRC

Tabla 40: Tarjeta CRC # 2

Tarjeta CRC	
Clase: Threshold	
Responsabilidades	Colaboradores
<ol style="list-style-type: none"> <li>1. Umbralado de Huang's.</li> <li>2. Umbralado de Shman.</li> <li>3. Umbralado de Tsai.</li> <li>4. Devolver umbral medio de los métodos calculados.</li> </ol>	<ol style="list-style-type: none"> <li>1. AllPictBiClassThr_Huang.</li> <li>2. AllPictBiClassThr_Khashman07</li> <li>3. AllPictBiClassThr_Tsai95</li> </ol>

Tabla 41: Tarjeta CRC # 3

Tarjeta CRC	
Clase: BlockThreshold	
Responsabilidades	Colaboradores
<ol style="list-style-type: none"> <li>1. Aplicar Threshold en procedimiento por bloques</li> <li>2. Devuelve imagen compuesta por bloques procesados por Threshold</li> </ol>	<ol style="list-style-type: none"> <li>1. blkproc</li> </ol>

Tabla 42: Tarjeta CRC # 4

Tarjeta CRC
Clase: AllPictBiClassThr_Huang

Responsabilidades	Colaboradores
1. Aplicar el método de umbralado AllPictBiClassThr_Huang.	1. Im2double 2. Im2unit8 3. Floor 4. Cofilt 5. Cumsum 6. Flplr 7. Flipud 8. Zeros 9. Max 10. Rowindex 11. Abs

Tabla 43: Tarjeta CRC # 5

<b>Tarjeta CRC</b>	
Clase: AllPictBiClassThr_Khashman07	
Responsabilidades	Colaboradores
1. Aplicar el método de umbralado AllPictBiClassThr_Khashman07.	1. Im2unit8 2. Imhist 3. Find 4. Mean 5. Abs

Tabla 44: Tarjeta CRC # 6

<b>Tarjeta CRC</b>	
Clase: AllPictBiClassThr_Tsai95	
Responsabilidades	Colaboradores
<ol style="list-style-type: none"><li>1. Aplicar el método de umbralado AllPictBiClassThr_Tsai95.</li></ol>	<ol style="list-style-type: none"><li>1. Im2double</li><li>2. Imhist</li><li>3. Findpeaks</li><li>4. Find</li><li>5. Max</li><li>6. Min</li><li>7. Sum</li><li>8. Isempy</li></ol>