

Proyecto Ojo De Bife

Extracción Automática de Información de Imágenes Color del Músculo *Longissimus* *Dorsi*

Documentación de Proyecto de Grado
Ingeniería Eléctrica

Martín Bianculli
Andrés Duffour
José Lezama

Tutores:
Pablo Cancela Alicia Fernández Álvaro Gómez

Facultad de Ingeniería
Universidad de la República

Montevideo, Uruguay
Marzo 2007

Agradecimientos

A Alessandro Rizzi, PhD, miembro del GIC - Graphic, Imaging and Color research group, Departamento de Tecnologías de la Información, Universidad de Milán, por la ayuda brindada.

A Lautaro Perez y Fernando Butín, quienes en representación del Frigorífico Colonia hicieron posible la realización de ensayos, mostrando en todo momento una excelente predisposición a ayudar.

Al Frigorífico San Jacinto, y en particular al Ing. Juan Brum, quien nos brindó la posibilidad de probar el sistema en dicho frigorífico.

A la gente de INCA, en especial al Ing. Hugo Oxoby, por habernos permitido utilizar las instalaciones y equipos de la compañía.

A nuestros tutores por la ayuda y disponibilidad que nos brindaron en todo momento.

A todas aquellas personas que de alguna manera ayudaron a llevar este proyecto a cabo.

A los compañeros de generación, por los momentos compartidos a lo largo de los años de carrera

A nuestras familias por el apoyo brindado a lo largo de toda la carrera.

Índice general

1. Introducción	7
1.1. Planteo del problema	7
1.2. Desarrollo del Proyecto	7
1.3. Organización del Documento	8
2. Descripción del Problema	10
2.1. Contexto del Problema	10
2.1.1. Contexto Nacional	10
2.1.2. Contexto Norteamericano	10
2.2. Descripción del Proyecto	11
2.3. Descripción del Ojo de Bife	12
2.4. Otros métodos de evaluación cárnica y trabajos previos	13
2.5. Descripción del proceso de faena	14
3. Algoritmo de Segmentación del músculo <i>Longissimus Dorsi</i>	16
3.1. Introducción	16
3.1.1. Estructura del algoritmo de segmentación	16
3.1.2. Análisis del desempeño del algoritmo de segmentación	17
3.2. Preprocesado de la Imagen	17
3.3. Algoritmo FCM	19
3.3.1. Introducción Teórica al Algoritmo FCM	21
3.3.2. Resultados obtenidos al aplicar FCM	23
3.4. Envolverte Convexa (Convex Hull) y Morfología Binaria	24
3.4.1. Convex Hull	24
3.4.2. Morfología	25
3.4.3. Proceso iterativo	25
3.5. Eliminación de penínsulas y músculo <i>Longissimus Costarum</i>	31
3.6. Situaciones en las que fallaba el algoritmo	32

4. Obtención de Medidas	36
4.1. Introducción	36
4.2. Área del Ojo de Bife	36
4.2.1. Cálculo del área	36
4.3. Porcentaje de Marmoreo	37
4.4. Espesor de Grasa Subcutánea	38
4.4.1. Atributos topológicos de una imagen	38
4.4.2. Algoritmo de cálculo del espesor de grasa	42
4.5. Medidas de color en el Ojo de Bife	45
5. Colorimetría y Calibración de color	46
5.1. Introducción	46
5.2. Metamerismo	47
5.3. Espacio de color RGB	48
5.4. Espacios de colores independientes del dispositivo	48
5.4.1. Espacio de color CIE XYZ	49
5.4.2. Espacio de color CIELab	51
5.5. Caracterización de dispositivos	52
5.5.1. Métodos numéricos de Caracterización de dispositivos	53
5.5.2. Linealización	56
5.6. Caracterización de la cámara	57
5.6.1. Resultados de la primera calibración	58
5.6.2. Fuentes de error	60
5.6.3. Resultados de la segunda calibración	61
5.6.4. Conclusiones	62
6. Hardware	66
6.1. Elección del equipo de captura	66
6.1.1. Cámara de video: Análogo o Digital	66
6.2. Tecnología de sensado de imagen	67
6.2.1. Principios de funcionamiento de las cámaras CCD	67
6.2.2. Cámara color con 3 CCD	67
6.2.3. Cámara color con 1 CCD	68
6.3. Selección del puerto de entrada	70
6.4. Determinación del campo de visión	71
6.4.1. Tipo de lente	72
6.4.2. Cámaras y Lentes	73
6.5. Cámara utilizada en el proyecto	75
6.6. Equipo de captura de imágenes	76

7. Calibración geométrica	83
7.1. Calibración explícita	83
7.2. Proceso de calibración	86
7.3. Resultados de la calibración	87
7.3.1. Influencia en las medidas obtenidas	90
7.4. Relación entre pixeles y centímetros	92
8. Arquitectura de Software: Planeación y Elaboración	93
8.1. Documento de Requerimientos	93
8.1.1. Introducción	93
8.1.2. Propósito	93
8.1.3. Clientes	94
8.1.4. Descripción de la Arquitectura	94
8.1.5. Actores	94
8.1.6. Atributos	94
8.1.7. Funciones básicas	95
8.2. Casos de uso	95
8.2.1. Caso de uso 1	95
8.2.2. Caso 2	96
8.2.3. Caso 3	97
8.2.4. Caso 4	98
8.2.5. Caso 5	99
8.3. Modelo Conceptual	101
8.4. Diagramas de secuencia del sistema	102
8.5. Contratos	103
8.6. Diagramas de colaboración	107
9. Validación	110
9.1. Introducción	110
9.1.1. Indicadores	111
9.1.2. Versiones	111
9.2. Resultados	112
9.2.1. Análisis de resultados	114
9.3. Estudio de Robustez de las Medidas	117
9.4. Validación del Marmoreo y del Espesor de grasa Subcutánea	119
10. Conclusiones	121
10.1. Conclusiones Generales	121
10.2. Conclusiones del desarrollo del Proyecto	121
10.3. Conclusiones sobre los algoritmos implementados	122
10.4. A futuro	123

A. Visitas	128
A.1. Frigorífico Colonia 24 de Mayo de 2006	128
A.2. Frigorífico San Jacinto 8 de Diciembre de 2006	128
A.3. Frigorífico Colonia 7 de Febrero de 2007	129
B. Contenido del CD Adjunto	130
C. Manual de Usuario del Programa Ojo de Bife	131
D. Planificación y Dedicación	140
D.1. Evaluación de la planificación	143

Capítulo 1

Introducción

1.1. Planteo del problema

El objetivo del proyecto es desarrollar un prototipo de un sistema de visión por ordenador, que tome una imagen del corte entre la 3a y 4ta¹ costilla de una res, segmente el músculo *Longissimus Dorsi* (Ojo de Bife) y brinde medidas objetivas de su área, porcentaje de marmoreo o grado de veteado, espesor de la grasa de cobertura, así como medidas colorimétricas del mismo. El sistema todo en su conjunto deberá ser simple en su operación y manipulación, brindando las antedichas medidas en tiempos breves respecto al funcionamiento normal de producción de los frigoríficos. Estas medidas deberán ser lo suficientemente confiables como para ayudar a los expertos que las realizan manualmente en el frigorífico, y poder así clasificar las carcasas de manera más rápida y efectiva. En la figura 1.1 se muestra una imagen fotográfica de una costilla y la segmentación del músculo *Longissimus Dorsi*.

La técnica manual permite conocer el tamaño del Ojo de Bife así como el espesor de la grasa subcutánea. Si se desean realizar otras medidas, es necesario recurrir a otros instrumentos. Así, para realizar una medida del color, es necesario contar con un colorímetro; para las medidas de porcentaje de marmoreo es necesario realizar análisis químicos del Ojo de Bife.

1.2. Desarrollo del Proyecto

Durante el proyecto se desarrolló un software y un hardware (equipo de toma de imágenes), que permite capturar una imagen de una costilla y segmentar el músculo *Longissimus Dorsi* (Ojo de Bife), realizando automáticamente las mediciones del área del músculo, porcentaje de marmoreo, espesor de la grasa subcutánea y medidas de

¹Dependiendo de donde se comience a contar, dichas costillas pueden ser contadas como la 12ava y la 13ava

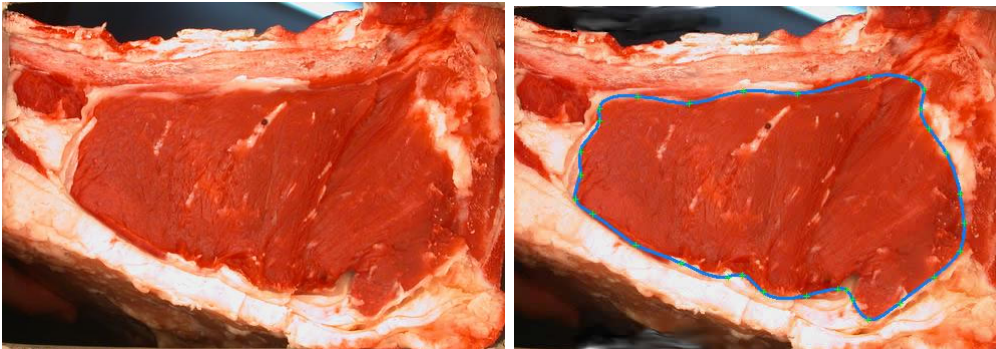


Figura 1.1: Imagen de un Ojo de Bife y su respectiva segmentación realizada manualmente

colorimetría. El software presenta una interfaz de usuario amigable, pensada para ser usada por un operario de un frigorífico. El hardware también está pensado para su operación en un ambiente de trabajo como el de un frigorífico.



Figura 1.2: Utilización en frigorífico de uno de los equipos desarrollados.

1.3. Organización del Documento

El capítulo 2 introduce el problema que se plantea en el proyecto y presenta una introducción sobre el proceso de faena y las medidas que se realizan en los animales

faenados.

En el capítulo 3 se explica el algoritmo de segmentación del músculo *Longissimus Dorsi*, desarrollando los conceptos teóricos que se utilizan en dicho algoritmo.

En el capítulo 4 se explican las medidas que se realizan una vez finalizada la segmentación del Ojo de Bife: área, marmoreo, espesor de grasa subcutánea y colorimetría.

El capítulo 5 explica todo lo referente al las medidas de colorimetría, desarrollando los conceptos teóricos y explicando la forma en que se calibra el equipo.

El capítulo 6 está dedicado al hardware del proyecto, incluye conceptos sobre cámaras digitales y presenta los equipos de toma de imágenes desarrollados.

El capítulo 7 trata sobre la calibración geométrica de la cámara, donde se mide el error introducido debido a imperfecciones del lente.

El capítulo 8 presenta una descripción en alto nivel de la arquitectura del software, en formato UML.

El capítulo 9 presenta el análisis de los resultados de la validación del algoritmo de segmentación del Ojo de Bife.

El último capítulo está dedicado a las conclusiones del proyecto y a los posibles caminos a seguir en un futuro.

Al final se incluyen los anexos en donde se describen las visitas a los frigoríficos realizadas, el manual de usuario del software desarrollado y un detalle acerca del desarrollo del proyecto.

Capítulo 2

Descripción del Problema

2.1. Contexto del Problema

2.1.1. Contexto Nacional

La industria cárnica del Uruguay es un componente relevante de la economía del país. Las exportaciones del sector representan aproximadamente el 24% del volumen total de las mismas. Uruguay, como país exportador de carnes rojas, tiene un perfil amplio en los mercados internacionales y actualmente exporta anualmente alrededor de 400.000 toneladas (peso carcasa) de carne vacuna, y 13.000 toneladas (peso carcasa) de carne ovina. Considerando todos los productos cárnicos, está presente en más de 70 países del mundo.¹

El Instituto de Ingeniería Eléctrica ha venido trabajando en proyectos de tratamiento de imágenes para valoración cárnica en el marco de un convenio con el INIA (Instituto Nacional de Investigación Agropecuaria). Se ha trabajado por ejemplo, en la detección del músculo *Longissimus Dorsi* en imágenes ecográficas [17].

2.1.2. Contexto Norteamericano

La industria de la carne vacuna es la más grande de las industrias de alimentos en Estados Unidos [6]. El organismo americano que regula la industria cárnica se llama USDA (United States Department of Agriculture). Esta institución fue fundada por Abraham Lincoln en 1862 y desde entonces se encarga del desarrollo rural, de la calidad de los alimentos, nutrición e investigación para el desarrollo de la tecnología agrícola entre otras cosas.

Para facilitar el marketing de la carne vacuna, la USDA tiene un estándar para calificar la calidad de las carcasas. Para la evaluación de una res, expertos de la USDA

¹Fuente: *INAC*, Instituto Nacional de Carnes del Uruguay, organismo encargado de regular la actividad de los frigoríficos y el comercio cárnico nacional e internacional

hacen un análisis visual del corte entre la 3a y la 4a costilla [19]. Según estudios, este corte es representativo de la calidad cárnica y la musculatura de toda la res. Para la evaluación se tiene en cuenta el área del Ojo de Bife, la grasa intramuscular (marmoreo o marbling score), el espesor de la grasa de cobertura (backfat) y la terneza de la carne que se puede medir por el color o la madurez fisiológica de la carne [18].

El hecho de que estas medidas deban ser realizadas por una persona trae consigo problemas ya que la velocidad de la línea en los frigoríficos es tal que el experto tiene entre 9 y 18 segundos para realizarlas en un ambiente con mala iluminación y muy baja temperatura [6], [5], [9]. Por otro lado, el cansancio y la situación emocional del experto pueden incidir en las medidas. Es por esto que un sistema de visión por computador con un algoritmo computacionalmente rápido para realizar las medidas promete ser el futuro para la evaluación de carcasas en los frigoríficos. La visión por computador ya ha aportado importantes soluciones en la agroindustria, por ejemplo calificando semillas o carcasas de cerdo.

Entre las características que tienen incidencia en la decisión de compra de la carne, se encuentra el color. La apariencia visual de la carne es uno de los factores de mayor importancia al momento de seleccionar la carne [43]

2.2. Descripción del Proyecto

El Proyecto se puede dividir en tres grandes partes temáticas:

- Equipo de adquisición de imágenes e interfaz gráfica con el usuario
- Algoritmo de Segmentación del Ojo De Bife, medidas de marmoreo y espesor de la grasa de cobertura
- Medidas objetivas de color

Las tres partes se encuentran vinculadas entre sí, de forma que la calidad en la captura de las imágenes afecta el desempeño del algoritmo de segmentación, mientras que la consideración de partes ajenas al Ojo de Bife como pertenecientes a él afectan a las restantes medidas.

Para la etapa de adquisición se estudiaron diferentes propuestas de dispositivos de captura de imágenes, esto es cámaras de video o cámaras fotográficas, así como distintas formas de iluminar el objeto a capturar.

La etapa de desarrollo del algoritmo de segmentación utilizó como base el algoritmo propuesto en el artículo “*Segmentation of longissimus dorsi for Beef Quality Grading using Computer Vision*” de S. Jeyamkondan, N. Ray, G.A. Kranzler y

S. Acton[6]. Se desarrolló un algoritmo que permite quitar el músculo *Longissimus Costarum*, el cual no forma parte del Ojo de Bife (ver figura 2.1).

La restante etapa requirió un estudio de los distintos espacios de colores y la conversión entre los mismos, así como la influencia de los elementos que participan en la adquisición de imágenes color.

2.3. Descripción del Ojo de Bife

El Ojo de Bife es una sección del músculo *Longissimus Dorsi*. Un esquema de las imágenes con las que trabajaremos se observa en la figura 2.1. [17]

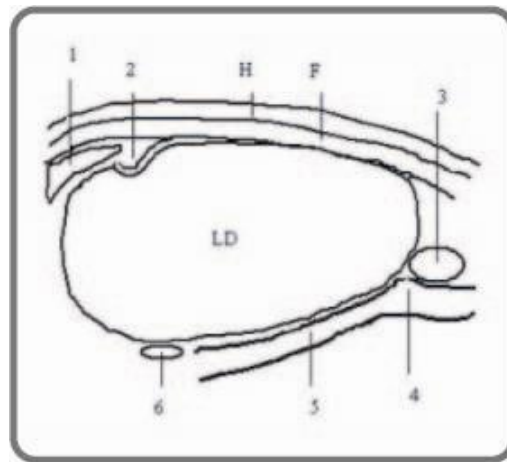


Figura 2.1: Descripción de las distintas partes que conforman el Ojo de Bife

- 1 Espina dorsal
 - 2 Veta de grasa que se introduce en el Ojo de Bife, llamada “hook”(no siempre está presente)
 - 3 Músculo *Longissimus Costarum*
 - 4 Quiebre intercostal
 - 5 Bordes del músculo intercostal, también llamado vías de tren
 - 6 Músculo *Quadratus Lumborum*
- H Piel del animal

F Capa de grasa subcutánea

LD Sección del músculo *Longissimus Dorsi* (Ojo de Bife)

2.4. Otros métodos de evaluación cárnica y trabajos previos

No hay hoy en el mundo un estándar para medir la terneza del bife [11]. No se sabe exactamente hasta qué punto están correlacionados con la terneza el color y el marmoreo o grado de veteado. La terneza de un bife cocido se mide con el índice Warner-Blatzer o WBSF (Warner-Blatzer Shear Force). Este mide la fuerza que debe ejercerse para cortar el churrasco cocinado a 70°C.

Otro tipo de pruebas como medir la resistencia electro-mecánica del bife al introducirle una punta de prueba se han llevado a cabo pero sin suficiente precisión para los intereses de la industria de la carne. En [11], se investigó la correlación entre la terneza y la lectura colorimétrica en el espacio CIEL*a*b observando la existencia de cierta correlación entre el valor de b y la fuerza del corte. Esto es lo que llevó a la universidad de Colorado a desarrollar el *Beefcam*.

Beefcam

El *Beefcam* es un hardware/software comercial que cumple las mismas funciones que tiene este proyecto como objetivo. *Beefcam* utiliza una cámara conectada por un cable de datos y energía a una computadora portable [7].

2.5. Descripción del proceso de faena

A continuación se describe el proceso que atraviesa una res, desde que ingresa al frigorífico hasta que se realiza el corte pistola a 3 costilla, el cual presenta el Ojo de Bife a analizar.

Baño Es obligatorio, se realiza con agua fría. Es necesario lavarlo porque el cuero es el contaminante exógeno más importante. El baño se realiza para limpiar el cuero, a su vez es un antiestresante para el animal (se considera que el agua fría tiene esta característica)

Cajón de insensibilización Se hace para respetar la faena humanitaria. Para no dañar el bulbo raquídeo, para que sigan funcionando los ciclos respiratorios y circulatorios. El bulbo tiene que mantenerse en movimiento. Existen distintos métodos, el más común es el realizado con una pistola neumática, insensibilización eléctrica. El animal, por medio de aturdimiento eléctrico, entra más relajado a la playa de faena.

Colgación Se cuelga al animal por su pata trasera. Es una zona de vómito y degüello.

Sacrificio Consiste de una incisión en la línea ventral del cuello, mientras que con otro cuchillo se hace el corte del gran vaso del cuello. El tiempo de sangrado oscila entre los 3 y 5 minutos. En dicho período se produce la muerte del animal.

Desollado Consiste en retirar el cuero de manera de no contaminar la carne. Luego se realiza una disección del esfínter anal y se engancha, se disecciona y se ata el recto para que no haya contaminación.

Evisceración Se le extraen todas las vísceras al animal y mediante la utilización de una sierra se corta el esternón. Se retiran las vísceras abdominales y respiratorias.

Aserrado y Dressing de la carcasa Consiste en cortar al medio los cuerpos vertebrados (división en dos carcasas y dos canales). Los dos medios canales son destinados a inspección veterinaria. Luego se realiza una secuencia operacional en playa de faena para dar una terminación adecuada a las carcasas.

Lavado de las medias reses El lavado se hace con agua a presión a 45 grados. Si se utiliza agua más caliente, se cambia el color de la carne.

Clasificación y tipificación de la carcasa Se realiza por sistemas oficiales del MGAP.² Se coloca un sello ovalado en el trasero y delantero, el cual da la seguridad de que el producto fué sometido a inspección durante la faena.

²MGAP: Ministerio de Ganadería Agricultura y Pesca

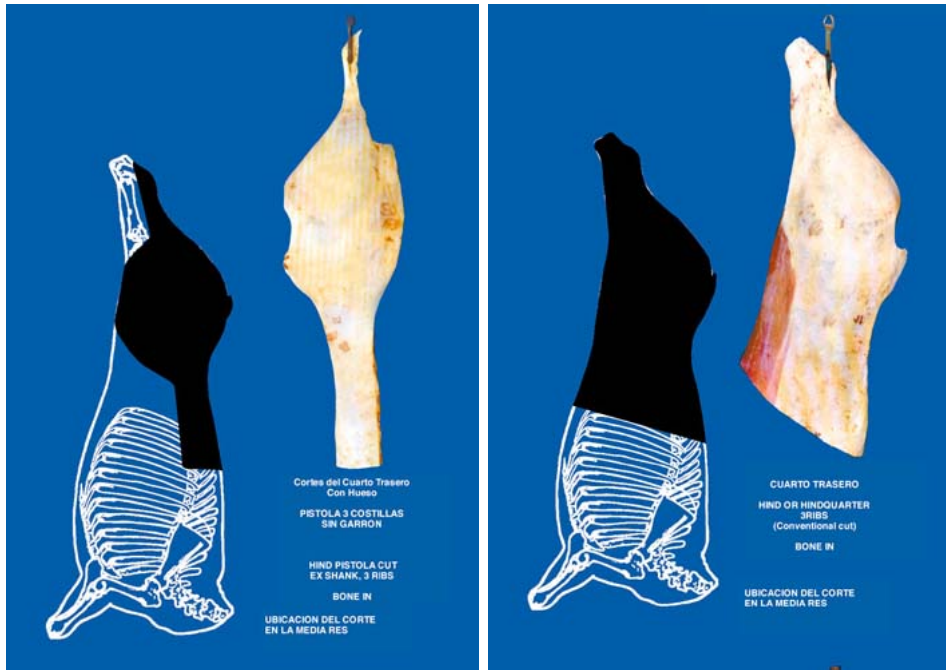


Figura 2.2: Corte Pistola a 3 costillas incompleto y cuarto trasero a 3 costillas

Refrigeración de las carcasas Durante 24 horas se mantienen las carcasas a una temperatura entre 4 y 7 grados celcius.

Destino Las carcasas son destinadas o bien a mercado interno (las medias reses son llevadas a las carnicerías), o van a la sala de procesamiento del frigorífico.

Se procede luego a realizar los cortes a la media res. El deshuesado se puede realizar de dos formas: separar en cortes individuales, o separar por mantas. Dentro del deshuesado por mantas se encuentra el corte Pistola a 3 costillas completo o incompleto, siendo la diferencia entre ambos la presencia o ausencia del garrón, y el cuarto trasero a 3 costillas.

Para la evaluación y clasificación, el equipo toma una imagen del caño de pistola, corte entre la 3a y la 4ta costilla, el cual es representativo del animal.

Capítulo 3

Algoritmo de Segmentación del músculo *Longissimus Dorsi*

3.1. Introducción

Como primer paso, previo a la implementación del algoritmo, se realizó una búsqueda intensiva para encontrar trabajos anteriores sobre segmentación en imágenes de Ojo de Bife. Se observó que la mayoría de los métodos de segmentación del Ojo de Bife se basan en el trabajo de S. Jeyamkondan [6] y [11]. Fue en el algoritmo descrito en dichos trabajos en que se basó nuestra investigación. En el comienzo del proyecto, se implementó un primer prototipo del algoritmo de segmentación del músculo *Longissimus Dorsi* en lenguaje Matlab que si bien no funcionaba correctamente para todas las imágenes, los resultados eran suficientemente satisfactorios como para considerarlo un buen punto de partida para abordar el problema.

3.1.1. Estructura del algoritmo de segmentación

La estructura del algoritmo cuenta básicamente con cuatro etapas que se desarrollarán más adelante, estas son:

Preprocesado de la Imagen

Usando un simple algoritmo de umbralización automática aplicado al canal rojo, se distingue la costilla del fondo que es irregular, como resultado obtenemos una imagen donde aparece sólo la costilla con el fondo negro. Esta será la imagen de partida para el algoritmo.

Umbralización usando el algoritmo FCM

Para distinguir la carne de la grasa se umbraliza en los canales verde y azul. Para tener un umbral adaptativo y robusto, se utiliza el algoritmo Fuzzy C-Means Clustering Algorithm (FCM) como se describirá más adelante. El resultado de esta etapa es una imagen binaria donde toman valores 1 y 0 la carne y la grasa respectivamente.

Envolverte Convexa (Convex Hull), Morfología binaria

Una vez que se tiene una imagen binaria donde aparece la carne y la grasa de la costilla, se debe encontrar cual de éstos se corresponde con el Ojo de Bife. Para esto hay que eliminar las componentes de carne que no pertenecen al músculo *Longissimus Dorsi*.

Eliminación de penínsulas y músculo *Longissimus Costarum*

A partir de la etapa anterior, se tiene un buen candidato a ser el contorno del Ojo de Bife. Sin embargo sucede muchas veces que penínsulas de carne quedan pegadas a este, que “sobreviven” al algoritmo de Convex Hull. Es muy común que aparezca el músculo *Longissimus Costarum* acoplado al Ojo de Bife y es necesario su identificación y extracción.

3.1.2. Análisis del desempeño del algoritmo de segmentación

Para medir el resultado del algoritmo de segmentación se comparan las imágenes segmentadas automáticamente por el algoritmo con las mismas imágenes segmentadas a mano por un experto. En aquellas imágenes para las cuales no contamos con la marca del Ojo de Bife por parte de un experto, procedimos a marcarlas manualmente siguiendo nuestro sentido común y basándonos en los criterios utilizados por los expertos.

El análisis de los resultados está descrito en el capítulo referente a Validación (capítulo 9).

3.2. Preprocesado de la Imagen

El preprocesado de la imagen consiste en una primera segmentación de la imagen, manteniendo sólo la región donde aparece la costilla entera. Se procede a segmentar el fondo de la imagen, que por lo general es oscuro pero irregular. Se usa el método de Otsu en el canal rojo. [41]

“El método de Otsu es un método no iterativo y se basa en asumir que todos los píxeles de la imagen pertenecen a una de dos clases: el fondo o el objeto. Una forma

de particionar una imagen en dos contenedores (clusters), es tratando de hacer cada cluster lo más firme posible” (la menor dispersión). Como se describe en [16], a medida que se varía el umbral, se aumenta la varianza de los pixeles de un cluster (el rango dinámico de los valores de intensidad de los pixeles aumenta), mientras que disminuye para el otro. El objetivo es, pues, elegir el umbral de modo que se minimice el rango dinámico combinado de ambos clusters. Se define entonces la varianza intra-clase (within-class variance) como la suma ponderada de las varianzas de cada cluster:

$$\sigma_{Within(T)}^2 = \eta_B \sigma_B^2(T) + \eta_O \sigma_O^2(T)$$

donde

$$\eta_B(T) = \sum_{i=0}^{T-1} p(i)$$

$$\eta_O(T) = \sum_{i=T}^{N-1} p(i)$$

$p(i)$ es la cantidad de pixeles que tienen el valor de intensidad i , $\sigma_B^2(T)$ es la varianza de los pixeles en el fondo (background) (por debajo del umbral) $\sigma_O^2(T)$ es la varianza de los pixeles en el frente (foreground) (por encima del umbral) y N es el rango dinámico de los niveles de intensidad.

Computar la within-class variance para cada posible nivel de intensidad es computacionalmente muy intenso e ineficiente. Si se resta la within-class variance de la varianza de la distribución combinada, se obtiene la llamada varianza inter-clase (between-class variance):

$$\sigma_{Between}^2(T) = \sigma^2 - \sigma_{Within}^2 = \eta_B(T)[\mu_B(T) - \mu]^2 + \eta_O(T)[\mu_O(T) - \mu]^2$$

donde σ es la varianza combinada y μ es la media combinada. Se observa que la between-class variance es simplemente la varianza ponderada de la media de cada cluster en torno a la media total de la imagen. Sustituyendo $\mu = \eta_B(T)\mu_B(T) + \eta_O(T)\mu_O(T)$ se obtiene

$$\sigma_{Between}^2(T) = \eta_B(T)\eta_O(T)[\mu_B(T) - \mu_O(T)]^2$$

Por lo tanto, para cada posible umbral T :

1. Se separan los pixeles en dos clusters de acuerdo al umbral
2. Se encuentra la media de cada cluster
3. Se eleva al cuadrado la diferencia entre las medias

4. Se multiplica la cantidad de pixeles en un cluster por la cantidad de pixeles en el otro

El umbral óptimo es aquel que maximiza la between-class variance (o lo que es igual, minimiza la within-class variance). La forma anterior envuelve muchas operaciones, dado que se tienen que realizar las operaciones para cada posible umbral. Sin embargo se demuestra que las operaciones no son independientes al pasar de un umbral a otro. Es posible actualizar los valores de $\eta_B(T)$ y de $\eta_O(T)$, y de las respectivas medias de cada cluster $\mu_O(T)$ y $\mu_B(T)$, a medida que se incrementa el umbral T . Utilizando relaciones de recurrencia se puede actualizar el valor de la between-class variance a medida que se testea cada umbral.

$$\begin{aligned}\eta_B(T + 1) &= \eta_B(T) + \eta(T) \\ \eta_O(T + 1) &= \eta_O(T) - \eta(T) \\ \mu_B(T + 1) &= \frac{\mu_B(T)\eta_B(T) + \eta(T)T}{\eta_B(T + 1)} \\ \mu_O(T + 1) &= \frac{\mu_O(T)\eta_O(T) - \eta(T)T}{\eta_O(T + 1)}\end{aligned}$$

donde $\eta(T) = p(T)$. Como resultado de esta primera etapa se tiene una imagen en que aparece el objeto costilla original con el fondo negro, es decir, todos los pixeles que pertenecen a la costilla con su valor original y todos los pixeles del fondo (los que no pertenecen a la costilla) con valor 0.

En la figura 3.1 se observa una imagen de partida para el algoritmo y la salida una vez quitado el fondo.

3.3. Algoritmo FCM

En esta etapa del algoritmo se debe separar la grasa de la carne a partir de la imagen obtenida del preprocesado en la cual se tiene la costilla entera con hueso, grasa y carne, esto sobre un fondo negro (pixeles valor 0).

Por una cuestión de sencillez, en esta etapa se decidió trabajar sobre los valores de color de los pixeles en el espacio RGB (rojo, verde, azul) sin entrar en el uso de coherencia espacial ni ninguna otra noción de la estructura del objeto (más adelante sí se usará). Por lo tanto en esta etapa se tomarán umbrales en los canales verde y azul, que separen los pixeles de la imagen que pertenecen a una región de carne y los pixeles que pertenecen a una región de grasa. La segmentación se realiza en los canales verde y azul porque es de los cuales se puede extraer más información ya que estos componentes se encuentran presentes en la grasa y no en la carne mientras el rojo esta presente en ambas partes. La figura 3.2 muestra los tres canales de una

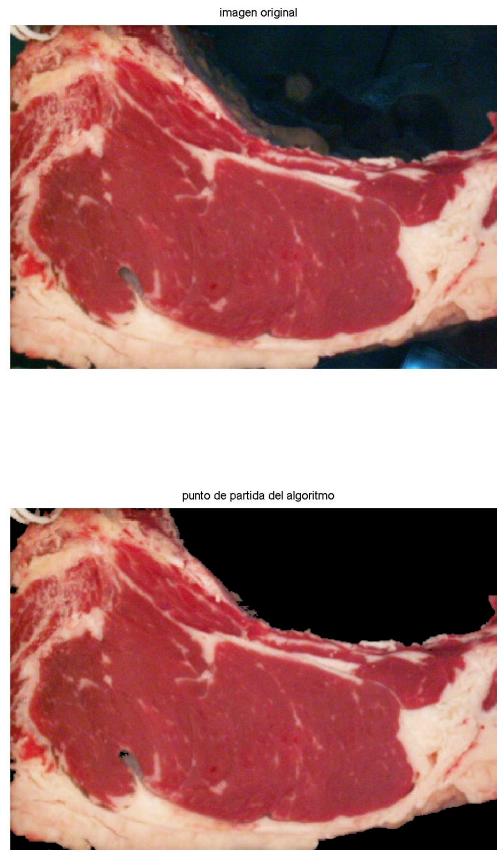


Figura 3.1: Imagen original e imagen luego de quitado el fondo

imagen ejemplo, en las cuales se observa que la diferencia entre grasa y carne es más notoria en los canales verde y azul.

Para que el algoritmo de segmentación del Ojo de Bife sea robusto, la elección de un buen umbral debe ser adaptativa, no pudiendo ser fija. Sin embargo, la tarea de umbralización automática se ve dificultada en esta etapa ya que los histogramas de los canales verde y azul presentan un solo pico, como se ve en la figura 3.3, que corresponde a la región de la carne, ya que es donde hay mayor cantidad de píxeles. Notar que el fondo (ceros) no se representa en el histograma.

Los algoritmos de umbralización tradicionales, que se basan en la presencia de dos picos fácilmente identificables en el histograma, no pueden ser de utilidad para solucionar este problema. Para asegurar la robustez de la segmentación, lo óptimo es



Figura 3.2: En la figura se observan los tres canales de una imagen ejemplo

usar un algoritmo que simule la percepción del ojo humano en esta etapa. Una buena aproximación a esto se puede hacer utilizando lógica Fuzzy.

Sin entrar en detalle, la lógica Fuzzy, es una extensión de la lógica Booleana que maneja el concepto de verdad parcial. Mientras que en la lógica clásica todo puede ser expresado en términos binarios (0 o 1, sí o no, carne o grasa), la lógica Fuzzy reemplaza los valores booleanos de verdad con valores que representan un determinado grado de verdad, por ejemplo valores continuos entre 0 y 1.[20]

Un algoritmo de lógica Fuzzy muy utilizado para la umbralización automática, es el Fuzzy C-Means Clustering Algorithm (FCM). El algoritmo FCM permite separar los píxeles en dos distintos clusters, de acuerdo a un criterio de mínimos cuadrados como se detalla más adelante [6].

3.3.1. Introducción Teórica al Algoritmo FCM

Asumamos que se quiere clasificar los píxeles de una imagen $I(x, y)$ en C clases. Cada píxel en la posición (x, y) tiene el valor de membresía fuzzy $u_i(x, y)$. La técnica “clustering” está basada en la minimización de una función objetiva que contabiliza la distancia entre los centros de cada cluster y los datos (el valor del píxel). En este caso, la función objetiva esta dada por:

$$J(U, M) = \sum_{x,y} \sum_{i=1}^C (u_i(x, y))^m |d_i(x, y)|^2 \quad (3.1)$$

donde: U es la partición de clases del fuzzy c-means, M es el conjunto de los centros de los clusters, m es el exponente fuzzy, valor mayor a 1.

Dado un valor de píxel $I(x, y)$, la medida

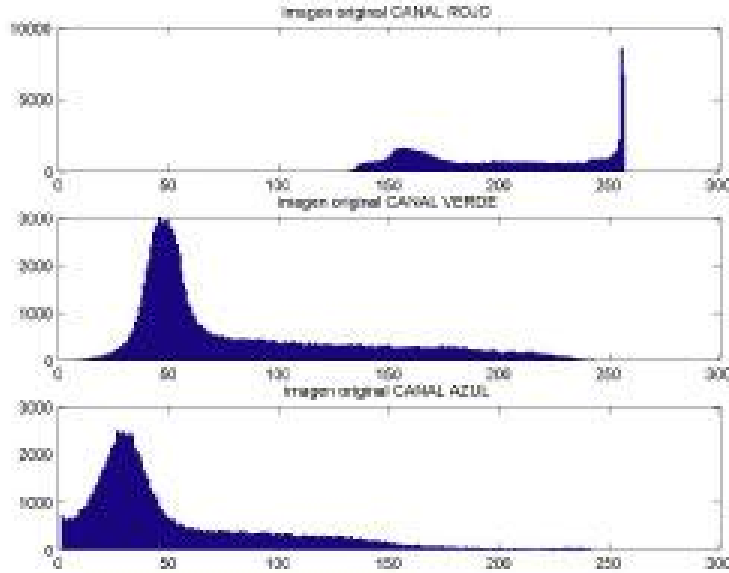


Figura 3.3: Histograma de los diversos canales de la imagen

$$|d_i(x, y)| = |I(x, y) - \mu_i| \quad (3.2)$$

es la distancia entre el valor del píxel y el centro del cluster u_i para la i -ésima clase. Esta distancia es ponderada en la función objetiva por el valor de membresía fuzzy para el píxel correspondiente. La función objetiva es iterativamente minimizada, sujeta a la siguiente condición:

$$\sum_{i=1}^C u_i(x, y) = 1$$

y

$$u_i > 0 \quad (3.3)$$

Al inicio de la iteración, tanto los valores de membresía como los centros de los píxeles son asignados aleatoriamente. A continuación, el esquema de iteración procede a contabilizar las siguientes cantidades, una a continuación de la otra, para cada paso:

$$u_i(x, y) = 1 / \left(\sum_{x,y} \sum_{l=i}^C \left(\frac{d_l(x, y)}{d_i(x, y)} \right)^{m/(m-1)} \right) \quad (3.4)$$

$$\mu_i = \frac{\sum_{x,y} (u_i(x,y))^m I(x,y)}{\sum_{x,y} (u_i(x,y))^m} \quad (3.5)$$

El esquema de salida es la partición U de los píxeles en C clases (es decir, el valor de membresía de cada píxel en cada uno de los cluster). Si bien el FCM es más robusto que la simple umbralización, es computacionalmente intenso, pues es un proceso iterativo. El tiempo necesario aumenta exponencialmente con el tamaño de la imagen, por lo que una buena práctica es realizar un submuestreo de la misma para reducir la cantidad de píxeles y por lo tanto la cantidad de cuentas.

3.3.2. Resultados obtenidos al aplicar FCM

En esta parte del algoritmo, se aplica el algoritmo FCM a los canales verde y azul. Cada canal de color de la imagen es una imagen de grises. Al algoritmo se le pasa a la entrada, un vector con todos los valores de los píxeles de la imagen de grises de los canales verde y azul, eliminando previamente los píxeles con valor 0. El algoritmo entonces separa los píxeles de estos canales en dos clusters y devuelve los mismos, así como sus centros.

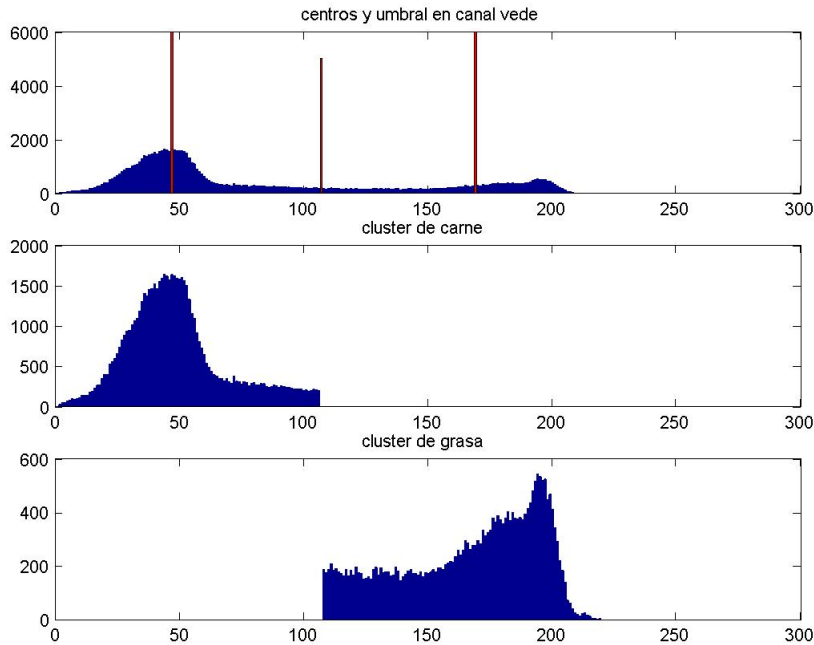


Figura 3.4: Separación en clusters por el algoritmo FCM

Una vez que se tienen los clusters, es fácil determinar el umbral que los separa. Los píxeles del cluster más oscuro (más cercano al valor cero) son los que representan la carne. Los píxeles del cluster más claro (más cercano a 255) son los que representan grasa. Por lo tanto se umbraliza la imagen y se obtiene una imagen binaria con valor 1 en donde hay carne y 0 en el resto de la imagen.

Se realiza la operación lógica AND de las imágenes binarias obtenidas de cada canal para trabajar con una sola imagen binaria. A modo de ejemplo, en la figura 3.5, se muestra el antes y el después del algoritmo FCM.

3.4. Envoltente Convexa (Convex Hull) y Morfología Binaria

En esta etapa se trabaja con el resultado del algoritmo FCM, que es una imagen binaria que representa todo lo que es carne en la imagen. Ahora se trata de identificar de todos los componentes de carne, cual de todos es el Ojo de Bife.

Como el Ojo de Bife es compacto y conexo, su identificación se lleva a cabo con una iteración que va erosionando la imagen binaria y manteniendo la parte conexas más grande hasta eliminar los pedazos de carne adyacentes al Ojo.

Esta es una de las etapas más críticas del algoritmo, debido a la existencia de parámetros que se deben fijar, como la máscara que se utilice para las operaciones binarias y el número de iteraciones. Estos son fundamentales para un buen resultado y son parámetros que el programa no encuentra automáticamente, por lo tanto deben ser elegidos previamente de forma experimental, de tal manera que se ajusten a la mayoría de las imágenes con las que se vaya a trabajar. Esto compromete considerablemente la robustez del algoritmo, pero con ayuda de la etapa posterior se logra complementar este compromiso y mejorar la robustez.

3.4.1. Convex Hull

El problema a solucionar consiste en, dado un conjunto de puntos, encontrar la envoltente o encontrar el menor polígono convexo que los contenga a todos, tal como se muestra en la figura 3.6. Un polígono convexo es un polígono cuyos ángulos interiores son todos menores o iguales a 180 grados.

Existen distintos algoritmos que solucionan el problema del cálculo del convex hull, sin embargo algunos resultan ser menos eficientes que otros en función del tiempo que toman para resolver el problema.

3.4.2. Morfología

A continuación se describen en forma somera los procesos de dilatación y erosión, según se encuentran en [16].

Dilatación

La *dilatación* es una operación que “agranda” o “engorda” objetos en una imagen binaria. La manera en que lo hace está controlada por la forma del *elemento estructural*. Computacionalmente, los elementos estructurales son representados típicamente como una matriz de ceros y unos. Además, el origen del elemento estructural debe ser claramente definido. La dilatación es el proceso de trasladar el origen del elemento estructural a través del dominio de la imagen, y chequear donde se superpone con valores de píxel uno. La imagen de salida tiene unos en cada lugar del origen donde el elemento estructural se superpone con al menos un píxel de valor uno de la imagen binaria de entrada.

Erosión

La erosión, en cambio, encoje un objeto en una imagen binaria. Al igual que en la dilatación, es controlada mediante el *elemento estructural*. La erosión de una imagen A por un elemento estructural B es el conjunto de todos los puntos tales que B trasladado desde el origen hasta esos puntos está contenido en A. Puede pensarse en cierto sentido como el proceso opuesto a la dilatación.

3.4.3. Proceso iterativo

Como ayuda para el proceso iterativo, en cada etapa se calcula el convex hull de la imagen.

Una buena medida de qué tan convexo es el objeto, es calcular el porcentaje de área que este ocupa dentro de su convex hull. Por lo tanto lo que hace este algoritmo es erosionar iterativamente la imagen hasta cierto punto y luego, el objeto que tuviera más porcentaje de área dentro de su convex hull es el que el algoritmo elegirá como segmentación del Ojo de Bife [6]. La iteración, a partir de la imagen binaria de los píxeles de carne, consiste en:

1. Tomar la región conexas más grande.
2. Rellenar agujeros en la imagen binaria.
3. Calcular el área del objeto.
4. Erosionar

5. Tomar la región conexas más grande
6. Calcular el convex hull
7. Calcular el área del objeto y el área del convex hull
8. Guardar estos datos y la imagen binaria
9. Repetir desde el paso 4 hasta que se cumpla alguno de los criterios de parada

Los criterios de parada, que se hallaron empíricamente son:

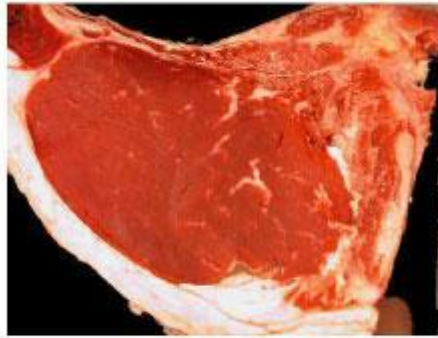
1. Cuando el objeto redujo su área inicial a menos del 80 %
2. 15 iteraciones.

Los criterios de parada, tanto como la máscara que se utilice para la erosión, son los parámetros críticos que comprometen la robustez del algoritmo y fueron elegidos a partir de resultados experimentales que se obtuvieron con la base de imágenes de prueba.

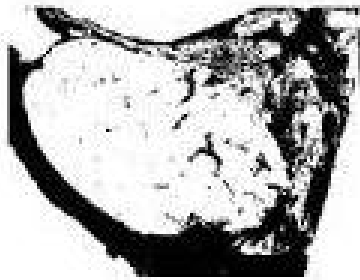
La máscara que se utilizó es un rombo de 3x3. En la figura 3.7 se muestra cómo en cada paso de la iteración se van eliminando trozos del objeto de carne que no forman parte del músculo que se busca.

Una vez terminada la iteración, se elige como resultado, de entre todas las imágenes binarias que se guardaron, el objeto binario cuya área ocupe el mayor porcentaje dentro de su convex hull (el objeto más compacto). Luego, para recomponer las partes del objeto que se redujeron por causa de la erosión, se aplica la misma cantidad de dilataciones que de erosiones efectuadas, utilizando el mismo elemento estructural. El contorno de este objeto es la primera segmentación del Ojo de Bife que se tiene. En la siguiente etapa se llegará a una segmentación definitiva, que en algunos casos como se verá coincide con esta.

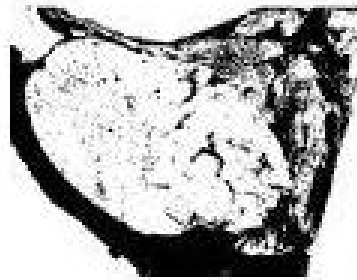
En la figura 3.8 se observa el diagrama de flujo del algoritmo.



Resultado del algoritmo FCM



Canal Verde



Canal Azul

AND de las dos imagenes binarias



Figura 3.5: En la figura se observa la imagen original, previa a la ejecución del algoritmo FCM, el resultado de la segmentación de los canales verde y azul, y el AND de los mismos obtenido con el algoritmo

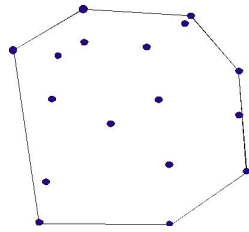


Figura 3.6: Menor polígono convexo que envuelve un grupo de puntos

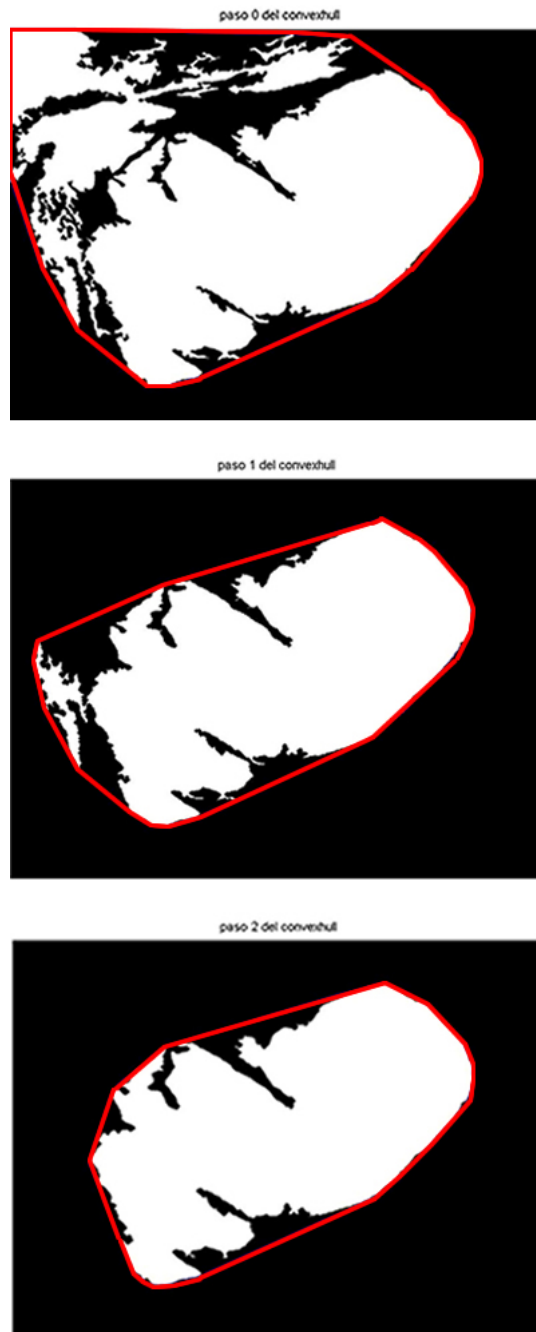


Figura 3.7: Tres pasos de la iteración del Convex Hull

CAPÍTULO 3. ALGORITMO DE SEGMENTACIÓN DEL MÚSCULO
LONGISSIMUS DORSI

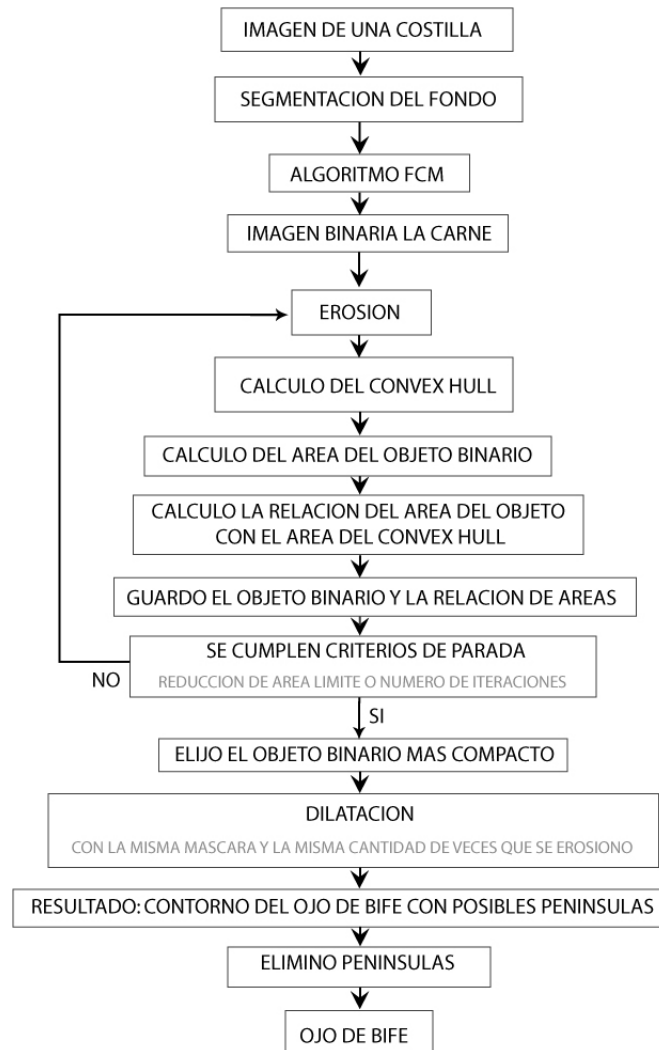


Figura 3.8: Diagrama de flujo del algoritmo de segmentación

3.5. Eliminación de penínsulas y músculo *Longissimus Costarum*

Esta etapa del algoritmo fue enteramente desarrollada por nosotros. En esta etapa se trabaja con información de la forma del contorno del músculo segmentado. Introduce una mejora sustancial en aquellos casos en los que el músculo *Longissimus Costarum*, aparece pegado al músculo *Longissimus Dorsi*, situación que el algoritmo del Convex Hull no solucionaba.

El objetivo de este algoritmo es eliminar las penínsulas que presente el objeto binario que se tiene como posible Ojo de Bife.

Para esto, primero se vectoriza el contorno del objeto, obteniendo un vector ordenado de las coordenadas de todos los puntos del contorno. Este vector se submuestra con una relación de 20 para aliviar la carga computacional. El vector presenta en promedio alrededor de dos mil puntos¹ y como se verá a continuación este algoritmo es de orden n^2

Una vez que se tiene los puntos que representan el contorno, procedemos a comparar todas las parejas de puntos midiendo entre éstos la distancia sobre la curva y la distancia euclídea. Para considerar una pareja de puntos como candidatos a ser el nacimiento de una península, deben cumplir dos condiciones:

1. La relación entre la distancia sobre la curva entre dos puntos y su distancia euclídea debe ser mayor a cierto parámetro que se hayó experimentalmente.
2. El área encerrada por la curva entre estos debe ser del mismo signo que el área del contorno entero. Esto significa que se está frente a una península saliente y no una entrante.

El algoritmo recorre el vector de los puntos del contorno y busca todas las parejas de puntos en que cumplen las mencionadas condiciones, es decir, todos los candidatos a ser penínsulas. Estos candidatos se almacenan en un vector y luego se evalúa cual de estas parejas de puntos tienen la menor distancia euclídea entre ellos, es decir los que están más cerca entre sí, que corresponden al nacimiento de la península. A continuación se eliminan todos los puntos del vector que están entre los dos puntos que cumplieron las condiciones. Estos pasan a ser adyacentes en el vector y por lo tanto la península queda eliminada del contorno.

Se repite el algoritmo hasta que no se encuentra ninguna pareja de puntos que cumplan las condiciones, es decir hasta haber eliminado todas las penínsulas.

La iteración consiste básicamente en:

1. Se vectoriza el contorno de la primer segmentación del Ojo de Bife (resultado del Convex Hull).

¹Para imágenes de 640x480 pixeles

2. Se submuestra este vector con una relación de submuestreo de 20
3. Se recorre el vector midiendo entre cada pareja de puntos la distancia siguiendo la curva y la distancia euclídea.
4. Si se cumple determinada relación entre éstas distancias, y el área de la curva es de signo tal que se trata de una península saliente, se guarda en un vector auxiliar la pareja de puntos y sus respectivas distancias.
5. Al terminar de recorrer el vector, se selecciona de las parejas de puntos halladas que cumplen la condición, la de menor distancia euclídea entre los dos puntos.
6. Se eliminan los puntos del vector del contorno que estén entre éstos dos puntos (se elimina la península).
7. Se vuelve al punto 3 hasta que no se encuentran más puntos que cumplan las condiciones.

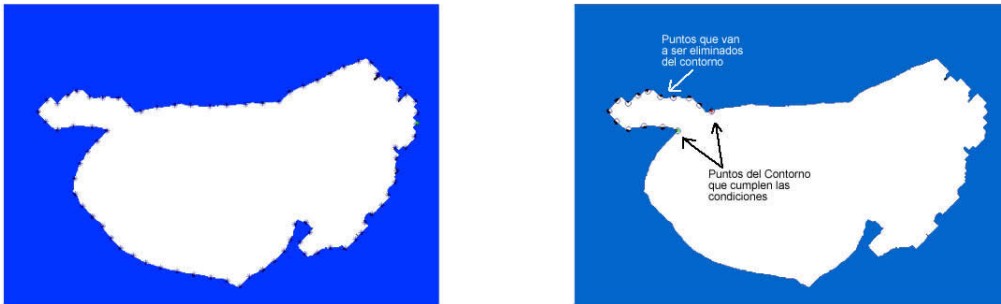


Figura 3.9: Vector de puntos del contorno y península a ser eliminada (submuestreado)

3.6. Situaciones en las que fallaba el algoritmo

En la figura 3.11 se muestran cuatro casos en los que el algoritmo funciona correctamente.

En la figura 3.12 se muestran cuatro casos en los que el algoritmo fallaba en sus primeras versiones. Las razones por las que fallaba el algoritmo eran:

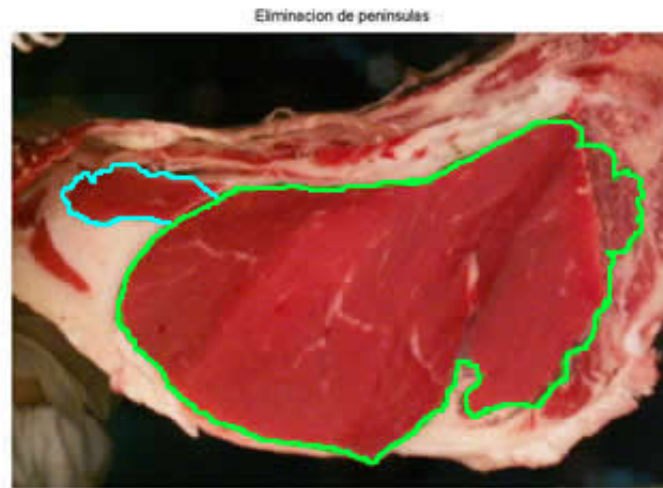


Figura 3.10: Segmentación antes y después de la eliminación de penínsulas

1. Falta de grasa para distinguir la frontera del Ojo de Bife (imagen superior izquierda).
2. Suciedad en la costilla debido al el aserrín que se produce al utilizar una sierra para el corte, que el programa confunde con grasa (imagen superior derecha).
3. Penínsulas acopladas al Ojo de Bife como el músculo *Longissimus Costarum* (imagen inferior izquierda).
4. Brillos en la carne debidos a la mala iluminación, los cuales hacen ver como “blanco” partes que deberían ser “rojas” (imagen inferior derecha).

Algunos de estos problemas se solucionaron en las siguientes versiones del algoritmo. Por ejemplo, con el algoritmo de eliminación de penínsulas se elimina el músculo *Longissimus Costarum* en los casos que se encuentra pegado al Ojo de Bife. Los brillos en la imagen se corrigieron cambiando el hardware, haciendo que la iluminación sea indirecta, en lugar de directa como lo hacía el primer prototipo. En cuanto a la suciedad debida al aserrín, la única manera de evitar este problema es limpiando la costilla antes de adquirir las imágenes.

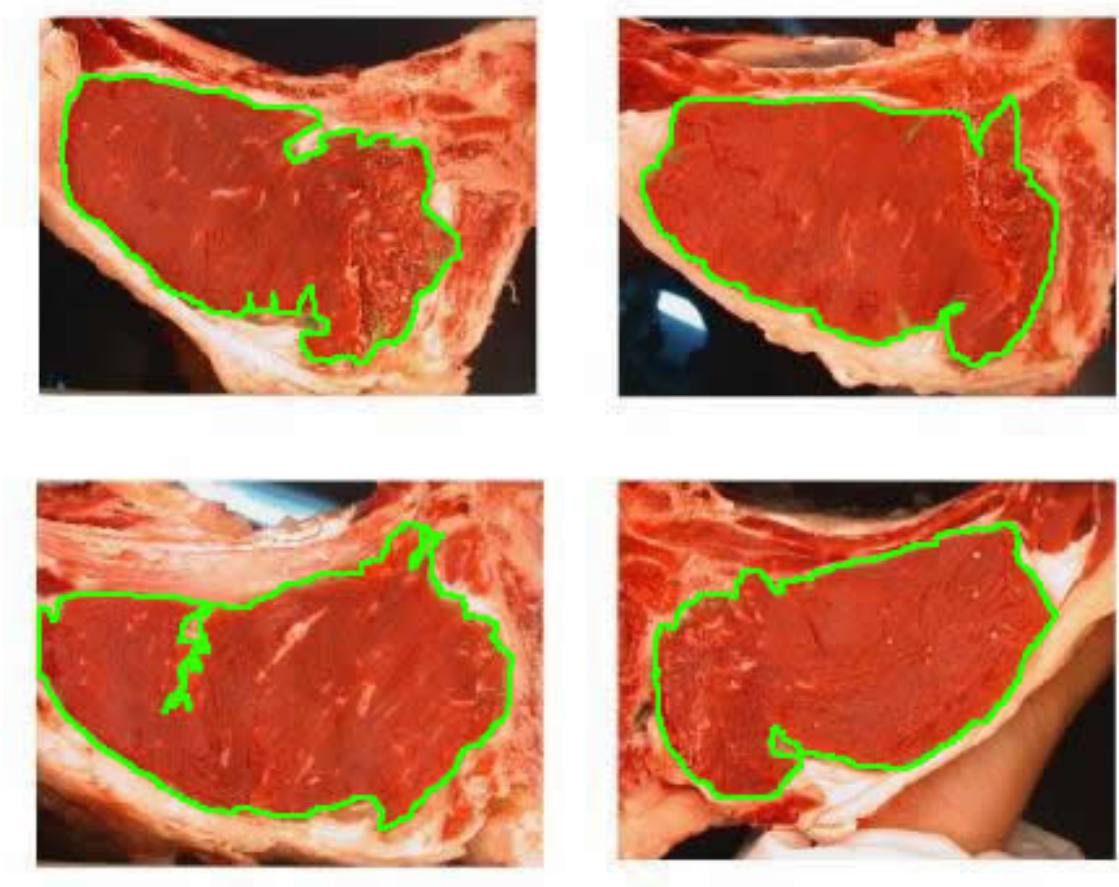


Figura 3.11: Ejemplos de resultados óptimos del algoritmo

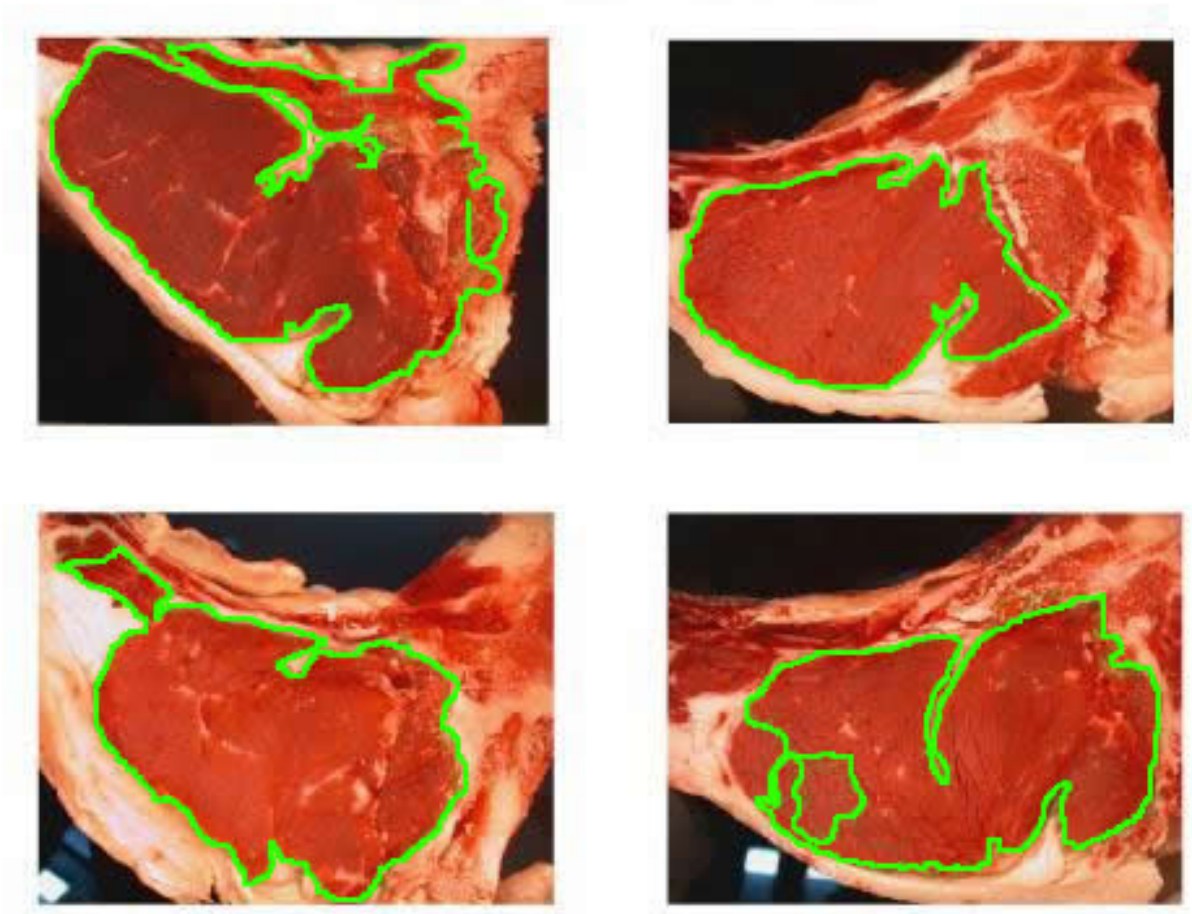


Figura 3.12: Ejemplos en los cuales el algoritmo falla

Capítulo 4

Obtención de Medidas

4.1. Introducción

Las medidas que el programa tomará automáticamente de la imagen del Ojo de Bife son:

1. Área del músculo *Longissimus Dorsi*.
2. Porcentaje del marmoreo.
3. Espesor de grasa subcutánea.
4. Color.

El programa realiza automáticamente dichas medidas, a partir de la segmentación del Ojo de Bife.

4.2. Área del Ojo de Bife

4.2.1. Cálculo del área

El área del músculo *Longissimus Dorsi* se calcula a partir de la imagen binaria del contorno del Ojo de Bife. Se obtiene de la cantidad de píxeles que tiene el objeto binario que representa al Ojo de Bife, resultado del algoritmo de segmentación del músculo. Debido a que la cámara está fija en el equipo de toma de imágenes y la distancia entre la cámara y la costilla es constante, se conoce la relación que hay entre píxeles y centímetros (ver capítulo referente a calibración geométrica). Con dicha relación se obtiene el área del músculo *Longissimus Dorsi* en centímetros cuadrados.

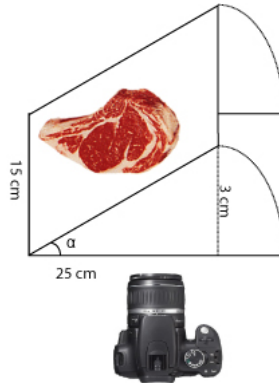


Figura 4.1: Cálculo del error debido al mal apoyo de la costilla.

Error introducido por el apoyo incorrecto de la costilla sobre el aparato de toma de imágenes

Dado que no se implementó un sistema de verificación automática del correcto apoyo de la costilla sobre el aparato de toma de imágenes, se calculó el error que se cometería en la medida del área en caso de que la costilla no estuviese apoyada correctamente sobre el aparato. Esto es, en el caso en que el plano de la costilla no estuviera paralelo al plano de la fotografía. Se tomó como referencia una separación de tres centímetros entre la parte más alejada de la costilla y el plano de la fotografía. Se calculó el coseno del ángulo que formarían el plano de la costilla con el plano de la fotografía (ángulo α en la figura 4.1). Sabiendo que el agujero por donde se toma la imagen es de 20 cm x 15 cm, el coseno de este ángulo es $\cos(\arcsin(3/20)) = 0,9886$. Tomando como referencia un área de Ojo de Bife promedio de 90 centímetros cuadrados, el error en el cálculo del área sería aproximadamente de 1 centímetro cuadrado. Este error es insignificante, tomando en cuenta que una separación de tres centímetros como se muestra en la figura sería un caso particular de utilización del aparato.

4.3. Porcentaje de Marmoreo

El marmoreo (del término inglés *marbling*) se le llama a los pequeños trozos de grasa que se encuentran dentro del músculo del Ojo de Bife. Esta grasa intramuscular está infiltrada, dispersa dentro del músculo, de ahí el término marmoreo. La grasa intramuscular hace que la carne sea más tierna y jugosa, por lo tanto más agradable para el consumidor final.

Dentro del músculo, también podemos encontrar venas que no deben ser confundidas con la grasa intramuscular. Dichas venas por lo general son más oscuras que la



Figura 4.2: Marmoreo del Ojo de Bife

grasa y tienen formas alargadas y más grandes que la grasa intramuscular.

El marmoreo puede ser clasificado según la cantidad de grasa intramuscular que se encuentra en el músculo. En la figura 4.3 se muestra la clasificación de la USDA según el nivel de marmoreo.

El programa discrimina según su forma (de acuerdo a los valores del perímetro y del área de los elementos clasificados como grasa intramuscular), los posibles trozos de grasa como se ve en la figura 4.4. Los que no cumplen determinadas condiciones de tamaño y forma (los mayores), no pueden ser considerados como parte de la grasa intramuscular (posiblemente se deba a venas que se prolongan dentro del Ojo de Bife) y no se tienen en cuenta al calcular el porcentaje.

El porcentaje del marmoreo es simplemente la relación entre el área que ocupan estos trozos de grasa intramuscular sobre el área total del Ojo de Bife. Según [42], los valores aceptables de marmoreo, que influyen positivamente en el sabor de la carne, son los que se encuentran entre 3% y 7%.

4.4. Espesor de Grasa Subcutánea

Antes de pasar al desarrollo del algoritmo utilizado para hallar la medida del espesor de grasa intramuscular, es necesario precisar algunos conceptos sobre los atributos topológicos de las imágenes. El marco teórico que se detalla a continuación fue extraído del trabajo de Marcos Martín [44]

4.4.1. Atributos topológicos de una imagen

Los atributos topológicos de imágenes binarias, son propiedades invariantes al objeto y permiten caracterizarlo de manera única. Sea una imagen binaria cuyos valores son

$$I(x, y) = \begin{cases} 1 & \text{si el punto pertenece al objeto} \\ 0 & \text{el punto pertenece al fondo} \end{cases} \quad (4.1)$$

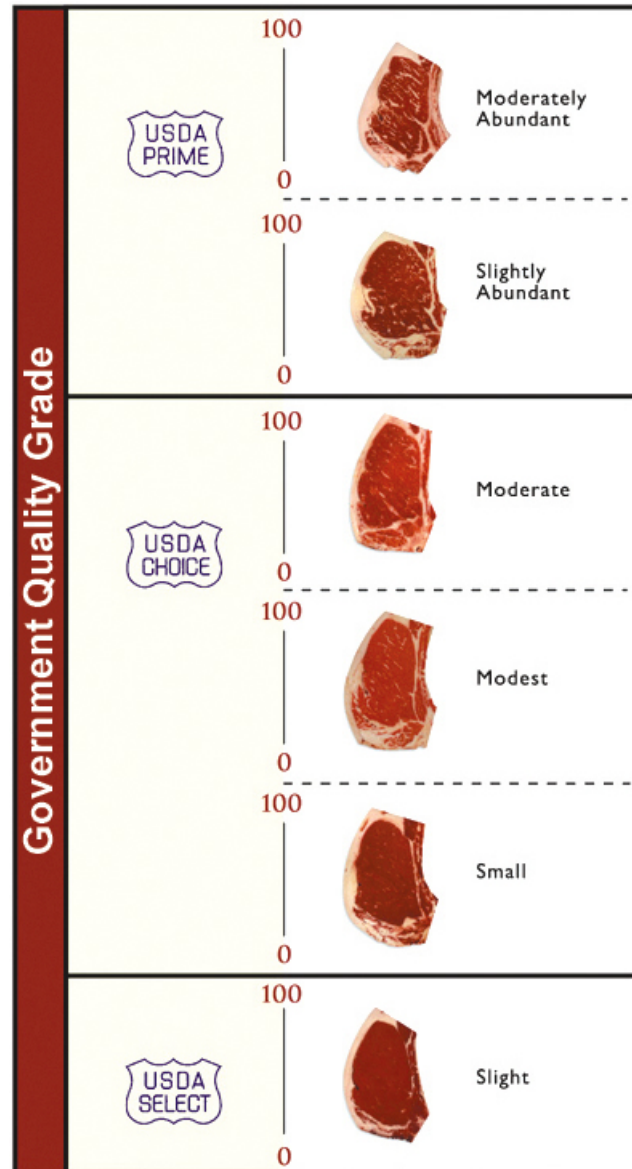


Figura 4.3: Clasificación de la USDA

Momentos

La geometría de una región plana se basa en el tamaño, la posición, la orientación y la forma. Todas estas medidas están relacionadas con una familia de parámetros llamada momentos. Los momentos de orden $p + q$ de una imagen $I(x, y)$ se definen como

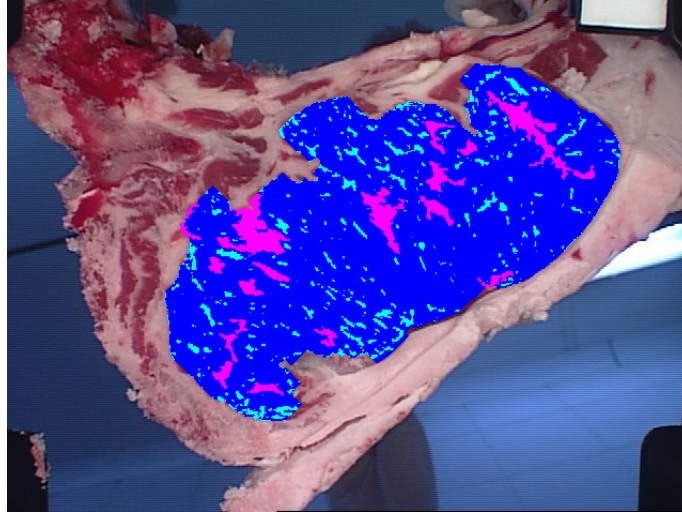


Figura 4.4: Discriminación entre lo que es grasa intramuscular y lo que no

$$M_{p,q} = \sum_{x=1}^N \sum_{y=1}^M x^p y^q I(x, y) \quad (4.2)$$

Por ejemplo, el momento de orden 0 es

$$M_{0,0} = \sum_{x=1}^N \sum_{y=1}^M I(x, y) = A \quad (4.3)$$

que corresponde al área del objeto.

Centro de Masa

La definición del centro de masa, desde el punto de vista de la física es el punto de un objeto que tiene la misma cantidad de objeto en cualquier dirección. En el caso del tratamiento de imágenes se utiliza como punto de referencia del objeto y viene dado por

$$\bar{x} = \frac{\sum_{x=1}^N \sum_{y=1}^M x I(x, y)}{A} \quad (4.4)$$

$$\bar{y} = \frac{\sum_{x=1}^N \sum_{y=1}^M y I(x, y)}{A} \quad (4.5)$$

lo cual puede reescribirse en función de los momentos

$$\bar{x} = \frac{M_{1,0}}{M_{0,0}} \quad (4.6)$$

$$\bar{y} = \frac{M_{0,1}}{M_{0,0}} \quad (4.7)$$

Momentos centrales

Para poder hacer una descripción independiente de la posición del objeto, los momentos pueden calcularse respecto al centro de gravedad. Son los *momentos centrales* o *momentos estadísticos*

$$\mu_{p,q} = \sum_{x=1}^N \sum_{y=1}^M (x - \bar{x})^p (y - \bar{y})^q I(x, y) \quad (4.8)$$

Si los momentos son conocidos, es fácil calcular los momentos centrales a partir de los primeros. Por ejemplo se tiene que

$$\begin{aligned} \mu_{0,0} &= M_{0,0} \\ \mu_{2,0} &= M_{2,0} - \frac{M_{1,0}^2}{M_{0,0}} \\ \mu_{1,1} &= M_{1,1} - \frac{M_{1,0}M_{0,1}}{M_{0,0}} \\ \mu_{0,2} &= M_{0,2} - \frac{M_{0,1}^2}{M_{0,0}} \end{aligned} \quad (4.9)$$

Ejes principales

Los ejes principales de un objeto se calculan a partir de los valores propios de la *matriz de inercia I*

$$I = \begin{bmatrix} \mu_{2,0} & \mu_{1,1} \\ \mu_{1,1} & \mu_{0,2} \end{bmatrix} \quad (4.10)$$

Los momentos principales son los valores propios de la matriz I que vienen dados por

$$\lambda_{1,2} = \frac{1}{2}(\mu_{2,0} + \mu_{0,2}) \pm \frac{1}{2}\sqrt{\mu_{2,0}^2 + \mu_{0,2}^2 - 2\mu_{2,0}\mu_{0,2} + 4\mu_{1,1}^2} \quad (4.11)$$

El ángulo del eje principal se calcula a partir del valor propio más grande, mientras que el ángulo del eje secundario es el perpendicular a este

$$\phi = \tan^{-1} \left(\frac{\lambda_{max} - \mu_{2,0}}{\mu_{1,1}} \right) \quad (4.12)$$

4.4.2. Algoritmo de cálculo del espesor de grasa

El cálculo del espesor de grasa subcutánea consiste en la aproximación de los contornos inferiores y superiores de la grasa subcutánea del Ojo de Bife, para luego realizar la medida de dicho espesor en dos puntos.

Se parte de la información que se obtiene de aplicar el algoritmo FCM a la imagen una vez quitado el fondo. El mismo separa en dos clusters, en uno predominantemente se encuentran los pixeles que representan carne y en el restante aquellos que representan grasa. En este último se encuentra la parte de la grasa subcutánea que se quiere medir. La figura 4.5 muestra el conjunto de pixeles clasificados como grasa.

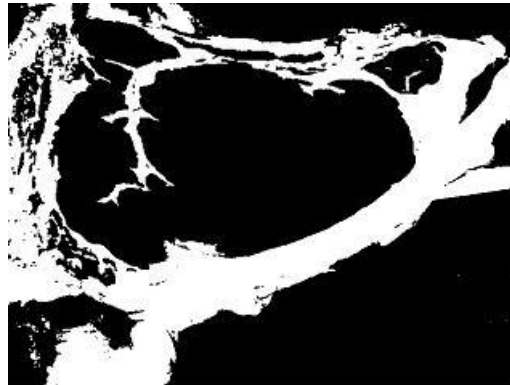


Figura 4.5: Pixeles clasificados como grasa

La grasa subcutánea se encuentra en la parte inferior de Ojo de Bife. Para determinar la parte media inferior del Ojo de Bife se calculan los Momentos Espaciales del mismo. A partir de ellos, se determina el centro de masa y el eje principal. Cortando según el eje principal el cluster de grasa y manteniendo únicamente la parte inferior del mismo, se obtiene una buena aproximación del borde de la grasa subcutánea o grasa de cobertura, como se muestra en la figura 4.6, en donde los valores del cluster de grasa fueron invertidos solamente con el propósito de la exposición de los resultados.

La aproximación del borde superior y el borde inferior se realiza utilizando una interpolación mediante Spline cúbicas. Para cada borde se utilizan tres puntos que son la intersección de tres rectas trazadas perpendiculares al eje principal del Ojo del Bife con el corte inferior del cluster de grasa, de forma de delimitar la zona que se desea medir.

Una vez que se obtienen las aproximaciones del borde superior e inferior de espesor de grasa subcutánea, se obtiene la distancia entre ambas curvas en dos puntos, cuyas medidas representan el espesor de grasa subcutánea. Los puntos en donde se realizan las medidas fueron obtenidos trazando rectas perpendiculares a la aproximación del borde superior del espesor de grasa subcutánea, en la mitad y en el cuarto superior

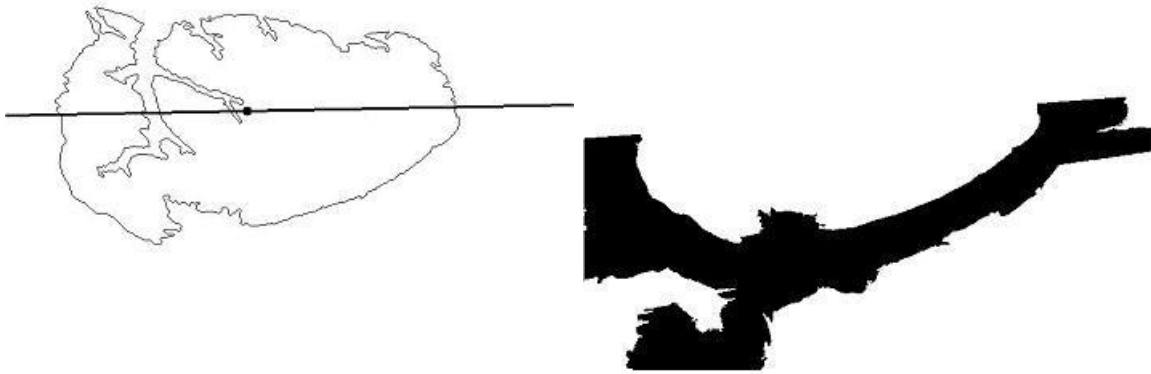


Figura 4.6: En las figuras se observan el eje principal y el baricentro del Ojo de Bife (izquierda), y la parte que resta luego de realizar el corte (derecha)

(el cuarto más cercano al músculo *Longissimus Costarum*)(figura 4.7). Las medidas realizadas en dichos puntos pretenden caracterizar el espesor de la grasa.

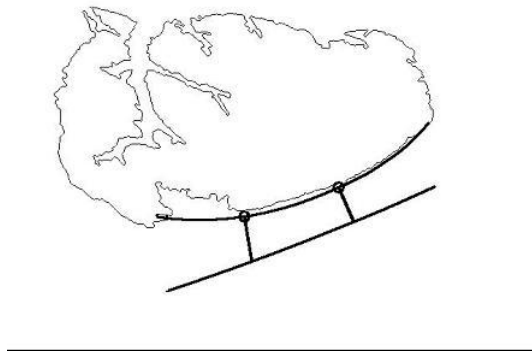


Figura 4.7: Resultado final del algoritmo de medida del espesor de grasa intramuscular

En la imagen 4.8 se muestra una captura del programa, donde marca las medidas del espesor de grasa y muestra el valor numérico de un promedio de ambas.

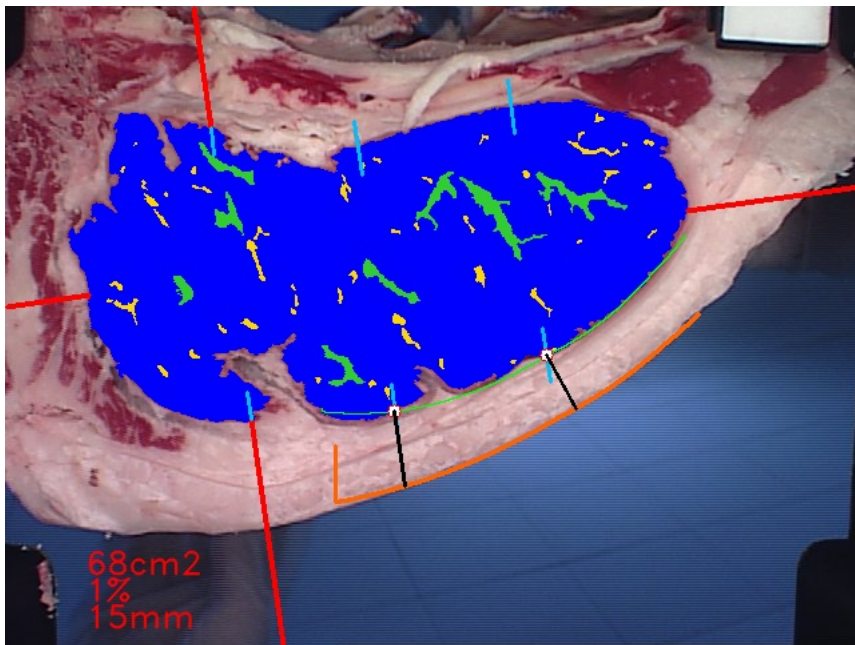


Figura 4.8: Captura del programa donde marca el espesor de grasa

4.5. Medidas de color en el Ojo de Bife

Para estimar las medidas colorimétricas en el Ojo de Bife, se aplica el algoritmo de estimación de los parámetros CIELab al Ojo de Bife luego de quitado el marmoreo del mismo, como se muestra en la figura 4.9, hallando un promedio de los valores resultantes de dicho proceso. Para la estimación de los valores CIELab se utiliza un polinomio de 10 términos para realizar el mapeo entre los espacios RGB y CIELab. Las medidas referentes al color se profundizan en el capítulo *Colorimetría y Calibración de Color*.

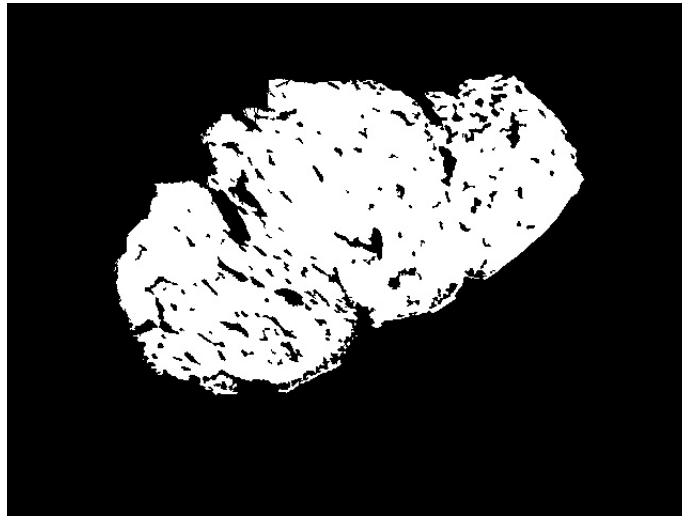


Figura 4.9: Imagen del Ojo de Bife luego de quitado el marmoreo

Capítulo 5

Colorimetría y Calibración de color

5.1. Introducción

En el presente capítulo se brinda un marco teórico en donde enmarcar el trabajo realizado en el área del color, extraído mayormente del capítulo 2 *Colour and imaging* de [21] y del capítulo 3 *Photometry and Colorimetry* de [28].

La Colorimetría es la rama de la ciencia del color que estudia las características cuantitativas del color, siendo éste un estímulo visual psicológico tal que:

- Cuando visto por un observador con visión de color normal, bajo idénticas condiciones de observación, dos estímulos con las mismas especificaciones se ven iguales
- Estímulos que se ven igual tienen las mismas especificaciones
- El número de especificaciones son funciones continuas de los parámetros físicos que definen la distribución de potencia espectral radiante del estímulo.

El color no existe por sí mismo. Este no es el resultado de la luz para una determinada longitud de onda, cómo proponía Newton, ni tampoco el resultado de la mente de un observador, según la concepción de Goethe, sino que es un compendio de ambas teorías. El proceso de visión consiste de tres elementos: una fuente de luz que irradia energía en el espectro visible, caracterizada por una Distribución Espectral de Potencia (SPD); un objeto que refleja una parte determinada de la energía que incide sobre él, determinadas por la reflectancia espectral del mismo, y un observador. La interacción entre la SPD de una fuente luminosa y la reflectancia de un objeto se muestra en la figura 5.1

Que el color no exista por si mismo se refiere al hecho que dos mismas SPD que llegan a un observador, no producen el mismo efecto psicofísico en él, y dos percepciones idénticas de un color no tienen necesariamente una misma SPD.

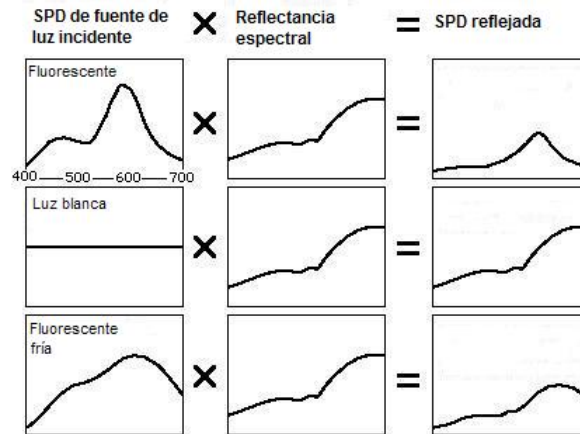


Figura 5.1: Distribución espectral de potencia resultante de un objeto al ser iluminado bajo distintos tipos de fuentes luminosas: distintas fuentes iluminando un mismo objeto dan como resultado distribuciones diferentes, generando estímulos diferentes en un observador

Según la teoría Tri-estímulo, iniciada por Young-Helmholtz (1867) y los experimentos posteriores de Wright (1928) y Guild (1931), es posible y conveniente representar los colores mediante un espacio tridimensional de vectores.

5.2. Metamerismo

Dos espectros distintos, f y g , correspondientes a dos colores distintos, pueden producir la misma respuesta en el ojo y ser percibidos idénticamente. El fenómeno por el cual dos espectros de colores con distribuciones diferentes, y que dan lugar al mismo efecto psicofísico (generan la misma percepción de color) al ser percibidos por un observador, se conoce con el nombre de metamerismo. El metamerismo implica que dos objetos que parecen tener exactamente el mismo color, pueden diferir en sus distribuciones espectrales de potencia. Este fenómeno permite recrear la percepción del color, mediante una distribución espectral diferente a la reflectancia del color que se busca reproducir, permitiendo que no halla que reproducir exactamente el mismo espectro del color buscado, lo cual posibilita la reproducción de los colores.

5.3. Espacio de color RGB

El espacio de color RGB¹ es un espacio aditivo, esto es que los colores del espacio se forman mediante la combinación de los componentes primarios RGB. Esto es, un color Q puede ser representado mediante la siguiente suma:

$$Q = \alpha_Q R + \beta_Q G + \gamma_Q B \quad (5.1)$$

Este modelo de color por sí mismo no define los colores primarios RGB, por lo que el resultado de la mixtura de ellos no es exacta (a menos que los colores primarios sean definidos).

Típicamente, las cámaras digitales color capturan imágenes que dependen del propio dispositivo así como de la iluminación. Esto es, los valores RGB capturados son específicos de la cámara y de las condiciones de iluminación bajo las cuales fueron obtenidas. La dependencia de la iluminación se da debido a que las cámaras describen el color de una imagen solamente cuando son vistas bajo la fuente de luz utilizada durante la captura. Por otro lado, la dependencia del dispositivo significa que dispositivos iguales, bajo las mismas condiciones de iluminación, no obtendrán idénticos valores de los canales RGB al capturar una misma imagen. Además, si se reproduce una misma imagen en monitores diferentes, los colores aparecerán distintos (debido a que las intensidades de emisión de los monitores son distintas). Esto significa que un color con valores RGB, deberá ser reproducido con valores R'G'B' en otro monitor de forma de que el color sea igualmente percibido. El modelo de color RGB es pues un espacio dependiente del dispositivo (*device-dependent*)

5.4. Espacios de colores independientes del dispositivo

La Commission Internationale de l'Eclairage (CIE), propuso un método estándar para la descripción de los colores bajo condiciones de iluminación controladas, basándose en la respuesta del sistema visual humano derivado de experimentos psicofísicos.

Debido a que la apariencia de un color depende fuertemente de la fuente de iluminación, resulta esencial en la especificación de un color la fuente de iluminación involucrada. La CIE definió varias fuentes de iluminación estándar, describiendo la distribución espectral de potencia de las mismas. La "clase D" de iluminante especifica una distribución relativa de energía que se corresponde muy estrechamente con la radiación emitida por lo que se llama un cuerpo negro (black body).

"En un cuerpo negro, cuando se aumenta la temperatura, hay un cambio en la radiación que se emite hacia longitudes de onda más breves. Un iluminante de tipo D

¹RGB: Red, Green and Blue. Rojo, Verde y Azul

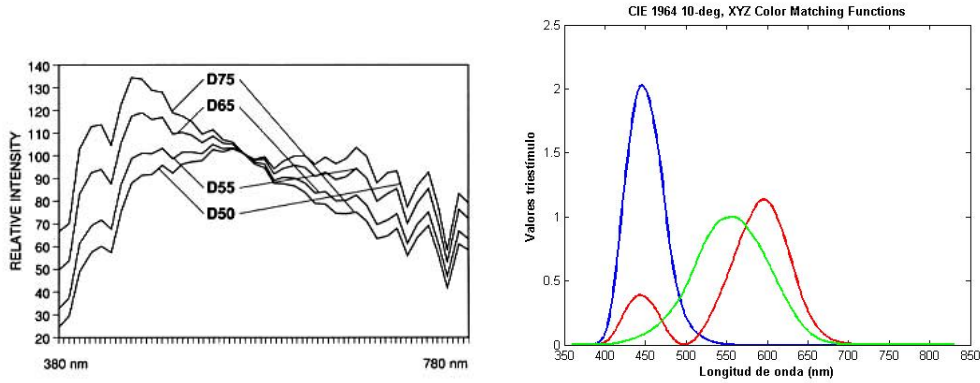


Figura 5.2: Distribución espectral de potencia relativa para algunas fuentes lumínicas definidas por la CIE, y las funciones de correspondencia de color CIE \bar{x} , \bar{y} , \bar{z}

se indica haciendo referencia a la temperatura absoluta, en grados Kelvin, del cuerpo negro al que más se aproxima, en este caso un cuerpo negro a 6.500 grados Kelvin. Por ejemplo, D_{65} representa luz de día estándar, mientras que D_{50} se acerca más a la temperatura de color de fuentes de luz interiores. Los subíndices 50 y 65 refieren a la temperatura de color² correlacionada³, e.g. D_{65} se correlaciona con una temperatura de color de 6500K.”[29]

5.4.1. Espacio de color CIE XYZ

La CIE definió los llamados observadores estándar, que consisten en la definición de tres funciones de emparejamiento (matching) de colores. Basándose en los experimentos de Wright (1928) y Guild (1932), se definieron los colores primarios r, g y b, mediante las curvas de distribución espectral de potencia \bar{r} , \bar{g} , \bar{b} . Dichas funciones fueron manipuladas matemáticamente de forma que las nuevas fuesen positivas en todo el espectro visible, y que una de las funciones resultantes, \bar{y} , coincidiese con la función de luminosidad (la forma en que los humanos perciben el brillo). Se definen pues dos variables de funciones de emparejamiento de color, conocidas como observadores estándares de 2 y 10 grados, siendo éste último el más ampliamente difundida, y es la utilizada en el presente trabajo. Las curvas resultantes para el

²La temperatura de color de una fuente de luz es la temperatura del cuerpo negro a la que su apariencia de color es similar a la de la fuente considerada. Unidad: grado Kelvin (K). Si no se trata de una fuente de luz que emite por termorradiación (lámparas incandescentes), se debe utilizar el concepto de temperatura de color correlacionada

³Temperatura de color correlacionada es aquella temperatura de color que corresponde al punto del lugar geométrico de Planck que está más próximo al que representa la cromaticidad del iluminante considerado.

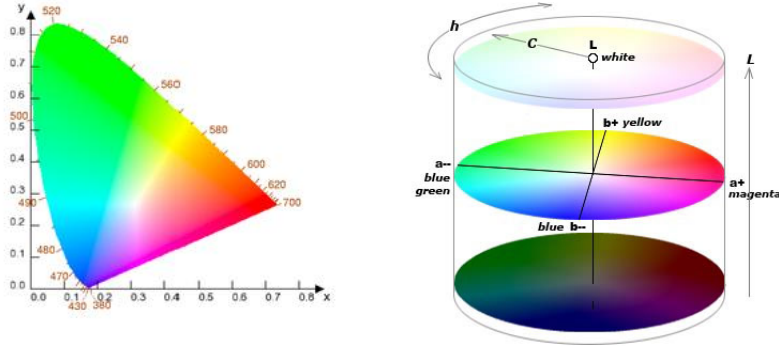


Figura 5.3: Diagrama de cromatismo CIE(x,y) y geometría del espacio CIELab.

observador estándar CIE 10 grados \bar{x} , \bar{y} y \bar{z} se observan en la figura 5.2. Dado un estímulo de color con distribución espectral de potencia radiante $f(\lambda)$, este puede ser representado mediante tres valores XYZ, definidos a partir de las siguientes fórmulas:

$$\begin{aligned}
 X &= \int_{\lambda_{min}}^{\lambda_{max}} f(\lambda) \bar{x} d\lambda \\
 Y &= \int_{\lambda_{min}}^{\lambda_{max}} f(\lambda) \bar{y} d\lambda \\
 Z &= \int_{\lambda_{min}}^{\lambda_{max}} f(\lambda) \bar{z} d\lambda
 \end{aligned} \tag{5.2}$$

en donde $f(\lambda)$ representa la distribución espectral de potencia (SPD) percibida por un observador, la cual es el producto de la energía incidente sobre el objeto, y la reflectancia $\rho(\lambda)$ del objeto.

Para visualizar gráficamente un color, se utiliza el espacio CIE xyY con su respectivo diagrama de cromatismo CIE(x,y), donde $x = \frac{X}{X+Y+Z}$, $y = \frac{Y}{X+Y+Z}$, $Y = Y$, cuya gama de colores, o *gamut*, puede observarse en la figura 5.3.

Sin embargo, la sensibilidad del ojo humano respecto a la luz es no lineal. El espacio de color XYZ definido por la CIE está relacionado linealmente con la distribución espectral de potencia de la luz coloreada.

Más, cuando se cambian los valores XYZ, un observador percibirá la diferencia entre colores luego de cierta cantidad, igual a la JND⁴ de acuerdo con la menor

⁴Just Noticeable Difference, diferencia mínima perceptible entre dos colores con valores de componentes distintos

diferencia perceptible. En el espacio XYZ, dicha cantidad depende de la posición del color en el espacio XYZ. El sistema XYZ y los diagramas de cromaticidad asociados no son perceptualmente uniformes.

Para remediar dichas desventajas, la CIE propuso dos espacios de colores pseudo-uniformes. Un espacio de color se dice uniforme, cuando iguales diferencias perceptibles entre colores son producidas por puntos equiespaciados en todo el espacio. Es decir, la diferencia de color percibida por un observador humano se puede aproximar por la distancia euclídea entre dos puntos en el espacio de color.

5.4.2. Espacio de color CIELab

El espacio de color CIELab se define a través de las cantidades L^* , a^* y b^* , de la forma que sigue:

$$\begin{aligned} L &= 116f\left(\frac{Y}{Y_n}\right) - 16 \\ a^* &= 500 \left[f\left(\frac{X}{X_n}\right) - f\left(\frac{Y}{Y_n}\right) \right] \\ b^* &= 200 \left[f\left(\frac{Y}{Y_n}\right) - f\left(\frac{Z}{Z_n}\right) \right] \end{aligned} \quad (5.3)$$

con

$$f = \begin{cases} \alpha^{1/3} & \text{si } \alpha \geq 0,008856 \\ 7,787\alpha + \frac{16}{116} & \text{si } \alpha < 0,008856 \end{cases}$$

X_n, Y_n y Z_n son los valores triestímulo del blanco de referencia utilizado. Para una fuente D_{50} , dichos valores son:

$$\begin{aligned} Y_n &= \int_{\lambda_{min}}^{\lambda_{max}} l_{D_{50}}(\lambda) \cdot \bar{y} d\lambda = 96,42 \\ X_n &= \int_{\lambda_{min}}^{\lambda_{max}} l_{D_{50}}(\lambda) \cdot \bar{x} d\lambda = 100,00 \\ Z_n &= \int_{\lambda_{min}}^{\lambda_{max}} l_{D_{50}}(\lambda) \cdot \bar{z} d\lambda = 82,49 \end{aligned} \quad (5.4)$$

donde $l_{D_{50}}(\lambda)$ representa la radiación espectral de la fuente iluminante para una longitud de onda λ . La geometría del espacio CIELab puede observarse en la figura 5.3

Al comparar dos colores, especificados como $[L_1^*, a_1^*, b_1^*]$ y $[L_2^*, a_2^*, b_2^*]$, una medida de la diferencia entre los colores definida por la CIE es la que computa la distancia Euclídea entre los colores:

$$\Delta E_{ab}^* = \sqrt{(L_1^* - L_2^*)^2 + (a_1^* - a_2^*)^2 + (b_1^* - b_2^*)^2} \quad (5.5)$$

Una regla práctica para interpretación de ΔE_{ab}^* cuando dos colores son comparados se presenta en el cuadro 5.1

ΔE_{ab}^*	Efecto
< 3	Difícilmente perceptible
$3 < 6$	Perceptible, pero aceptable
> 6	inaceptable

Cuadro 5.1: Regla para la interpretación práctica de ΔE_{ab}^* , para medir la diferencia entre dos colores puestos lado a lado

Cabe destacar, sin embargo, que al ser CIELab un espacio de color no completamente uniforme (pseudo-uniforme), la diferencia entre colores ΔE_{ab}^* no es perfecta. En 1994 la CIE definió otra forma de medir la diferencia entre colores, el modelo de diferencia CIE94, ΔE_{94}^* . Se definieron condiciones de referencia bajo las cuales la nueva métrica se espera funcione bien. Esto es: los especímenes son homogéneos en color, la diferencia de color ΔE_{ab}^* es menor que 5; son iluminados con 100 lux, bajo un fondo gris uniforme, con componente $L^* = 50$; la fuente iluminante es del tipo D_{65} ; entre otros. La métrica ΔE_{94}^* se define como

$$\Delta E_{94}^* = \sqrt{\left(\frac{\Delta L^*}{k_L S_L}\right)^2 + \left(\frac{\Delta C^*}{k_C S_C}\right)^2 + \left(\frac{\Delta H^*}{k_H S_H}\right)^2} \quad (5.6)$$

donde ΔL^* , ΔC^* y ΔH^* son las diferencias de luminosidad, croma y tonalidad. Los funciones S_L , S_C y S_H varían según la croma del espécimen de referencia, de la siguiente forma:

$$S_L = 1, \quad S_C = 1 + 0,045C^*, \quad S_H = 1 + 0,012C^*$$

Las variables k_L , k_C y k_H son llamados factores paramétricos, y son introducidos para poder corregir posibles desviaciones de las condiciones experimentales. Bajo las condiciones descritas anteriormente, todas las constantes tienen valor 1.

5.5. Caracterización de dispositivos

La imagen capturada por una cámara depende de tres factores: el contenido físico de la imagen, la iluminación, y las características de la cámara. Esto significa que, manteniendo los dos primeros parámetros constantes, dos cámaras distintas obtendrán imágenes distintas (valores de los canales RGB distintos). Por tanto, para lograr una reproducción consistente y de alta calidad de los colores es necesario un entendimiento de las características de los dispositivos involucrados, de como estos reproducen los

colores. Caracterizar un dispositivo es hallar la relación entre las coordenadas de color del dispositivo, generalmente RGB o CMYK, y un espacio de color independiente del dispositivo, como ser el CIE XYZ.

Siguiendo el enfoque dado en [24], existen tres grandes métodos para lograr la caracterización de una cámara [24]: modelos físicos de dispositivo, tablas de correspondencia (look-up tables) y métodos numéricos.

Cabe mencionar que una caracterización efectiva es practicable solamente si el dispositivo en cuestión no realiza balance de punto blanco automático (automatic white-point balancing). Este es el proceso mediante el cual el parche (patch) más brillante de cada imagen capturada es denotado como blanco (normalmente R=G=B=255 para cámaras de 8 bits), y los valores RGB de cada píxel son transformados. Automatic white-point balancing es útil cuando el propósito es obtener una imagen placentera sin preocuparse por la iluminación, pero para propósitos de colorimetría dicha acción debe ser evitada. En el presente trabajo se realiza un balance de blancos fijo, lo que lleva a que en la práctica se halle un mapeo entre la salida de la cámara (valores RGB de la cámara transformados según el balance de blanco) y el espacio de color CIELab bajo un iluminante D_{65} .

5.5.1. Métodos numéricos de Caracterización de dispositivos

La siguiente descripción se basa en el trabajo realizado por Cheung y Westland [25]. Este método de caracterización consiste de dos etapas: la primera realiza una linealización de los valores; la segunda transforma los valores linealizados (típicamente RGB) en los valores triestímulo CIE XYZ.

A pesar de que la respuesta física de los sensores CCD es lineal en relación a la intensidad de luz incidente sobre ellos, la salida de los distintos canales RGB de una cámara son generalmente no lineales en relación con los valores triestímulo XYZ de la escena en cuestión. La respuesta cruda (raw) de los canales es seguida de procesamiento en la electrónica de las cámaras, lo cual ocasiona que la intensidad de la luz percibida por la cámara y la respectiva salida RGB sea generalmente no lineal. Es, por tanto, común considerar correcciones de la no linealidad como la primera etapa del procedimiento en la caracterización de una cámara. Una posible relación es la siguiente:

$$C_c = (R'_c)^p \tag{5.7}$$

donde R'_c es la respuesta cruda del canal c , p es un exponente para el canal, y C_c es la respuesta de la cámara transformada para el canal en cuestión, de forma que sea lineal con la entrada. Para determinar el exponente p , generalmente se utilizan “parches” grises (muestras de valores de grises), los cuales presentan una reflectancia plana a lo largo del espectro visible. La respuesta a la salida de la cámara es determinada bajo una fuente de luz determinada. Los valores XYZ de la escala de grises pueden



Figura 5.4: Ejemplo de tablas de colores disponibles comercialmente de Gretag Macbeth ColorChecker. Las mismas brindan información sobre los valores XYZ de las distintas muestras de colores bajo diversas fuentes de iluminación estándares.

ser computados si se conocen los valores de la reflectancia espectral de los mismos y la distribución espectral de potencia de la fuente de luz incidente o ser medidos directamente utilizando un colorímetro. Otra posible forma es utilizando los valores XYZ o de otro espacio que proporcionan algunas tablas de colores comerciales como ser Gretag Macbeth ColorChecker Chart, las cuales brindan los datos bajo ciertas fuentes de iluminación determinadas, e.g. D_{65} . Para utilizar los datos brindados por ellas, es pues necesario reproducir la fuente de luz debida. La linealización se lleva a cabo buscando un valor de p tal que exista una relación lineal entre C_c y el valor CIE Y del conjunto de muestras de grises.

La segunda etapa en el proceso de caracterización de una cámara requiere encontrar un mapeo entre los valores RGB linealizados de la cámara y los valores triestímulo CIE XYZ. Un ejemplo de mapeo puede observarse a continuación:

$$\begin{aligned} X &= a_{11}R + a_{12}G + a_{13}B \\ Y &= a_{21}R + a_{22}G + a_{23}B \\ Z &= a_{31}R + a_{32}G + a_{33}B \end{aligned} \quad (5.8)$$

donde los coeficientes $a_{11} - a_{33}$ deben ser determinados.

Dadas n muestras con sus respectivos valores de respuesta del dispositivo (e.g. RGB) y sus correspondientes valores triestímulo, en términos de álgebra matricial es necesario encontrar la matriz A , tal que:

$$T = CA \quad (5.9)$$

donde T representa una matriz $n \times 3$ con los valores triestímulo XYZ, C es la matriz $n \times 3$ que contiene los valores de salida de la cámara y A es la matriz que contiene los coeficientes $a_{11} - a_{33}$. La matriz A puede hallarse directamente:

$$A = C^{-1}T \quad (5.10)$$

A	Matriz
3 x 3	$[R\ G\ B]$
3 x 4	$[R\ G\ B\ 1]$
3 x 5	$[R\ G\ B\ RGB\ 1]$
3 x 7	$[R\ G\ B\ RG\ RB\ GB\ 1]$
3 x 8	$[R\ G\ B\ RG\ RB\ GB\ RGB\ 1]$
3 x 10	$[R\ G\ B\ RG\ RB\ GB\ R^2\ G^2\ B^2\ 1]$
3 x 11	$[R\ G\ B\ RG\ RB\ GB\ R^2\ G^2\ B^2\ RGB\ 1]$
3 x 14	$[R\ G\ B\ RG\ RB\ GB\ R^2\ G^2\ B^2\ RGB\ R^3\ G^3\ B^3\ 1]$
3 x 16	$[R\ G\ B\ RG\ RB\ GB\ R^2\ G^2\ B^2\ RGB\ R^2G\ G^2B\ B^2R\ R^3\ G^3\ B^3]$
3 x 17	$[R\ G\ B\ RG\ RB\ GB\ R^2\ G^2\ B^2\ RGB\ R^2G\ G^2B\ B^2R\ R^3\ G^3\ B^3\ 1]$
3 x 19	$[R\ G\ B\ RG\ RB\ GB\ R^2\ G^2\ B^2\ RGB\ R^2G\ G^2B\ B^2R\ R^2B\ G^2R\ B^2G\ R^3\ G^3\ B^3]$
3 x 20	$[R\ G\ B\ RG\ RB\ GB\ R^2\ G^2\ B^2\ RGB\ R^2G\ G^2B\ B^2R\ R^2B\ G^2R\ B^2G\ R^3\ G^3\ B^3\ 1]$

Cuadro 5.2: Distintos polinomios utilizables para realizar la caracterización numérica

donde C^{-1} denota la matriz inversa de C . Si C es una matriz cuadrada, entonces existe una única solución a la ecuación 5.10. Para sistemas sobredeterminados, una solución de la antedicha ecuación puede ser obtenido computado la matriz pseudo-inversa de C , denotada como C^+

$$A = C^+T \quad (5.11)$$

En la caracterización numérica de la cámara, el sistema es por lo general sobredeterminado (es necesario utilizar una muestra “grande” de colores para realizar un mapeo efectivo). La transformación lineal mostrada en la ec. 5.9 es una del conjunto de transformaciones polinómicas. Transformaciones de orden superior (no lineales) pueden ser utilizadas, como ser

$$\begin{aligned} X &= a_{11}R + a_{12}G + a_{13}B + a_{14}R^2 + a_{15}G^2 + a_{15}B^2 \\ Y &= a_{21}R + a_{22}G + a_{23}B + a_{24}R^2 + a_{25}G^2 + a_{25}B^2 \\ Z &= a_{31}R + a_{32}G + a_{33}B + a_{34}R^2 + a_{35}G^2 + a_{35}B^2 \end{aligned} \quad (5.12)$$

que puede representarse matricialmente de la forma $T = DA$, donde D es una matriz $n \times 6$ y cada columna contiene 6 términos $[R\ G\ B\ R^2\ G^2\ B^2]$. La forma de resolver esta ecuación es la misma que la usada para la ec. 5.9

Otros posibles polinomios a usar en la caracterización se pueden ver en el cuadro 5.2. El polinomio óptimo a utilizar debe ser determinado experimentalmente, caracterizando y evaluando para diferentes polinomios.

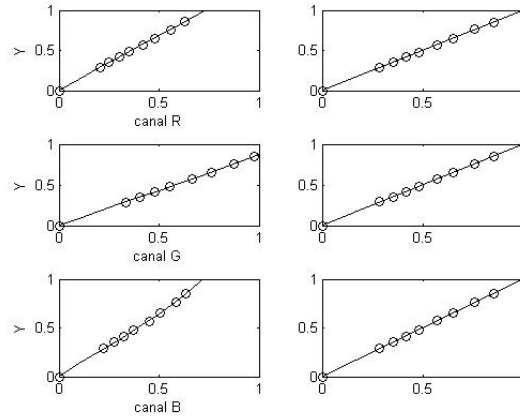


Figura 5.5: La columna de la izquierda muestra el gráfico de los canales de salida de una cámara y el ajuste polinomial de los mismos. La columna de la derecha se observa la respuesta de la cámara transformada y, para comparación, la respuesta lineal ideal

5.5.2. Linealización

La primera etapa en la caracterización de una cámara es la linealización de los valores RGB medidos. Para realizarla, se toman imágenes de muestras de distintos valores de grises y se halla la reflectancia media Y de cada muestra. Se supone un valor de $Y = 0$ para valores de $R=G=B=0$, lo cual implica que la cámara da una respuesta de 0 para una señal de entrada nula. La linealidad de la respuesta frente a los distintos estímulos de grises dependerá de la cámara a utilizar. La linealización se lleva a cabo utilizando polinomios de orden bajo, el cual se utiliza luego para transformar la respuesta cruda de la cámara. De forma de ilustrar lo dicho anteriormente, se utilizaron datos fictos, tomados de [26].

Al realizar la linealización, los datos de los tres canales son balanceados para las muestras neutrales (grises), lo cual se puede observar en la figura 5.5. Para el ejemplo, los valores del parche de color más brillante son [160, 249, 162]. Ello no es lo ideal, pues dicho parche no se observará neutro al ser desplegado en un monitor. Al realizar la transformación de los valores RGB de la cámara, son, sin embargo, [217, 218, 217]. Este proceso se conoce como balance de canal⁵, donde los desbalances resultan de la combinación de las propiedades de la cámara y la distribución espectral de la luz bajo las cuales las imágenes son tomadas.

En el presente trabajo, se modela como lineal el sensor para la primera calibración,

⁵El concepto de balance de canal no debe ser confundido con el de balance por punto blanco, en donde el punto más brillante capturado en una imagen es transformado a [255,255,255]

debido a que se disponía de pocas muestras de colores, y los grises disponibles presentaban una reflectancia alta en la parte del espectro correspondiente al rojo. El haber modelado el sensor como lineal es fuente de error en las medidas.

5.6. Caracterización de la cámara

La cámara fotográfica utilizada en el presente trabajo, no brinda la posibilidad de no realizar balance de punto blanco. Por tanto, se deja fijo el sistema de corrección de color de la cámara,⁶ el cual utiliza un blanco prefijado transformando los valores de los pixeles siempre de la misma forma. Por tanto, se realiza una calibración del color del conjunto formado por la cámara y el software específico de la misma que realiza el balance de punto blanco fijado, encontrándose un mapeo entre el espacio RGB de salida de la cámara y el espacio CIELab bajo iluminante D_{65} . De cambiar la cámara, el software utilizado o el balance de blancos, se deberá realizar una nueva calibración.

En la práctica, los cálculos realizados fueron los siguientes:

$$\begin{aligned} X &= \frac{1}{N} \int_{\lambda} \bar{x}(\lambda) S(\lambda) I(\lambda) d\lambda \\ Y &= \frac{1}{N} \int_{\lambda} \bar{y}(\lambda) S(\lambda) I(\lambda) d\lambda \\ Z &= \frac{1}{N} \int_{\lambda} \bar{z}(\lambda) S(\lambda) I(\lambda) d\lambda \\ N &= \int_{\lambda} \bar{y}(\lambda) I(\lambda) d(\lambda) \end{aligned} \quad (5.13)$$

La potencia espectral recibida por un observador, tal como lo indica la eq. 5.2, se compone de la reflectancia espectral de la muestra de color, $S(\lambda)$, multiplicada por la distribución espectral de potencia de la fuente iluminante $I(\lambda)$, lo cual representa la situación mostrada en la fig. 5.1

Los distintos valores de reflectancia de la tabla de color, $S_i(\lambda)$, fueron medidos utilizando un espectrofotómetro⁷. Las medidas espectrales pueden realizarse incluyendo o excluyendo la componente especular de la reflectancia espectral. La luz que se refleja en un ángulo opuesto al de la luz incidente se llama reflectancia especular, mientras que el resto recibe el nombre de reflectancia difusa, como puede observarse gráficamente en la figura 5.6. Para medir el color de un espécimen del mismo modo que se ve, la reflectancia especular deberá excluirse y sólo deberá medirse la reflectancia

⁶Se utiliza el balance de punto blanco para iluminación con lámparas de tungsteno (blanco prefijado según iluminado bajo una lámpara de tungsteno)

⁷Los valores de reflectancia fueron en parte gentileza de Alessandro Rizzi, y en parte medidos gracias a la colaboración de INCA

difusa. El color de un objeto puede parecer diferente debido a las diferencias en el nivel de la reflectancia especular.[30]

La caracterización se realizó utilizando el iluminante estándar D_{65} , el cual representa un promedio de la luz de día en el espectro visible.

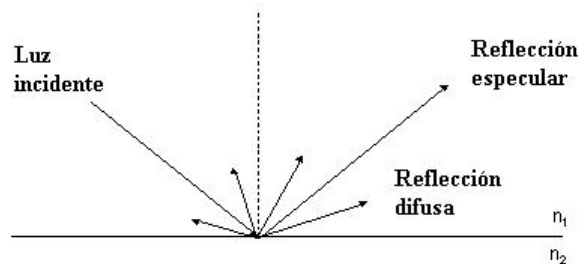


Figura 5.6: Componentes especular y difusa de la reflectancia total de un objeto

Con los valores de reflectancia y la distribución espectral de la fuente luminosa, se realizaron los cálculos de los valores X , Y y Z , según la ecuación 5.13. Por otra parte se capturó una fotografía de la tabla de colores bajo la luz utilizada para iluminar el Ojo de Bife de modo de obtener los valores R , G y B percibidos por la cámara, la cual se configuró utilizando balance de punto blanco fijo.

La caracterización de la cámara, o mejor dicho el mapeo entre la salida de la misma (conjunto Hardware-Software debido a la realización de balance de punto blanco), se llevó a cabo utilizando Métodos Numéricos. Según un estudio publicado en [25], a medida que la complejidad del modelo y el número de muestras aumenta, el error cometido en la estimación disminuye, alcanzando un mínimo y luego manteniéndose a medida que satura el número de muestras de color agregadas. En el enfoque polinómico, es necesario buscar el polinomio que mejor realiza la caracterización, lo cual puede ser alcanzado mediante rigurosa experimentación.

A continuación se describen los resultado obtenidos de dos experimentos distintos, los cuales incluyeron distinta cantidad de muestras. Durante los mismos, la posición de la cámara así como la iluminación permanecieron fijos.

5.6.1. Resultados de la primera calibración

En una primera calibración se utilizaron 19 muestras de colores, cuya reflectancia espectral puede observarse en la figura 5.7. Se estudió realizar la caracterización de

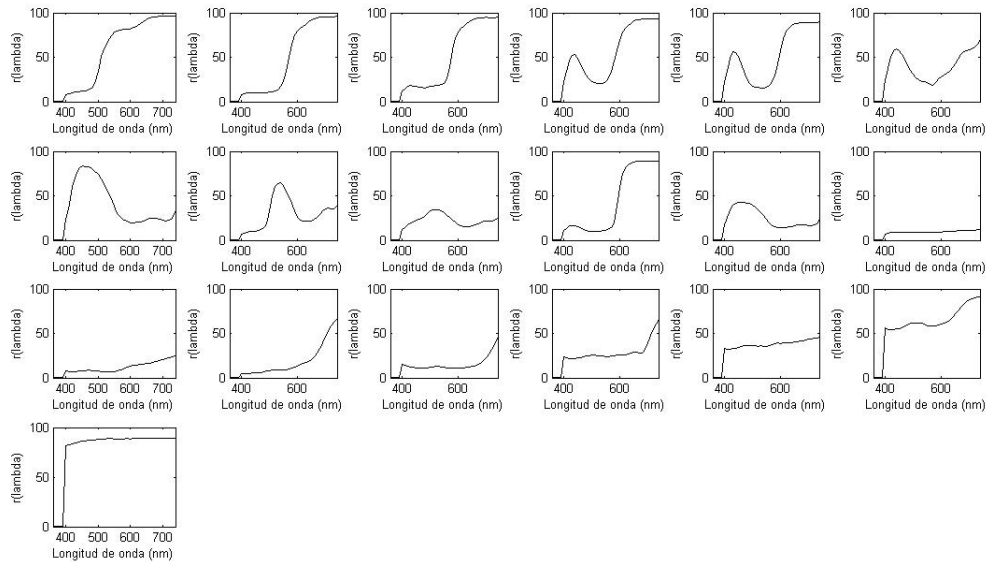


Figura 5.7: Reflectancias espectrales para los colores utilizados en la primera calibración

la cámara empleando polinomios de 3, 4, 10 y 20 términos⁸, cuya estructura puede observarse en el cuadro 5.2. Utilizando las 19 muestras de color disponibles para la calibración y comparando contra los valores CIELab de los mismos, el gráfico 5.8 muestra que los polinomios de 10 y 20 términos son los que realizan la mejor estimación de valores utilizados en la calibración.

Sin embargo, como puede observarse en el gráfico 5.9, si se utilizan 18 muestras para la calibración y se compara contra la muestra que se dejó fuera, y se promedian los datos, se puede ver que el polinomio que menor error arroja es el de 10 términos. (Para el polinomio de 20 términos el error resultó ser muy alto, por lo que no se muestra en la gráfica). Para utilizar el polinomio de 10 términos propuesto, es necesario estimar 30 coeficientes utilizando tan solo 19 muestras de colores, mientras que para el polinomio de 20 términos el número de coeficientes a estimar trepa hasta 60. El error cometido es excesivamente alto, debido principalmente a que 19 muestras de color no resultan suficientes para realizar un mapeo efectivo entre la salida de la cámara y el espacio de color CIELab bajo iluminante D_{65} .

⁸Si bien resultan el número de muestras disponible resulta ser bajo para la cantidad de coeficientes que se pretende hallar, la calibración fué llevada acabo de forma de obtener resultados parciales

5.6.2. Fuentes de error

Entre las distintas fuentes que introducen error en las medidas colorimétricas encontramos:

Número de muestras El número de muestras utilizadas para calibrar el polinomio resultó ser excesivamente bajo para realizar un mapeo efectivo entre la entrada a la cámara y el espacio CIELab de los colores vistos bajo luz D_{65} .

Linealización Se utilizó un modelo lineal de los canales de salida R, G y B de la cámara con respecto a la intensidad de luz incidente sobre el sensor CCD. Como se mencionó anteriormente, dicha relación es por lo general no lineal.

1 sólo sensor CCD La cámara de fotos utiliza un solo sensor CCD en lugar de 3, lo cual obliga a realizar una estimación de los valores R, G o B de los píxeles.

Formato JPG JPEG es un algoritmo de compresión con pérdidas, lo cual ocasiona que al hallar los valores RGB iniciales antes de realizada la compresión, dichos valores no son los mismos. Se recomienda trabajar con los valores crudos (RAW) de la cámara en la medida de lo posible.⁹

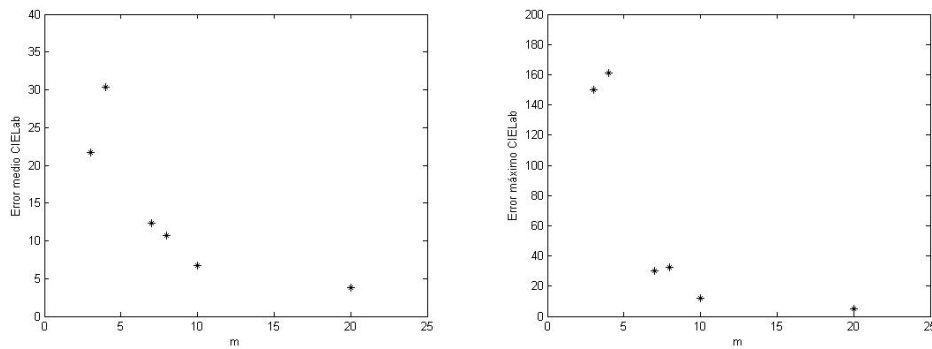


Figura 5.8: Distancia entre los valores de la tabla de colores calculados utilizando los distintos polinomios y los valores Lab de los mismos: error CIE Lab medio (izquierda) y error CIE Lab máximo (derecha).

⁹Se optó por trabajar con formato JPG debido a que el tiempo de transferencia entre la cámara y el ordenador eran bajos en comparación a utilizar un formato RAW

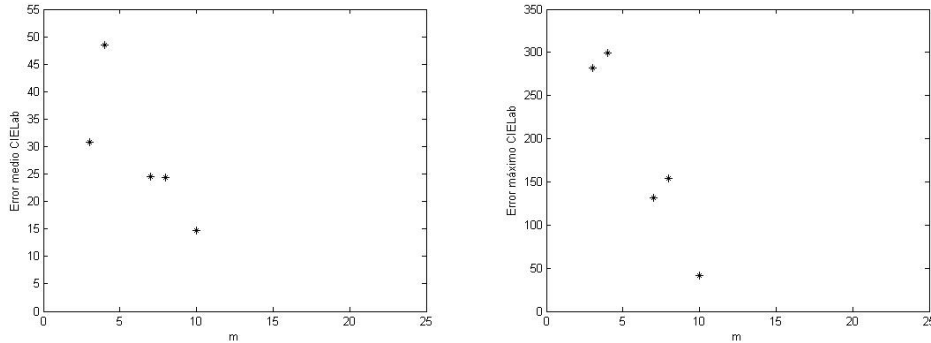


Figura 5.9: Error CIELab medio (izquierda) y error CIELab máximo (derecha) al realizar la calibración con 18 muestras y comparar contra la muestra no utilizada. Los errores cometidos para el polinomio de 20 términos son muy altos por lo que no se muestran en el gráfico

5.6.3. Resultados de la segunda calibración

En un segundo proceso de calibración, se obtuvieron 150 muestras de color, cortesía de INCA S.A. De las mismas, se utilizaron hasta 110 muestras para calibrar el dispositivo (esto es, hallar los coeficientes del polinomio de calibración). En esta etapa se probaron polinomios de 3, 4, 7, 8, 10, 11, 14, 16, 17, 19 y 20 términos, los cuales se muestran en la tabla 5.2. En la tabla 5.3 se muestran los valores CIELab calculados teóricamente y los estimados utilizando el polinomio de 17 términos, así como las distancias y el código de color¹⁰ al cual corresponde, para los colores que arrojan las mayores y las menores distancias. Dado que se disponía de muestras de colores “grises”, se realizó el paso de linealización del sensor, el cual no se efectuó en la primera calibración. Los resultados de la ejecución de dicho proceso se observan en la figura 5.10.

Como se muestra en la misma, existe una fuerte dispersión de los valores utilizados para la linealización, en donde no siempre corresponde un aumento en los valores canales R G y B al aumentar la luminancia Y. Ello se puede deber a que la reflectancia espectral de las muestras utilizadas para la linealización presentan variaciones respecto a las curvas espectrales típicas de grises “puros” (figura 5.11). A modo de ejemplo, el “gris” con código 9154, quien ocupa la posición (2, 3) del gráfico, es quien presenta las mayores variaciones en el espectro. Sin embargo, los resultados obtenidos excluyendo dicha muestra empeoran, por lo cual aún siendo grises con componentes contaminantes, los resultados mejoran. Por otra parte, si se comparan los resultados obtenidos realizando o no la linealización, las diferencias son notorias, indicando que

¹⁰El código de color corresponde a la nomenclatura de INCA

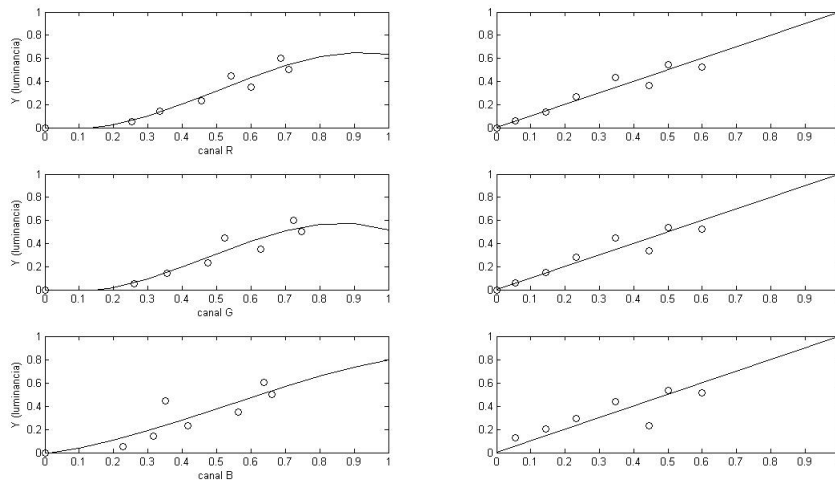


Figura 5.10: En la columna de la izquierda puede observarse la respuesta de los distintos canales R, G y B para los distintos parches de grises utilizados, y la aproximación polinómica de los mismos. La columna de la izquierda muestra la respuesta de la cámara transformada en comparación con una respuesta lineal óptima

el proceso de linealización del sensor, especialmente para cámaras de baja calidad, resulta un paso importante en la caracterización colorimétrica de una cámara. En la figura 5.12 se muestran los resultados de la calibración al efectuar o no el paso de linealización del sensor.

De forma de observar la influencia del número de muestras utilizadas para la calibración, se hallaron los errores medios y máximos para el polinomio de 19 términos, aumentando paulatinamente el número de muestras utilizadas para la calibración. Los resultados se pueden observar en la figura 5.13. Se observa una tendencia a la baja en el error medio CIELab a medida que el número de muestras de color utilizadas para la calibración aumenta.

5.6.4. Conclusiones

Los resultados obtenidos para ambos procesos arrojaron valores altos tanto en los errores medios como máximos para las distancias en el espacio CIELab.

Los resultados expuestos en la primera calibración resultaron ser alentadores, dando cuenta que si bien la cantidad de muestras utilizadas para la calibración era baja, sería posible obtener un error menor en las medidas colorimétricas si se aumentaba el número de muestras para entrenar el polinomio de mapeo. Sin embargo, tal como

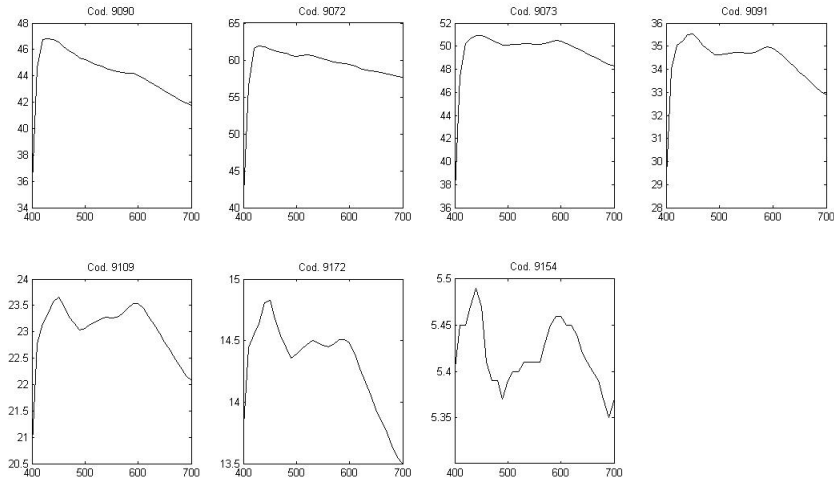


Figura 5.11: Reflectancias espectrales de las muestras de grises utilizadas para la linealización. Se observa como dichos parches de colores presentan contaminaciones de otros colores, no siendo grises puros

lo indica el segundo proceso de calibración, el simple incremento en el número de muestras utilizadas para calibrar el mapeo entre los espacios de colores no resultó ser determinante. En cuanto al proceso de linealización refiere, si bien las muestras de grises utilizadas son aproximaciones de dichos colores, se observa igual un comportamiento no lineal en el sensor de la cámara, como fué mostrado. Por otro lado, existen muestras para las cuales el error entre el valor estimado y el valor colorimétrico calculado teóricamente es alto, por lo que cabría estudiar criterios para la elección de las muestras a utilizar para la calibración¹¹. No se estudió la influencia de la compresión JPEG en los valores RGB de la cámara, para lo cual es necesario obtener los valores RAW de los parches de colores. De modo de realizar correcciones espaciales para minimizar el efecto de falta de uniformidad en la intensidad de la iluminación o de la sensibilidad del sensor CCD de la cámara, *Cheung y Westland*[25] proponen realizar una corrección para cada canal, lo cual podría implementarse en un posterior experimento.

¹¹Las muestras utilizadas para la calibración así como las utilizadas para la validación fueron elegidas al azar

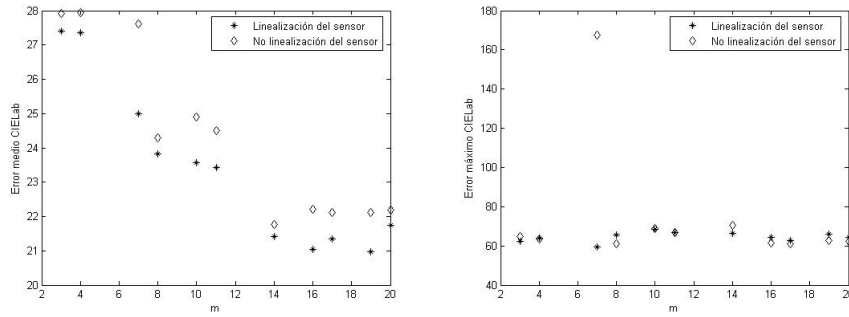


Figura 5.12: Errores medio y máximo utilizando 110 muestras de colores para la calibración y 40 para la validación, en donde se incluye la comparación de efectuar o no el paso de linealización

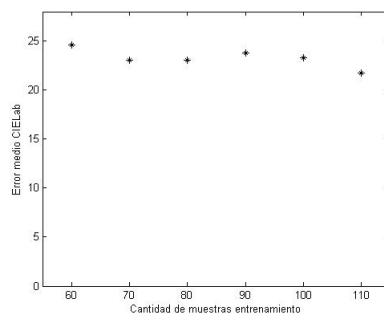


Figura 5.13: Efecto del número de términos utilizados para hallar los coeficientes de calibración para el polinomio de 17 términos

L	a	b	L_{est}	a_{est}	b_{est}	Error	Cod. color
48.3407	45.9147	-85.0168	60.8520	47.4766	-54.1711	56.4742	9309
96.4988	-18.3134	-160.2667	96.6095	-11.3993	-135.0103	28.5253	9559
93.4690	-18.3378	-142.2540	89.2202	-14.6858	-118.2433	26.5478	7184
82.2871	-44.1230	-120.0655	79.7077	-34.6565	-97.0728	21.8498	6990
58.5467	-6.4183	-159.3521	58.0888	-13.4860	-163.0525	6.5707	9528
89.8535	2.0270	-140.9164	92.7329	0.3023	-137.7866	4.8125	8610
93.5152	3.0867	-24.4474	93.3996	7.1424	-23.4885	3.8021	5473

Cuadro 5.3: Algunos resultados del proceso de calibración 2. En el mismo se observan los valores teóricos CIELab calculados utilizando las curvas de reflectancia espectral, los valores CIELab estimados utilizando el polinomio de 17 términos, el error cometido y el código de color al cual corresponde. Se muestran aquellas entradas que ocasionaron los mayores y los menores errores

Capítulo 6

Hardware

6.1. Elección del equipo de captura

6.1.1. Cámara de video: Análogo o Digital

Entre los posibles dispositivos de captura de imágenes encontramos las cámaras de video y las cámaras fotográficas. Entre las cámaras de video se encuentran las analógicas y las digitales, siendo estas últimas preferidas, dado que no es necesario la conversión a formato digital, lo cual da lugar a imágenes con mayor ruido, se obtienen imágenes con resoluciones más altas, mayor velocidad, rangos dinámicos mayores, lo cual permite discernir cambios de brillo más sutiles. Ruido proveniente de equipos electrónicos es introducido en las cámaras analógicas antes de realizar la conversión digital en el frame grabber y es dificultoso de remover. En cambio, el ruido eléctrico que se introduce en los cables cuando se transmite la imagen al ordenador es fácilmente quitado. Para las cámaras digitales, éstas deberán de ser del tipo matricial (area-scan).

En el caso de los cuadros digitales, los requerimientos del sistema consisten en poder visualizar un video en vivo en pantalla, para lo cual se requeriría una velocidad de al menos 24 fps, de entre los que se captura uno para su posterior procesamiento. La consistencia de los diferentes pixeles que componen la imagen bajo idénticas condiciones de iluminación deberá ser lo más uniforme posible de forma de optimizar la segmentación y la detección del color. La opción de balance de punto blanco automático (white-point balancing), el cual realiza una transformación de los valores RGB de cada píxel, es preferible pueda ser deshabilitada, de forma de poder caracterizar efectivamente el dispositivo. La resolución mínima deberá ser de 640 x 480 pixeles, de forma de trabajar con tamaños considerables. Esto es, si se considera que las dimensiones del cuadro a observar son de aproximadamente 20cm x 15cm, una resolución de 640 x 480 arroja una media de 32 pixeles/cm en ambas direcciones.

6.2. Tecnología de sensado de imagen

Comparando las tecnologías CCD y CMOS, los puntos relevantes son la uniformidad, el rango dinámico y la velocidad.

Performance	CCD	CMOS
Rango dinámico	Alto	Moderado
Uniformidad	Alta	Baja a Moderada
Velocidad	Moderada a Alta	Alta

Cuadro 6.1: Comparación de la performance de algunas características de las tecnologías CCD y CMOS[31]

Dado los requerimientos de calidad de imagen ambos sensores son de utilidad, mas es preferible una cámara con sensor CCD debido a sus mejores prestaciones en cuanto a calidad.

6.2.1. Principios de funcionamiento de las cámaras CCD

La información presente en esta sección, en donde se explican los principios de funcionamiento de las cámaras con sensores CCD, así como las imágenes mostradas, fue extraída en su mayor parte de [34].

La idea de funcionamiento de los chips CCD se basa en que los fotones de la luz inciden en las celdas de memoria, fotodiodos sensibles a la luz, liberando, por causa del efecto fotoeléctrico, electrones en dichas celdas, siendo estos proporcionales a la intensidad de los fotones incidentes. La longitud de onda de los fotones no es “transmitida” a los electrones, debido a que los fotodiodos son dispositivos monocromáticos: no distinguen entre distintas longitudes de onda de la luz, son “ciegos al color”.

Para una cámara monocromática, debido al efecto fotoeléctrico, los rayos incidentes en el chip crean cargas negativas en los mismos. Luego de transcurrido el tiempo de exposición, las cargas son leídas para ser pre-procesadas por la electrónica de la cámara. La imagen digital resultante es obtenida a la salida del procesamiento electrónico que se realiza. Un esquema de funcionamiento se muestra en la figura 6.1

6.2.2. Cámara color con 3 CCD

Los fotones tienen, además de una cierta energía, una longitud de onda. Ésta información, como se mencionó anteriormente, no es captada por los sensores. La mayor parte de las cámaras trabaja con tres canales de color, R , G y B . La cámara debe devolver un valor para el canal rojo, un valor para el canal verde y otro para el azul. Por tanto, es necesario tener un chip CCD para cada uno de los colores primarios,

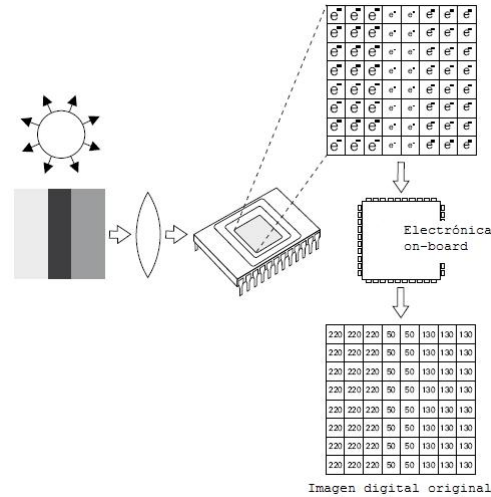


Figura 6.1: Principio de funcionamiento de una cámara monocromática

en donde cada uno de los chips obtiene “fotones filtrados”: un CCD sensorá aquellos fotones que fueron filtrados utilizando el filtro rojo, otro los filtrados mediante el verde, y el restante los fotones resultantes de atravesar el filtro azul. Los filtros son creados utilizando un prisma, como puede observarse en la figura 6.4. Con este tipo de cámaras se consigue los mejores resultados en cuanto a calidad, debido a que se obtiene información de los tres canales directamente de la fuente de luz incidente, en contraposición con las cámaras que trabajan con un solo CCD.

6.2.3. Cámara color con 1 CCD

Si la cámara utiliza solamente un chip CCD, los filtros de color deben ser, pues, esparcidos a lo largo de los píxeles del chip. Este concepto se conoce con el nombre de *filtro Mosaico* o *filtro Bayer*, patentado por el Dr. Bryce Bayer en 1976. Otro posible arreglo (array) de filtro de color, o CFA¹ por sus siglas en inglés, es el CMGY. La disposición de los mencionados filtros se observa en la figura 6.3

Por lo tanto, un píxel contendrá información del canal rojo, o del canal verde, o del canal azul. Sin embargo, es necesario que la imagen de salida proporcione valores para los tres primarios para cada píxel.

Para remediar la falta de información de los píxeles, se utiliza una solución llamada *Interpolación de Color Espacial*, la cual es llevada a cabo por la electrónica de la cámara. El algoritmo estima los valores faltantes analizando los valores de los píxeles

¹Color Filter Array

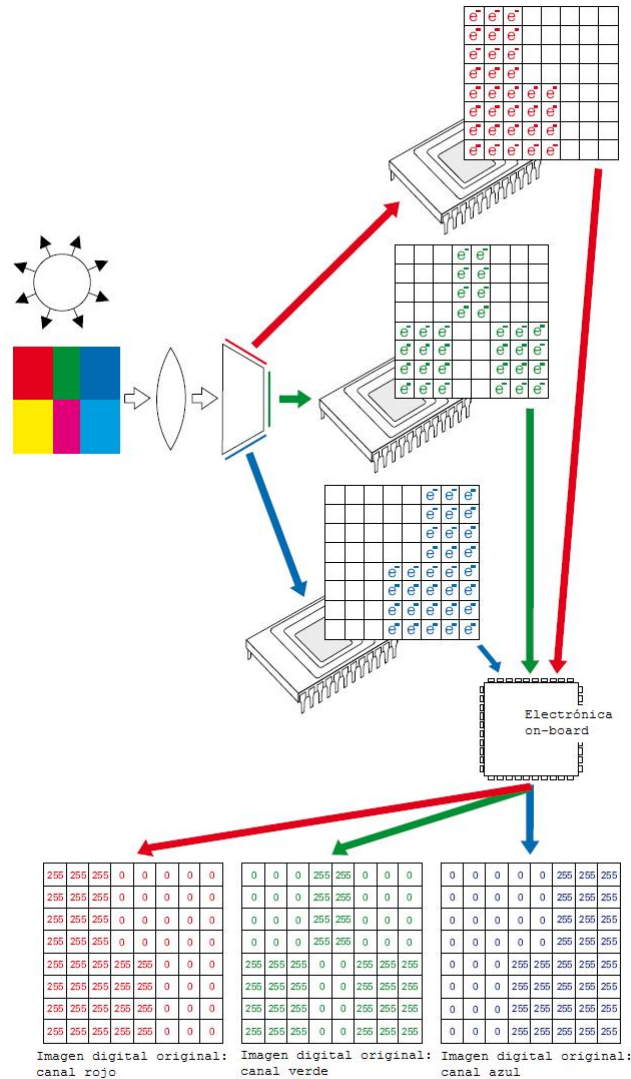


Figura 6.2: Principio de funcionamiento de una cámara color utilizando 3 chips CCD

vecinos. Entre los varios tipos de algoritmos de interpolación, se mencionan dos formas básicas de realizar la interpolación:

Replicación del vecino más cercano Es la forma más simple de rellenar la información faltante. Simplemente se copian los valores de los canales faltantes de los vecinos más próximos. Este método, si bien es de rápida ejecución y puede

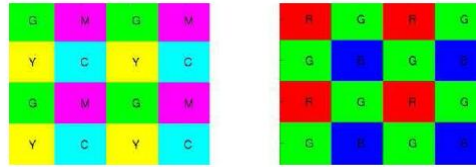


Figura 6.3: Patrones de arrays de filtros de color: CMGY y el Bayer RGB

ser útil para video streaming, conlleva las más bajas prestaciones en cuanto a calidad de imagen

Interpolación bilineal Realiza un promedio de los valores faltantes de los píxeles cercanos. Trae asociado consigo comportamiento de pasa-bajos, y por consiguiente un suavizamiento en la forma de los bordes de la imagen.

Al comparar visualmente dos imágenes distintas obtenidas con cámara que utilizan un solo CCD o 3 CCD, pueden parecer muy similares. Sin embargo, las cámaras que utilizan un solo CCD traen aparejados inconvenientes, como ser efectos pasa-bajos no deseados, perdiéndose información de las altas frecuencias, como ser cambios abruptos en los colores. Por consiguiente, las imágenes resultantes presentan un aspecto más borroso, con menor definición que las imágenes obtenidas con cámaras con 3 CCD. La interpolación espacial, además, sobrecarga al procesador de la cámara.

6.3. Selección del puerto de entrada

La cantidad de datos mínima a ingresar al ordenador, cuando se muestra el video en pantalla, son imágenes de 640x480, de 24 bits, a una tasa de 24 cuadros por segundo, lo cual significa velocidades de 177Mbps, sin compresión. USB 1.1 permite una velocidad máxima de 12Mbits por segundo, mientras que con USB 2.0 se pueden alcanzar velocidades de hasta 470Mbits/sec². Por otro lado FireWire permite alcanzar velocidades de hasta 400 Mbps. Por tanto es recomendable que la cámara de video posea conexión USB 2.0 o FireWire, de forma de evitar el uso de un Frame Grabber Digital.³⁴

²La tasa efectiva es de 320 Mbps con solamente un periférico utilizando el bus

³Por consiguiente también el PC donde se utilice la cámara deberá poseer entrada USB 2.0 o Fire Wire, los cuales no se encuentran en placas madre antiguas.

⁴Para el caso de cámaras analógicas, la utilización de una tarjeta de adquisición (frame grabber) es obligatoria debido a que se debe digitalizar la imagen.

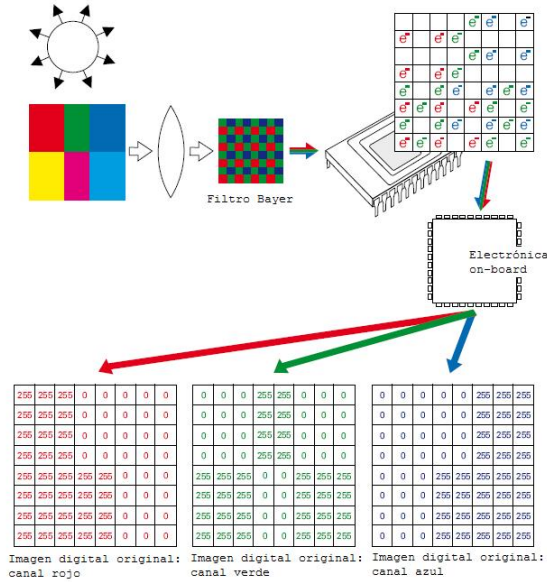


Figura 6.4: Principio de funcionamiento de una cámara color que utiliza solamente 1 chip CCD

6.4. Determinación del campo de visión

El campo de visión o Field-of-View (FOV), en una dirección dada se puede determinar como:

$$FOV = (D_p + L_v)(1 + P_a)$$

donde:

D_p es la longitud máxima del objeto a observar en la dirección del FOV

L_v es la variación máxima de la posición y orientación del objeto

P_a es el factor por el cual el FOV debe ser expandido de forma de permitir variaciones.

El objeto a observar es el Ojo de Bife, a través de una apertura rectangular de 15cm x 20cm. El objeto se observa a través de una apertura fija. A P_a se le atribuye un valor de 10 %.

$$FOV_{vert} = (15cm + 0cm)(1 + 0,1) = 16,5cm \quad (6.1)$$

$$FOV_{horiz} = (20cm + 0cm)(1 + 0,1) = 22,0cm \quad (6.2)$$

El campo de visión tendrá por consiguiente una relación de 4/3. La cámara a seleccionar tendrá una relación diferente; más 4:3 es muy común en las cámaras comerciales.

6.4.1. Tipo de lente

Elegir el lente implica seleccionar un tipo de lente y la distancia focal del mismo. En cuanto al tipo de lente, predominan 3 tipos [32]:

CCTV (C-Mount) Para una buena performance se recomiendan los CCTV de precios moderado o altos, debido a que los de precios bajos generalmente son diseñados para cámaras de vigilancia y son de baja performance.

35mm camera Una de las mejores opciones en cuanto a la relación calidad-precio. Una de las desventajas de estos lentes, es que poseen montura de bayoneta, y por tanto los lentes se mueven cuando son sometidos a golpes. Sin embargo existen bayonetas atornilladas para mantener los lentes en posición.

Amplificadores Estos lentes son óptimos para trabajar con distancias pequeñas. La disponibilidad de rangos de distancias focales es limitada. Son los que presentan el costo más elevado.

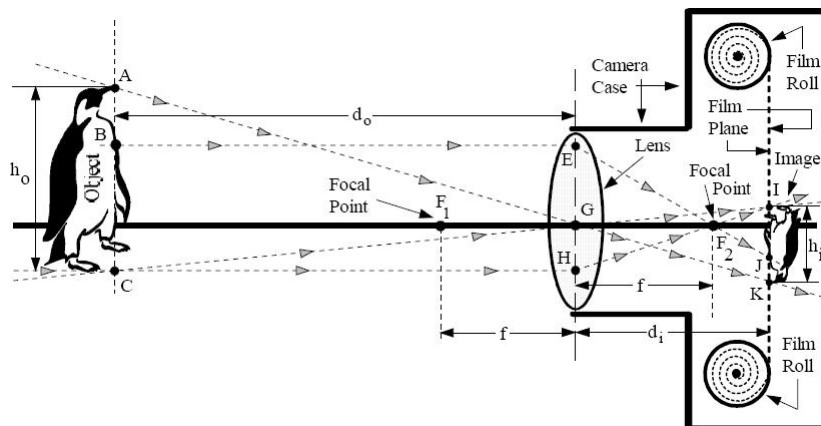


Figura 6.5: Imagen de un objeto captada en una cámara fotográfica

Las fórmulas para el cálculo de lentes son las siguientes [32]:

$$M_i = \frac{H_i}{H_o} = \frac{D_i}{D_o} \quad (6.3)$$

$$F = \frac{D_o M_i}{1 + M_i} \quad (6.4)$$

$$LE = D_i - F \quad (6.5)$$

donde:

M_i es la magnificación de la imagen













Imager Sizes	Formats (Type)	~Diag.
	1/7" - 1.85 x 1.39mm	2.3
	1/6" - 2.15 x 1.61mm	2.7
	1/5" - 2.55 x 1.91mm	3.2
	1/4" - 3.2 x 2.4mm	4.0
	1/3.6" - 4.0 x 3.0mm	5.0
	1/3.2" - 4.536 x 3.416mm	5.678
	1/3" - 4.8 x 3.6mm	6.0
	1/2.7" - 5.27 x 3.96mm	6.592
	1/2" - 6.4 x 4.8mm	8.0
	1/1.8" - 7.176 x 5.319mm	8.932
	2/3" - 8.8 x 6.6mm	11.0
	1" - 12.8 x 9.6mm	16.0

Figura 6.6: Formato y tamaño de sensores de imágenes

H_i es la altura de la imagen

H_o es la altura del objeto

D_i es la distancia a la imagen

D_o es la distancia al objeto

F es la distancia focal del lente (DF)

LE es la distancia entre el lente y la imagen de forma de lograr enfocar la misma

Los cálculos se realizan tomando la dirección en la cual el FOV es mayor. En el caso de estudio, $H_i = 220mm$, es decir el ancho del FOV calculado.

Con una distancia al objeto de 230mm (fija), la distancia focal F_{opt} del lente a utilizar dependerá del tamaño del sensor. Siguiendo las ecuaciones 6.3 y 6.4 las distancias focales óptimas se muestran en el cuadro 6.2

Para la elección de la combinación cámara-lente, es necesario considerar una cámara determinada, y en base a dicha elección seleccionar un lente que mejor aproxime a la distancia focal, calculada en función de la distancia al objeto a observar, según el cuadro 6.2

6.4.2. Cámaras y Lentes

En el cuadro 6.3 se listan algunas posibles cámaras digitales, con sus atributos principales. Las cámaras con interfaz Firewire requieren la utilización de una fuente

Formato	Distancia Focal(mm)
1/7"	1.91
1/6"	2.22
1/5"	2.63
1/4"	3.29
1/3.6"	4.10
1/3.2"	4.64
1/3"	4.91
1/2.7"	5.38
1/2"	6.5
1/1.8"	7.2
2/3"	8.84
1"	12.64

Cuadro 6.2: Relaciones entre el tamaño del sensor y la distancia focal óptima del lente

Marca	Modelo	Sensor	Montura	FPS	Res.	Interfaz	Precio
Sentech	TC-C33USB	1/3" CCD	C/CS	60	692x504	USB 2.0	
QImaging	Rolera-MGi	2/3" CCD	C	?	512x512	Firewire	
PicSight	P34B-USB	1/4" CCD	C	60	659x494	USB 2.0	
Videology	20K14XUSB	1/4" CCD	C	30	720x480	USB 2.0	
Imaging	DFK 21F04	1/4" CCD	C/CS	30	640x480	Firewire	USD 260
Imaging	21AF04	1/4" CCD	C/CS	60	640x480	Firewire	USD 350
Lumenera	LU 075M	1/3" CMOS	C	60	640x480	USB 2.0	USD 945
Basler	A601FC	1/2" CMOS	C	60	640x480	Firewire	USD 995

Cuadro 6.3: Cámaras

de alimentación externa, mientras que las que tienen una interfaz de tipo USB no⁵. La elección de la cámara condiciona la distancia focal óptima del lente, de modo que el FOV se registre lo más justo en la cámara. Algunos tipos de lentes se pueden observar en el cuadro 6.4

Si bien las cámaras con 3 CCDs son las de mejores prestaciones y por lo tanto las recomendadas, no se tomaron en consideración debido al alto precio que presentan. Una elección aconsejable sería una cámara Sentech STC-C33USB-B⁶, la cual utiliza un sensor CCD de 1/3" y un lente Computar T0412CS-3⁷, cuya distancia focal de 4mm es la que mejor se aproxima a la distancia focal óptima.

⁵Para la gran mayoría de los casos

⁶<http://www.sentechamerica.com/STC-C33USB-B.pdf>, visitada el 23 de Marzo de 2007

⁷http://www.cbcamerica.com/cctvprod/computar/mono/mono_pdf/T0412CS-3.pdf, visitada el 23 de Marzo de 2007

Marca	Modelo	DF(mm)	Montura	D(mm)	Largo(mm)	Precio
Imaging	T 2314 FICS-3	2.3	CS	34.50	33.00	USD 113
Imaging	H 2616 FICS-3	2.6	CS	35.00	33.00	USD 181
Imaging	H 3616 FICS-3	3.6	CS	34.50	33.00	USD 110
Imaging	T 0412 FICS-3	4.0	CS	32.00	30.00	USD 39
Computar	T0412CS-3	4.0	CS	39.00	39.40	USD 72
Imaging	T 0812 FICS-3	8.0	CS	32.00	33.00	USD 72
Computar	TO812CS	8.0	CS			USD 37
Computar	H0612FI	6.0	C	30.50	33.00	USD 52

Cuadro 6.4: Lentes

No obstante, la cámara y lente finalmente utilizados en el proyecto fueron distintos a cualquiera de los anteriores, debido a razones económicas.

6.5. Cámara utilizada en el proyecto

Debido a las limitaciones económicas, en este trabajo se utilizó una cámara fotográfica *Canon PowerShot G1*. Las especificaciones más relevantes de la misma se listan en el cuadro 6.5.

Resolución máxima	2048 x 1536
Resolución mínima	640 x 480
Razón ancho/alto	4:3
Tamaño CCD	1/1.8"
Array del filtro de color CCD	CYGM
Distancia focal	7(panorámico) - 21(teleobjetivo) mm
Auto Foco	Single or Continuous AF + Low Light Assist Lamp
Balance de blancos	Automático, 5 preestablecidos, personalizado
Formato sin compresión	Sí, RAW (propietario de Canon)
Formato comprimido	JPEG
Interfaz con computadora	Universal Serial Bus (USB), RS-232C/RS-422
Temperatura de operación	0-40°C

Cuadro 6.5: Propiedades relevantes de la cámara fotográfica Canon PowerShot G1

Canon brinda una librería propietaria llamada CD-SDK⁸, que provee una interfaz para acceder a las cámaras digitales Canon. Dicha librería posibilita comandar remotamente la cámara utilizada en este proyecto. Sin embargo, la mencionada librería funciona solamente con algunos modelos de cámaras Canon, y no todos permiten

⁸Canon Digital Camera Software Developers Kit por sus siglas en inglés

controlar la cámara remotamente: para algunos modelos es posible solamente realizar una descarga en el ordenador de las fotos tomadas, pero no comandar la cámara⁹. La utilización de dicho software ocasiona que el sistema desarrollado funcione solamente con cámaras Canon o compatibles que soporten la librería CD-SDK. Los requerimientos mínimos del ordenador en donde se corran aplicaciones que utilicen la antedicha librería son:

Hardware: IBM PC/AT compatible¹⁰

Sistema Operativo: Windows 95/98 (incluyendo la 2da edición), Windows NT 4.0 (Service Pack 3 o superior), Windows 2000, Windows XP

CPU: Pentium o superior

Interfaz: USB, RS-232C: puerto Serial o COM

RAM: 64 MB or superior (excepto Windows 2000/XP), 128MB o superior (Windows 2000/XP)

Resolución del monitor: 800x600 pixeles, 256 colores (8 bit) o superior.

6.6. Equipo de captura de imágenes

La iluminación en fotografía es muy importante, incidiendo tanto en la calidad de la fotografía (siendo un factor físico imprescindible), como en la percepción de los colores (ver figura 5.1). La dirección de la luz y la altura desde la que incide tiene una importancia decisiva en el aspecto general de la fotografía. El objetivo es obtener una luz uniforme en la superficie del Ojo de Bife, de forma de observar un color uniforme en el mismo y evitar deslumbramientos

De forma de tener una iluminación controlada y conocida, se utiliza un equipo de captura de imágenes para adquirir las fotografías del Ojo de Bife. Se desarrollaron dos equipos, A y B, los cuales utilizan técnicas de iluminación distintas.

El equipo de captura A, cuya arquitectura puede observarse en las figuras 6.7 y 6.8, utiliza una iluminación semi-directa, combinando la iluminación directa con difusores para alumbrar el Ojo de Bife. El flujo de la luz se dirige casi completamente sobre la zona a iluminar. El equipo de captura B utiliza un sistema de iluminación indirecta, reflejando la luz sobre una pantalla blanca. En la figura 6.9 y 6.10 puede observarse la disposición del mismo. En la figura 6.11 se muestran fotografías captadas con uno

⁹Cámaras que pueden ser controladas remotamente utilizando la librería CD-SDK: PowerShot G1, G2, S30, S40, S100, Pro90IS; Powershot A10, A20, A30, A40, A100, S110, S200, S300, S300a; DIGITAL IXUS v, v2, 300, 330; IXY DIGITAL 200, 200a, 300, 300a;

¹⁰No soporta ordenadores Macintosh

y otro equipo. Es de notar una mayor uniformidad en el color al capturar una imagen utilizando el equipo B, el cual evita también deslumbramientos no deseados los cuales se observan con facilidad en la imagen captada por el equipo A.

La arquitectura de ambos equipos lleva a que la componente especular de la iluminación sea captada, por tanto la técnica de medida espectométrica SCI es la que mejor se adapta a lo captado por los equipos.

Ambos equipos utilizan luminarias incandescentes halógenas, conocidas comúnmente como dicroicas. Dichas lámparas funcionan con voltaje de entrada de 220V o 12V, requiriendo éstas últimas el uso de transformadores 220V/12V. De forma de evitar descargas peligrosas para la integridad el operador, se recomienda el uso de lámparas dicroicas de 12V. Dado que las lámparas se encuentran conectadas en paralelo y funcionan a 12V, con un consumo individual de 50W, es necesario un transformador capaz de soportar una corriente de salida de $I = 16,7$ A. Seguidamente se muestran las especificaciones de un adaptador que se ajusta a las necesidades del sistema[40]:

Marca	Modelo	Entrada	Watts	Salida	I salida	Precio
V-Infinity	V SP-200-12	90-264VAC	200 W	12 V	16.7 A	\$ 63.33
V-Infinity	VU-SP-225-12A	127-370VAC	225 W	12 V	18.7 A	\$ 99.17

Protección de los equipos

El ambiente en el cual se desempeña el equipo es un ambiente de producción húmedo, expuesto a múltiples factores que pueden perjudicar al computador. Tanto el ordenador, el monitor y el transformador de corriente 220V AC/ 12V DC deben ser protegidos del ambiente, por ejemplo resguardándolos en un carro móvil cerrado.

Entre las posibilidades se recomienda la utilización del producto *Mobile Work Station* de Strongarm Inc.[38], el cual puede observarse en la figura 6.12. Entre las características más salientes encontramos:

- Material de acero inoxidable tipo 304
- Compartimiento interno para ordenador e impresora
- Pantalla táctil de 17.1", 19.1", 21.3"

Los cables de alimentación que unen el carro con el tomacorriente deben protegerse dentro de un cable de PVC resistentes a los golpes y a la humedad, los cuales pueden hallarse en el mercado local e internacional. A modo de ejemplo se cita el tubo WELLFLEX PVC 111 de la empresa NORRES[39]

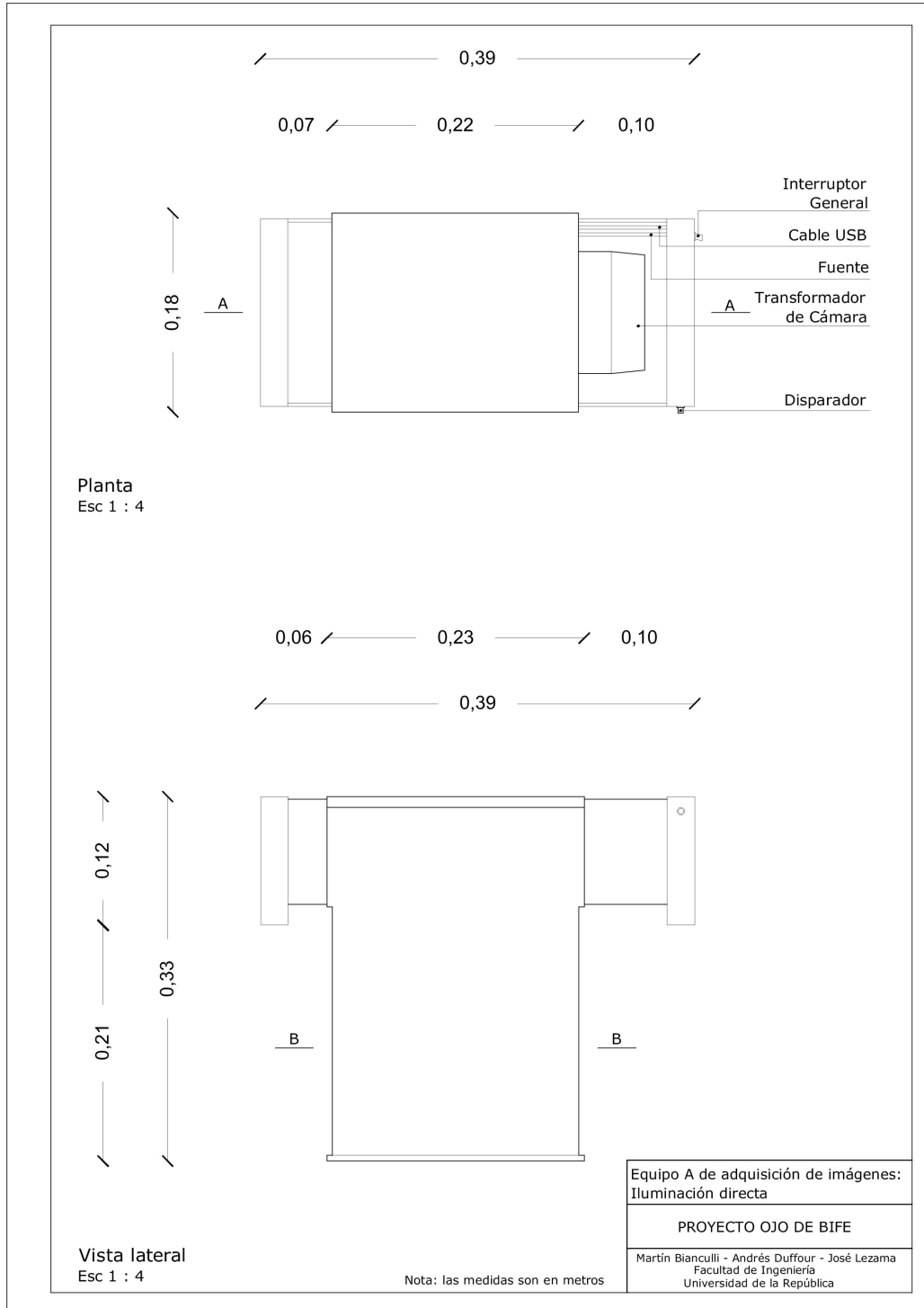


Figura 6.7: Planta y vista lateral del equipo A de captura de imágenes

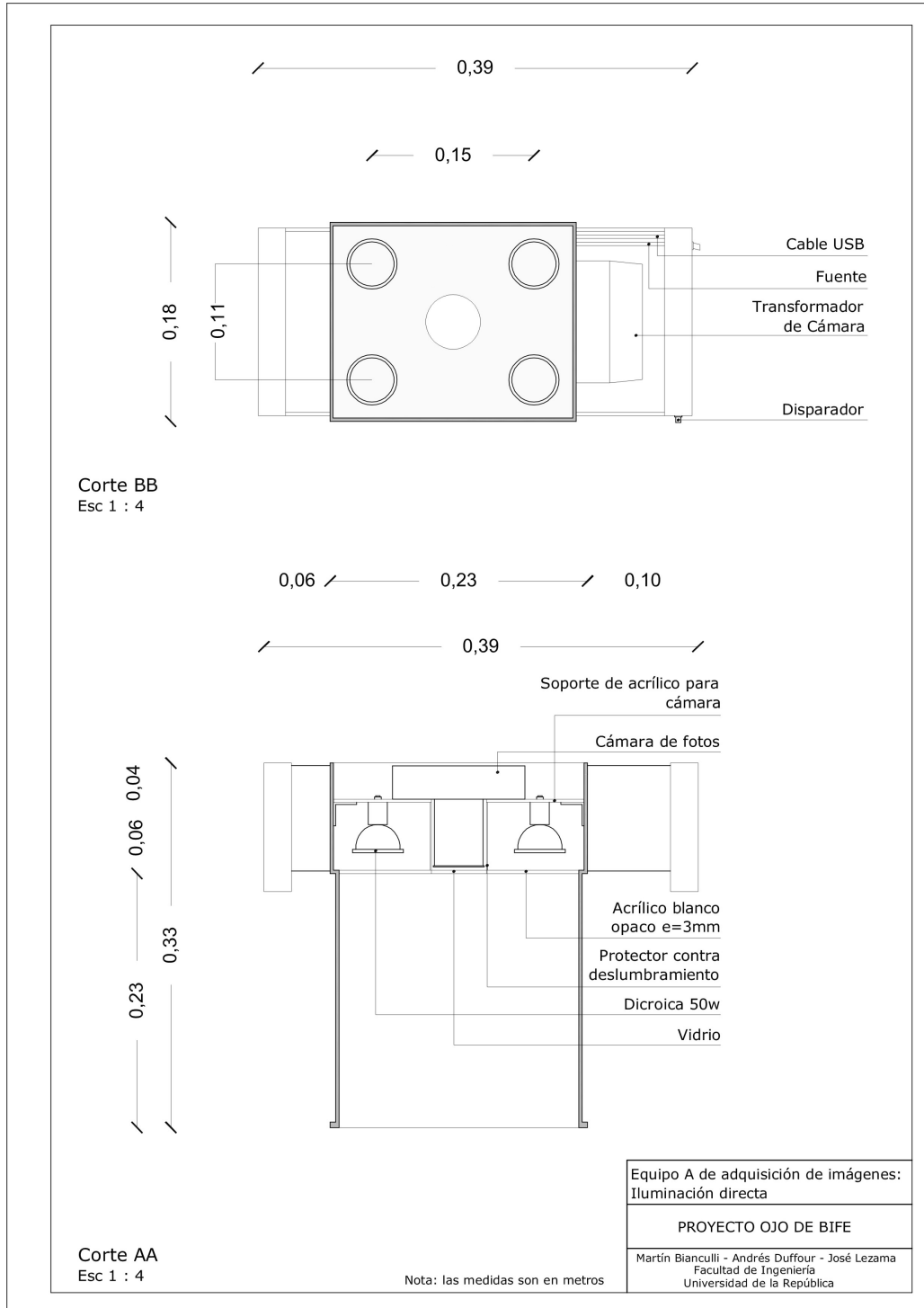


Figura 6.8: Cortes longitudinales del equipo A de captura de imágenes

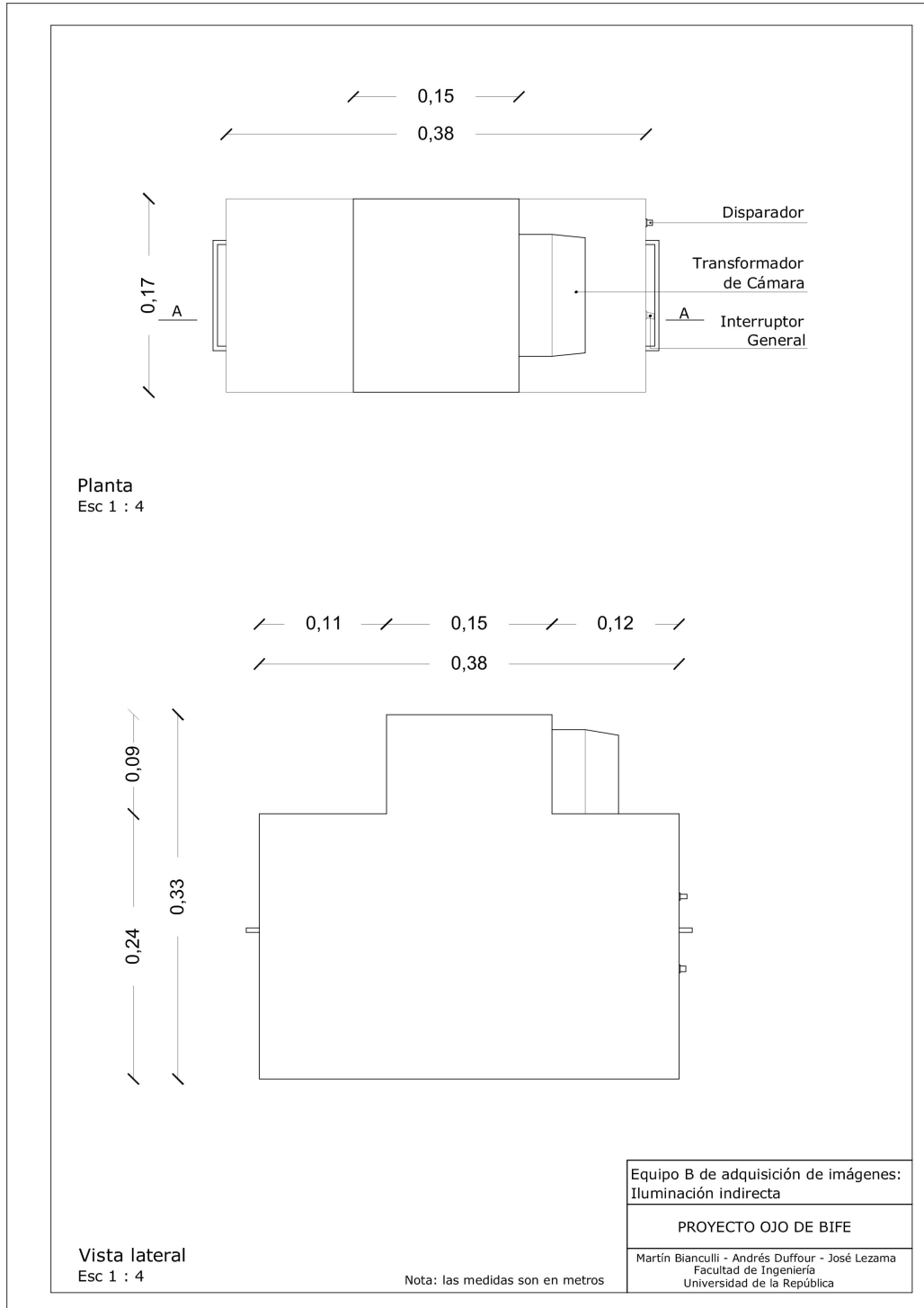


Figura 6.9: Planta y vista lateral del equipo B de captura de imágenes

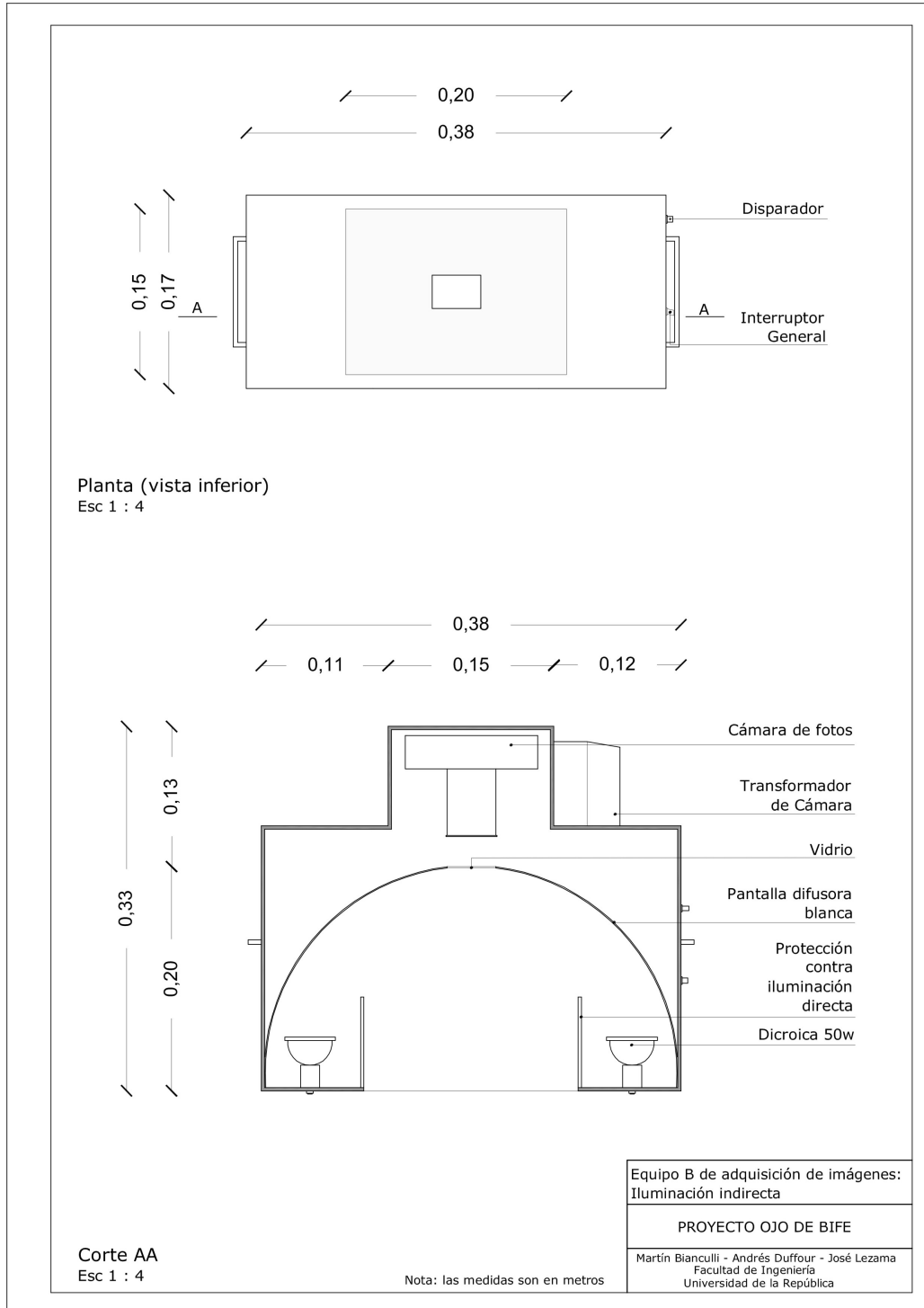


Figura 6.10: Cortes longitudinales del equipo B de captura de imágenes



Figura 6.11: Ejemplo de una costilla captada con el equipo A (izquierda), en el cual se observan deslumbramientos debido a la iluminación directa, y utilizando el equipo B (derecha), con el cual se logra una mayor uniformidad en la iluminación



Figura 6.12: Estación de trabajo móvil de la compañía Strongarm

Capítulo 7

Calibración geométrica

Para ajustar, con la mayor exactitud posible, las indicaciones del instrumento de captura con los valores de la magnitud que ha de medir, se realizaron dos clases de calibraciones: calibración de color y calibración geométrica. Lo concerniente a la calibración de color se trata en el capítulo referente al color.

En la calibración geométrica de la cámara el objetivo es determinar un conjunto de parámetros de la cámara que describan el mapeo entre las coordenadas de referencia en 3-D y las coordenadas en 2-D de la imagen. Se intentan determinar la geometría interna de la cámara y sus características ópticas (parámetros intrínsecos) y la posición y orientación en 3-D del cuadro de la cámara referente a un sistema de coordenadas fijo (parámetros extrínsecos). La calibración y la descripción del marco teórico brindados en este capítulo se basan en los trabajos de James Heikkilä y Olli Silvén [36] y K. Strobl, W. Sepp, S. Fuchs, C. Paredes and K. Arbter [37]

7.1. Calibración explícita

Los parámetros físicos de una cámara se pueden clasificar en intrínsecos y extrínsecos. Los parámetros intrínsecos especifican las características propias de la cámara, como ser la distancia focal (la distancia entre el lente de la cámara y el plano de la imagen), la posición del centro de la imagen (en coordenadas de píxel) y el tamaño efectivo del píxel. Los parámetros extrínsecos describen la transformación entre la cámara y el marco de referencia exterior y constan de una matriz de rotación y un vector de traslación. Estos últimos son necesarios para transformar las coordenadas de un objeto a coordenadas centradas en el cuadro de la cámara. El modelo “pinhole” de una cámara se basa en el principio de colinealidad, en donde cada punto del espacio del objeto es proyectado mediante una línea recta a través del centro de proyección al plano de la imagen, como puede observarse en la figura 7.1. El origen del sistema de coordenadas de la cámara se encuentra en el centro de proyección en la

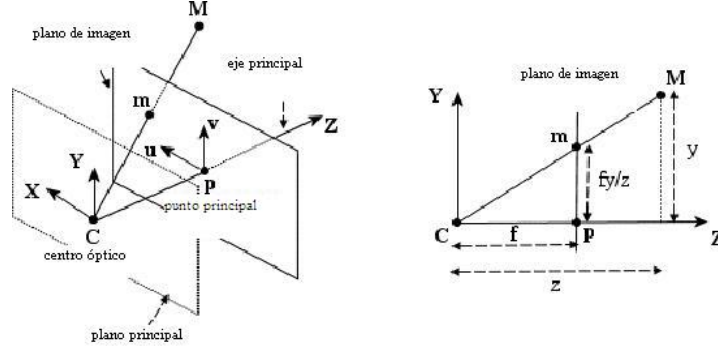


Figura 7.1: El modelo de cámara Pinhole

posición (X_0, Y_0, Z_0) con respecto al sistema de coordenadas del objeto, y el eje z del cuadro de la cámara es perpendicular al plano de imagen. La rotación es representada utilizando los ángulos de Euler $(\omega, \varphi, \kappa)$. La orientación de un cuerpo rígido puede especificarse completamente considerando un sistema de coordenadas fijo a él. Pero es necesario especificar la relación entre un sistema de ejes fijo en el espacio y uno fijo en el cuerpo. Es posible construir la transformación entre estos dos sistemas con tres rotaciones sucesivas del cuerpo rígido, introduciendo los ángulos de Euler. Las rotaciones son aquí representadas en sentido horario, primero en torno al eje x , luego en torno al eje y y finalmente en torno al eje z .

De forma de expresar la posición de un objeto arbitrario M en la posición (X_i, Y_i, Z_i) en coordenadas de imagen (cuyo origen es el punto principal P situado en (u_0, v_0)), es necesario transformarlo previamente a coordenadas de cámara (x_i, y_i, z_i) . Dicha transformación se compone de una traslación y una rotación, y puede ser alcanzada utilizando la siguiente matriz de proyección de cámara:

$$\begin{bmatrix} x_i \\ y_i \\ z_i \end{bmatrix} = \begin{bmatrix} m_{11} & m_{12} & m_{13} \\ m_{21} & m_{22} & m_{23} \\ m_{31} & m_{32} & m_{33} \end{bmatrix} \begin{bmatrix} X_i \\ Y_i \\ Z_i \end{bmatrix} + \begin{bmatrix} x_0 \\ y_0 \\ z_0 \end{bmatrix} \quad (7.1)$$

donde

$$\begin{aligned} m_{12} &= \sin\omega \sin\varphi \cos\kappa - \cos\omega \cos\kappa & m_{11} &= \cos\varphi \cos\kappa \\ m_{22} &= \sin\omega \sin\varphi \sin\kappa + \cos\omega \cos\kappa & m_{21} &= \cos\varphi \sin\kappa \\ m_{13} &= \cos\omega \sin\varphi \sin\kappa + \sin\omega \sin\kappa & m_{31} &= -\sin\varphi \\ m_{23} &= \cos\omega \sin\varphi \sin\kappa - \sin\omega \cos\kappa & m_{32} &= \sin\omega \cos\varphi \\ m_{33} &= \cos\omega \cos\varphi \end{aligned} \quad (7.2)$$

Los parámetros intrínsecos incluyen la distancia focal f , un factor de escala s_u y el centro de la imagen (u_0, v_0) . Las unidades de las coordenadas de la imagen son

dadas en píxeles, por tanto es necesario la utilización de los coeficientes D_u y D_v para realizar el pasaje de unidades métricas a píxeles. Dichos coeficientes son linealmente dependientes de la distancia focal f y del factor de escala s_x . Siguiendo el modelo de cámara pinhole, la proyección del punto (x_i, y_i, z_i) al plano de imagen se expresa como:

$$\begin{bmatrix} \tilde{u}_i \\ \tilde{v}_i \end{bmatrix} = \frac{f}{z_i} \begin{bmatrix} x_i \\ y_i \end{bmatrix} \quad (7.3)$$

Las coordenadas de imagen en píxeles (u'_i, v'_i) se obtienen de la proyección de $(\tilde{u}_i, \tilde{v}_i)$ mediante la aplicación de la siguiente transformación:

$$\begin{bmatrix} u'_i \\ v'_i \end{bmatrix} = \begin{bmatrix} D_u s_u \tilde{u}_i \\ D_v \tilde{v}_i \end{bmatrix} + \begin{bmatrix} u_0 \\ v_0 \end{bmatrix} \quad (7.4)$$

En resumen, la relación entre un punto M del espacio y su proyección m en el plano imagen se puede representar mediante la siguiente fórmula:

$$m = A[Rt]M$$

donde A es la matriz intrínseca de la cámara:

$$A = \begin{bmatrix} f_x & 0 & c_x \\ 0 & f_y & c_y \\ 0 & 0 & 1 \end{bmatrix}$$

donde (c_x, c_y) son las coordenadas del punto principal, (f_x, f_y) son las componentes de los ejes x e y de la distancia focal, R, t son los parámetros extrínsecos: la matriz de rotación y el vector de traslación.

$$R = \begin{bmatrix} r_{11} & r_{12} & r_{13} \\ r_{21} & r_{22} & r_{23} \\ r_{31} & r_{32} & r_{33} \end{bmatrix}, t = \begin{bmatrix} t_1 \\ t_2 \\ t_3 \end{bmatrix}$$

Distorsión del lente

El modelo pinhole es una aproximación de la proyección real de la cámara, el cual se extiende con correcciones que toman en cuenta la sistemática distorsión de las coordenadas de la imagen introducidas por imperfecciones en el lente. La corrección más comúnmente hallada es para la distorsión radial del lente, que causa que el punto imagen sea desplazado radialmente en el plano de la imagen. La distorsión radial se puede aproximar mediante la siguiente expresión:

$$\begin{bmatrix} \delta u_i^{(r)} \\ \delta v_i^{(r)} \end{bmatrix} = \begin{bmatrix} \tilde{u}_i(k_1 r_i^2 + k_2 r_i^4 + \dots) \\ \tilde{v}_i(k_1 r_i^2 + k_2 r_i^4 + \dots) \end{bmatrix} \quad (7.5)$$

donde k_1, k_2, \dots son los coeficientes que modelan la distorsión radial, y $r_i = \sqrt{\tilde{u}_i^2 + \tilde{v}_i^2}$. Típicamente bastan uno o dos coeficientes para compensar la distorsión radial.

El hecho de que los centros de curvatura de la superficie de los lentes no sean estrictamente colineales, introduce otro tipo de distorsión, la cual tiene una componente radial así como una tangencial. La expresión para introducir este tipo de distorsión es:

$$\begin{bmatrix} \delta u_i^{(t)} \\ \delta v_i^{(t)} \end{bmatrix} = \begin{bmatrix} 2p_1 \tilde{u}_i \tilde{v}_i + p_2 (r_i^2 + 2\tilde{u}_i^2) \\ p_1 (r_i^2 + 2\tilde{v}_i^2) + 2p_2 \tilde{u}_i \tilde{v}_i \end{bmatrix} \quad (7.6)$$

donde p_1 y p_2 son los coeficientes para la distorsión tangencial.

Otros tipos de errores pueden ser considerados, sin embargo en la mayoría de los casos el error introducido es pequeño y la componente de distorsión es despreciable. Por tanto no se considerarán otro tipo de errores que los mencionados anteriormente.

Un modelo de cámara para realizar una calibración que tome en cuenta tanto el error radial como el error tangencial se deriva de combinar el modelo pinhole de la cámara con las componentes de distorsión radial y tangencial:

$$\begin{bmatrix} u_i \\ v_i \end{bmatrix} = \begin{bmatrix} D_u s_u (\tilde{u}_i + \delta u_i^{(r)} + \delta u_i^{(t)}) \\ D_v (\tilde{v}_i + \delta v_i^{(r)} + \delta v_i^{(t)}) \end{bmatrix} + \begin{bmatrix} u_0 \\ v_0 \end{bmatrix} \quad (7.7)$$

En resumen, sea (u, v) las coordenadas reales del píxel (con proyección ideal, sin distorsión), y (\tilde{u}, \tilde{v}) los pixeles observados (distorsionados). De igual forma, sean (x, y) y (\tilde{x}, \tilde{y}) las coordenadas sin y con distorsión físicas de la imagen. Así,

$$\begin{aligned} \tilde{x} &= x + x[k_1 r^2 + k_2 r^4] + [2p_1 xy + p_2 (r^2 + 2x^2)] \\ \tilde{y} &= y + y[k_1 r^2 + k_2 r^4] + [2p_1 xy + p_2 (r^2 + 2y^2)] \end{aligned} \quad (7.8)$$

donde $r^2 = x^2 + y^2$. El centro de distorsión radial es el mismo que el punto principal. Dado que $\tilde{u} = c_x + f_x u$ y $\tilde{v} = c_y + f_y v$, el sistema resultante puede reescribirse como:

$$\begin{aligned} \tilde{u} &= u + (u - c_x)[k_1 r^2 + k_2 r^4 + 2p_1 y + p_2 (r^2/x + 2x)] \\ \tilde{v} &= v + (v - c_y)[k_1 r^2 + k_2 r^4 + 2p_2 x + p_1 (r^2/y + 2y)] \end{aligned} \quad (7.9)$$

En el modelo, el conjunto de parámetros intrínsecos f, s_u, u_0, v_0 se corrige con los coeficientes de distorsión $k_1, \dots, k_n, p_1, p_2$. Estos parámetros se conocen también como parámetros físicos de la cámara, debido a que poseen un significado físico. En general, el objetivo de la calibración explícita de la cámara es determinar los valores óptimos para los mencionados parámetros, basándose en imágenes de un objeto 3-D conocido.

7.2. Proceso de calibración

La calibración realizada en este trabajo se basa en [37] y consta básicamente de dos pasos: un primero de inicialización, y luego un paso de optimización no lineal.

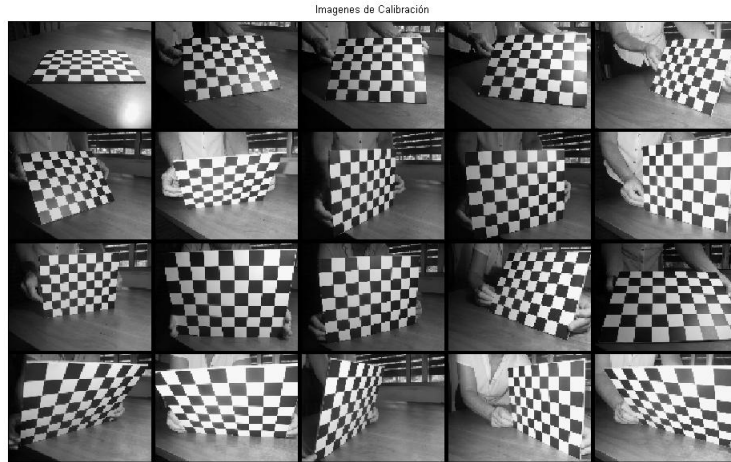


Figura 7.2: Imágenes utilizadas para la calibración geométrica de la cámara

El paso de inicialización computa una solución para los parámetros de calibración en donde se excluyen las distorsiones causadas por el lente. La optimización no lineal minimiza el error de proyección, en un sentido de mínimos cuadrados, sobre todos los parámetros de calibración. La optimización es hecha mediante el método gradiente descendiente iterativo.

El procedimiento para hallar los parámetros de calibración y la estimación de la deformación causada por el lente se realizó utilizando la Camera Calibration Toolbox for Matlab, realizada por Klaus Strobl, Wolfgang Sepp, Stefan Fuchs y Cristian Paredes[37]. El procedimiento consiste en adquirir imágenes de un damero cuadrado en distintas posiciones e inclinaciones, de forma de comparar lo que la cámara efectivamente capta, con lo que teóricamente debería captar (las dimensiones del damero son conocidas). Manualmente se le indican los cuatro ángulos para realizar una predicción de donde se deberían encontrar las esquinas de los cuadrados, así como el tamaño de los mismos. Las fotos utilizadas para dicho procedimiento pueden observarse en 7.2. Las esquinas de los cuadrados son extraídas automáticamente, con una precisión de 0.1 píxel.

En la figura 7.4 puede observarse la reproyección de los píxeles sobre la imagen, la cual es calculada utilizando los parámetros extrínsecos e intrínsecos de la cámara .

7.3. Resultados de la calibración

Los parámetros de calibración obtenidos fueron los siguientes:

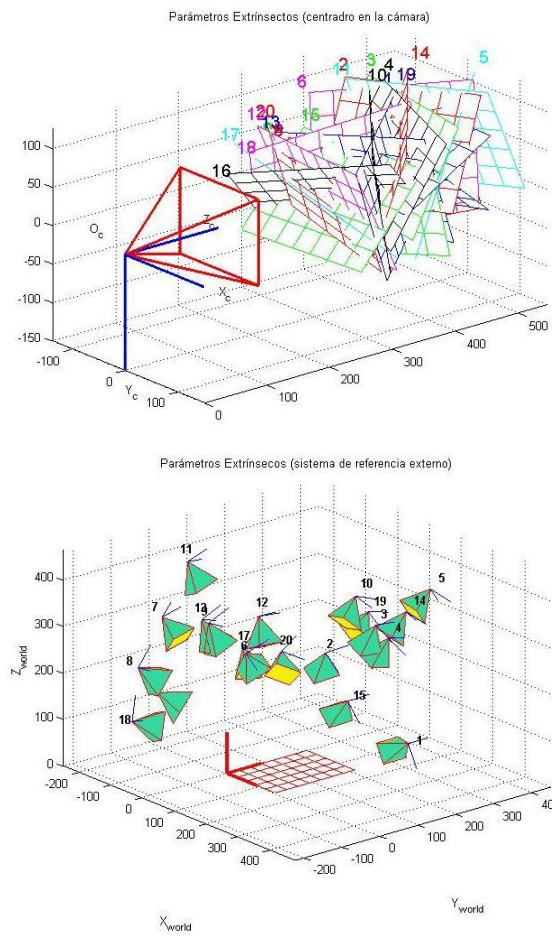


Figura 7.3: Posiciones de la grilla utilizadas para la toma de imágenes de la grilla de calibración: vista desde la cámara y vista desde un sistema de referencia externo

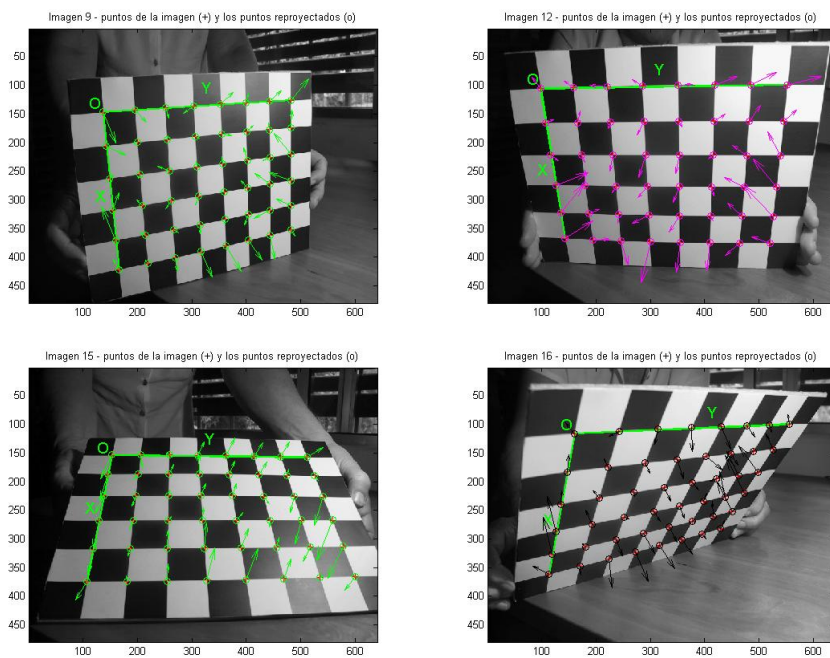


Figura 7.4: En las distintas imágenes se observan los ángulos detectados y la reproyección de los mismos sobre la imagen. Cada flecha representa el desplazamiento efectivo de los píxeles introducidos por el lente.

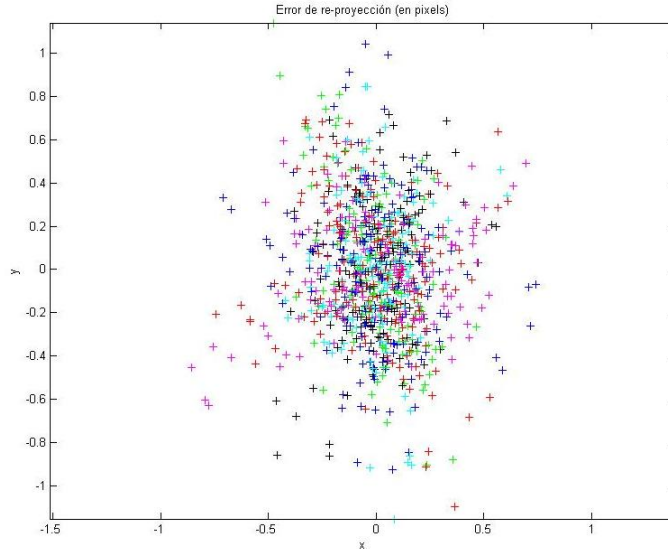


Figura 7.5: Error de reproyección de los píxeles en la imagen

$$\begin{array}{l}
 \text{Distancia focal:} \\
 \text{Punto Principal:} \\
 \text{Distorsión:}
 \end{array}
 \left| \begin{array}{l}
 fc = [644,99769 ; 641,81830] \pm [1,42775 \ 1,47776] \\
 c = [316,92825 \ 246,71124] \pm [3,67352 \ 2,52671] \\
 kc = [-0,19330 \ 0,20465 \ -0,00077 \ -0,00082 \ 0,00000] \\
 \pm [0,01416 \ 0,06358 \ 0,00089 \ 0,00112 \ 0,00000]
 \end{array} \right.$$

Los valores de la distancia focal, el cual es un único valor en mm, se encuentran expresados en unidades de píxeles horizontales y verticales. La razón entre dichos coeficientes se conoce como relación de aspecto (aspect ratio), y el que esta no sea igual a uno se debe al hecho de que el sensor trabaja con píxeles no exactamente cuadrados. Los primeros dos términos de distorsión corresponden a los términos de segundo y cuarto orden de la distorsión radial, k_1 y k_2 . El quinto término corresponde al coeficiente de distorsión de sexto orden k_3 . El tercer y cuarto término corresponden a los coeficientes de distorsión tangencial p_1 y p_2 , como se puede observar en 7.5 y 7.6

Los mapas del modelo de distorsión tangencial, radial, y total puede observarse en la figura 7.6. Obsérvese que la mayor distorsión se produce en las esquinas, en donde la misma trepa hasta los 14 píxeles.

7.3.1. Influencia en las medidas obtenidas

Como se observa en la figura 7.6, las mayores distorsiones se dan en el borde del lente. Considerando que la imagen del Ojo de Bife se encuentra centrada, los mayores errores ocurrirán en el borde de la imagen, no afectando considerablemente a las

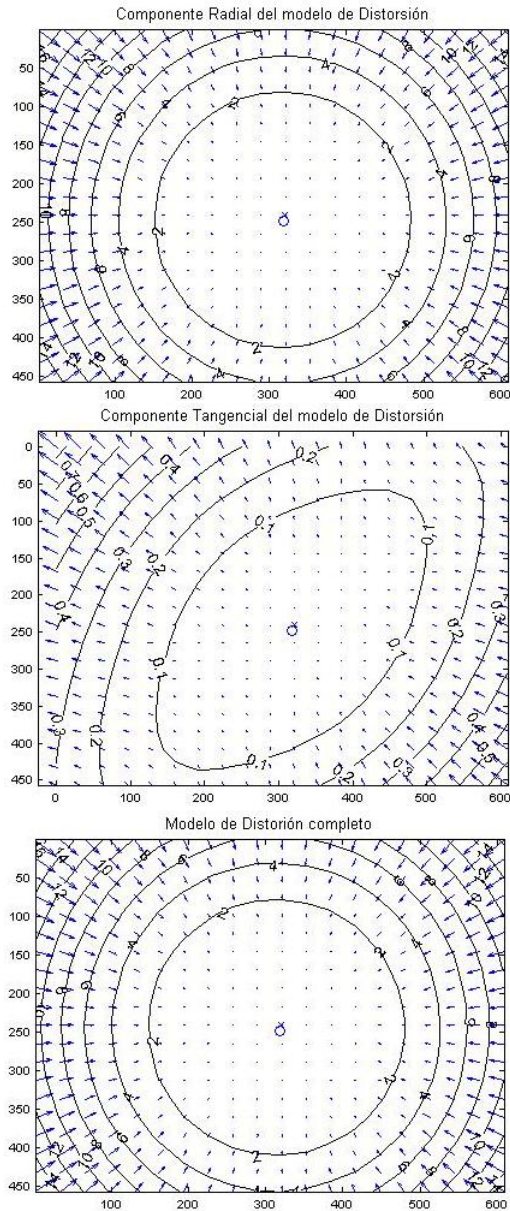


Figura 7.6: Mapa de los modelos de distorsión radial, tangencial y total causada por las imperfecciones de la cámara. Los valores que aparecen en las líneas, las cuales indican zonas de igual desplazamiento, están en pixeles. Cada flecha representa el desplazamiento efectivo de los pixeles introducidos por el lente

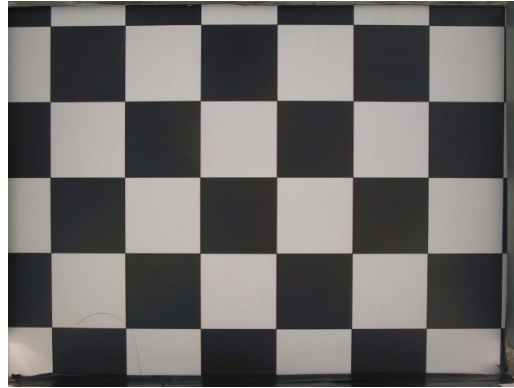


Figura 7.7: Fotografía que se utilizó para hallar la relación entre píxeles y centímetros. Los cuadrados miden 3 centímetros por 3 centímetros.

medidas de área.

Las medidas de espesor de grasa subcutáneas se toman en la parte inferior de las imágenes, en donde los desplazamientos oscilan entre los 2 y 8 píxeles. Considerando que 1 píxel corresponde aproximadamente a 0.3mm (ver siguiente sección), y tomando como ejemplo un espesor de 2cm, el error cometido puede llegar hasta los 0.24cm, correspondiendo al 12%; para espesores de 1cm el error en cambio puede trepar hasta el 24%.

De lo antedicho se desprende que es necesario corregir las deformaciones del lente. Para ello se utilizan los parámetros hallados en las secciones anteriores.[37] [36]

7.4. Relación entre píxeles y centímetros

Para calcular la medida en centímetros de lo que representa un píxel, se tomó una fotografía del damero cuyos cuadrados miden 3x3 cm, luego con un programa de edición de imágenes se contaron en distintas partes de la imagen cuantos píxeles abarcaban los cuadrados de 3x3. Haciendo un promedio se llegó que cada píxel representa 0,031579 centímetros. En la figura 7.7 se muestra la imagen que se tomó del damero.

Capítulo 8

Arquitectura de Software: Planeación y Elaboración

8.1. Documento de Requerimientos

Ojo De bife	Versión: 1.0
Arquitectura de Software	Fecha: 16 de abril de 2007

8.1.1. Introducción

En este capítulo se presenta una descripción de alto nivel de la arquitectura del software de Ojo De Bife. Se presenta una descripción de las metas, los casos de uso soportados por el sistema, y los componentes que han sido utilizados para alcanzar los objetivos de los casos de uso.

8.1.2. Propósito

El Objetivo es crear un sistema amigable que permita adquirir y procesar imágenes del Ojo de Bife de una res, ubicado entre la 3a y la 4ta costilla (comúnmente aparece en el corte pistola a 3 costillas) de forma sencilla, debiendo permitir consultar los datos obtenidos de cada imagen. La información a extraer de la costilla consiste en:

- área del Ojo de Bife
- porcentaje de marmoreo
- espesor de grasa intramuscular
- grado de terneza

En términos generales, la meta es introducir automatización en las medidas, las cuales son efectuadas manualmente por un operario (experto). Se busca introducir mayor rapidez, y un grado similar de exactitud en las medidas.

8.1.3. Clientes

Frigoríficos

8.1.4. Descripción de la Arquitectura

Ojo De Bife fue desarrollado en C\C++, debido a la gran eficiencia que dicho lenguaje brinda, lo cual es un requisito para la implementación de algoritmos que se ejecuten en tiempo real. El entorno de desarrollo utilizado fue Visual Studio 6.0, las librerías utilizadas fueron MFC 6.0, OpenCv (para procesamiento de imágenes), SQLite 3.3.7 (para Base de Datos)

8.1.5. Actores

A continuación se muestra el actor a interactuar con el sistema.

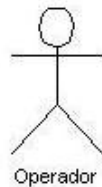


Figura 8.1: Actores del sistema

8.1.6. Atributos

Facilidad de uso: El operador deberá indicar el momento de la captura, debiendo presionar simultáneamente dos botones, uno para activar la iluminación y otro para ordenar la captura. El botón de iluminación deberá permanecer presionado hasta escuchar un sonido que indica la finalización de la captura. Ambos botones se encuentran en el equipo de adquisición.

Metáfora de interfaz: Amigable con el usuario

Tiempo de respuesta: El sistema deberá ser capaz de capturar y procesar la imagen en un tiempo menor a 15 seg.

Plataforma del sistema operativo: Windows XP

Tolerancia a fallas:

Base de datos: Se mantendrá un registro de las imágenes adquiridas así como de los datos de ellas extraídas

8.1.7. Funciones básicas

Ref.	Función	Categoría
R 1.1	Registrar el ingreso de un cuarto	Evidente
R 1.2	Capturar una imagen	Evidente
R 1.3	Segmentar el Ojo de Bife	Oculto
R 1.4	Calcular área del Ojo de Bife	Evidente
R 1.5	Medir espesor de la grasa intramuscular	Evidente
R 1.6	Medir porcentaje de marmoreo	Evidente
R 1.7	Dar medidas objetivas de color	Evidente
R 1.8	Guardar los datos en base de datos	Evidente
R 1.9	Consultar las medidas realizadas	Evidente
R 1.10	Calibrar colorimetría cámara	Evidente

8.2. Casos de uso

8.2.1. Caso de uso 1

Caso de uso: Conectar cámara

Actores: Operador

Propósito Conectar la cámara que se encuentra en el equipo de adquisición y el software, de modo de poder establecer una comunicación entre el software y la cámara

Resumen Previo a poder realizar las medidas, es necesario conectar el software del sistema con la cámara de fotos.

Tipo Primario, esencial

Referencias cruzadas R 1.2, R 1.3, R 1.4, R 1.5, R 1.6, R 1.7

Curso normal de eventos:

Acción del actor	Acción del sistema
1 El operador indica al sistema que desea conectar la cámara de fotos	Se buscan las cámaras que se encuentran conectadas al ordenador y se despliega una ventana mostrando las mismas
2 Se elige la cámara	
3	Se conecta la cámara y habilita la posibilidad de capturar y procesar imágenes

Curso alternativo de eventos:

Acción del actor	Acción del sistema
Linea 2	No se encuentran cámaras conectadas, se despliega en pantalla un mensaje informativo
Linea 3	La opción de capturar y procesar una imagen permanece deshabilitada

8.2.2. Caso 2

Caso de uso: Medir Ojo de Bife

Actores: Operador, Res

Propósito Capturar una foto y realizar las medidas del Ojo de Bife

Resumen El operador registra el ingreso de un nuevo cuarto de res, ingresando un número de identificación. Luego le indica al sistema que desea capturar y procesar una imagen, indicando el momento en que se deberá realizar la captura presionando simultáneamente dos botones, uno para activar la iluminación y otro para capturar. El sistema procede a capturar una imagen, segmentar el Ojo de Bife y calcular su área, medir el porcentaje de marmoreo, calcular los valores colorimétricos CIELab, y calcular el grosor de la grasa subcutánea. El identificador, el área del Ojo de Bife, el porcentaje de marmoreo, el espesor de grasa subcutánea, los valores CIELab de colorimetría, y el directorio donde se encuentran las fotos (la original y la analizada) se guardan en una base de datos.

Tipo Primario, real

Referencias cruzadas R 1.1, R 1.2, R 1.3, R 1.4, R 1.5, R 1.6, R 1.7, R 1.8, R 1.9

Curso normal de eventos:

Acción del actor	Acción del sistema
1 Operador ingresa el identificador de un cuarto de res en pantalla	
2 Se indica el momento de capturar y procesar una imagen, presionando dos botones simultáneamente.	Se saca una fotografía
3	Se segmenta el Ojo de Bife y se calcula su área
4	Se calcula el porcentaje de marmoreo
5	Se calcula el espesor de grasa de cobertura
6	Se calculan los valores colorimétricos CIELab
7	Se guarda el directorio de la imagen analizada y los datos de la misma en la base de datos
8	Se muestra la imagen analizada y sus datos en pantalla
9	La cámara permanece conectada, y queda habilitada la posibilidad de realizar una nueva captura y procesamiento

Curso alternativo de eventos:

- Línea 2 El operador indica capturar una imagen sin haber indicado un identificador. El sistema da una alerta, sin realizar la captura.

8.2.3. Caso 3

Caso de uso: Seleccionar propiedades de captura

Actores: Operador

Propósito: Seleccionar la carpeta en donde guardar las imágenes adquiridas

Resumen: El operador, una vez conectada la cámara, puede indicarle al sistema la carpeta en donde desea guardar las imágenes a adquirir.

Tipo: secundario

Referencias cruzadas: R 1.2, R 1.9

Curso normal de eventos

Acción del actor	Acción del sistema
1 Indica que desea seleccionar las propiedades de captura, accediendo al menú correspondiente	Se despliega en pantalla una ventana que permite al operador seleccionar la carpeta
2 Se selecciona la carpeta en la cual se quieren guardar las imágenes	
3 Se presiona OK	

Curso alternativo de eventos

8.2.4. Caso 4

Caso de uso: Consultar base de datos

Actores: Operador

Propósito Consulta los datos previamente adquiridos.

Resumen El operador indica al sistema para consultar la base de datos, pudiendo utilizar distintos criterios de consulta(excluyentes): por valor de identificador, valor de área de ojo, porcentaje de marmoreo. Presionando el botón consultar, se despliega en pantalla los datos de la imagen así como el directorio en donde se encuentra guardada la misma.

Tipo Primario, esencial

Referencias cruzadas R 1.8, R 1.9

Curso normal de eventos

Acción del actor	Acción del sistema
1 Indica que desea consultar la base de datos presionando un botón	Se despliega en pantalla una ventana para realizar la consulta
2 Se ingresan los criterios de búsqueda (exclusivos) y se indica consultar la base	Se despliegan los valores en pantalla

Curso alternativo de eventos

8.2.5. Caso 5

Caso de uso: calibrarColor

Actores: Operador

Propósito Hallar los elementos necesarios para realizar el mapeo entre los distintos espacios de colores

Resumen El operador indica al sistema que desea calibrar el color, en donde el programa responde desplegando un cartel que indica al operador que sitúe la tabla de colores en posición y encienda las luces del equipo. Una vez listo, el operador presiona ok, y el sistema captura una imagen de la tabla. El sistema realiza una validación de los datos, de forma de corroborar que la tabla fue puesta en posición correcta. De no ser así, se solicita al operador que realice la captura nuevamente.

Tipo Primario, esencial

Referencias cruzadas R 1.5, R 1.8, R 1.10

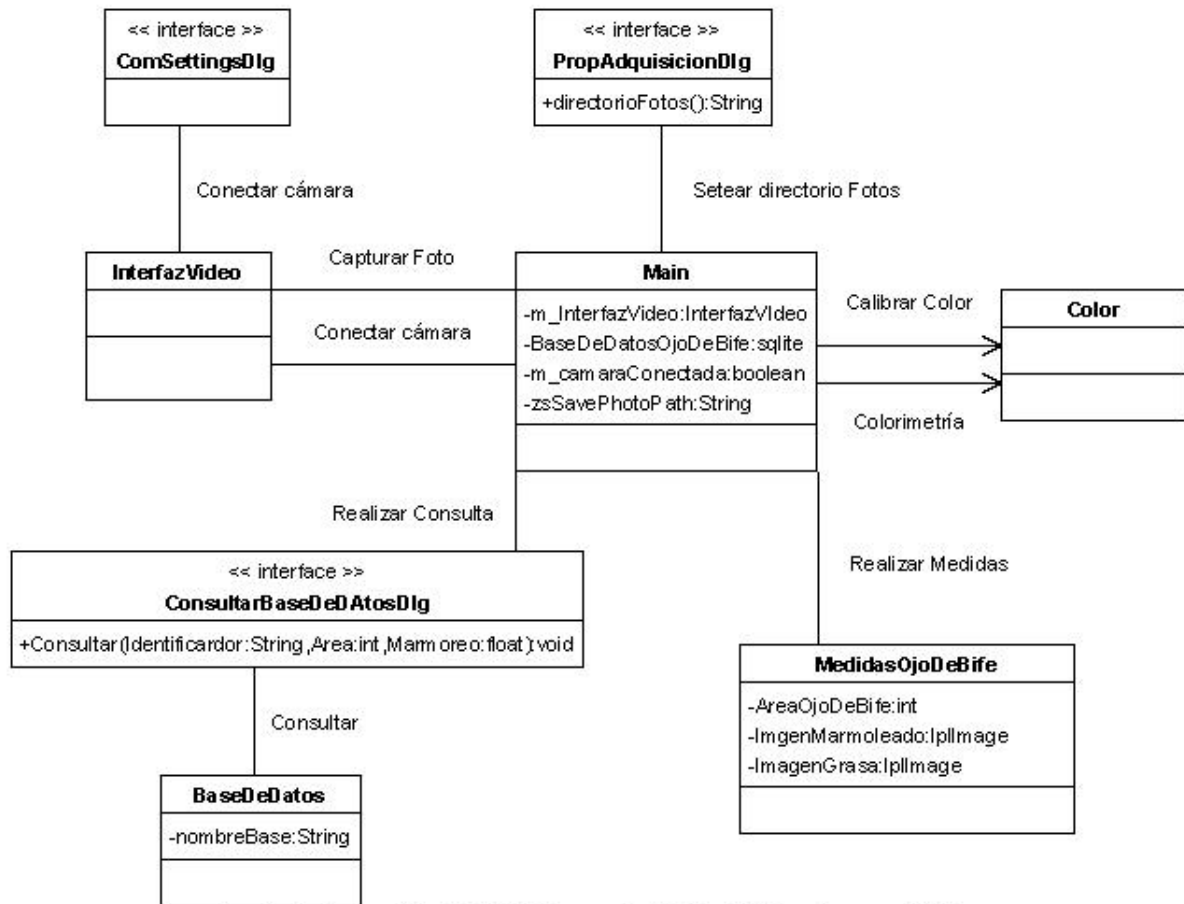
Curso normal de eventos

Acción del actor	Acción del sistema
1 Se indica que desea calibrar la colorimetría de la cámara, mediante un ítem del menú Calibración	Despliega en pantalla indicación para que el operador sitúe la tabla de colores en posición y encienda la iluminación
2 Se coloca la tabla de datos en posición y se presiona OK	Se toma una fotografía y se muestra la misma en pantalla
3	Se realiza una validación de la toma, y se calibra la matriz de color

Curso alternativo de eventos

- Linea 3 El sistema invalida la imagen de entrada, solicitando al operador de situar la tabla en posición y realizar el procedimiento nuevamente

8.3. Modelo Conceptual



Created with Poseidon for UML Community Edition. Not for Commercial Use.

Figura 8.2: Modelo conceptual del sistema

8.4. Diagramas de secuencia del sistema



Figura 8.3: Diagrama de secuencia para el caso de uso Conectar cámara

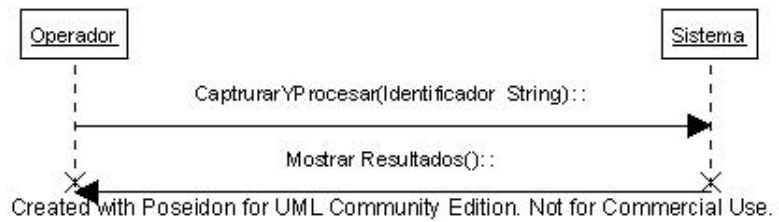


Figura 8.4: Diagrama de secuencia para el caso de uso Medir Ojo de Bife

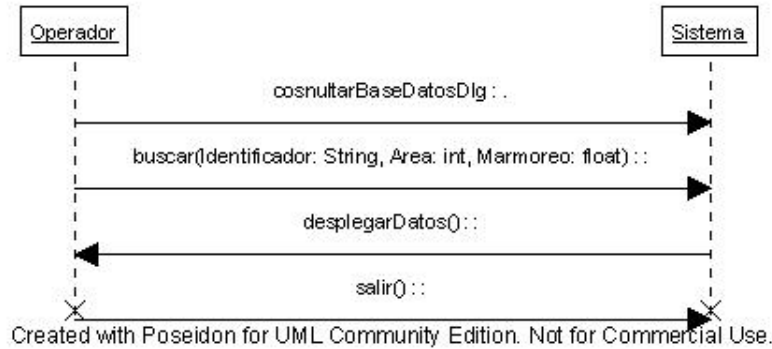


Figura 8.5: Diagrama de secuencia para el caso de uso Consultar Base de Datos



Figura 8.6: Diagrama de secuencia para el caso de uso Seleccionar Propiedades de Captura

8.5. Contratos

Contrato

Nombre: conectarCámara()

Responsabilidades: Conectar la cámara de adquisición con el sistema

Tipo: Sistema

Referencias Cruzadas: R 1.2

Notas:

Excepciones: No existen cámaras conectadas al ordenador; drivers no instalados o en mal funcionamiento

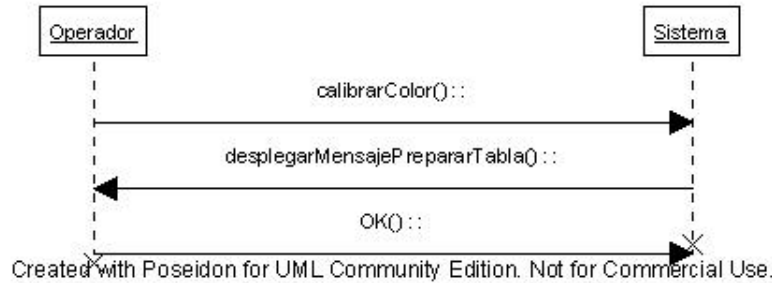


Figura 8.7: Diagrama de secuencia para el caso de uso calibrarColor

Salida:

Precondiciones: Controladores (drivers) instalados en el ordenador, cámara conectada correctamente al ordenador

Postcondiciones:

- Si existe cámara conectada y compatible, se conectó la cámara y se habilitó captura y procesamiento de imágenes
- Si no existen cámaras conectadas, captura y procesamiento permanecen inhabilitados

Contrato

Nombre: setearPropiedadesCaptura()

Responsabilidades: Fijar la carpeta en donde se almacenarán las imágenes capturadas por el programa

Tipo: Sistema

Referencias Cruzadas: R 1.2

Notas:

Excepciones:

Salida:

Precondiciones: La cámara se encuentra conectada

Postcondiciones:

- Se fija el directorio en donde guardar las imágenes adquiridas

Contrato

Nombre: capturarProcesar()

Responsabilidades: Capturar una imagen y desencadenar la realización de todas las medidas sobre la costilla: segmentación del ojo de bife y cálculo del área, estimación del grado de terneza, porcentaje de marmoreado, espesor grasa subcutánea. Dibujar el contorno del Ojo de Bife. Validar la correcta segmentación de la imagen. Guardar la información obtenida en la base de datos.

Tipo: Sistema

Referencias Cruzadas:

Notas: Todas las medidas deberán realizarse en un tiempo menor a 15 segundos

Excepciones:

Salida: Imagen del Ojo de Bife marcado y las medidas realizadas

Precondiciones: Equipo de captura conectado al equipo

Postcondiciones:

- Se atribuyeron los siguientes datos a la imagen: identificador, valor área del Ojo de Bife, porcentaje de marmoreo, espesor grasa de cobertura, valores colorimétricos CIELab
- Se marcó el Ojo de Bife
- Se guardaron los datos adquiridos y el directorio y nombre de la imagen analizada en la base de datos
- Se mostraron los datos y la imagen analizada en pantalla
- El sistema se encuentra listo para capturar y procesar una nueva imagen

Contrato

Nombre: consultarBaseDeDatos()

Responsabilidades: Desplegar los valores de acuerdo a los criterios de selección desplegados por el usuario

Tipo: Sistema

Referencias Cruzadas: R1.2, R1.3, R1.4 ,R1.5, R1.6, R1.7, R1.8, R1.9

Notas:

Excepciones:

Salida: Datos adquiridos mostrados en pantalla

Precondiciones:

Postcondiciones:

- El contenido de la base de datos permanece incambiado

Contrato

Nombre: calibrarColor()

Responsabilidades: Hallar la matriz necesaria para realizar la conversión entre los espacios de color

Tipo: Sistema

Referencias Cruzadas: R1.7, R1.8, R1.10

Notas: Se utiliza un polinomio de 10 términos

Excepciones:

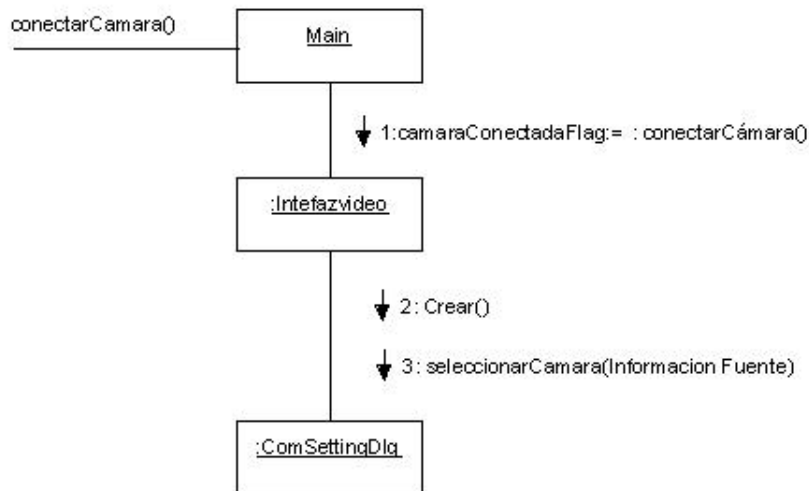
Salida: Foto adquirida mostrada en pantalla, matriz de calibración

Precondiciones: La cámara se debe encontrar conectada al sistema

Postcondiciones:

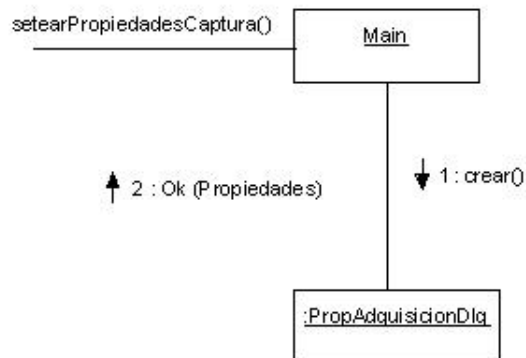
- La matriz de calibración es hallada

8.6. Diagramas de colaboración



Created with Poseidon for UML Community Edition. Not for Commercial Use.

Figura 8.8: Diagrama de colaboración conectarCamara



Created with Poseidon for UML Community Edition. Not for Commercial Use.

Figura 8.9: Diagrama de colaboración setearPropiedadesCaptura

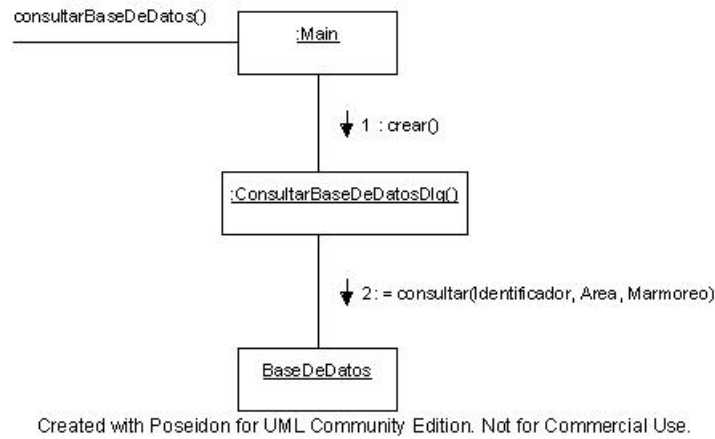


Figura 8.10: Diagrama de colaboración consultarBaseDeDatos

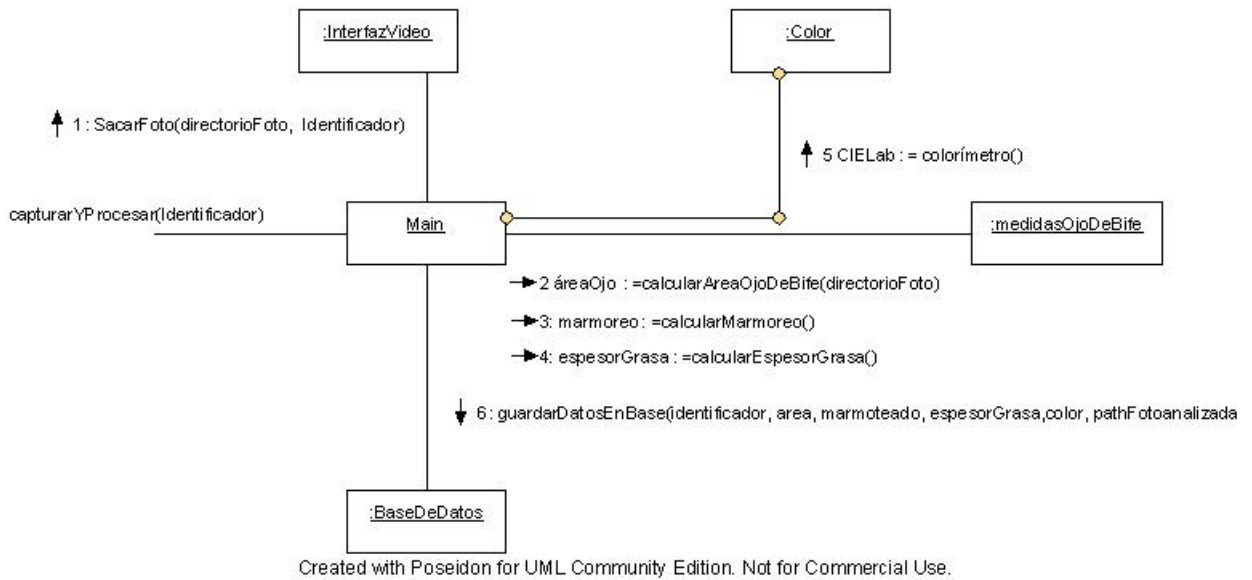
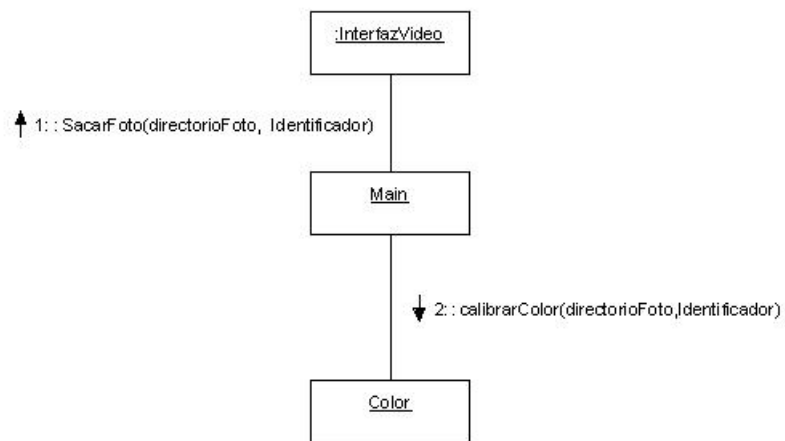


Figura 8.11: Diagrama de colaboración capturarProcesar



Created with Poseidon for UML Community Edition. Not for Commercial Use.

Figura 8.12: Diagrama de colaboración `calibrarColor`

Capítulo 9

Validación

9.1. Introducción

Durante la etapa de desarrollo del algoritmo de segmentación surge la necesidad de evaluar el algoritmo de forma objetiva.

La forma de validar la segmentación lograda por el algoritmo es comparar ésta con una segmentación manual hecha por un experto, o en caso de no contar con la misma, comparar con una segmentación manual aproximada del Ojo de Bife hecha por los propios estudiantes. Por lo tanto se desarrolló un programa de validación del algoritmo de segmentación en lenguaje Matlab, que compara ambas marcas y arroja determinados indicadores que se explicarán más adelante. A su vez, se compararon los resultados de la primera versión del algoritmo y de la última versión de este.

Se trabajó con una base de 64 imágenes marcadas por un experto y 272 imágenes marcadas por los estudiantes del proyecto. En total se utilizaron 336 imágenes distintas de costillas para la validación del algoritmo.

Con la base de imágenes marcadas por un experto se comparó el desempeño de la primera versión que se desarrolló del algoritmo y la versión definitiva de este. Las imágenes marcadas por los estudiantes fueron validadas sólo con la última versión del algoritmo. Las imágenes disponibles se realizaron con distintos equipos y bajo distintas condiciones, pudiendo ser divididas en cinco grupos:

- Grupo A: **122** Imágenes tomadas con una primera versión del equipo, en Frigorífico Colonia en mayo de 2006.
- Grupo B: **44** Imágenes proporcionadas por el Frigorífico Colonia, cortesía de Lautaro Perez.
- Grupo C: **47** Imágenes tomadas con la segunda versión del equipo en Frigorífico Colonia en febrero de 2007.

- Grupo C.bis: **59** Imágenes de prueba tomadas con la primera y segunda versión del equipo en Frigorífico Colonia en febrero de 2007.
- Grupo D: **64** Imágenes tomadas por el IIE en 2005, cortesía de Álvaro Gómez.

9.1.1. Indicadores

Se calcularon en principio dos indicadores, pero se podrían utilizar otros indicadores para la validación del algoritmo de segmentación. Los indicadores que se calcularon fueron los mismos utilizados en el proyecto [17], y estos son:

Indicador 1: Área de la unión sobre área de la intersección: Dadas dos marcas el indicador se calcula como $1 - A_i/A_u$, donde A_i corresponde al área de la intersección de las marcas y A_u al área de la unión. En el caso en que las dos marcas coincidan (y sólo en este caso) el indicador vale 0 y aumenta a medida que disminuya el solapamiento entre ambas marcas.

Indicador 2: Diferencia entre áreas: Dadas dos marcas el indicador se calcula como $|\frac{A_1 - A_2}{A_1}|$, donde A_1 corresponde al área marcada por un experto y A_2 al área marcada por el algoritmo. En el caso en que las dos marcas coincidan (y sólo en este caso) el indicador vale 0 y aumenta a medida que disminuya el solapamiento entre ambas marcas

9.1.2. Versiones

- Version 1.0: Primera versión del algoritmo de segmentación del Ojo de Bife
 - Implementaba incorrectamente el algoritmo FCM pues introducía los ceros en los cálculos de los clusters, lo que llevaba a un resultado incorrecto. Esto fue solucionado quitando los ceros del vector de entrada del algoritmo FCM.
 - Faltaban datos experimentales para fijar los parámetros del algoritmo de Convex Hull.
 - No implementaba la función de eliminar penínsulas.
- Version 1.1: Versión final del algoritmo
 - Se corrigió el algoritmo FCM
 - Se optimizaron los parámetros del algoritmo de Convex Hull.
 - Se implementó la función de eliminar penínsulas utilizando información de forma.

9.2. Resultados

En los siguientes cuadros se detalla, para cada indicador y para cada base de imágenes, el porcentaje de las mismas que caen en cada rango del indicador.

Rangos	Num. referencia	Grupo A	Grupo B	Grupo C
0.00-0.05	1	4 %	5 %	23 %
0.05-0.10	2	33 %	66 %	32 %
0.10-0.15	3	21 %	18 %	11 %
0.15-0.20	4	9 %	9 %	6 %
0.20-0.25	5	9 %	2 %	9 %
0.25-0.30	6	9 %	0 %	0 %
0.30-1	7	15 %	0 %	19 %

Cuadro 9.1: Resultados del Indicador 1

Rangos	Referencia	Grupo A	Grupo B	Grupo C
0.00-0.05	1	40 %	62 %	53 %
0.05-0.10	2	17 %	23 %	11 %
0.10-0.15	3	5 %	11 %	9 %
0.15-0.20	4	7 %	2 %	6 %
0.20-0.25	5	8 %	2 %	4 %
0.25-0.30	6	12 %	0 %	2 %
0.30-1	7	12 %	0 %	15 %

Cuadro 9.2: Resultados del Indicador 2

Los siguientes cuadros muestran las diferencias entre la primera versión y la versión definitiva del algoritmo de segmentación para los indicadores 1 y 2.

Rangos	Referencia	Primera Versión	Última Versión
0.00-0.05	1	0 %	3 %
0.05-0.10	2	12.96 %	36 %
0.10-0.15	3	9.26 %	17 %
0.15-0.20	4	9.26 %	13 %
0.20-0.25	5	12.96 %	11 %
0.25-0.30	6	11.11 %	6 %
0.30-1	7	44.44 %	14 %

Cuadro 9.3: Diferencia entre distintas versiones del algoritmo con imágenes marcadas por un experto en el indicador 1

Rangos	Referencia	Primera Versión	Última Versión
0.00-0.05	1	11.11 %	31 %
0.05-0.10	2	14.81 %	20 %
0.10-0.15	3	5.56 %	14 %
0.15-0.20	4	1.85 %	6 %
0.20-0.25	5	5.56 %	8 %
0.25-0.30	6	12.96 %	8 %
0.30-1	7	46.3 %	13 %

Cuadro 9.4: Diferencia entre distintas versiones del algoritmo con imágenes marcadas por un experto en el indicador 2

9.2.1. Análisis de resultados

Lo primero que salta a la vista es que los mejores resultados se obtuvieron con las imágenes proporcionadas por Lautaro Pérez (Grupo B), luego en las imágenes tomadas por la última versión del equipo desarrollada por el grupo de proyecto en Febrero de 2007 (Grupo C) y por último, los peores resultados se ven en las imágenes tomadas con la primera versión del equipo en Mayo de 2006 (Grupo A). Ello indica que una buena calidad en la captura resulta muy importante para el desempeño del algoritmo propuesto.

Para el indicador 1 ($1 - A_i/A_u$) se obtuvo:

- un 89 % de las imágenes del Grupo B se encuentran en el rango 0-0,15
- en el caso de los grupos A y C, en este rango se encuentran el 58 % y el 66 %.

Para el indicador 2 ($|\frac{A_1 - A_2}{A_1}|$) se obtuvo que:

- el 85 % de las imágenes del grupo B se encuentran en el rango 0-0,1 y el 96 % en el rango 0-0,15 lo cual se puede calificar como un muy buen resultado.
- el 79 % de las imágenes del grupo C se encuentran en el rango 0-0,2 y dentro de este mismo rango el 69 % de las imágenes del grupo A.

En cuanto a las distintas versiones del programa, se ve claramente en el cuadro 9.3 y 9.2 que los resultados mejoran considerablemente con la nueva versión del algoritmo.

En las siguientes figuras se muestran de forma gráfica los resultados obtenidos para las distintas bases de imágenes y en el caso de las imágenes marcadas por un experto (Grupo D) las diferencias entre la primera y la última versión del algoritmo.

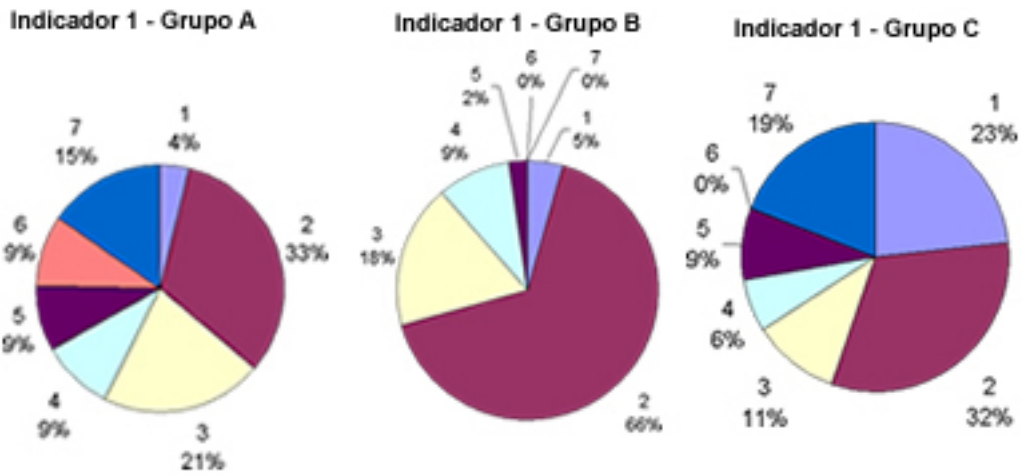


Figura 9.1: Indicador 1: $1 - A_i/A_u$ para las tres bases de imágenes marcadas por los estudiantes.

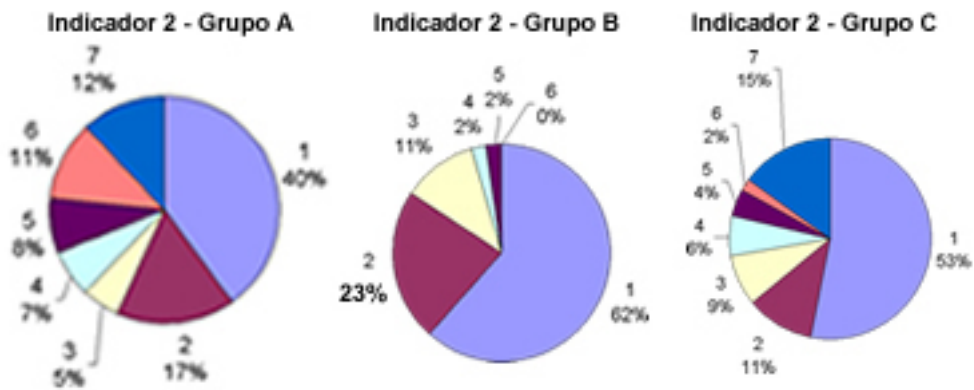


Figura 9.2: Indicador 2: $|\frac{A_e - A_s}{A_s}|$ para las tres bases de imágenes marcadas por los estudiantes.

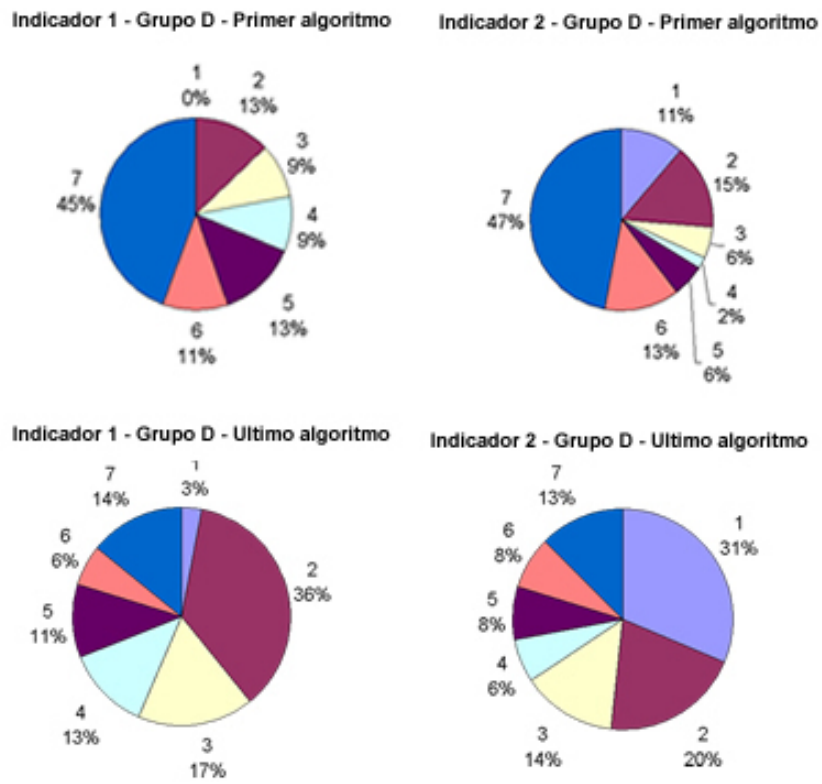


Figura 9.3: Comparación entre la primera y la última versión del algoritmo, para las imágenes marcadas por un experto.

9.3. Estudio de Robustez de las Medidas

Se realizó un estudio estadístico de la estabilidad de las medidas que efectúa el programa procesando varias imágenes de una misma costilla. Para esto se utilizó una costilla sin lomo comprada en una carnicería. Se capturaron y procesaron 21 imágenes de esta costilla y se calcularon la media, la varianza y la desviación estándar de cada medida.

Los resultados muestran que los valores de dispersión menores son los de área, marmoreado y de las componentes de color a y b. En el cuadro 9.5 se muestran los 21 resultados de las medidas sobre la misma costilla.

ID Res	Área	Marmoreado	Grasa sub. 1	Grasa sub. 2	L	a	b
1	64	3,55	0,31	1,53	-3,19	44,59	46,97
2	60	3,62	0,54	0,25	-4,14	45,25	50,68
3	62	3,11	0	1,01	-5,12	44,59	46,97
4	64	2,95	0	0,95	-1,19	44,59	46,97
5	69	2,99	0,21	0,99	-1,43	45,91	45,63
6	64	3,13	0,03	0,03	-3,5	41,33	45,63
7	68	3,50	0,39	1,38	-3,19	44,59	46,97
8	66	2,94	0,23	1,05	-2,9	47,59	48,26
9	67	2,72	0,28	1,13	-5,12	44,59	46,97
10	60	3,50	0,21	1,41	-3,5	45,91	45,63
11	66	2,77	1,09	0,03	-7,43	41,33	45,63
12	63	2,61	0,82	3,47	-5,5	41,33	45,63
13	65	3,41	0,16	1,63	-3,19	44,59	46,97
14	65	3,04	0,15	1,84	-5,12	44,59	46,97
15	66	3,35	0,23	1,44	-7,43	45,91	45,63
16	65	3,07	0,12	1,18	-5,5	41,33	45,63
17	65	2,73	0,15	2,11	-2,9	43,38	48,26
18	64	2,38	0,79	0,03	-4,76	43,38	48,26
19	65	2,77	0,19	1,47	-4,76	39,05	48,26
20	67	2,86	0,27	0,8	-4,76	47,59	48,26
21	68	3,03	0,58	0,09	-5,12	44,59	46,97

Cuadro 9.5: Estudio estadístico de las medidas sobre una misma costilla

Sobre estos resultados se calculó la media, varianza y desviación estándar, estos valores se muestran en el cuadro 9.6

ID Res	Área	Marmoreado	Grasa sub. 1	Grasa sub. 2	L	a	b
Media	64,90	3,05	0,32	1,13	-4,27	44,09	47,01
Varianza	5,59	0,11	0,08	0,66	2,63	4,73	1,72
Desv. Est.	2,36	0,33	0,28	0,81	1,62	2,17	1,31

Cuadro 9.6: Resultados estadísticos del estudio

9.4. Validación del Marmoreo y del Espesor de grasa Subcutánea

No se realizó la validación del marmoreo ya que no se contaba con medidas objetivas de este.

Lo mismo ocurre con el espesor de grasa de cobertura, no se tienen medidas objetivas de este y por otro lado el espesor depende de donde se tome la medida ya que el mismo es irregular en distintas partes de la costilla. De todas maneras se realizó un estudio de validación, con la información de espesor de grasa de 40 de las imágenes que fueron proporcionadas por Lautaro Pérez. Se midió manualmente el espesor de grasa de las costillas intentando tomar las medidas en los dos puntos donde las toma el programa. Se compararon estas medidas con las obtenidas por el programa. Como era de esperar, debido a la irregularidad de la grasa subcutánea y al hecho que no coincide el lugar donde se mide, la dispersión del error cometido es grande, como se muestra en el cuadro 9.7.

Rangos de error	Referencia	Resultados
0.00-0.05	1	8 %
0.05-0.10	2	13 %
0.10-0.20	3	27 %
0.2-0.30	4	15 %
0.30-0.40	5	19 %
0.40-0.50	6	3 %
0.50-1	7	15 %

Cuadro 9.7: Rangos de error cometido al comparar medidas de espesor de grasa subcutánea entre medidas manuales y medidas del programa. Se utilizó una base de 40 imágenes.

Capítulo 10

Conclusiones

10.1. Conclusiones Generales

Como conclusión general, se trató de un proyecto muy interesante, que plantea un problema de tratamiento de imágenes por computadora que consistía en un verdadero desafío de ingeniería. Por otro lado, es un proyecto completo, que abarca tanto software como hardware y la integración de estos dos. Esta característica hace que el proyecto sea más motivante en comparación con un proyecto puramente de software. En el trabajo desarrollado, el hardware influía considerablemente en el software a implementar, ya que el algoritmo de segmentación depende ampliamente de las condiciones en que sea tomada la fotografía de la costilla.

Se cumplió con los objetivos del proyecto, se realizó un software que, conectado a un equipo de toma de imágenes, captura una imagen de una costilla, segmenta el músculo del Ojo de Bife y realiza las medidas de área de este, marmoreado, espesor de la grasa subcutánea y colorimetría. El software, por otro lado, presenta una interfaz de usuario amigable, pensada para ser usada por un operario de un frigorífico.

En cuanto al desarrollo de la solución del problema, se hicieron una cantidad considerable de pruebas de ensayo y error, buscando las mejores condiciones de la toma de imágenes y los mejores parámetros para el algoritmo de segmentación. El desarrollo de este último, resultó ser la parte más difícil del proyecto, ya que lograr una solución robusta, dada las diferencias que pueden haber entre distintas imágenes de distintas costillas, fue una tarea que requirió varias iteraciones de ensayo y error y se extendió a lo largo de todo del proyecto.

10.2. Conclusiones del desarrollo del Proyecto

Durante el desarrollo se adquirieron conocimientos, así como también experiencia práctica en desarrollo de software, que consideramos que es un hecho de gran valor

ya que es importante tener este tipo de práctica para ir conociendo los tiempos de cada uno a la hora de desarrollar software.

Al tratarse de un proyecto de hardware, se obtuvieron conocimientos y experiencia en gestión de recursos en un proyecto, búsqueda de materiales en el mercado y fabricación del mismo hardware, integración entre los diversos componentes.

Las visitas a los frigoríficos fueron de gran interés, permitiéndonos ver el funcionamiento de estos, que constituyen una de las principales industrias de nuestro país. Como experiencia fue sin duda un muy valioso aporte.

Finalmente cabe destacar que esta experiencia nos ha enseñado que la validación y documentación del proyecto es una parte muy importante del mismo y que requiere de mucho trabajo. Esta etapa se llevó a cabo paralelamente al transcurso del proyecto, previendo que no se acumulara para el final del mismo. Esta estrategia dio buenos resultados ya que la carga horaria de ésta parte del proyecto se vio sólo levemente incrementada en la etapa final.

10.3. Conclusiones sobre los algoritmos implementados

En cuanto al algoritmo de segmentación del músculo *Longissimus Dorsi*, cabe mencionar que fue de gran ayuda el trabajo realizado por S. Jeyamkondan [6], el cual aportó una importante base para el desarrollo del algoritmo de segmentación del Ojo de Bife.

En este proyecto, se llegó más lejos que el trabajo de S. Jeyamkondan, al implementar la etapa de eliminación de penínsulas que no pertenecen al Ojo de Bife, etapa que dicho trabajo no realizaba y que aporta significativas mejoras en muchos casos.

El algoritmo de segmentación fue la parte del proyecto que requirió mayores iteraciones de ensayo y error, buscando las mejores combinaciones de funciones y parámetros del algoritmo para encontrar una solución robusta y que funcionara de la mejor manera posible para la mayor cantidad de imágenes posible.

El hecho de que las imágenes difieren mucho entre sí, sobre todo entre distintos lotes de animales, requirió que se enfocara gran parte del trabajo en el desarrollo de este algoritmo. Cabe mencionar aquí también, que el diseño del hardware fue igual de importante, ya que el algoritmo de segmentación depende fuertemente de las características con que haya sido tomada la foto. Por lo tanto, el hardware debía estar correctamente diseñado de forma de lograr una iluminación uniforme y sin brillos sobre la costilla. Con tal propósito se diseñaron e implementaron dos equipos, los cuales arrojaron resultados muy diversos.

La validación del algoritmo de segmentación se realizó comparando la segmentación hecha por el programa contra la segmentación hecha a mano. Se obtuvieron resulta-

dos satisfactorios en la segmentación del Ojo de Bife, alcanzando un error de menos del 10 % para aproximadamente el 70 % de las imágenes.

En lo que a colorimetría refiere, fue este un primer acercamiento al tema, dado que no se encontraban trabajos previos realizados en dicho campo en el IIE, por lo cual se requirió un amplio estudio de la teoría así como de las formas de implementación de la misma, arrojando resultados alentadores, proponiéndose posibles mejoras.

10.4. A futuro

El proyecto quedó abierto a trabajos futuros en los que se puede mejorar el proyecto, apuntando hacia un producto totalmente robusto y capaz de competir mano a mano con los productos similares disponibles en el mercado, que alcanzan costos altos en el mercado local.

Algunos items que creemos son importantes para la continuación del proyecto son:

- Seguir mejorando el algoritmo de segmentación, agregando por ejemplo, que el algoritmo pueda aprender de segmentaciones anteriores y de esta manera autocalibrarse.
- Mejorar la calibración, de acuerdo a lo planteado en dicha sección.
- Implementar un sensor que determine automáticamente si el aparato de toma de imágenes está correctamente apoyado en la costilla
- Implementar un algoritmo que analice la distribución de la grasa intramuscular en el Ojo de Bife, comparándola por ejemplo con una serie de patrones.

Desde nuestro punto de vista, el desarrollo de tecnologías para la industria de la carne a nivel nacional es muy importante y pensamos que este proyecto nos demostró que se puede desarrollar una solución a un problema de importancia para la industria a un costo relativamente bajo, que si bien no funciona de manera perfecta, se puede seguir mejorando hasta alcanzar un producto altamente confiable.

Bibliografía

- [1] B.R. Schutte, N. Biju, G.A. Kranzler and H.G. Dolezal. *Color Video Image Analysis for Augmenting Beef Carcass Grading*. 1998 Animal Science
- [2] R.C. Cannell, J. D. Tatum, K.B. Belk, G.C. Smith, J. Wise B.R. Schutte, H.G. Dolezal, J.B. Morgan and G.A. Kranzler. 1999. *Evaluations of the Canadian Vision System, the HunterLab Miniscan, the Minolta Colorimeter and the Hunter-Lab/CSU BeefCam Technologies for Augmenting Application of USDA Yield and Quality Grades for Sorting Carcasses for Expected Tenderness and/or Palatability of Their Steaks*
- [3] D. M. Wulf, and J. W. Wise. 1999. *Measuring Muscle Color on Beef Carcasses Using the $L^*a^*b^*$ Color Space*
- [4] D. M. Wulf, S. F. O'Connor, J. D. Tatum, and G. C. Smith. *Using Objective Measures of Muscle Color to Predict Beef Longissimus Tenderness*. Department of Animal Sciences, Colorado State University, Fort Collins 80523. J. Anim. Sci. 1997.
- [5] R. C. Cannell, K. E. Belk, J. D. Tatum, J. W. Wise, P. L. Chapman, J. A. Scanga, and G. C. Smith. 2002. *Online evaluation of a commercial video image analysis system (Computer Vision System) to predict beef carcass red meat yield and for augmenting the assignment of USDA yield grades*
- [6] S. Jeyamkondan, N. Ray, G.A. Kranzler, S. Acton. *Segmentation of longissimus dorsi for Beef Quality Grading using Computer Vision*
- [7] [http : //www.rmsusa.com/htm/products/beefcam.asp](http://www.rmsusa.com/htm/products/beefcam.asp). Visitada el 23-3-07
- [8] K. E. Belk, J. A. Scanga, J. D. Tatum, J. W. Wise, and G. C. Smith. 1998. *Simulated Instrument Augmentation of USDA Yield Grade Application to Beef Carcasses*
- [9] Allen Trenkle, Chris Iiams. 1999. *Use of a Digital Camera to Collect Carcass Data from Experimental Cattle*. 1999 Beef Research Report. Iowa State University

-
- [10] J. Subbiah, N. Ray, G. A. Kranzler, S. T. Acton. *Computer Vision Segmentation of the Longissimus Dorsi for Beef Quality Grading*. Transactions of the ASAE
- [11] S. Jeyamkondan, G. A. Kranzler, A. Lakshmikanth. 2001. *Predicting Beef Tenderness with Computer Vision*. An ASAE Meeting Presentation
- [12] A.M. Wyle, R.C. Cannell, K.E. Belk, M. Goldberg, R. Riffle, and G.C. Smith. 1998. *An Evaluation of the Prototype Portable Hunterlab Video Imaging System (Beefcam) as a Tool to Predict Tenderness of Beef Carcasses Using Objective Measures of Lean and Fat Color*
- [13] D. J. Vote, K. E. Belk, J. D. Tatum, J. A. Scanga, and G. C. Smith. 2003. *Online prediction of beef tenderness using a computer vision system equipped with a BeefCam module*
- [14] Keith E. Belk, John A. Scanga, Aaron M. Wyle and Gary C. Smith. 2000. *Prediction Of Beef Palatability Using Instruments*
- [15] T. L. Wheeler, D. Vote, J. M. Leheska, S. D. Shackelford, K. E. Belk, D. M. Wulf, B. L. Gwartney, and M. Koochmaraie. 2002. *The efficacy of three objective systems for identifying beef cuts that can be guaranteed tender*
- [16] Rafael C. Gonzalez, Richard E. Woods. 2002. *Digital Image Processing Second Edition*. Prentice Hall
- [17] Pablo Arias, Alejandro Pini, Gonzalo Sanguinetti, Pablo Sprechmann. *Segmentación con Información a Priori de Forma Aplicada a Sistema de Valoración Cárnica*. Proyecto de fin de carrera, IIE-Facultad de Ingeniería. Universidad de la República
- [18] Dan S. Hale, Kyla Goodson, and Jeff W. Savell. *Beef Quality and Yield Grades*. Department of Animal Science Texas Agricultural Extension Service
- [19] *United States Standards for Grades of Carcass Beef*. United States Department of Agriculture
- [20] A tutorial on Clustering Algorithm
www.elet.polimi.it/upload/matteucc/Clustering/tutorial_html/. Visitada el 24-3-07
- [21] Hardeberg, Jon Y. *Acquisition and Reproduction of Color Images: Colorimetric and Multispectral Approches*. USA: Dissertation.com, 1999.
- [22] Balasubramanian, Raja. *Device characterization*. Chapter 5 in Digital Color Imaging Handbook, G. Sharma, Ed., CRC Press, 2003.

-
- [23] Barnard, Kobus and Funt, Brian. *Camera Characterization for Color Research*. Wiley Periodicals, 2002. ISBN: 0471486884
- [24] Green, Phil. *Colour Engineering*. Eds. Green, P y MacDonald, L. W. Nueva York: Jhon Wiley and Sons, 2002.
- [25] Cheung, Vien; Westland, Stephen; Connah, David and Ripamonti, Caterina. *A comparative study of the characterization of colour cameras by means of neural networks and polynomial transforms*
- [26] Westland, Stephen and Ripamonti, Caterina, *Characterization of Cameras*. Chapter 8 in *Computational Colour Science Using MATLAB*. John Wiley & Sons, Ltd. ISBN: 0 470 84562 7
- [27] Rizzi, Alessandro. *Color, from computation to perception and back to computation*. Transparencias del curso.
- [28] William K. Pratt. *Digital Image Processing* Third Edition. John Wiley and Sons, Inc. 2001. ISBN 0-471-37407-5.
- [29] [http : //gusgsm.com/color_faq](http://gusgsm.com/color_faq). Visitada el 23-3-07
- [30] Konica-Minolta. *Precise Color Communication*
[http : //www.konicaminoltaeurope.com/pcc/en/part3/02.html](http://www.konicaminoltaeurope.com/pcc/en/part3/02.html). Visitada el 23-3-07
- [31] Litwiller, Dave “CCD vs. CMOS: Facts and Fiction”. *Photonics Spectra*. Laurus Publishing Co. Inc. January 2001. En línea:
[http : //www.dalsa.com/shared/content/Photonics_Spectra_CCDvsCMOS_Litwiller.pdf](http://www.dalsa.com/shared/content/Photonics_Spectra_CCDvsCMOS_Litwiller.pdf). Visitada el 23-3-07
- [32] West, Perry. *A Roadmap for Building a Machine Vision System*. Automated Vision Systems, Inc.
- [33] Kennedy, John. *The Basic Geometry Behind A Camera Lens And A Magnifying Glass*. Mathematics Departmet, Santa Monica College, Santa Monica, CA.
- [34] *How Color Cameras Work*. The Imaging Source Europe GmbH. En línea:
www.theimagingsource.com/en/resources/whitepapers/download/howcolcamswp.en.pdf. Visitada el 23-3-07
- [35] *Canon Digital Camera Software Development Kit, Software Developer's Guide*. CANON INC, Ver. 7.3

-
- [36] Heikkilä, J. and Silvén, O. *A Four Step Camera Calibration Procedure with Implicit Image Correction*. Infotech Oulu and Department of Electrical Engineering. University of Oulu. Oulu, Finland.
- [37] K. Strobl, W. Sepp, S. Fuchs, C. Paredes and K. Arbter *Camera Calibration Toolbox for Matlab*. Institute of Robotics and Mechatronics. German Aerospace Center (DLR). En línea:
[http : //www.vision.caltech.edu/bouguetj/calib_doc/](http://www.vision.caltech.edu/bouguetj/calib_doc/). Visitada el 23-3-07
- [38] Strongarm Designs. Inc.
[http : //www.strongarm.com/ProductsMobileStationSpecs.htm](http://www.strongarm.com/ProductsMobileStationSpecs.htm). Visitada el 23-3-07
- [39] NORRES Industrial Hoses
[http : //www.norres.de/norres/hm/10](http://www.norres.de/norres/hm/10). Visitada el 23-3-07
- [40] V-Infinity, a division of CUI INC.
Catálogo en línea: [http : //www.v - infinity.com/](http://www.v-infinity.com/). Visitada el 23-3-07
- [41] *A threshold selection method from gray-level histogram*. N. Otsu, IEEE Trans. Sys., Man., Cyber., vol. 9, pp. 62-66, 1979.
- [42] *Effect of Marbling Degree on Beef Palatability in Bos taurus and Bos indicus Cattle*. U.S. Meat Animal Research Center, ARS, USDA (September 9, 1994).
- [43] . C. Jiménez de Aréchaga, M. I. Pravia y M. Xavier. *Caracterización de la ternera en el proceso de producción de carne vacuna en el Uruguay y su predicción utilizando las principales técnicas POS MORTEM: pH, temperatura y color*. Tesis de Grado. Montevideo: Facultad de Agronomía, Universidad de la República. 2002
- [44] Martín, Marcos.
Descriptores de Imagen. En línea: [http : //lmi.bwh.harvard.edu/papers/pdfs/2003/martin-fernandezCOURSE03e.pdf](http://lmi.bwh.harvard.edu/papers/pdfs/2003/martin-fernandezCOURSE03e.pdf). Visitada el 23-3-07

Apéndice A

Visitas

Se realizaron visitas a los frigoríficos Colonia y San Jacinto. El Frigorífico Colonia se encuentra en el kilómetro 30 de la ruta 22, cerca de la localidad de Tarariras, en el departamento de Colonia. Las visitas a Frigorífico Colonia fueron guiadas por Lautaro Pérez. El Frigorífico San Jacinto está ubicado en el kilómetro 59,5 de la ruta 7, próximo a la localidad de San Jacinto en el departamento de Canelones. Las visitas a este último fueron guiadas por el Ing. Juan Brum.

A.1. Frigorífico Colonia 24 de Mayo de 2006

Esta primera visita tuvo como objetivo tomar imágenes de costillas, para empezar a construir la base de datos de imágenes con las que se desarrollaría el proyecto. Las imágenes se tomaron con un prototipo desarrollado en el IIE, previo a la iniciación de este proyecto.

Por otro lado con esta visita se buscaba conocer el funcionamiento interno del frigorífico y en particular que tipo de mediciones se realizan en las reses. Se concluyó que las medidas a las que se les da mayor importancia son a la medida del área del Ojo de Bife y al porcentaje de grasa intramuscular o marmoreo.

El lugar donde se tomaron las imágenes es un pasillo donde circulan, colgados de un riel, los cuartos de reses que salen de la cámara frigorífica. Se tomaron 122 imágenes de la 3a costilla en esta visita.

A.2. Frigorífico San Jacinto 8 de Diciembre de 2006

Esta visita estuvo a cargo del ingeniero de planta del Frigorífico San Jacinto, Juan Brum. El objetivo era probar el primer prototipo que se tenía del equipo, en una etapa que se considera bastante avanzada y con la mayoría de las funcionalidades implementadas. El hardware que se llevó en esta visita fue la primera versión del

equipo de toma de imágenes. Lamentablemente en este frigorífico no se realizaba el corte pistola 3 costillas, sin embargo, los operarios, muy generosamente se ofrecieron a realizar especialmente 100 cortes pistola 4 costillas para que se pudiera probar el prototipo.

Sin embargo, los resultados de ésta prueba fueron pésimos, aunque no dejó de ser una visita positiva, ya que se pudo detectar donde fallaba el sistema y se encontraron soluciones a los problemas hallados. Entre los problemas que se detectaron se destacan los siguientes:

- Exceso de brillos en las imágenes.
- Fotos fuera de foco.
- Falla del algoritmo FCM.

A partir de esta visita se cambió totalmente el hardware, realizando un segundo equipo que utiliza iluminación indirecta a diferencia del primer equipo que utilizaba iluminación directa. Se ajustaron los parámetros de la cámara para que enfocara correctamente al centro de la imagen. Con respecto al algoritmo FCM, se corrigió de manera que no se incluyeran los pixeles de valor 0 en vector de entrada del algoritmo, como se explica en la sección correspondiente a este algoritmo.

A.3. Frigorífico Colonia 7 de Febrero de 2007

Una vez que se corrigieron los defectos encontrados en la visita realizada al Frigorífico San Jacinto, se realizó una segunda visita al Frigorífico Colonia. En esta ocasión se llevó el segundo equipo de toma de imágenes, que utiliza iluminación indirecta. También se llevó el primer equipo, para comparar ambos diseños de hardware. Se tomaron imágenes tanto en el pasillo como en la propia cámara frigorífica, puesto que las reses de “feedlot”, las que son más grasas y en las que tiene más sentido medir el marmoreo, no iban a ser procesadas ese día y se debió proceder a fotografiarlas en la misma cámara. Se obtuvieron 47 imágenes, que juzgamos muy satisfactorias, de las reses de “feedlot”, tomadas con el segundo equipo de captura de imágenes. En el pasillo se probaron ambos equipos, con reses de animales que eran muy magros, en los que el programa no siempre realizaba una correcta segmentación, ya que apenas había grasa en estas costillas y por lo tanto la frontera entre el músculo *Longissimus Dorsi* y el resto de los trozos de carne estaba indefinida en gran parte de la costilla. Como se esperaba, el segundo equipo funcionó mucho mejor que el primero y por lo tanto se decidió usarlo como definitivo. A partir de esta visita se conocieron los últimos ajustes que debían hacerse tanto en hardware como en software, para la culminación del proyecto.

Apéndice B

Contenido del CD Adjunto

El CD adjunto es parte del proyecto y contiene:

- Imágenes con que se trabajaron (carpeta **imagenes**)
- Marcas hechas por el programa (dentro de la carpeta **imagenes** y en la carpeta **resultados C++**)
- Marcas hechas por versiones anteriores del algoritmo (en la carpeta **resultados viejos Matlab**)
- Marcas hechas manualmente (dentro de la carpeta **imagenes**)
- Algoritmos desarrollados en Matlab (dentro de la carpeta **scripts Matlab**, se encuentran las carpetas **Segmentacion**, **Calibracion Color** y **Calibracion Geometrica**)
- Programas desarrollados en C++ (carpeta **scripts C++**)
- Planillas de Resultados de validación (carpeta **validacion**)
- Papers que se estudiaron para el proyecto (carpeta **papers**)
- Instalador del programa (carpeta **Instalador Ojo de Bife**)
- Drivers de la cámara (carpeta **Camera Drivers**)
- Documentación (documento **ProyectoOjodeBife.pdf**)

Apéndice C

Manual de Usuario del Programa Ojo de Bife

Instalación

Para instalar el programa se debe ejecutar el archivo Setup.exe que se encuentra dentro del CD adjunto en la carpeta **Instalador Ojo de Bife**.

IMPORTANTE: Se deben instalar los drivers de la cámara para que funcione el programa.

Los drivers de la cámara Canon Powershot G1 que se utilizó en el proyecto se encuentran en la carpeta **Camera Drivers** del CD adjunto.

Abrir el Programa

Para abrir el programa, se debe ejecutar el archivo **Ojodebifeproto.exe** que se encuentra en la carpeta de instalación del programa. Durante la instalación se crean accesos directos en el Menú Inicio de Windows y en el Escritorio. En la siguiente figura se ve una captura del programa.

APÉNDICE C. MANUAL DE USUARIO DEL PROGRAMA OJO DE BIFE

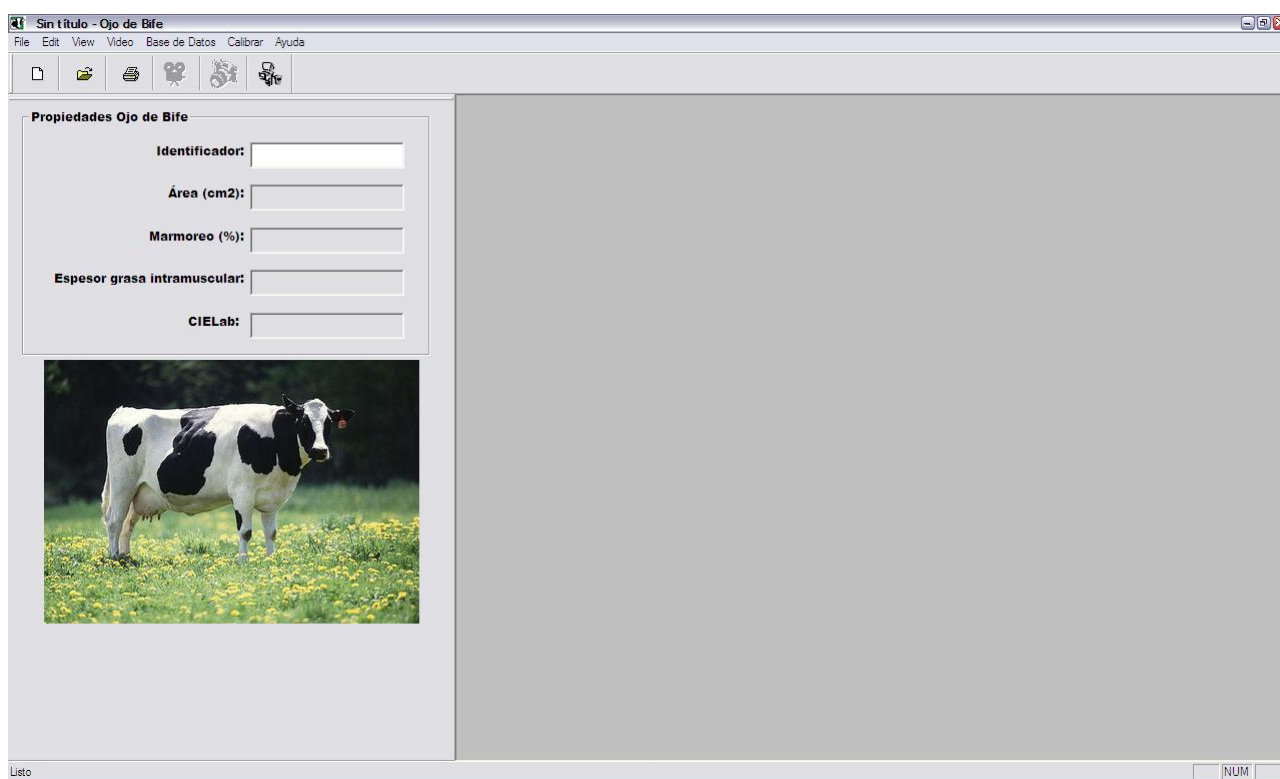


Figura C.1: Captura del programa Ojo de Bife

Funciones del Programa

El programa es muy sencillo de usar y sus funciones son básicamente las siguientes:

- Conectar la cámara.
- Calibración del color.
- Tomar una imagen y procesarla.
- Abrir una imagen y procesarla.
- Acceder a la base de datos.
- Realizar búsquedas en la base de datos.

La interfaz de usuario cuenta con 6 botones de acceso directo que son los siguientes:



Figura C.2: Botones del programa

1. Nuevo: limpia la pantalla.
2. Abrir: permite abrir una imagen del disco duro y procesarla
3. Imprimir: permite imprime la imagen.
4. Capturar Video: permite capturar video, no es posible en esta versión del programa.
5. Capturar y Procesar: captura una imagen y la procesa.
6. Consultar la Base de Datos: accede a la base de datos.

Conectar la Cámara



Figura C.3: Conectar la Cámara

Una vez que la cámara está conectada al puerto USB del ordenador, el programa debe conectarse a ella para poder darle instrucciones y descargar de ésta las imágenes. Para conectar la cámara, se debe seleccionar, en el menú **Video**, **Conectar Cámara**. Posteriormente, como muestra la figura C.3 se selecciona la cámara y se da **OK**.

Calibración del Color

Para la calibración del color del programa, se debe seleccionar dentro del menú **Calibrar**, **Calibrar Color**, como se muestra en la figura C.4. El programa pedirá que se sitúe la tabla de calibración de color en posición de tomar la fotografía y una vez que se encuentre en posición se debe dar **OK**. En caso de que la tabla haya quedado mal posicionada, el programa dará un mensaje de error y se pedirá que se efectúe de nuevo la calibración.



Figura C.4: Calibrar color

Capturar una Fotografía

Previo a la captura, se debe anotar en el campo **Identificador**, el identificador de la res que se va a capturar. Si este campo se encontrara vacío el programa no permitirá la captura.

Para tomar una fotografía, hay tres posibilidades.

- Presionando el botón en el equipo de toma de imágenes.
- Efectuando un clic en el botón de capturar imagen (botón 5).
- Seleccionando dentro del menú **Video**, la opción de **Capturar y Procesar**.

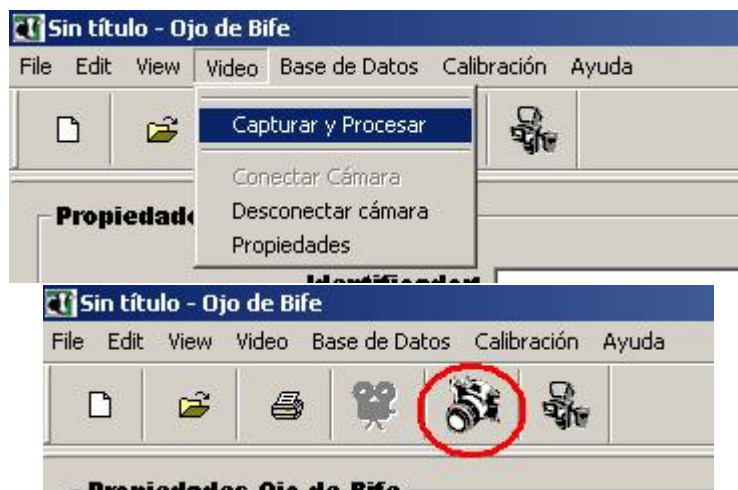


Figura C.5: Capturar y procesar una imagen

Propiedades de Adquisición

Para establecer las propiedades de la adquisición, se debe seleccionar bajo el menú Video, la opción Propiedades. En esta versión del programa solo se permite elegir el directorio donde se guardarán las imágenes.



Figura C.6: Propiedades de Adquisición

Resultados del Procesamiento

Una vez que se captura una imagen, se mostrará en pantalla el resultado del procesamiento. A la izquierda de la pantalla se mostrarán los campos Área, Marmoreo, Espesor de grasa subcutánea y CIELab.



The image shows a software window titled "Propiedades Ojo de Bife". It contains five rows of data, each with a label on the left and a value in a text box on the right:

Propiedad	Valor
Identificador:	eq1_cam_WB011origin
Área (cm2):	93.00
Marmoreo (%):	1.792
Espesor grasa subcutánea	0.73 ; 1.88
CIELab:	-8.45 ; 28.42 ; 42.7632

Figura C.7: Resultados del procesamiento

Base de Datos

Cada vez que se toma una imagen, el resultado del procesamiento se guarda en la base de datos, junto con los valores de las medidas. Se puede acceder a la base de datos para ver los resultados de imágenes anteriores, con los valores de todas las medidas.

Para acceder a la base de datos, hay dos posibilidades, una es con el botón de acceder a la base de datos (botón 6) y la otra es a través del menú **Base de Datos**, eligiendo la opción **Realizar Consulta**. Esto se muestra en la figura C.8.



Figura C.8: Consultar la base de datos

Apéndice D

Planificación y Dedicación

Todo proyecto requiere de una planificación inicial, de forma de fijar plazos y establecer los diferentes objetivos a desarrollar. Si al iniciar el trabajo se desconocen las tareas a desarrollar, existirá una discrepancia natural entre lo planificado y lo realmente llevado a cabo.

El plan del proyecto se realizó el 27-10-06, se fijó como fecha de finalización del mismo el día 15-12-06, con una carga horaria de 2112 horas por integrante, dando un gran total de 3960 horas. A continuación se muestra una comparación entre la planificación y la ejecución del proyecto.

APÉNDICE D. PLANIFICACIÓN Y DEDICACIÓN

Hito	Tarea	Finalización	Horas hombre
a	Primer algoritmo de Segmentación (Matlab)	15/12/05	300
	Equipo de captura de imágenes conectado al Pc		350
	Indicadores de Performance del algoritmo de segmentación		200
b	Propuesta del conjunto Hardware-software	31/3/06	300
	Propuesta de iluminación del equipo de captura		250
	Protecciones del sistema contra el ambiente del frigorífico		80
	Propuesta de manipulación del sistema		40
	Sistema operativo, bibliotecas a utilizar, lenguaje de desarrollo		60
c	Primer prototipo, que integre captura y procesamiento de imágenes	30/6/06	350
	Medidas de área, espesor de grasa, marmoreo y colorimétricas avanzadas		380
	Migración del software de desarrollo (Matlab) al lenguaje escogido		220
	Interfaz gráfica amigable desarrollada		140
d	Prototipo pronto para probar en planta	30/9/06	250
	El sistema deberá realizar todas las medidas		400
	Interfaz de usuario concluída		50
e	Prueba en planta realizada	15/12/06	40
	Resultados analizados		50
	Documentación final		500

Figura D.1: Plan Inicial del Proyecto

APÉNDICE D. PLANIFICACIÓN Y DEDICACIÓN

Hito	Tarea	Finalización	Horas hombre
a	Primer algoritmo de Segmentación (Matlab)	15/12/05	300
	Equipo de captura de imágenes conectado al Pc	01/7/06	300
	Indicadores de Performance del algoritmo de segmentación	15/12/05	300
b	Propuesta del conjunto Hardware-software	01/6/06	250
	Propuesta de iluminación del equipo de captura 01/02/07		180
	Protecciones del sistema contra el ambiente del frigorífico	31/1/06	24
	Propuesta de manipulación del sistema	31/1/07	40
	Sistema operativo, bibliotecas a utilizar, lenguaje de desarrollo	31/12/05	50
c	Primer prototipo, que integre captura y procesamiento de imágenes	30/6/06	370
	Medidas de área, espesor de grasa, marmoreo y colorimétricas avanzadas	30/10/06	300
	Migración del software de desarrollo (Matlab) al lenguaje escogido	28/2/07	300
	Interfaz gráfica amigable desarrollada	31/7/06	240
d	Prototipo pronto para probar en planta	30/9/06	250
	El sistema deberá realizar todas las medidas	31/1/07	350
	Interfaz de usuario concluída	31/7/06	30
e	Prueba en planta realizada	06/2/07	100
	Resultados analizados	28/2/07	200
	Documentación final	16/3/07	600

Figura D.2: Implementación real del Proyecto

D.1. Evaluación de la planificación

En líneas generales el Proyecto se retrasó en el entorno de los 3 meses. Los primeros corrimientos se debieron a la elección del equipo de adquisición de imágenes, lo cual trajo consigo el consecuente retraso del armado del prototipo.

Debido a un deficiente desempeño de la iluminación del primer equipo de captura de imágenes, el cual no fue lo debidamente probado previo a la realización de la primera prueba en planta, se debió rediseñar el sistema de iluminación, y realizar una prueba en planta extra, acarreado consigo retrasos no previstos.

Se dedicó mucho más tiempo del previamente estipulado a mejorar y depurar el algoritmo de segmentación, tarea que se realizó prácticamente durante toda la duración del proyecto

En lo que a las medidas colorimétricas refiere, se encontraron dificultades a la hora de obtener los elementos necesarios para la calibración del sistema.

Por último, cabe resaltar que el presente Proyecto fué financiado en su totalidad por sus integrantes.