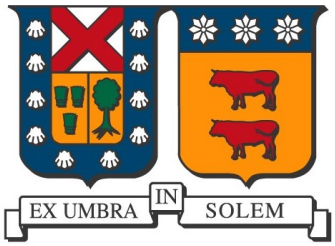
2018

SEGMENTACIÓN AUTOMÁTICA DE LA GLOTIS EN VIDEOS ENDOSCÓPICOS DE ALTA VELOCIDAD UTILIZANDO COLORES Y FORMAS CARACTERÍSTICAS DE LAS REGIONES GLOTALES

SALAZAR CERDA, LUCAS FIDEL DE JESUS

UNIVERSIDAD TÉCNICA FEDERICO SANTA MARÍA
DEPARTAMENTO DE ELECTRÓNICA
VALPARAÍSO - CHILE



SEGMENTACIÓN AUTOMÁTICA DE LA GLOTIS EN VIDEOS ENDOSCÓPICOS DE ALTA VELOCIDAD UTILIZANDO COLORES Y FORMAS CARACTERíSTICAS DE LAS REGIONES GLOTALES

LUCAS FIDEL DE JESÚS SALAZAR CERDA

MEMORIA DE TITULACIÓN PARA OPTAR AL TÍTULO DE INGENIERO CIVIL ELECTRÓNICO MENCIÓN COMPUTADORES

PROFESOR GUÍA: MATÍAS ZAÑARTU

AGOSTO - 2018

Resumen

Keywords: Laringoscopía, Glotis, Videos de alta velocidad, Patologías vocales, Segmentación de imágenes, Descriptores de Fourier, Canales de color

Para analizar y diagnosticar enfermedades y disfunciones vocales es indispensable analizar visualmente las cuerdas vocales en acción. El uso de cámaras de alta velocidad es la mejor forma de capturar el ciclo de fonación de las cuerdas vocales en gran detalle, pero debido a la inmensa cantidad de datos generado por estas cámaras se vuelve necesario desarrollar técnicas automáticas para procesar estos videos de alta velocidad (HSV), en particular técnicas de segmentación automática de la glotis.

Se plantean como objetivos para esta memoria implementar un método de segmentación automática de la glotis en HSV, evaluar cuantitativamente dicho método y compararlo con otros métodos existentes, investigar la utilidad del uso de la información de color de los videos en la segmentación, e investigar la utilidad del uso de técnicas de machine learning en la segmentación.

En esta memoria se implementa el paper [14] que describe un método automático de segmentación de la glotis en HSV. Este método se basa en la aplicación de un umbral flexible, comparación de descriptores de Fourier, aplicación de contornos activos, machine learning y cálculo de una imagen de probabilidad a partir de las propiedades de color de las glotis segmentadas, entre otras cosas. Se proponen varias mejoras: Cambios en la comparación de descriptores de Fourier, en la comparación de las propiedades de color, resolución de colisiones en la segmentación y el cálculo de una ROI inicial a partir de la varianza de cada pixel a lo largo del video. También se desarrolló una versión para videos en escala de grises, y se evaluó cuantitativamente el algoritmo utilizando el coeficiente Dice y el error de área.

Los resultados muestran que el algoritmo original no entrega muy buenos resultados, pero al implementar las modificaciones propuestas se logran mejoras significativas. Se concluye que la información de color de los videos no debiera descartarse ya que puede ayudar a la segmentación, y que con la cantidad de datos de entrenamiento limitada que se tiene, la parte de machine learning del algoritmo no funciona lo suficientemente bien.

Abstract

Keywords: Laryngoscopy, Glottis, High speed video, Vocal pathologies, Image segmentation, Fourier Descriptors, Color channels

Visual analysis of the vocal cords in action is essential for the diagnosis of vocal pathologies. The use of high speed cameras is the best way to capture the vocal cords' phonation cycle in detail, but due to the huge amount of data generated by these cameras it becomes necessary to develop automatic processing techniques for the captured videos, particularly automatic glottis segmentation techniques.

The objectives set for this thesis are to implement an automatic glottis segmentation method for laryngeal high-speed videos (HSV), to quantitatively evaluate the method's performance and compare it with other existent methods, to investigate the usefulness of the videos' color information, and to investigate the usefulness of machine learning techniques in glottis segmentation.

The paper [14] is chosen for implementation; this paper describes an automatic glottis segmentation method in HSV based on a flexible thresholding technique, Fourier descriptors comparison, active contours, machine learning and a probability image calculation based on color properties of already segmented glottis, amongst other things. Many modifications are proposed: Changes in Fourier descriptor comparison, changes in color properties comparison, collision resolution during frame-by-frame segmentation and an initial ROI calculation from the video's pixels variance. A version of the algorithm for grayscale videos was developed, and a quantitative evaluation of the algorithm's performance was made using the Dice coefficient and area error.

Results show that the original algorithm does not give very good results, but the implementation of the proposed modifications significantly improves performance. It is concluded that the video's color information should not be discarded because it can be helpful for glottis segmentation, and that with the limited amount of data available, the machine learning part of the algorithm does not work well enough.

Glosario

- Cuerdas vocales: Consisten en dos membranas mucosas ubicadas en la laringe. Durante la fonación las cuerdas vibran modulando el aire que pasa a través de ellas y produciendo sonido.
- Efecto Lombard: Es la tendencia involuntaria de los hablantes a aumentar el esfuerzo vocal al hablar en un ambiente ruidoso.
- Estroboscopía: Es una técnica para la visualización de las cuerdas vocales en movimiento basada en el efecto estroboscópico, aprovechando la periodicidad de la vibración de las cuerdas.
- Escala de grises: Una imagen en escala de grises es una en donde cada pixel tiene un solo valor que indica la intensidad de la luz sobre él. No hay información de color.
- GIE: *Glottal Image Explorer*, herramienta open-source para la segmentación de la glotis en videos endoscópicos de alta velocidad de las cuerdas vocales, propuesta en [3].
- Glotis: La apertura entre las cuerdas vocales.
- GND: *Glottal Neighborhood Descriptor*, descrito en la sección 1.3.2 del desarrollo del tema.
- HSV: *High Speed Video*, video de alta velocidad.
- Laringe: Es un órgano ubicado en la parte superior del cuello asociado principalmente a la producción de la voz. Contiene las cuerdas vocales.
- Laringoscopía: Es una endoscopía de la laringe, es decir un procedimiento médico para observar la laringe.
- PCA: *Principal Component Analysis*, técnica estadística que transforma un conjunto de variables posiblemente correlacionadas en otro conjunto de variables no correlacionadas llamadas *componentes principales*.
- ROI: *Region Of Interest*, región de interés.

- Segmentación: En el contexto de imágenes segmentación se refiere a particionar la imagen en segmentos o regiones, generalmente con el objetivo de identificar objetos o fronteras en la imagen.

Índice general

Parte I

Motivación y Objetivos

La laringe, que aloja a las cuerdas vocales y es el órgano responsable de la producción de la voz, es sumamente delicada y vulnerable a ser afectada por una gran variedad de enfermedades y disfunciones. Para poder diagnosticar y posteriormente tratar estas enfermedades es indispensable analizar visualmente las cuerdas vocales en acción y sus características vibratorias. Esto se puede hacer de varias formas, pero la mejor y más confiable es el uso de cámaras de alta velocidad. Estas cámaras son capaces de grabar videos a tasas del orden de decenas de miles de cuadros por segundo, más que suficiente para capturar en gran detalle la vibración de las cuerdas que no es visible ante el ojo humano debido a su alta frecuencia.

Un problema que surge del uso de estas cámaras es la inmensa cantidad de datos generada. A una tasa normal de 10000 cuadros por segundo, un video de unos pocos segundos de duración contendrá varias decenas de miles de cuadros. Analizar esta cantidad de datos de forma manual es inviable, por lo que se vuelve necesario desarrollar técnicas automáticas para procesar estos videos de alta velocidad (HSV), en particular técnicas de segmentación automática de la glotis.

Se han propuesto muchos métodos distintos para la segmentación de la glotis en HSV, pero el tema aún no está completamente resuelto y sigue siendo investigado. No existe un benchmark oficial para evaluar objetivamente los resultados de los métodos, por lo que es difícil saber si una técnica es mejor que otra sin implementarla directamente. También llama la atención que la mayoría de los métodos propuestos hasta

ahora trabajan con videos en escala de grises, siendo que para los clínicos el color del video es un factor importante para la identificación de lesiones en el tejido de las cuerdas vocales. Y además también llama la atención el poco uso de técnicas de machine learning ya que estas técnicas son muy usadas para el reconocimiento y clasificación de objetos en imágenes, y pueden ser de ayuda para la segmentación.

En base a todo lo anterior, los objetivos que se plantean para esta memoria son:

1) Implementar un método automático existente para segmentar la glotis en videos endoscópicos de alta velocidad.

2) Evaluar cuantitativamente el método y comparar su rendimiento con otros métodos existentes.

3) Investigar la utilidad del uso del color del video en la segmentación de la glotis.

4) Investigar la utilidad del uso de técnicas de Machine Learning en la segmentación de la glotis.

La motivación de los objetivos 2, 3 y 4 es ayudar a mejorar las actuales técnicas de segmentación; el objetivo 2 en particular busca contribuir a hacer una evaluación más objetiva de los algoritmos, mientras que los objetivos 3 y 4 buscan confirmar la utilidad de ciertas técnicas para la segmentación. El objetivo 1 surge a partir de los otros 3, ya que para completarlos claramente es necesario implementar un algoritmo de segmentación primero.

Para completar el objetivo 1 se implementará el paper [14]: *"Fully automated glottis segmentation in endoscopic videos using local color and shape features of glottal regions"*, brevemente descrito en la sección 1.2. Una descripción más detallada se hará en la sección III de Desarrollo del tema. Este paper en particular fue elegido porque es el único paper que se encontró que trabaja con el color del video y además aplica técnicas que podrían clasificarse como machine learning.

Para completar el objetivo 2 también será necesario implementar otra técnica de segmentación; el paper [3]: *"GlottalImageExplorer–An open source tool for glottis segmentation in endoscopic high-speed videos of the vocal folds"* descrito brevemente en

1.2 fue elegido para este propósito. Se eligió porque los investigadores que escribieron el paper dejaron públicamente disponible una aplicación para Windows que se puede usar para segmentar la glotis en videos laringoscópicos utilizando su algoritmo y permite exportar los contornos calculados. Por lo tanto no es necesario hacer la implementación desde cero y además no queda duda de que el algoritmo se está utilizando tal cual lo pensaron sus creadores. El detalle de cómo se evaluarán los algoritmos, y de cómo se completarán lo objetivos 3 y 4 se explicará en la siguiente sección.

Parte II

Introducción y Estado del Arte

La voz humana se origina dentro de la laringe, cuando las cuerdas vocales vibran a causa del flujo de aire pulmonar a través de ellas. Dicha vibración consiste en ciclos de apertura y cierre de la glotis, que se define como el espacio entre las cuerdas vocales. En la voz hablada la glotis completa en promedio 125 y 200 ciclos de apertura y cierre en hombres y mujeres respectivamente, y puede completar más todavía al cantar o hacer otro tipo de vocalizaciones, alcanzando frecuencias mucho más altas de las que el ojo humano es capaz de captar.

La laringe es un órgano sumamente delicado que requiere de constante cuidado y supervisión, ya que puede ser afectada por una gran variedad de enfermedades y disfunciones que pueden traer serias consecuencias físicas y emocionales. Dichas enfermedades generalmente se ven reflejadas en alteraciones o cambios en el ciclo normal de oscilación de las cuerdas vocales. Por lo tanto el análisis y observación de las cuerdas vocales en acción es sumamente importante para que los clínicos puedan identificar y posteriormente tratar estas enfermedades.

Pero poder observar el ciclo de vibración de la glotis en detalle no es una tarea simple debido a las altas frecuencias a las que ésta vibra, por lo que para hacerlo se requiere el uso de técnicas especiales. La técnica más común es la *estroboscopía*, una técnica que permite obtener una vista en ”cámara lenta” de la vibración de las cuerdas vocales. La estroboscopía generalmente se hace con un endoscopio rígido que se inserta a través de la boca del paciente y llega hasta la faringe. Desde allí se observa

la laringe, sin necesidad de entrar a la garganta. Además se necesita de un estetoscopio ubicado en el cuello del paciente que mida la frecuencia fundamental de vibración de las cuerdas vocales. Luego se configura una luz estroboscópica que parpadea a una frecuencia varias veces más lenta y ligeramente de-sincronizada con la frecuencia de fonación. Esto permite capturar imágenes secuenciales de distintas partes del ciclo de oscilación que al combinarse forman un video en ”cámara lenta” de la vibración de las cuerdas vocales. En la figura 1.1 se ilustra el principio detrás de esta técnica.

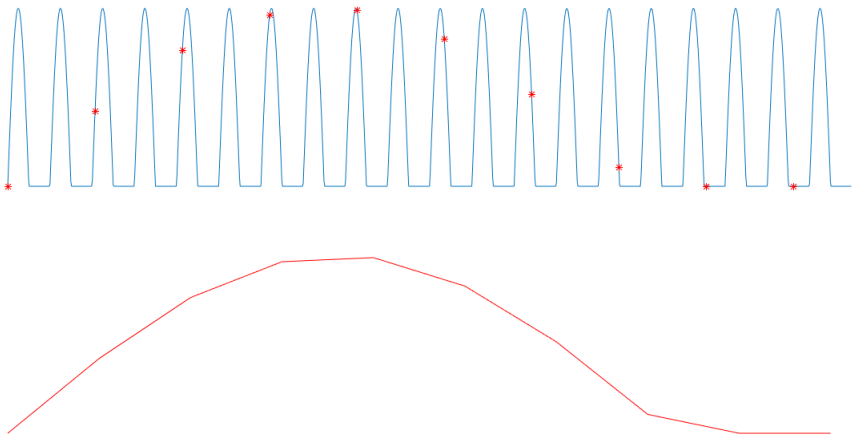


Figura 1.1: Ilustración del principio tras la estroboscopía. La señal azul de arriba representa la apertura de una glotis en el tiempo. Los puntos rojos representan los instantes en que se toman imágenes como muestras, a una tasa de muestreo mucho menor a la frecuencia fundamental de vibración de la glotis. La señal roja de abajo representa una versión en ”cámara lenta” de la vibración reconstruida a partir de las muestras tomadas.

La estroboscopía tiene varias limitaciones. Una es que requiere que la frecuencia de fonación sea estable para que la luz estroboscópica se sincronice. Por esta razón sólo será posible grabar vídeos de pacientes haciendo fonación a una frecuencia constante, y no es posible evaluar la voz hablada o comportamientos transientes de las cuerdas. Pacientes con disfunciones severas pueden tener fonaciones aperiódicas o con rápidos cambios de frecuencia, por lo que en esos casos la técnica no es aplicable. Además se requieren de varios segundos para que la luz se sincronice, y si el paciente no puede mantener una fonación constante durante ese tiempo tampoco será posible aplicar la técnica. Otra limitación está asociada al uso del laringoscopio rígido; puede ocurrir que

no se logre obtener una vista adecuada de la laringe debido a la anatomía del paciente o porque éste no tolere la examinación debido al reflejo faríngeo. En esos casos se puede utilizar un laringoscopio flexible que se inserta por la nariz, previamente anestesiada, y de ahí hasta la laringe, pero la calidad de video resultante es generalmente peor ya que la forma flexible del laringoscopio distorsiona la imagen.

Sumado a todo lo anterior está el tema de que el análisis de las imágenes grabadas mediante estroboscopía generalmente es de tipo cualitativo y depende del criterio del que lo analiza. Ya que el video no captura el ciclo de vibración entero y hay información que se pierde, es difícil hacer mediciones objetivas y cuantitativas de la vibración. Por lo tanto el análisis es subjetivo y no muy confiable.

Otra técnica más moderna es el uso de cámaras de alta velocidad para capturar completamente el ciclo de fonación de las cuerdas vocales. Hoy en día existen cámaras capaces de grabar a frecuencias de hasta 1,000,000 de cuadros por segundo, más que suficiente para capturar el ciclo de fonación entero en gran detalle. Gracias a esto los videos de alta velocidad solucionan varios de los problemas que presenta la estroboscopía. No es necesario que la fonación sea periódica y por lo tanto es posible capturar comportamientos aperiódicos y grabar a pacientes que no pueden mantener una fonación estable. Además la mayor precisión en la captura del ciclo de fonación permite hacer análisis objetivos y cuantitativos ya que todas las características de la vibración son potencialmente medibles, como el área o borde exacto de la glotis en todo momento. Sin embargo, sigue siendo posible que hayan dificultades para obtener vistas adecuadas de la glotis o que el paciente no tolere la examinación con un laringoscopio rígido. El laringoscopio flexible también sigue teniendo las mismas limitaciones.

Si bien es posible hacer mediciones objetivas a partir de videos de alta velocidad, surge otro problema al hacerlo: Las cámaras de alta velocidad típicamente graban a tasas de varios miles de cuadros por segundo, lo cual al grabar un video de una duración de unos pocos segundos se traduce en varios miles de imágenes capturadas. Es decir la cantidad de imágenes que se requiere analizar es inmensa, y es inviable hacerlo de forma manual. Se hace necesario el desarrollo de técnicas y algoritmos que puedan hacer estos análisis y mediciones de forma automática.

El tema central de esta memoria está relacionado con la medición automática de parámetros o características de la vibración de las cuerdas vocales, en particular la seg-

mentación automática de la glotis en videos de alta velocidad (HSV). A continuación se hará una revisión del estado del arte en este tema.

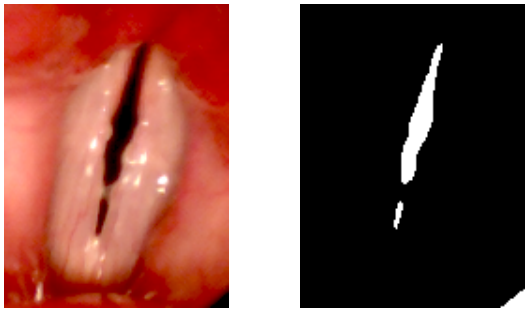
1.1. Técnicas comunes de segmentación de la glotis

La segmentación de imágenes digitales consiste en particionar la imagen en múltiples segmentos de acuerdo a algún criterio. Cada segmento corresponde a un conjunto de pixeles en la imagen. En este caso se desea separar los pixeles de la imagen en dos segmentos: aquellos que son parte de la glotis, y aquellos que no. A continuación se presentan algunas de las técnicas más comúnmente usadas para este propósito.

1.1.1. Thresholding

Thresholding, o método del valor umbral, es la técnica de segmentación más simple. En su forma más básica, cada pixel cuya intensidad iguale o supere el valor de un umbral T quedará dentro de la región segmentada, mientras que los pixeles cuya intensidad sea menor quedarán fuera. La elección del valor T del umbral se puede hacer en base a muchos criterios, unos más complejos que otros. Por ejemplo se podría obtener un umbral a partir de la intensidad promedio de la imagen, o de un análisis del histograma de intensidades de la imagen. Con imágenes a color también es posible elegir umbrales distintos para cada canal.

Esta técnica puede ser útil para la segmentación de la glotis ya que esta última generalmente es más oscura que el resto de la imagen (es decir los pixeles que la componen tienen una intensidad de gris menor que el resto de la imagen) y por lo tanto es posible filtrar algunos pixeles pertenecientes a la glotis aplicando un umbral. La figura 1.2 es un ejemplo de esto; note que se logra delinear bastante bien la glotis en este ejemplo. Pero esta técnica no es muy robusta; puede que la glotis no tenga una intensidad de gris tan uniforme como en la figura y no sea posible capturar a todos los pixeles glotales sin sobre-segmentar regiones adicionales. De hecho note que en la figura 1.2 también se segmentó una pequeño grupo de pixeles en la esquina inferior derecha; de alguna forma habría que eliminar esa segmentación errónea. También está el problema de cómo



(a) Ejemplo de una imagen glotal de un HSV

(b) Imagen segmentada con umbral

Figura 1.2: Ejemplo de aplicación de un umbral de intensidad de gris igual a 30. En la imagen (b), los pixeles cuya intensidad es menor o igual a 30 tienen color blanco, y los que tienen una intensidad mayor son negros

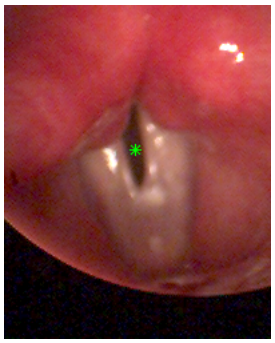
elegir el valor del umbral. Por estas razones, esta técnica casi nunca se aplica por sí sola sino acompañada o como complemento de otra técnica más elaborada.

1.1.2. Crecimiento de regiones

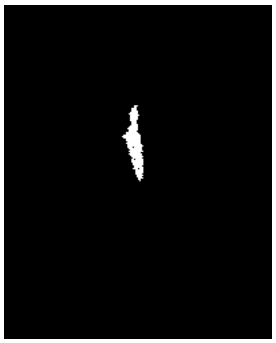
En estos métodos se selecciona un pequeño conjunto de pixeles iniciales llamados "*puntos semilla*", y luego se determina si los puntos adyacentes deben ser añadidos a la región o no según algún criterio. Criterios simples pueden ser la intensidad de los pixeles o la similitud de color. También generalmente se debe decidir cuándo unir dos regiones, para lo cual se pueden utilizar criterios más complejos como la textura o momentos espaciales, por ejemplo. Se pueden desarrollar muchas variantes del método dependiendo de qué criterios de expansión y unión se utilicen. La elección de los puntos iniciales también se puede hacer de muchas formas y afectará el desempeño general del algoritmo.

Para la segmentación de la glotis una posibilidad es utilizar como criterio de crecimiento la intensidad de los pixeles, aprovechando que los pixeles de la glotis son típicamente más oscuros que los de regiones circundantes. En la figura 1.3 se muestra un ejemplo. Pero esto asume que se conocen los puntos iniciales, y obtenerlos generalmente no es una tarea trivial. Incluso en el ejemplo, sabiendo los puntos iniciales, el

resultado no es tan ”bonito”, por lo que para obtener mejores resultados probablemente haya que elegir un criterio de selección más elaborado.



(a) Ejemplo de una imagen glotal de un HSV



(b) Imagen segmentada a través de crecimiento de regiones

Figura 1.3: Ejemplo de aplicación de un método de crecimiento de regiones. En la imagen (a) el punto semilla está marcado con verde. El criterio para añadir puntos a la región es que la diferencia entre las intensidades de gris de los pixeles adyacentes sea menor a 4

1.1.3. Watershed Transform

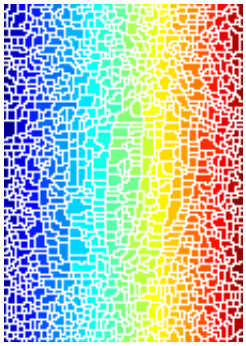
Los métodos basados en la *Watershed Transform*, o transformación divisoria, generalmente trabajan con la imagen en escala de grises. Se interpreta a la imagen como un mapa topográfico, con la intensidad de cada pixel representando su altura. Se consideran tres tipos de puntos: (a) puntos que son mínimos locales, (b) puntos en los cuales si se dejara caer una gota de agua, llegaría hasta uno de los mínimos locales, y (c) puntos donde si se dejara caer una gota habrían probabilidades iguales de que llegue hasta más dos o más mínimos. Para cada mínimo local, los puntos (b) asociados son la ”cuenca hidrográfica” de ese mínimo y representarán una región individual en la segmentación final. Los puntos (c) corresponden a las divisorias entre las cuencas y posteriormente serán los límites entre las regiones segmentadas [15]. Es más común que se trabaje sobre el gradiente de la imagen que sobre la imagen en sí, ya que de esa forma se obtienen más fácilmente los bordes de los objetos de interés.

La aplicación directa de este método típicamente lleva a una sobre-segmentación de

la imagen como se observa en la figura 1.4, particularmente si ésta es ruidosa. Para solucionar esto es necesario hacer algo más; algunas soluciones posibles son pre-procesar la imagen, definir algún criterio para unir regiones luego de la segmentación, o limitar el número de regiones en el resultado final.

(a) Ejemplo de una imagen glotal de un HSV en escala de grises

(b) Gradiente de la imagen



(c) Resultado de la transformada watershed

Figura 1.4: Ejemplo de cálculo directo de la transformada watershed sobre el gradiente. El resultado es una sobre-segmentación extrema.

Para ejemplificar, se intentará eliminar el ruido de la imagen para obtener una versión más ”plana” de la misma, con menos divisorias. En la figura 1.5 (a) se muestra el resultado de aplicar una apertura y cierre por reconstrucción. La imagen resultante es efectivamente más ”plana” con menos mínimos locales los cuales se pueden encontrar fácilmente como se muestra en la figura 1.5 (b). En la figura (c) se aplica un umbral con el objetivo de identificar el fondo de la imagen, el cual se usará después para separar los objetos que se segmentarán. La figura (d) son las divisorias entre los objetos detectados en la figura (c).

(a) Apertura y cierre por reconstucción

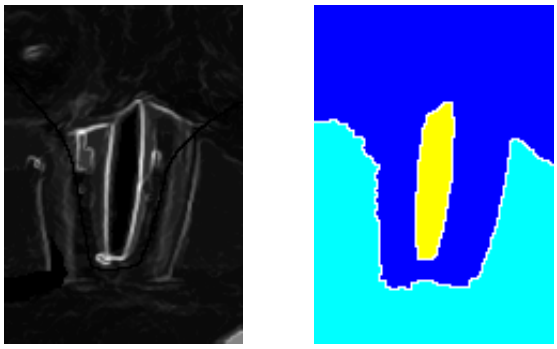
(b) Mínimos regionales, luego de cierre y erosión para eliminar ruido

(c) Aplicación de umbral a la imagen (a)

(d) Divisorias de la imagen (c)

Figura 1.5: Pre-procesamiento para reducir la sobre-segmentación

Finalmente se utilizan los mínimos regionales y las divisorias para limitar los mínimos regionales del gradiente sólo a esas zonas. En la figura 1.6 (a) los mínimos regionales sólo están ubicados en las regiones marcadas por las figura 1.5 (b) y (d). Note que esas regiones se oscurecieron en la figura 1.6 (a). La figura 1.6 (b) muestra el resultado de aplicar la transformada watershed.



(a) Gradiente con mínimos regionales restringidos

(b) Resultado de aplicar la transformada watershed a imagen (a)

Figura 1.6: Resultado final de aplicar la transformada watershed a la imagen procesada. Ahora el resultado es mucho más aceptable

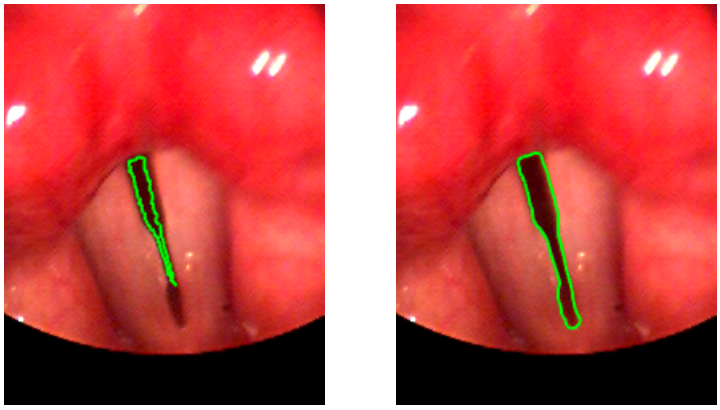
En imágenes de las cuerdas vocales, al aplicar este método la glotis típicamente quedará como una cuenca hidrográfica por sí sola como en la figura anterior. Sin embargo,

este método en general no podrá identificar la glotis por sí solo ya que la imagen queda dividida en más de dos regiones. Se debe aplicar otra técnica posterior para identificar cuál de las regiones segmentadas corresponde a la glotis.

1.1.4. Contornos Activos

Los modelos de contorno activos, también llamados *snakes* o *active contour models* en inglés, son *splines* (funciones polinomiales definidas por trozos) que buscan minimizar una función de energía lo cual los atrae hacia características como líneas y bordes de la imagen [17]. Típicamente se minimiza la suma de dos energías: una energía interna, asociada a las deformaciones del contorno, y una energía externa, que generalmente depende de las características de la imagen y de fuerzas introducidas por el usuario. Esta minimización se hace generalmente de forma iterativa mediante *gradient descent* u otro método similar, y por lo tanto se debe definir algún criterio de convergencia para terminar el algoritmo. Comparado con los métodos anteriormente mencionados, los contornos activos son más flexibles e insensibles a problemas como ruido o intensidades no homogéneas. Sin embargo, igual que para el crecimiento de regiones, se necesita definir un contorno inicial que puede ser no trivial de obtener. A pesar de esto, debido a las ventajas mencionadas anteriormente son una de las técnicas más populares para segmentar la glotis en videos laríngeos.

En los anexos A y B se detallan ejemplos de implementaciones de contornos activos. La figura 1.7 muestra el resultado de aplicar el contorno activo descrito en [29] y en el Anexo A a una imagen glotal.



(a) Imagen de ejemplo de un HSV con el contorno iniicial marcado en verde

(b) Resultado al aplicar el modelo de contorno activo

Figura 1.7: Ejemplo de contorno activo. Note que el contorno resultante marca el borde de la glotis de forma bastante precisa

1.2. Papers recientes

A continuación se hará una revisión rápida de papers recientes que traten el tema de la segmentación de la glotis en videos de las cuerdas vocales.

1.2.1. Glottis segmentation with a high-speed glottography: a fully automatic method (2009) [10]

Técnicas: LoG filtering, Crecimiento de regiones, Contornos activos

En este paper primero se eligen los cuadros del video donde la glotis está más abierta. Esto se hace calculando la intensidad de gris promedio de la imagen para todos los cuadros; aquellos cuadros donde la glotis está abierta tendrán menor intensidad promedio. Se hace un análisis frecuencial de la intensidad promedio de los cuadros a lo largo del video para obtener la frecuencia fundamental de oscilación de las cuerdas vocales, y luego se encuentra el cuadro con la glotis más abierta (i.e. con la menor intensidad promedio) para cada ciclo.

Para segmentar la glotis en los cuadros elegidos se aplica un método de crecimiento de regiones; los puntos iniciales para el crecimiento de regiones se obtienen a través de un método basado en *Laplacian of Gaussian filtering* (LoG) descrito en [18]. Un filtro

LoG es básicamente un Laplaciano, que es una representación de la derivada, que se aplica a una imagen previamente suavizada por un filtro Gaussiano para eliminar ruido. Ya que ambas operaciones se pueden expresar como filtros con un kernel, se pueden convolucionar entre sí dando como resultado un solo filtro, el llamado filtro LoG, que se aplica sólo una vez a la imagen.

La idea de aplicar este filtro es encontrar regiones elípticas y más oscuras que la región circundante. Para esto la gaussiana del filtro no será redonda sino elíptica. El filtro se aplicará para todas para todas las posiciones posibles de la imagen; la máxima respuesta entregada por el filtro será considerada como la ubicación de la glotis. Pero la glotis puede tener distintos tamaños en distintos videos, por lo que se hace un análisis multidimensional: la dimensión del filtro aumentará progresivamente para detectar objetos de distintos tamaños. Es decir el filtro LoG se aplica para todas las posiciones en cada dimensión. Los resultados se normalizan para que sean comparables entre sí.

El procedimiento anterior da como resultado el centro de la glotis y una estimación de su área; con estos datos se procede a aplicar un método de crecimiento de regiones. El criterio de expansión es un umbral que es igual a la intensidad de gris promedio del area encontrada, mientras que el punto inicial es el centro de la glotis calculado anteriormente. Una vez que la región deja de crecer, se actualiza la media, se aumenta el umbral y se repite el proceso. El algoritmo termina cuando el área resultante supera el área de la región encontrada con el filtro LoG en el paso anterior multiplicada por un factor.

Una vez terminado el algoritmo de crecimiento de regiones, el contorno resultante puede todavía ser algo irregular. Para corregir esto se aplica un modelo de contorno activo descrito en [24]. Luego se segmentan los cuadros adyacentes en dirección hacia adelante y hacia atrás utilizando el mismo contorno activo, tomando como contorno inicial el cuadro anterior ya segmentado.

Los resultados se presentan sólo de forma cualitativa.

1.2.2. Automatic high-speed video glottis segmentation using salient regions and 3D geodesic active contours (2015) [23]

Técnicas: Saliency detection, Análisis de componentes conectados, Contornos activos

El primer paso del algoritmo es un pre-procesamiento del video. Se aplica un méto-

do de reducción de ruido descrito en [6] a cada canal de color de forma separada con el objetivo de eliminar elementos como reflejos y otros distractores manteniendo la información importante de los bordes intacta. También se aplica otro procedimiento con el objetivo de compensar los movimientos de la cámara a lo largo del video: cada 10 cuadros se hace una comparación basada en la suma de diferencias absolutas entre los colores de pixeles en las imágenes, que es una versión modificada del *método de minimización de diferencia de magnitud* propuesto en [9]. Esta comparación permite estimar la rotación y traslación sufrida por los cuadros del video, lo cual permite hacer las correcciones necesarias.

Luego se aplica una técnica de *saliency detection* o detección de prominencia para obtener un contorno inicial que luego será expandido mediante un contorno activo. La detección de prominencia busca identificar regiones de la imagen que tengan alta probabilidad de ser importantes en base a la intensidad de sus pixeles. Las regiones rodeadas por ”fondo” y que no están conectadas a los bordes de la imagen serán identificadas como regiones prominentes; la glotis califica ya que generalmente es una región oscura rodeada de tejido más claro y ubicada hacia el centro de la imagen. Esta técnica de *saliency detection* está basada en lo descrito en el paper [32]: se utilizan *mapas booleanos*, los cuales son una representación espacial de la imagen que la divide en dos regiones: primer plano (1) y fondo (0), y se obtienen aplicando un umbral a cada canal de color de la imagen. El umbral no será único, sino que se aplicarán varios umbrales de forma iterativa aumentando el valor según un *step size* fijo. En cada iteración se calcula un *mapa de atención* a partir del mapa booleano, el cual identifica objetos buscando regiones del fondo que estén completamente rodeadas por el primer plano y que no estén conectadas a los bordes de la imagen. Todos los mapas de atención calculados se promedian, obteniendo un mapa final al cual se le aplica un umbral para filtrar señales débiles.

Esta técnica de detección de prominencia se utiliza para calcular una ROI. Se elige el cuadro con la menor intensidad de gris promedio, el cual será el que tiene la mayor apertura glotal. Luego se calcula el mapa de prominencia para ese cuadro y se hace un análisis de componentes conectados donde cada componente se multiplica por una versión en escala de gris de la imagen. Se hace un ranking tomando en cuenta el área y suma de intensidades de cada componente para identificar aquel que corresponda a la glotis. El objeto elegido se utiliza como ROI para todo el video y para calcular un cuadro delimitador.

El último paso es la segmentación como tal, la cual se hace con un contorno activo *geodésico*. El término geodésico hace referencia a una implementación particular de un modelo de contorno activo descrita en [27]. Para obtener el contorno inicial se aplica la misma técnica de detección de prominencia descrita anteriormente a la imagen que se desea segmentar. El mapa de atención resultante se multiplica por la ROI y el objeto resultante se utiliza para inicializar el contorno. Esto se hace para todos los cuadros del video, segmentándolo de forma completa.

Se hace al final una evaluación numérica de los resultados utilizando el coeficiente Dice [11] obtenido al comparar con glotis manualmente segmentadas.

1.2.3. Automatic glottal segmentation using local-based active contours and application to glottovibrography (2012) [16]

Técnicas: Filtro Sobel, Análisis de componentes conectados, Contornos activos

De forma similar a [10], se encuentran primero los cuadros donde la glotis está más abierta, llamados cuadros de referencia, uno por cada ciclo. Luego a cada cuadro de referencia se le aplica un filtro de Sobel para detectar bordes verticales, seguido de una operación de cerrado morfológico para conectar pequeñas regiones relacionadas. Se separan los objetos detectados mediante un análisis de componentes conectados utilizando la metodología descrita en [22], y se selecciona aquel con la mayor área y orientación vertical. Alrededor de ese objeto se calcula una caja rectangular, con el objetivo de reducir la cantidad de datos a procesar. La parte del video que se encuentra dentro de la caja se recorta; para todo el procesamiento posterior se trabajará exclusivamente con ese recorte. Para ubicar la glotis con mayor precisión, se repite todo el proceso descrito pero ahora aplicado a la versión recortada de los cuadros elegidos. El resultado será una caja rectangular más compacta y precisa. De esta forma se obtendrá una ROI rectangular para cada ciclo lo cual compensa los movimientos que podría tener la cámara a lo largo del video.

También se necesita determinar si la glotis existe en todos los cuadros o no, para lo cual se aplican dos técnicas. Primero, se calcula la mínima intensidad de pixel de la imagen (el recorte) y se evalúa si esta intensidad es mayor a un umbral definido como la mediana de las intensidades de pixel de toda la secuencia. Segundo, se calcula la caja rectangular para el cuedro en cuestión de la misma forma descrita en el párrafo anterior y se evalúa si la caja resultante está centrada lejos de la ROI del ciclo actual (o si no

existe). Cuando ambas condiciones ocurren simultáneamente, se excluye al cuadro de más procesamiento.

También se aplica una técnica de mejora de contraste descrita en [33] para obtener más detalle en el área glotal evitando añadir más ruido. La técnica está basada en ecualización de histogramas.

Finalmente la glotis se segmenta utilizando un modelo de contorno activo descrito en [7]. Para obtener el contorno inicial se proponen dos métodos: el primero consiste en hacer un análisis de histograma de la ROI para obtener un valor de umbral de intensidad de pixel. Este análisis de histograma es descrito en [13] y es llamado "mode method". El segundo método consiste en dibujar un elipse de tamaño proporcional a la ROI y utilizarlo como contorno inicial. Ambos métodos se usan para inicializar un contorno activo; aquel resultado que logre la mayor diferencia de intensidad entre la región segmentada y el fondo es elegido como la segmentación correcta. Para segmentar los cuadros adyacentes se utiliza el borde ya segmentado del cuadro anterior como inicialización para el contorno activo. Los cuadros iniciales serán los cuadros de referencia obtenidos al inicio del método.

Se hace un análisis cualitativo de los resultados pidiéndole a 13 participantes que evaluaran la calidad de la segmentación en una escala de 1 a 5 y que en caso de haber errores los corrigieran utilizando una herramienta desarrollada en MATLAB. Estas correcciones fueron utilizadas para calcular el error promedio del área de segmentación medido en cantidad de pixeles para cada video.

1.2.4. An automatic method to detect and track the glottal gap from high speed videoendoscopic images (2015) [2]

Técnicas: ROI basada en cambios de intensidad durante el video, Watershed transform, Contornos activos

Este paper trabaja con imágenes en escala de grises. Se hace primero un pre-procesamiento aplicando una transformación no lineal a la intensidad pixel para mejorar el contraste que consiste en:

$$I_{out}(x,y) = \begin{cases} 255 & I(x,y) > L_i \\ 255 \cdot (\frac{I(x,y)}{L_i})^{\zeta} & I(x,y) \leq L_i \end{cases} \tag{1.1}$$

$$L_i = \frac{1}{m\beta} \sum_{j=1}^{m} I(x_j, y)$$

Donde $I(x,y)$ es la intensidad de gris de un pixel de la imagen, L_i es la intensidad promedio de la fila i, m es el número de columnas, β es un factor que ajusta el contraste y se deja en 1.3 y ζ es un coeficiente que se deja en 1.8.

Luego se calcula una ROI basada en los cambios de intensidad de cada pixel a lo largo del video. El cálculo se hace en base a la variación total de intensidad de cada fila y de cada columna del video. La transformación no lineal aplicada anteriormente reduce la influencia de reflejos y otros ruidos que pudieran afectar a los cambios de intensidad de pixeles a lo largo del video. La ROI además se va actualiza cada N cuadros para compensar los movimientos que pueda sufrir la cámara a lo largo del video.

Luego se aplica la transformada watershed, seguida de una fusión de regiones basada en un criterio a su vez basado en JND (*Just Noticeable Difference*) definido en [25] para reducir la sobre-segmentación. Primero se aplica un umbral de intensidad bajo de 10 para eliminar ruido en la imagen de gradiente, luego se aplica la transformada watershed y se fusionan las regiones resultantes según este criterio. La JND busca medir cuantitativamente la máxima diferencia de intensidad de gris que puede ser diferenciada por el ojo humano a partir de cierto valor de intensidad, y se define como:

$$JND(k) = \begin{cases} D_0 \cdot (1 - (k/127)^{0,5}) + 3 & k \leq 127 \\ \gamma(k - 127) + 3 & e.o.c. \end{cases} \tag{1.2}$$

Donde k es un valor de intensidad, D_0 es el umbral de visibilidad cuando el fondo es 0 dejado en 17 y γ es la pendiente de la recta que modela el JND a intensidades más altas dejada en 3/128. El criterio para determinar si se unen regiones es la siguiente función de costo:

$$F_c = [\,|\bar{R}_1 - \bar{R}_2| - min(JND(\bar{R}_1), JND(\bar{R}_2))\,] \tag{1.3}$$

Donde $\bar{R}_1$ y $\bar{R}_2$ son las intensidades promedio de cada región. Las regiones se unirán si la función de costo tiene un valor menor a 256.

Luego se realiza otra fase de fusión de regiones basada en el coeficiente de correlación normalizado con modelos de glotis segmentadas a mano. El paso final consiste en aplicar el modelo de contorno activo descrito en [7] para delinear mejor la glotis.

Se hace un análisis cuantitativo de los resultados utilizando el índice Pratt (PI) [1] y el error de consistencia a nivel de objetos (*object-level consistency error*, OCE) [21] al comparar con glotis segmentadas manualmente.

1.2.5. Snake based automatic tracing of vocal-fold motion from high-speed digital images (2012) [31]

Técnicas: Comparación de formas, Contornos activos

Se comienza aplicando un umbral global a la imagen, el cual tiene un valor igual a un 60 % de la máxima intensidad de ésta. Los pixeles bajo ese umbral serán igual a 1, el resto 0. Luego se calcula la suma de los pixeles de cada columna para la mitad superior y la mitad inferior de la imagen binaria resultante. El resultado serán dos señales de largo n, donde n es el número de columnas, las cuales se multiplican entre sí. Esto está basado en la suposición de que la glotis estará aproximadamente al medio de la imagen orientada de forma vertical, y en las columnas donde ésta se encuentre la suma de pixeles será alta. La multiplicación eliminará falsas regiones no simétricas entre la mitad superior e inferior de la imagen. También se calcula la suma de pixeles para cada fila. Por lo tanto tendremos dos señales: la suma de intensidades por columna (el resultado de la multiplicación) y la suma de intensidades por fila. Se calcula el máximo de cada una de esas señales, lo cual resultará en las coordenadas del centro de la glotis. Sobre esas coordenadas se construirá un elipse con una forma que aproxime la forma del objeto detectado. El ancho y alto de este elipse se obtienen a partir del ancho del ”peak” encontrado en cada una de las dos señales calculadas anteriormente. Luego este elipse se utilizará como región inicial para un contorno activo que delineará correctamente la glotis. El contorno activo utilizado se describe en [17]. Para segmentar los cuadros adyacentes el contorno inicial se calcula un nuevo elipse basado en el contorno segmentado del cuadro anterior, con el cual se inicializa el mismo contorno activo.

Se presentan sólo resultados cualitativos de la segmentación.

1.2.6. Full-Automatic Glottis Segmentation With Active Shape Models (2011) [5]

Técnicas: Crecimiento de regiones, Active shape models

Se comienza con un algoritmo de crecimiento de regiones, cuyo valor inicial se obtiene a partir de una simple relación lineal entre la intensidad de gris promedio de la imagen y el valor inicial óptimo. Al resultado del crecimiento de región se le aplica una apertura morfológica, y luego se filtran los objetos resultantes en base a su área; se eliminan objetos demasiado pequeños o grandes.

Sobre cada objeto que quede se inicializará un modelo de forma activo (*active shape model* o ASM) para intentar segmentar la glotis descrito en [8]. Los ASM son muy similares a los contornos activos, ya que también son contornos que se deforman iterativamente, pero en este caso están restringidos a tomar ciertas formas predefinidas en un set de entrenamiento. Para determinar cuál de los contornos resultantes es el correcto, se calcula un puntaje de confiabilidad basado en la ubicación de ciertos *landmarks* definidos también en el set de entrenamiento, el cual se describe en [26]. Es decir, debe haber un set de entrenamiento que contenga glotis segmentadas correctamente con sus landmarks marcados.

Los resultados se expresan como el error de área medido en cantidad de pixeles al comparar con glotis manualmente segmentadas.

1.2.7. GlottalImageExplorer–An open source tool for glottis segmentation in endoscopic high-speed videos of the vocal folds (2016) [3]

Técnicas: Crecimiento de regiones

Este paper presenta un software gratis y de código abierto llamado *GlottalImageExplorer* para la segmentación semi-automática de la glotis en HSV. Permite segmentar videos en formato `.avi` y de tamaño 256×256. El algoritmo utilizado para la segmentación fue descrito originalmente en el paper [19].

El algoritmo trabaja con imágenes en escala de grises; es decir al cargar un video a color este automáticamente se convierte a escala de grises. Como se mencionó anteriormente, el método es semi-automático y por lo tanto requiere intervención del usuario; éste selecciona tres puntos ubicados dentro de la glotis: Uno en el extremo superior, otro en el extremo inferior, y otro en el centro. Además, para cada uno de esos puntos el usuario deberá definir el valor de un umbral de intensidad de gris. Luego se aplicará un método de crecimiento de regiones donde los puntos iniciales serán los puntos se-

leccionados por el usuario y el criterio de expansión será un umbral de intensidad, pero éste será distinto para cada fila de la imagen. Para aquellos puntos ubicados en filas superiores a la fila donde se ubica el punto superior (incluyendo a esa fila), se utiliza el umbral definido para el punto superior. La misma lógica se aplica para las filas ubicadas debajo del punto inferior. Para las filas ubicadas entre la fila del punto superior y la del punto central, se realiza una interpolación lineal entre el umbral superior y el umbral central. Lo mismo para las filas ubicadas entre el punto central y el inferior.

El usuario elige un cuadro inicial donde se definen los puntos iniciales y los umbrales, idealmente uno donde la glotis se encuentre en su punto máximo de apertura. Una vez definidos los puntos y umbrales la aplicación procederá a segmentar todos los cuadros del video utilizando el método de crecimiento de regiones inicializado con los mismos puntos y umbrales. Si el usuario observa que en cierto cuadro del video el algoritmo falla, puede redefinir los puntos iniciales y umbrales para ese cuadro y los cuadros subsiguientes.

El software cuenta con una interfaz gráfica para visualizar los cuadros del video y definir los puntos iniciales y umbrales. Se encuentra disponible de forma gratuita en [4]. En la figura 1.8 se muestra la aplicación en funcionamiento.

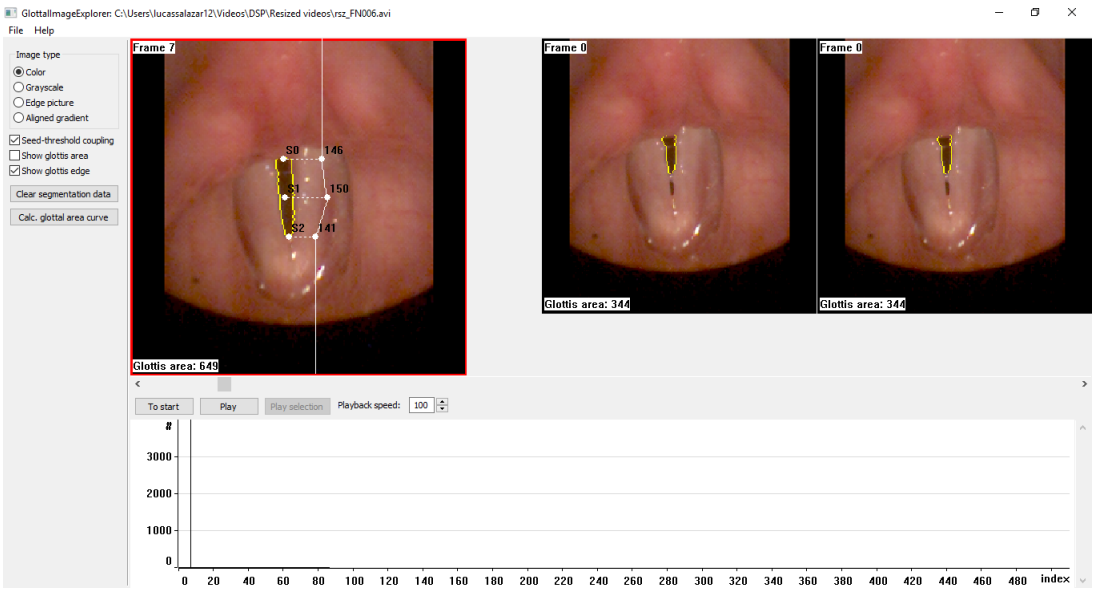


Figura 1.8: Interfaz gráfica de la aplicación GlottalImageExplorer

1.2.8. Fully automated glottis segmentation in endoscopic videos using local color and shape features of glottal regions (2015) [14]

Técnicas: Comparación de formas, Comparación de color, Contornos activos

En este algoritmo se tiene un dataset de entrenamiento que contiene información de la forma y propiedades de color de glotis manualmente segmentadas. Se comienza haciendo aplicando un umbral a la imagen, para luego hacer una comparación de la forma de todos los objetos detectados con el set de entrenamiento y filtrar aquellos que no sean similares. Luego se refina el borde con un contorno activo descrito en [29] y se comparan las propiedades de color con las del set de entrenamiento, filtrando nuevamente aquellos objetos con propiedades no similares. La detección resultante se marca como glotis. Esto se repite para todos los cuadros del video; de haber una detección de glotis, se guarda su nivel de similitud con el set de entrenamiento. Luego se ordenan todas las detecciones por su nivel de similitud, y comenzando por el que tenga el mayor nivel, se segmentan los cuadros adyacentes en base a la similitud de color con el cuadro anterior utilizando el contorno activo descrito en [7].

Los resultados se expresan de forma numérica utilizando el coeficiente Dice y el error de área en comparación a glotis segmentadas manualmente.

* * *

El hecho de que hayan papers recientes sobre el tema indica que la segmentación automática de la glotis en HSV todavía no es un tema completamente resuelto. Algunos problemas comunes que sufren los métodos son la alta sensibilidad al ruido, a la mala iluminación de la imagen o a desplazamientos de la cámara, que frecuentemente causan errores en la segmentación de la glotis como un mal delineamiento del contorno, detección de falsos positivos o derechamente la no detección de la glotis. Otras fuentes de problemas son la existencia de puentes mucosos en las cuerdas vocales (que causa que la glotis se encuentre separada en dos secciones), el que la glotis esté parcialmente tapada en el video, o la presencia de nódulos u otras patologías que alteran la apariencia de la glotis y pueden dificultar su correcta segmentación. Todas las situaciones nombradas ocurren con frecuencia en videos de alta velocidad, particularmente aquellos grabados con laringoscopios flexibles, y por lo tanto los métodos de segmentación deben lidiar

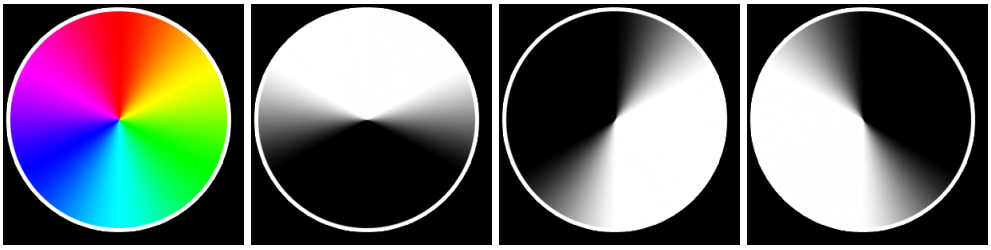
con ellas de una u otra manera.

Otro problema que puede dificultar la investigación de este tema es que en general no se pueden comparar directamente los resultados obtenidos por un paper con los de otro. Los videos utilizados siempre serán distintos en cada paper, y la metodología usada para evaluar los resultados no siempre es la misma; varios papers sólo presentan los resultados de forma cualitativa. No existe un benchmark oficial para evaluar cuantitativamente la calidad de un algoritmo. A esto se suma el hecho de que es bastante común que un paper reporte obtener buenos resultados, pero luego otro paper más nuevo lo cite y contradiga esa afirmación, diciendo que el método antiguo tiene problemas y que el nuevo método propuesto es mejor.

Además de lo anterior, hay dos puntos que llaman la atención en la mayoría de los algoritmos propuestos: Lo primero es que la mayoría trabaja sólo con imágenes en escala de grises, siendo que actualmente existen cámaras capaces de grabar videos de alta velocidad a color, y el color es importante para los clínicos ya que los ayuda a identificar lesiones en el tejido de las cuerdas vocales. Lo segundo es que se usan muy poco técnicas de Machine Learning, siendo que en los últimos años este tipo de técnicas se han vuelto muy populares en el campo de computer vision, del cual la segmentación de imágenes es parte. A continuación se hará una breve explicación del tema del color y de Machine Learning.

Sobre el color

La forma más común de representar el color en videos e imágenes es a través del modelo RGB. RGB es un modelo aditivo en el cual es posible representar un color mediante la suma de tres colores primarios que son rojo (R), verde (G) y azul (B). Cada imagen de un video a color estará compuesta por tres canales, cada uno representando la intensidad de un color primario. El color de un pixel en la imagen estará definido por los valores que tenga cada canal en ese pixel.



(a) Imagen de ejemplo. Fuente: [20] (b) Canal rojo (c) Canal verde (d) Canal azul

Figura 1.9: Imagen de ejemplo junto a cada uno de sus tres canales por separado

Tradicionalmente, la mayoría de los videos de alta velocidad eran grabados de forma monocromática, es decir en blanco y negro. El uso de sistemas HSV a color para la evaluación laríngea es un desarrollo relativamente reciente. En consecuencia, la mayoría de las técnicas de análisis de HSV existentes trabajan solamente con videos monocromáticos, lo cual es contradictorio ya que para los clínicos el color es importante para la identificación de lesiones en el tejido de las cuerdas vocales.

Sobre Machine Learning

Machine learning es un sub-campo de Inteligencia Artificial que utiliza técnicas para permitir que los computadores "aprendan" directamente de los datos y hagan predicciones o tomen decisiones basadas en ellos, sin ser explícitamente programados. Los algoritmos de machine learning mejoran su rendimiento a medida que aumenta la cantidad de datos disponibles para el aprendizaje.

Los algoritmos de machine learning generalmente se dividen en dos grupos: Aprendizaje supervisado y Aprendizaje no supervisado. En aprendizaje supervisado, al computador se le presentan ejemplos de entradas junto con la salida deseada para cada entrada. El objetivo es que el computador desarrolle a partir de estos ejemplos una regla que le permita mapear las entradas a las salidas de forma correcta. Ejemplos típicos son clasificadores de texto o imágenes. Por otro lado, en aprendizaje no supervisado los datos que se le entregan al computador no están clasificados ni etiquetados de ninguna manera (no hay asociación de entradas con sus salidas correctas) y el computador debe

encontrar la estructura de los datos y ajustarles un modelo por sí solo. En este caso no hay una forma establecida de evaluar el desempeño de los algoritmos.

La segmentación de la glotis es una tarea que caería principalmente en el área de aprendizaje supervisado, ya que se tiene una entrada (el video o imagen de las cuerdas vocales) y una salida (la región de la imagen que corresponde a la glotis) definidas y que pueden ser evaluadas. Pero también es posible utilizar técnicas de machine learning en algún paso intermedio de la segmentación, con otras entradas y salidas.

Sólo uno de los métodos de segmentación listados anteriormente utiliza técnicas que podrían ser clasificadas como aprendizaje supervisado, lo cual llama la atención ya que este tipo de algoritmos son muy usados para el reconocimiento y clasificación de imágenes, y pueden ser de ayuda en la segmentación.

Parte III

Desarrollo del Tema

Capítulo 1

Algoritmo de segmentación original

En este primer capítulo se describirá la implementación del paper [14]: *"Fully automated glottis segmentation in endoscopic videos using local color and shape features of glottal regions"* en su forma original. En el segundo capítulo se describirán las mejoras y modificaciones que se hicieron al algoritmo para completar los otros objetivos de la memoria, mientras que el tercer capítulo describe cómo se evaluarán cuantitativamente los algoritmos.

1.1. Resumen del algoritmo

El algoritmo puede dividirse en tres partes principales: Entrenamiento, Reconocimiento y Segmentación. Para la parte de Entrenamiento se requiere tener un dataset de imágenes laríngeas donde la glotis se encuentre segmentada de forma manual y correcta; a partir del dataset se extrae información sobre la forma de las glotis, sus propiedades

de color y las propiedades de color de su tejido adyacente. El objetivo de la segunda parte de Reconocimiento es encontrar aquellos cuadros del vídeo que tienen la mayor probabilidad de contener una glotis abierta. Se marcan todos los cuadros donde se detecta una glotis, y se guarda el contorno detectado junto con un indicador de similitud con las glotis de entrenamiento, según el cual los cuadros serán ordenados de mayor a menor en una lista. En la tercera parte del algoritmo se hace la segmentación como tal. Los cuadros guardados en la lista son utilizados como puntos iniciales para segmentar la glotis en los cuadros adyacentes de forma sucesiva. Esta segmentación de cuadros adyacentes continúa hasta que se detecta que la glotis está cerrada, momento en el que se elige el siguiente cuadro no segmentado de la lista para continuar la segmentación. El proceso sigue hasta que no quedan más cuadros en la lista, idealmente habiendo segmentado todos los ciclos de vibración glotal presentes en el video. En la figura 1.1 se muestra un diagrama que resume el método, sacado directamente del paper.

1.2. Pre-procesamiento

Todos los videos con los que se trabajó fueron grabados utilizando una cámara *Photron SA-X2*. La salida directa de la cámara genera un video a color con 12 bits de profundidad por canal, y en formato *.raw*. La tasa de cuadros por segundo varía de video a video; los videos grabados con laringoscopio flexible están a 4000fps, mientras que el resto de los videos algunos están grabados algunos 8000fps y otros a 10000fps.

El formato en que se encuentran los videos sin procesar (*.raw* y 12 bits por canal de color) hace que sea difícil trabajar directamente con ellos, ya que la mayoría de las aplicaciones no leen ese tipo de formato, incluido MATLAB que es la aplicación que se utilizó para el desarrollo del trabajo. Es por esto que los videos fueron convertidos a formato *.avi* y su profundidad de color se redujo a 8 bits, que es lo más normal. Para reducir la profundidad de bits del color se hace un *bit-shift*, que consiste en elegir cuáles de los 12 bits originales se considerarán para hacer el mapeo a 8 bits. En el formato original cada dato puede tomar valores entre 0 y $2^{12} - 1 = 4095$, mientras que en formato 8 bits pueden tomar valores entre 0 y $2^8 - 1 = 255$. Un bit-shift de 0 significa que se considerarán los 12 bits y el mapeo es $[0, 4095] \rightarrow [0, 255]$, es decir simplemente se multiplica por $255/4095$ y se redondea. Un bit-shift de 1 significa que

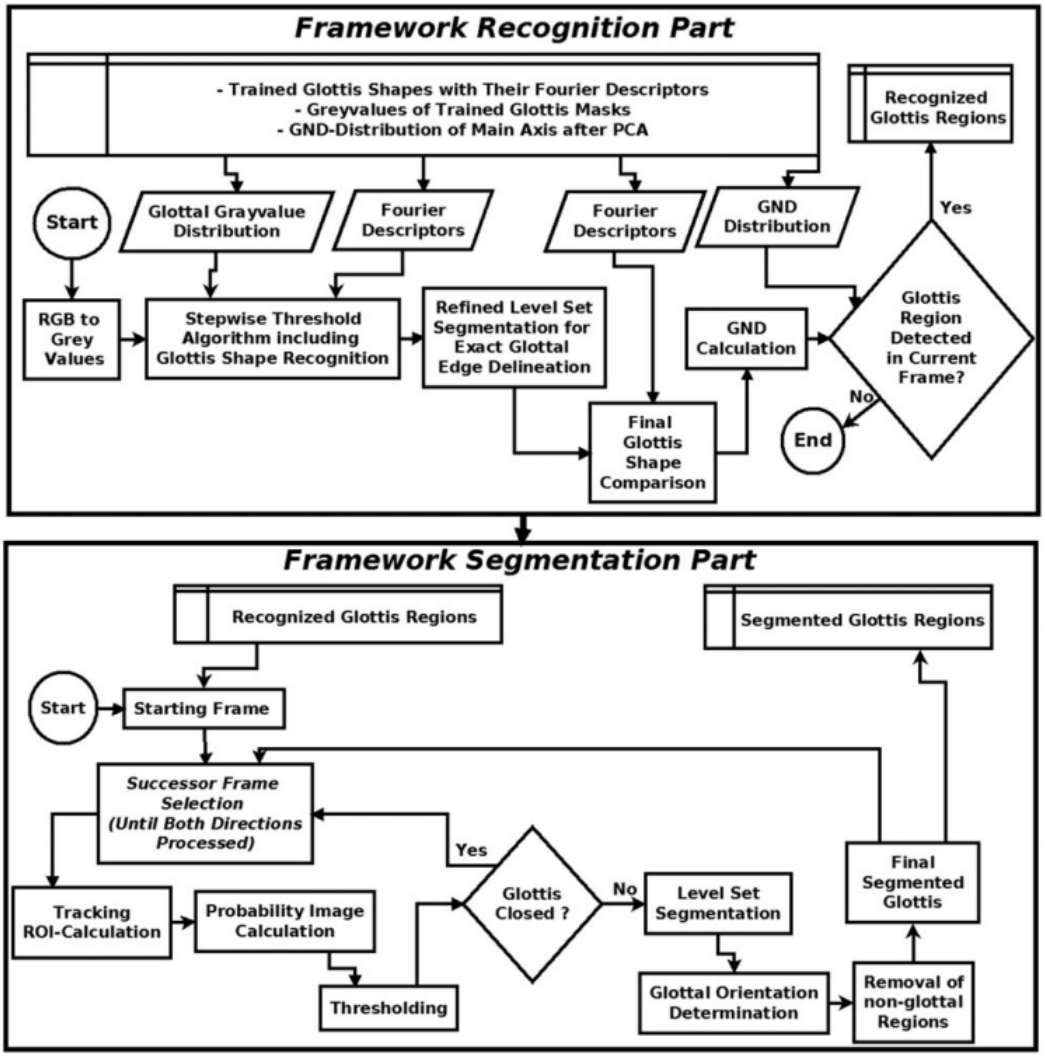


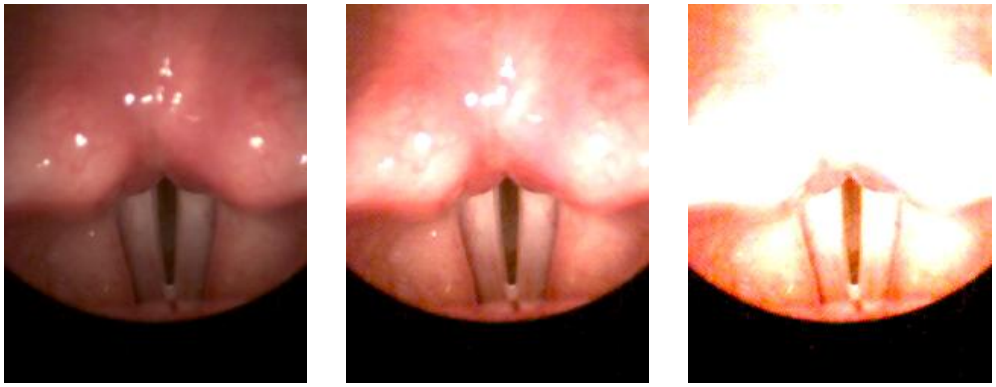
Figura 1.1: Diagrama que resume el algoritmo. Fuente: [14]

se considerarán los 11 bits más bajos para hacer el mapeo, es decir $[0, 2047] \rightarrow [0, 255]$. Aquellos datos que tengan valores superiores a 2047 se mapearán a 255. Otra forma de entender el bit-shift es que se define un umbral tal que los valores que estén por sobre ese umbral se mapearán a 255, y los que no se mapearán a $[0, 255]$ de la forma descrita anteriormente. Cambiar el valor del bit-shift significa cambiar el valor de ese umbral.

Bit-shift	**Valor máximo**	**Mapeo**
0	4095	$[0,4095] \rightarrow [0,255]$
1	2047	$[0,2047] \rightarrow [0,255]$
2	1023	$[0,1023] \rightarrow [0,255]$
3	511	$[0,511] \rightarrow [0,255]$
4	255	$[0,255] \rightarrow [0,255]$

Tabla 1.1: Mapeos de bit-shift

Es posible llegar hasta un bit-shift de 11, donde sólo el bit menos significativo se tomaría en cuenta, y los valores posibles para cada dato serían sólo dos: 0 o 255. A la mayoría de los videos se les aplica un bit-shift entre 0 y 2, dependiendo de cómo quede mejor la iluminación de la imagen. Aplicar un bit-shift siempre aumentará el brillo aparente y modificará el contraste de la imagen, lo cual generalmente es ventajoso porque se acentúan más las diferencias de iluminación entre la glotis y el resto de la imagen, facilitando la segmentación. Pero siempre en moderación; un bit-shift mayor a 2 generalmente aumenta demasiado el brillo y se pierde demasiada información de la imagen. A continuación se muestran algunos ejemplos:



(a) Imagen original, 0 bit-shift

(b) Imagen luego de 1 bit-shift

(c) Imagen luego de 2 bit-shift

Figura 1.2: Bit-shifting

1.3. Entrenamiento

El primer paso del algoritmo es formar un dataset de imágenes laríngeas donde la glotis se encuentre correctamente segmentada y que esté aprobado por expertos médicos. En esta implementación el dataset consiste en 67 imágenes de glotis. Luego se extraerá del dataset información sobre la forma y las propiedades de color que tienen las glotis de entrenamiento y que las diferencian del tejido adyacente. La forma de las glotis será descrita mediante *Descriptores de Fourier*, mientras que la información de color se describirá a través de un método original propuesto por los investigadores del paper y que ellos llaman *Glottal Neighborhood Descriptor* (GND). Antes de seguir con el algoritmo es necesario explicar cómo funcionan estas técnicas.

1.3.1. Descriptores de Fourier

Los descriptores de Fourier son una técnica para codificar la forma de un objeto a través de la transformada de Fourier. El proceso para calcularlos es:

1. Encontrar las coordenadas del contorno del objeto, en orden contrario a las manecillas del reloj.
2. Convertir cada coordenada en un número complejo de la forma: $(3,4) \rightarrow 3+4j$. El arreglo de coordenadas se convierte en una señal compleja.
3. Calcular la DFT de la señal compleja.

El resultado son los descriptores de Fourier del contorno del objeto. Se puede hacer una analogía con la transformada de Fourier de una señal cualquiera; si se calcula la DFT inversa de los descriptores volveremos al contorno original. Si calculáramos la DFT inversa de los primeros N descriptores (o más precisamente los primeros y últimos N/2 debido a la estructura de la DFT) el resultado se podría interpretar como aplicarle un filtro pasa-bajos al contorno, es decir lo suavizaríamos. Otras propiedades relevantes son:

- Traslación: Sólo el primer descriptor de Fourier es afectado por la traslación. Es decir, ignorando este descriptor es posible comparar dos contornos de forma invariante a traslación utilizando sus descriptores.
- Escalamiento: Si el contorno del objeto es escalado por un factor, los descriptores de Fourier serán escalados por ese mismo factor.

En la sección 1.4 de Reconocimiento del algoritmo se compararán los descriptores de Fourier de las glotis de entrenamiento con los de contornos detectados en cuadros del video que se desea segmentar; aquellos que sean más similares a las glotis de entrenamiento se considerará que tienen mayor probabilidad de efectivamente ser una glotis. Para hacer la comparación se utilizará simplemente la norma cuadrada de la diferencia entre los descriptores de los contornos de la forma:

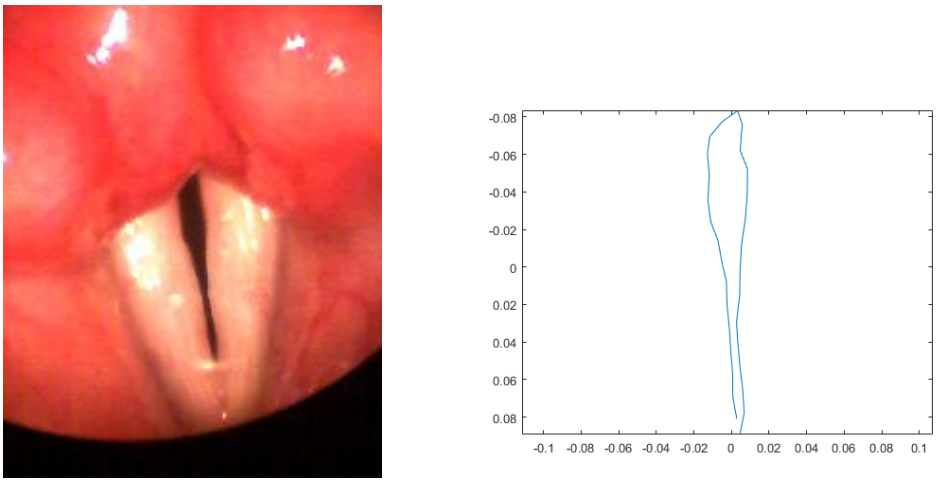
$$D(F_1, F_2) = \|F_1 - F_2\|^2 \tag{1.1}$$

Donde F_1 y F_2 son los descriptores de fourier a comparar.

Para que esta comparación sea válida, debemos comparar de forma invariante a traslación, escala, rotación y punto inicial. La invarianza a traslación se logra simplemente multiplicando el primer descriptor de fourier por cero. La invarianza a escala se logra dividiendo todos los descriptores de fourier por la magnitud del segundo descriptor de fourier que siempre será distinto de cero ya que el contorno describe un área distinta de cero [12]. Para obtener invarianza a rotación podemos aprovechar el hecho de que las glotis generalmente tienen una forma elíptica y alargada, y aproximar el eje principal de la glotis como la recta entre los dos puntos más distantes del contorno. Luego se rota el contorno según el ángulo entre ese eje principal y el eje vertical de la imagen. Y finalmente para obtener invarianza a punto inicial, se elige el punto con la coordenada más baja en el eje vertical como inicio. Todos estos cálculos deben hacerse al calcular el descriptor de Fourier de un contorno.

Se calcularán y se guardarán los descriptores de Fourier de todas las glotis de entrenamiento. Sólo los primeros 30 descriptores de cada contorno serán guardados, ya que se considera que eso es suficiente para preservar la información más importante del contorno. A continuación se muestra una imagen de una glotis junto con el contorno

generado a partir de los primeros 30 descriptores de Fourier y después de aplicar todos los procesamientos mencionados anteriormente.



(a) imagen original

(b) Contorno obtenido a partir de los primeros 30 descriptores de Fourier

Figura 1.3: Descriptores de Fourier

1.3.2. Glottal Neighborhood Descriptor

El *Glottal Neighborhood Descriptor* (GND) busca medir las diferencias de color entre la glotis y el tejido adyacente de la cuerda vocal. Para calcularlo, primero se seleccionan 8 puntos base del contorno de la glotis. Los puntos más distantes del contorno (que ya fueron encontrados para calcular los descriptores de Fourier) son elegidos como los primeros dos puntos, mientras que los 6 restantes se distribuirán de forma equidistante a lo largo del contorno, tres en el lado izquierdo y tres en el derecho. Serán ordenados según la dirección de las manecillas del reloj, partiendo con el punto más alto. Esto se hace para que puntos de contornos distintos puedan ser comparados entre sí.

Para cada punto base se calcula el vector de color medio ponderado por distancia en una vecindad del punto, tanto para la región glotal como para la región no glotal o fondo. El peso que se le asignará a cada pixel está dado por la siguiente función:

$$\omega(x_i, y_i) = e^{-\frac{(x_i - x_b)^2 + (y_i - y_b)^2}{\sigma^2}} \tag{1.2}$$

Donde (x_b, y_b) son las coordenadas del punto base. Para σ se usó un valor de 20 en el paper original, encontrado de forma empírica. El uso de esta función de peso significa que los pixeles más cercanos a los puntos base tendrán una mayor influencia en el GND.

Luego el vector de color medio ponderado por distancia al punto base para puntos dentro de la glotis se calcula como:

$$\vec{V}_{mean,in} = \frac{\sum_{in} \omega(x_i, y_i) \cdot \vec{V}(x_i, y_i)}{\sum_{in} \omega(x_i, y_i)} \tag{1.3}$$

Donde $\vec{V}$ es el vector de color RGB del pixel y $\sum_{in}$ es una sumatoria sobre todos los puntos dentro de la glotis. El mismo cálculo se hace para los puntos fuera de la glotis:

$$\vec{V}_{mean,out} = \frac{\sum_{out} \omega(x_i, y_i) \cdot \vec{V}(x_i, y_i)}{\sum_{out} \omega(x_i, y_i)} \tag{1.4}$$

Sin embargo, para reducir el tiempo de cálculo sólo se consideraron puntos dentro de una ventana cuadrada de tamaño $(2\sigma + 1) \times (2\sigma + 1)$ centrada en el punto base. Puntos fuera de la ventana están muy lejos del punto base y por lo tanto tienen pesos y contribuciones mínimas al resultado. De esta forma se ahorra algo de tiempo de cálculo.

Una vez calculados los vectores de color medio ponderado, se calcula la norma de la diferencia entre los dos vectores. La idea de hacer esto es obtener una medida de la diferencia de color entre los pixeles de la región glotal y los de la región no-glotal. En fórmula:

$$DMCL = |\vec{V}_{mean,out} - \vec{V}_{mean,in}| \tag{1.5}$$

Esta *diferencia media de color local* se obtiene para cada uno de los 8 puntos base del contorno, quedando como resultado un vector de tamaño 1×8 que es el GND. Todo

este cálculo se hace para cada una de las N imágenes de entrenamiento, resultando una matriz de tamaño $N \times 8$ con todos los GND. El siguiente paso es aplicar *análisis de componentes principales* (PCA) para reducir la dimensionalidad de los datos. Con las primeras dos dimensiones generadas por PCA es suficiente, ya que representan más del 94 % de la información en este caso.

Una vez reducida la matriz de GND a dos dimensiones, los datos se proyectan a un histograma 2D y luego son suavizados utilizando un kernel Gaussiano, con el objetivo de llegar a una función continua que represente la distribución probabilística de los GND del set de entrenamiento (reducidos a dos dimensiones). El procedimiento de suavizado es el siguiente: Se inicializa el resultado en cero. Luego por cada punto del histograma se suma una gaussiana bivariada centrada en el punto. Es decir:

$$f(x,y) = \sum_{i=1}^{N} e^{-(\frac{(x_i-x)^2}{2\sigma_x^2} + \frac{(y_i-y)^2}{2\sigma_y^2})} \tag{1.6}$$

Donde x_i e y_i son las coordenadas de cada uno de los N puntos del histograma, y σ_x y σ_y son los anchos de banda para cada variable. Los anchos de banda elegidos son 80 y 180, encontrados empíricamente. Una vez calculada la distribución ésta se normaliza para tener valores en el rango $[0,1]$. A continuación se muestra una imagen de la distribución resultante:

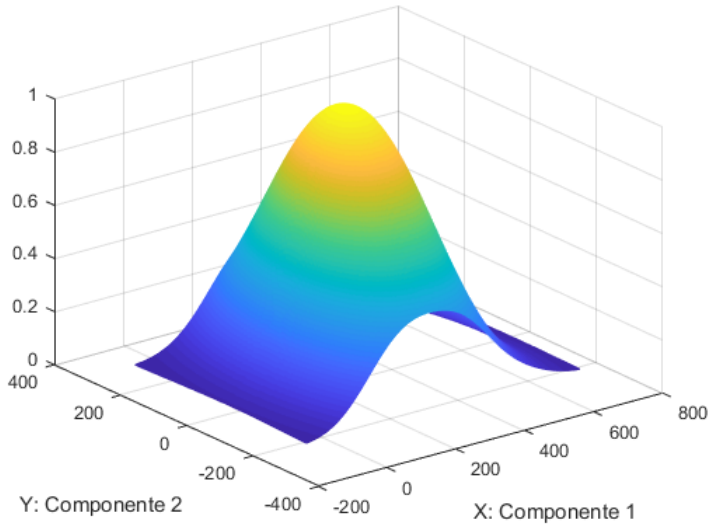


Figura 1.4: Distribución de GND reducido a dos dimensiones

Esta distribución, junto con los coeficientes PCA calculados, se guardarán para ser utilizados posteriormente en la sección de Reconocimiento. En esa parte del algoritmo, cuando se tenga un contorno candidato a glotis se calculará su GND de la forma descrita aquí, se reducirá su dimensionalidad a 2 utilizando los coeficientes PCA guardados y se obtendrá el valor $f(x,y)$ de la distribución de GND guardada. Este valor nos dirá qué tan probable es que el contorno corresponda a una glotis basado en las diferencias de color entre su interior y exterior.

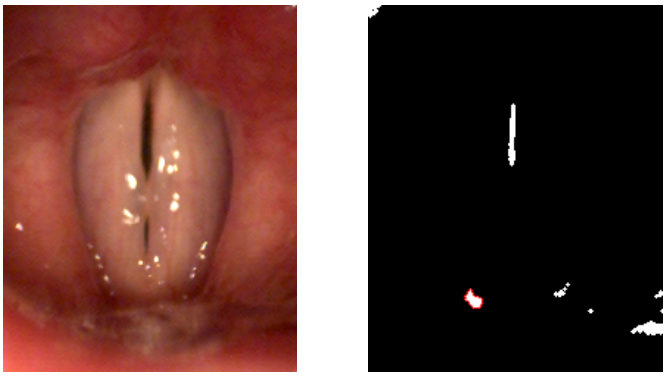
Como comentario, este procedimiento es uno de los puntos que más llama la atención del algoritmo ya que puede ser considerado como una técnica de machine learning. El hecho de que utilice datos de entrenamiento y que por lo tanto su efectividad mejore mientras más datos se tengan disponibles lo convierte en un método de aprendizaje supervisado. La tarea que se quiere lograr cae dentro de *clasificación de una clase*, ya que el dataset de entrenamiento consiste sólo en ejemplos positivos de una sola clase.

1.4. Reconocimiento

Una vez guardados los descriptores de Fourier y los GND de las glotis de entrenamiento, se procede a la parte de Reconocimiento. Aquí se trabaja con un video laríngeo, procesando todos los cuadros del video en orden.

1.4.1. Reconocimiento de potenciales regiones glotales con descriptores de Fourier

Se comienza trabajando sobre el primer cuadro del video. Se obtiene una copia en escala de grises del cuadro original que luego se convierte a una imagen binaria aplicando un umbral. Aquellos pixeles cuya intensidad de gris sea menor o igual al umbral tendrán un valor de 1, mientras que aquellos con intensidad mayor tendrán un valor de 0. El objetivo de esto es encontrar regiones que podrían corresponder a la glotis, aprovechando que ésta generalmente tiene una intensidad de gris menor que el área que la rodea. El valor del umbral no puede ser fijo, ya que las condiciones de iluminación de cada video serán distintas, por lo que se probará iterativamente con umbrales entre 1 y 80. La elección de estos valores es empírica; simplemente se encontró que los pixeles de la glotis generalmente tienen intensidades de gris menores a 80.



(a) Ejemplo de un cuadro del video

(b) Imagen binaria resultante

Figura 1.5: Aplicación de umbral de intensidad igual a 40 y apertura morfológica. El contorno de uno de los objetos detectados está marcado en rojo. Todos los objetos serán analizados por separado.

En cada iteración del umbral se aplica una apertura morfológica a la imagen binaria resultante para reducir el ruido y eliminar sobre-segmentaciones. Un ejemplo del resultado se muestra en la figura 1.5. Luego se hace un análisis de componentes conectados; se calculan los descriptores de Fourier del contorno de cada objeto detectado y luego se procesan y comparan con los descriptores de las glotis de entrenamiento como se describió en la sección 1.3.1. Recordemos que para comparar los descriptores de Fourier de dos contornos se utiliza la norma cuadrada de la diferencia:

$$D(F_1, F_2) = \|F_1 - F_2\|^2 \tag{1.7}$$

El resultado de esta comparación lo llamaremos *disimilitud*. Para el contorno de cada objeto detectado en la imagen binaria, se calculará la disimilitud promedio del contorno con todos los contornos de las glotis de entrenamiento:

$$D_{mean}(F_i) = \frac{1}{M} \sum_{k=1}^{M} D(F_i, F_{trained,k}) \tag{1.8}$$

Mientras menor sea este valor de disimilitud, se considerará que el contorno en cuestión tiene mayores probabilidades de corresponder a la glotis. Si la disimilitud tiene un valor debajo de cierto umbral fijo, el contorno se marcará como una potencial glotis y se avanzará a la siguiente parte del algoritmo. Si no, se procederá a evaluar el resto de los objetos detectados. Si ninguno tiene una disimilitud menor al umbral fijo, se aumentará el umbral de intensidad de gris de la imagen en 1 y se repetirá el proceso hasta que se encuentre un contorno cuya disimilitud sea lo suficientemente baja. Si se pasa del umbral 80 y no se encuentra ningún objeto, no se marca ninguna glotis para ese cuadro y se procede a trabajar en el siguiente cuadro. El umbral fijo de disimilitud usado en esta parte es de 0.32, elegido empíricamente.

1.4.2. Modelo de contorno activo

Simplemente aplicar un umbral no es una técnica lo suficientemente refinada como para delinear bien el contorno de la glotis; lo más probable es que el contorno no cubra completamente la glotis y haya que ajustarlo. Eso siempre que el contorno correspon-

da efectivamente a la glotis, lo cual puede no ser el caso. Para ajustar el contorno se aplica un *modelo de contorno activo*, utilizando la implementación descrita en [29]; los detalles de la implementación se explican en el Anexo A. Los autores de este método dejaron públicamente disponible una implementación en MATLAB del contorno activo en [28]; este código se utilizó directamente para implementar el contorno activo durante el desarrollo de la memoria.

Una vez que el contorno converja o llegue al límite de iteraciones permitidas (350, valor empírico), el contorno resultante se vuelve a comparar con las glotis de entrenamiento, calculando su disimilitud promedio. Si su disimilitud sigue estando por debajo del umbral fijo, se avanza a la siguiente parte del algoritmo. Si no, se descarta y el algoritmo termina de trabajar con el cuadro actual, pasando al siguiente cuadro. Este paso es útil ya que si el contorno no corresponde a la glotis, lo más probable es que al intentar ajustarlo se deforme a tal punto que su disimilitud pase a estar por sobre el umbral y se descarte.

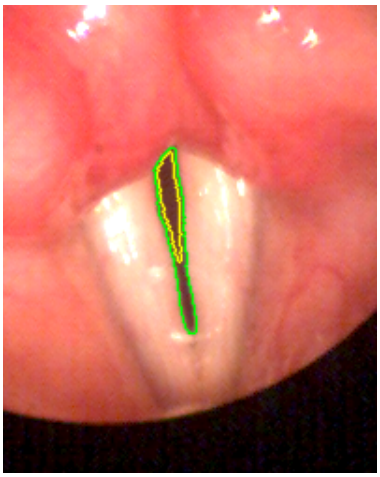


Figura 1.6: Ejemplo de aplicación de modelo de contorno activo. El contorno inicial se muestra en amarillo, el ajustado en verde.

1.4.3. Cálculo de GND

El GND de la potencial región glotal se calcula de la forma descrita en la sección 1.3.2, utilizando los coeficientes PCA guardados en la fase de entrenamiento. El GND de dos dimensiones resultante se mapea a la distribución también guardada previamente

para obtener un valor dentro de $[0, 1]$. Un valor alto indica que las diferencias de color entre pixeles al interior y exterior del contorno son similares a las que se encontraron en el set de entrenamiento, y por lo tanto es probable que el contorno en cuestión corresponda a la glotis. Se define un umbral para filtrar contornos no-glotales de 0.7; si el GND del contorno se mapea a un valor por sobre ese umbral, el contorno se confirma como correspondiente a la glotis y se guarda para su posterior uso en la siguiente parte de Segmentación. Junto con el contorno se guarda su disimilitud promedio calculada al comparar los descriptores de Fourier; este valor más adelante nos permitirá ordenar los contornos detectados según su similitud a las glotis de entrenamiento. Si el GND se mapea a un valor menor al umbral, el contorno se descarta. De cualquier forma, una vez terminada la evaluación se termina de trabajar en el cuadro actual y se pasa al siguiente.

Una vez que todos los cuadros del video hayan sido procesados, se avanza a la siguiente parte del método.

1.5. Segmentación

En esta parte es donde finalmente ocurre la segmentación como tal. Todos los contornos detectados en la parte anterior del método se ordenan según su disimilitud a las glotis de entrenamiento de menor a mayor. El primer contorno en la lista, es decir aquel con la menor disimilitud, es el que tiene las mejores probabilidades de corresponder a la glotis, y el cuadro al que pertenece es el punto de inicio óptimo para la segmentación cuadro a cuadro. Comenzando con este cuadro, se segmentarán los cuadros adyacentes del video en orden descendente hasta que se llegue a un cuadro donde no se detecte glotis, probablemente debido a que esta está cerrada. Al ocurrir esto se regresa al cuadro inicial y se procede a segmentar los cuadros adyacentes en dirección ascendente, hasta que nuevamente no se detecte glotis. Todos los cuadros visitados son marcados como segmentados. Luego se elige como nuevo cuadro inicial el siguiente cuadro de la lista que no haya sido visitado, y se segmentan los cuadros adyacentes de la misma forma. El algoritmo termina cuando todos los cuadros de la lista hayan sido visitados.

El cuadro elegido como cuadro inicial se considera como ya segmentado y su contorno de glotis queda como definitivo. Por lo tanto, el primer cuadro a procesar en esta parte es el cuadro previo al inicial, el que se encuentra a su “izquierda”. Desde ahora,

nos referiremos a este cuadro como el "actual", mientras que al cuadro inicial lo llamaremos "cuadro anterior", ya que fue el que se segmentó justo antes del cuadro actual. Una vez segmentado el cuadro, el siguiente cuadro a segmentar pasará a ser el nuevo cuadro actual, mientras que el cuadro segmentado pasará a ser el cuadro anterior.

1.5.1. Cálculo de ROI

En base al contorno segmentado del cuadro anterior se calculará una ROI con el objetivo de reducir el tiempo de cálculo y reducir sobre-segmentaciones hacia regiones no-glotales. Se hace el supuesto de que los desplazamientos de la glotis entre cuadros adyacentes es mínima, lo cual es válido para HSV. La ROI se calcula utilizando la *distancia de Mahalanobis* de cada pixel en el cuadro actual con los pixeles glotales del cuadro anterior. La distancia de Mahalanobis se puede entender como una medida de cuántas desviaciones estándar un punto se encuentra de cierta distribución. En este caso se busca medir qué tan lejos está cada punto de la imagen de la distribución formada por las coordenadas de los pixeles de la glotis. La distancia se calcula de la siguiente forma:

$$D_M(\vec{x}) = \sqrt{(\vec{x}-\vec{\mu})^T \cdot \Sigma^{-1} \cdot (\vec{x}-\vec{\mu})} \tag{1.9}$$

Donde $\vec{\mu}$ y Σ son la media y la matriz de covarianza de la distribución de todas las coordenadas de pixeles glotales en el cuadro anterior, y $\vec{x}$ es un pixel del cuadro actual. Se define a la ROI como aquellos puntos cuya distancia de Mahalanobis a los pixeles glotales del cuadro anterior sea menor a 4.5. A continuación se muestra una ROI de ejemplo:

Figura 1.7: Ejemplo de ROI calculada con la distancia de Mahalanobis

1.5.2. Imagen de Probabilidad

Se selecciona un número de puntos de referencia del contorno de la glotis del cuadro anterior que sean equidistantes entre sí. El número de puntos seleccionados fue 6, valor escogido empíricamente. Para cada punto de referencia se calculan dos histogramas tridimensionales ponderados por distancia para los valores rojo, verde y azul de los pixeles dentro de la ROI. El primer histograma corresponde a pixeles del cuadro anterior que están dentro de la glotis, mientras que el segundo histograma corresponde a pixeles fuera de la glotis. Una vez calculados, los histogramas se suavizan para obtener una distribución de color de los pixeles dentro y fuera de la glotis. A continuación se muestra un ejemplo de un histograma tridimensional que luego será suavizado, correspondiente a pixeles dentro de la glotis:

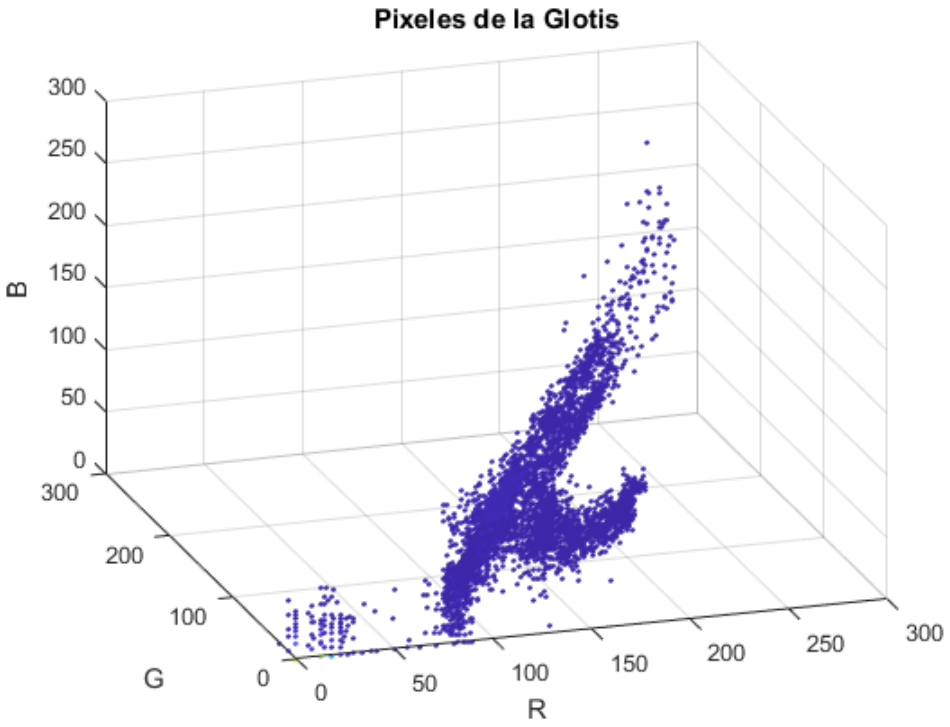


Figura 1.8: Histograma 3D de pixeles glotales, sin suavizar

La ponderación por distancia de cada pixel es muy similar a la que se hizo en la sección 1.3.2 para calcular el GND. Se utilizó la siguiente función de peso:

$$\omega(x_i, y_i) = e^{-\frac{(x_i - x_b)^2 + (y_i - y_b)^2}{2\sigma^2}} \tag{1.10}$$

Donde (x_b, y_b) son las coordenadas del punto de referencia respecto al cual se está calculando la distancia. El valor de σ utilizado en esta ponderación es de 40, elegido empíricamente.

Por un tema de tiempo de cálculo, los valores del histograma se mapearon a enteros entre $[0, 127]$. Para suavizar el histograma, éste se filtró utilizando un filtro gaussiano cúbico de tamaño 13 y $\sigma = 3$. El tamaño del filtro y de σ se eligieron empíricamente.

Los dos histogramas 3D resultantes representan la distribución de color ponderada por distancia en la vecindad de cada uno de los puntos del contorno de la glotis del cuadro anterior, tanto para la región glotal como para la región no-glotal. Se asume

que la distribución de color no cambia mucho de un cuadro a otro, por lo que estas distribuciones se usarán para generar una *imagen de probabilidad* para los pixeles en el cuadro actual. Para cada pixel del cuadro actual que esté dentro de la ROI se encontrará el punto de referencia más cercano y se mapearán los valores de color del pixel a los dos histogramas. Llamando $\vec{I}(x,y)$ a los colores del pixel, los dos valores resultantes serán interpretados como las probabilidades condicionales $P(\vec{I}|glotis)$ y $P(\vec{I}|fondo)$ (donde fondo se refiere a no-glotal), a partir de las cuales se obtendrá una probabilidad *a posteriori* de si el pixel pertenece a la glotis o no usando la regla de Bayes:

$$P_{glotis} = P(glotis|\vec{I}) \tag{1.11}$$

$$= \frac{P(\vec{I}|glotis) \cdot P(glotis)}{P(\vec{I}|glotis) \cdot P(glotis) + P(\vec{I}|fondo) \cdot P(fondo)} \tag{1.12}$$

A $P(glotis)$ y $P(fondo)$ se les asignó valores de 0.4 y 0.6 respectivamente. Las probabilidades resultantes para cada pixel de la ROI tendrán valores entre 0 y 1, por lo que serán multiplicadas por 255 para formar una imagen de probabilidad. Los pixeles que se encuentren dentro de la glotis en el cuadro actual tendrán valores altos, mientras que los que se encuentran fuera tenderán valores bajos. En la figura 1.9 se muestra un ejemplo. Si se observa de cerca la imagen de probabilidad se nota algo de ruido alrededor de la parte blanca; el resultado no es simplemente una imagen binaria.

(a) Cuadro actual (b) Imagen de probabilidad

Figura 1.9: Cálculo de Imagen de probabilidad

1.5.3. Segmentación con contorno activo

Se utiliza un modelo de contorno activo para obtener el contorno de la glotis a partir de la imagen de probabilidad. La implementación del contorno usada es la descrita en [7]. Igual que en la sección 1.4.2, los detalles de la implementación están fuera del alcance de esta memoria, pero se detallan en el Anexo B. Se utilizó código disponible en [30] para implementarlo.

La región inicial del contorno activo se obtiene aplicando un umbral de intensidad de pixeles a la imagen de probabilidad de 220. El resultado de la segmentación es un contorno que puede ser convertido a una imagen binaria.

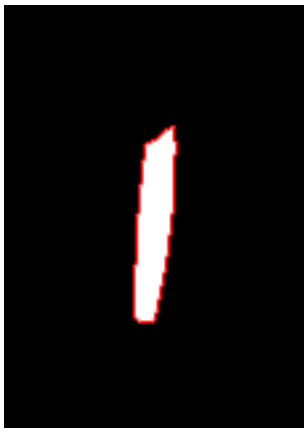


Figura 1.10: Imagen binaria resultante de la segmentación con contorno activo. El contorno está marcado en rojo

1.5.4. Eliminación de regiones no-glotales

La generación de la imagen de probabilidad en algunos casos puede causar que se iluminen regiones no pertenecientes a la glotis, causando que al segmentar con el contorno activo se incluyan regiones no-glotales que deben ser eliminadas. Es frecuente que estas regiones se encuentren fuera de lo que el paper llama la *región glotal rectangular*, definida por el eje principal de la glotis y su ancho. Para encontrar esta región se utiliza *análisis de componentes principales* (PCA) sobre las coordenadas de todos los pixeles ubicados dentro de la glotis en el cuadro anterior. La primera componente será

un vector cuya orientación coincidirá con el eje principal de la glotis, mientras que la segunda componente será un vector perpendicular al primero, es decir el eje normal a la glotis. Se proyectan todas las coordenadas de los pixeles glotales del cuadro anterior sobre la segunda componente principal (eje normal). Estas proyecciones forman el área glotal rectangular: mientras más lejos un pixel se encuentre del eje principal de la glotis, mayor será su proyección sobre el eje normal. Aquellos objetos segmentados en el cuadro actual que se encuentren completamente fuera de la región glotal rectangular (es decir que no se crucen con las proyecciones de los pixeles glotales del cuadro anterior) serán eliminados, ya que es casi seguro que corresponden a regiones no-glotales.

1.5.5. Elección de nuevo cuadro a segmentar

Con el paso anterior termina la segmentación del cuadro actual, y el siguiente paso es elegir un nuevo cuadro a segmentar, lo cual se hace de la forma ya explicada en la sección 1.5. Se elige un cuadro inicial y se segmenta en dirección descendente hasta no encontrar glotis, luego se vuelva al cuadro inicial y se segmenta en dirección ascendente hasta nuevamente no encontrar glotis. La no detección de una glotis en un cuadro nuevo ocurre porque no hubieron regiones lo suficientemente iluminadas en la imagen de probabilidad como para inicializar el contorno activo. Posteriormente se elige un nuevo cuadro inicial de la lista que no haya sido segmentado y se repite el proceso hasta que todos los cuadros de la lista hayan sido segmentados. De esta forma idealmente se habrá segmentado el video entero.

Capítulo 2

Modificaciones y mejoras al algoritmo original

En este capítulo se describen las modificaciones que se hicieron al algoritmo. Todas ellas buscan mejorar el rendimiento, excepto la versión para imágenes en escala de grises que se hizo para cumplir uno de los objetivos planteados. Si bien se hicieron muchas modificaciones y todas ellas se detallan en esta sección, hay dos que son más importantes que el resto: el cálculo de ROI con imagen de varianza (sección 2.5) y la versión para imágenes en escala de grises (sección 2.6). Para estas dos modificaciones hay apartados separados en la sección de resultados.

2.1. Cambios en la comparación de descriptores de Fourier

En la sección 1.4.1 se explica que el primer paso del algoritmo consiste en aplicar umbrales incrementales entre 1 y 80, y comparar los descriptores de Fourier de cada objeto detectado con los contornos de entrenamiento. Cuando se obtenga un contorno cuya disimilitud esté por debajo de cierto umbral fijo, el contorno se marca como una potencial glotis y se avanza a la siguiente parte del algoritmo.

Aquí se hicieron varios cambios. Primero, se decidió ignorar aquellos objetos cuyo contorno fuera demasiado pequeño. Es común que al aplicar un umbral a la imagen aparezcan muchos objetos pequeños que desde el punto de vista del algoritmo son sólo ruido, y filtrándolos se ahorra todo el tiempo de cálculo gastado al tener que procesar cada uno de ellos. Se filtró aquellos contornos cuyo largo fuera menor a 30, lo cual es bastante más pequeño que el tamaño usual de las glotis. También se ignoró aquellos objetos que se encuentren al borde de la imagen. Este filtro es muy útil ya que la glotis siempre se encuentra cerca del centro de la imagen, o al menos así es en todos los videos que se tienen disponibles. Esto se puede observar en la figura 2.1: Los puntos

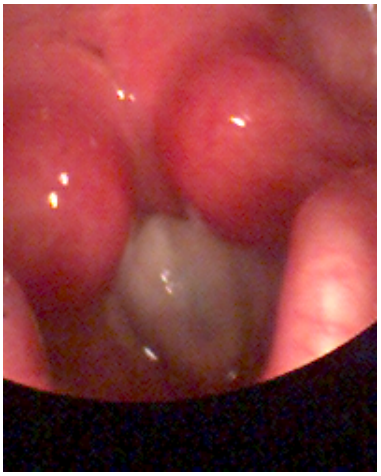
pequeños son eliminados, junto con la región blanca grande en la parte inferior de la imagen y también una más pequeña en el extremo superior izquierdo.

(a) Imagen original

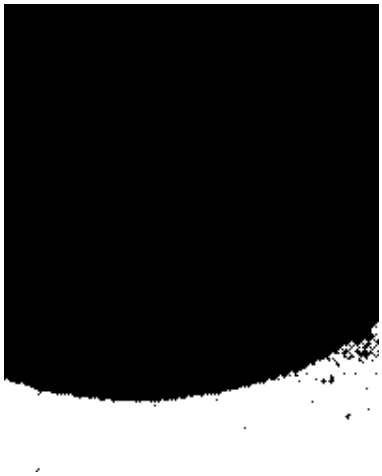
(b) Resultado de aplicar un umbral de intensidad

Figura 2.1: Eliminación de objetos pequeños y que toquen los bordes de la imagen

En la figura 2.1 también se observa que la región blanca inferior en la imagen binaria es en gran parte formada por la zona negra inferior que hay en la imagen original. También puede suceder que aparezcan objetos en esta zona que no toquen los bordes de la imagen. Para evitar procesar estos objetos, antes de empezar a aplicar los umbrales se intentará encontrar las regiones negras en los bodes de la imagen y eliminar cualquier objeto que está completa o parcialmente dentro de esa zona. Para encontrar las zonas negras se aplica un umbral de intensidad de gris fijo de 15 (empírico), tal que los pixeles cuya intensidad es menor al umbral se iluminen. Luego se analiza cada uno de los objetos detectados; aquellos que no tocan el borde de la imagen son eliminados. El resultado es una imagen binaria que marca sólo las zonas negras en los bordes de la imagen, como en la figura 2.2.



(a) Imagen original



(b) Resultado de aplicar umbral de intensidad igual a 15 y eliminar objetos que no tocan el borde de la imagen

Figura 2.2: Detección de zonas negras en los bordes de la imagen

Luego cuando se apliquen los umbrales incrementales, antes de analizar cada objeto detectado se calculará la intersección del objeto con esta imagen binaria que contiene a la región negra. Si la intersección no es vacía, eso significa que el objeto está por lo menos parcialmente dentro de la zona negra y por lo tanto se descarta.

Y por último, también se cambió la forma en que se detecta finalmente un potencial contorno de glotis. En el algoritmo original se avanza a la siguiente parte del algoritmo apenas se detecta un contorno cuya disimilitud sea menor a un umbral fijo. Aquí se decidió siempre aplicar todos los umbrales desde 1 a 80, y guardar el contorno que obtenga la menor disimilitud de todos los contornos analizados. Luego si ese contorno tiene una disimilitud menor al umbral fijo se avanza a la siguiente sección. Se encontró que este método es más robusto y produce menos falsos positivos.

2.2. Cambios en comparación de GND

En la sección 1.4.3 se explica que el GND calculado de un potencial contorno de glotis se reduce a dos dimensiones con PCA y luego se mapea a una distribución guar-

dada desde la fase de entrenamiento. La idea de esto es verificar que las diferencias de color entre los pixeles interiores y exteriores sea similar a las que hay en los contornos de entrenamiento. Sin embargo, se dan casos en los cuales la diferencia de colores es *mayor* a las que se tienen guardadas, ya que la glotis se ve muy oscura y el tejido circundante se ve muy iluminado. Esto debería hacer más fácil la segmentación, pero sucede lo contrario ya que un contorno así sería descartado por la comparación.

Para evitar esto, se intentó detectar casos en que la diferencia de intensidades entre los pixeles internos y externos es muy alta, y considerar esos casos como válidos. Cuando se hace la reducción del GND a dos dimensiones, la primera dimensión corresponde a una suma ponderada de cada uno de los 8 valores del GND. Esto se explica porque la primera columna de los coeficientes PCA son todos positivos, como se muestra en la tabla 2.1. Note que estos valores están asociados al set de videos con el que se entrenó.

Columna 1	**Columna 2**
0.2055	0.5606
0.3627	0.3118
0.4155	-0.0077
0.3723	-0.3740
0.2850	-0.4765
0.3490	-0.3161
0.4117	0.0777
0.3774	0.3399

Tabla 2.1: Coeficientes PCA

Al multiplicar los 8 valores del GND por la primera columna de los coeficientes PCA se obtiene una suma ponderada del GND. Y recordemos que cada uno de los 8 valores del GND representa las diferencias de color entre pixeles interiores y exteriores cercanos a cada punto base, por lo que la primera dimensión del GND reducido se puede entender como una estimación de la diferencia de color que hay entre los pixeles interiores y exteriores de toda la glotis. Luego si este valor es alto, se puede deducir que hay una alta diferencia de color entre los pixeles internos y externos y por lo tanto el contorno en cuestión será considerado como válido. El umbral para determinar si el valor es considerado como alto se definió como el mínimo valor de la primera compo-

nente con el cual es posible obtener un valor superior a 0.7 en la distribución de GND guardada. En la figura 2.3 se muestra un gráfico de la distribución que puede facilitar la visualización de este umbral.

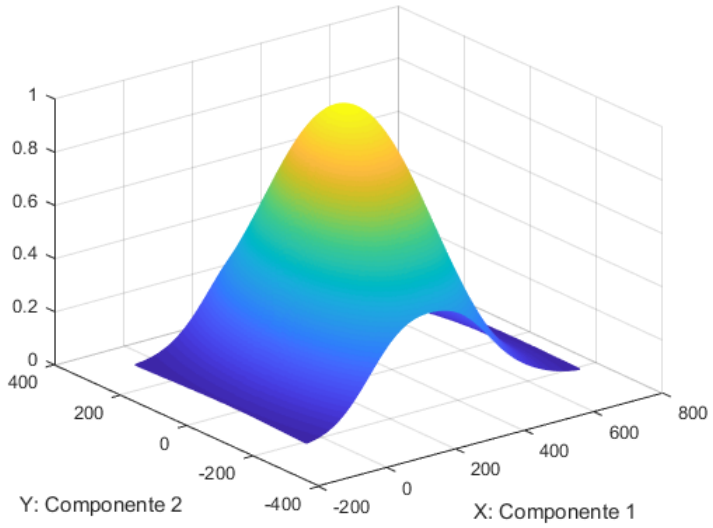


Figura 2.3: Distribución de GND reducido a dos dimensiones

2.3. Resolución de colisiones en segmentación cuadro a cuadro

En la sección 1.5 se explica que se hace una segmentación cuadro a cuadro primero en dirección descendente y luego en dirección ascendente a partir de un cuadro inicial. Es posible que en este proceso se encuentre un cuadro que ya ha sido segmentado, probablemente debido a un error en el algoritmo. Puede suceder, por ejemplo, que por alguna razón no se haya segmentado completamente un ciclo de la glotis y que luego se elija como nuevo cuadro inicial uno que pertenezca al mismo ciclo, provocando una colisión. O puede que se tome por error un cuadro donde la glotis esté cerrada como cuadro inicial. En ambos casos hay que resolver la colisión.

Lo que se hace es segmentar de todas formas el cuadro de forma normal, y una vez que se tenga el contorno final, compararlo con la segmentación ya existente. Si

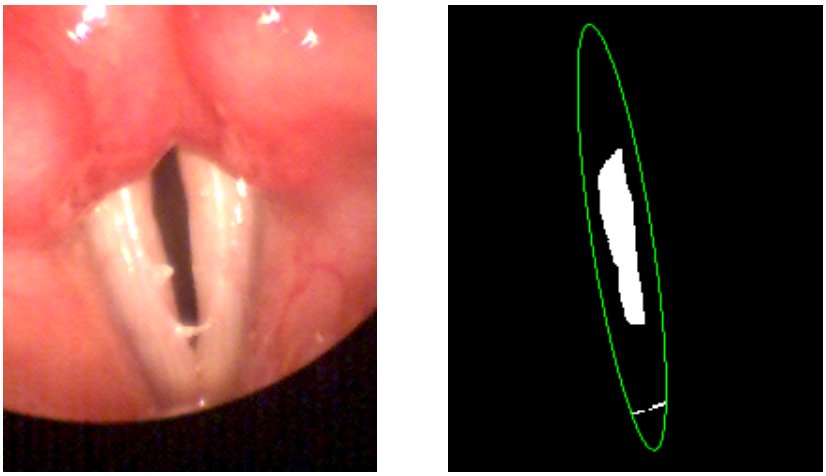
la intersección de las regiones formadas por los contornos no es vacía, se considerará que ambas segmentaciones son correctas. La segmentación final que se dejará para ese cuadro será la que se hizo primero; el siguiente cuadro a segmentar será el que se hubiera elegido si en el cuadro actual no se hubiera detectado glotis abierta. Si la intersección es vacía, las segmentaciones no calzan y por lo tanto al menos una de las dos es incorrecta. Se tomará como correcta la primera segmentación que se hizo, mientras que todos los contornos segmentados en la ronda actual serán descartados. Los cuadros visitados se dejarán marcados como tal para no segmentarlos de nuevo.

2.4. Criterios de eliminación de regiones no-glotales

En la sección 1.5.4 se explicó que se descartaban aquellos objetos segmentados cuya proyección sobre el eje normal de la glotis no se cruce con las proyecciones de los pixeles glotales del cuadro anterior. En la mayoría de los casos donde se segmentaron regiones no-glotales por error, este criterio de eliminación fue insuficiente y no sirvió como buen filtro. En la figura 2.4 se muestra una imagen de ejemplo con errores de segmentación que no hubieran sido eliminados con el criterio original. Por lo tanto el criterio se cambió y se agregaron varios otros criterios más estrictos para hacer más robusta la detección de errores. Los criterios establecidos son:

- En vez de eliminar objetos cuya proyección no se cruce con la proyección de la glotis del cuadro anterior, se eliminarán los objetos cuya proyección promedio esté muy lejos del eje central de la glotis. La distancia máxima se definió como un 35 % del ancho de la glotis del cuadro anterior. Este criterio es mucho más estricto y efectivo para filtrar segmentaciones erróneas que el original.
- Se filtran objetos demasiado pequeños cuyo borde tiene largo menor a 10.
- Se filtran objetos cuyo ancho es más que el doble de su altura, ya que la glotis en general es alargada y no ancha. El ancho se obtiene proyectando las coordenadas del objeto sobre el eje normal de la glotis y calculando el intervalo sobre el que caen las proyecciones. La altura se calcula igual pero proyectando sobre el eje principal de la glotis.

- Se filtran objetos cuyo ancho sea mayor a su alto **y** cuyo ancho sea mayor al ancho de la glotis del cuadro anterior. Este criterio es similar al anterior; elimina objetos cuya forma es demasiado distinta a la de una glotis normal.
- Se filtran objetos que toquen el borde de la ROI. La ROI da más que suficiente espacio para capturar la apertura gradual de la glotis cuadro a cuadro, por lo que si un objeto toca la ROI es casi seguro que es un error de segmentación.



(a) Cuadro actual

(b) Resultado de la segmentación con contorno activo. La ROI se muestra en verde.

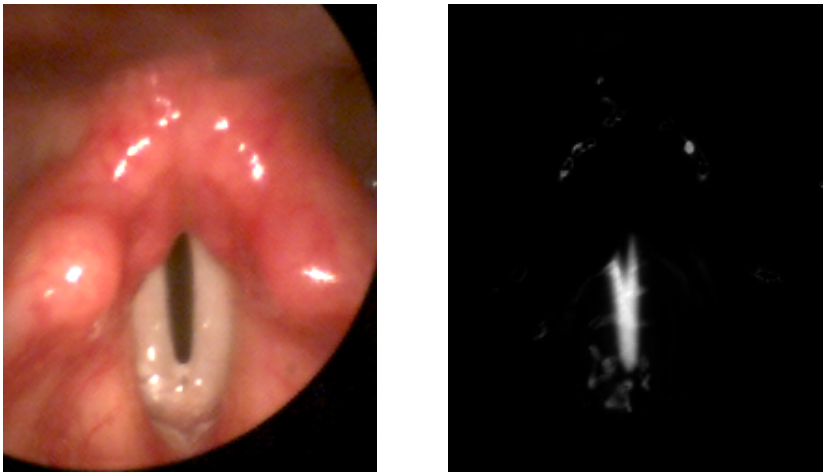
Figura 2.4: En la imagen (b), los pequeños objetos blancos en la parte inferior de la imagen son errores de segmentación que deben ser eliminados. El objeto de la izquierda se elimina por ser muy pequeño, mientras que el más grande de la derecha se elimina por colisionar con la ROI y por ser demasiado ancho.

2.5. Cálculo de ROI inicial con imagen de varianza

Cuando una persona observa un video de las cuerdas vocales, una de las cosas que más llama la atención y que ayuda a identificar fácilmente la ubicación de la glotis es que ésta se está abriendo y cerrando a lo largo del video. Es decir, los pixeles que abarca la glotis cambian su intensidad considerablemente durante el video. El algoritmo original no toma en cuenta esta información, ni tampoco lo hacen la mayoría de los otros

algoritmos estudiados. Esta extensión al algoritmo busca aprovechar esta información con el objetivo de localizar rápidamente el área de interés donde se encuentra la glotis y reducir detecciones erróneas.

El procedimiento es el siguiente: Como primer paso del algoritmo, antes de lo que se describe en la sección 1.4.1, se tomarán los primeros N cuadros del video, donde N será igual a 100 o igual al largo del video si éste tiene menos de 100 cuadros. Luego para cada pixel del video se calculará un arreglo de largo N que contendrá las intensidades de gris que ese pixel toma durante los N cuadros seleccionados. Se calculará la varianza de cada uno de esos arreglos. Una vez calculadas todas, se normalizarán sus valores para que queden dentro del intervalo $[0, 255]$ y se obtendrá una *imagen de varianza* a partir de ellas. El valor de cada pixel en esta imagen será igual a la varianza del arreglo normalizada correspondiente a cada pixel. En la figura 2.5 se muestra un ejemplo del resultado.

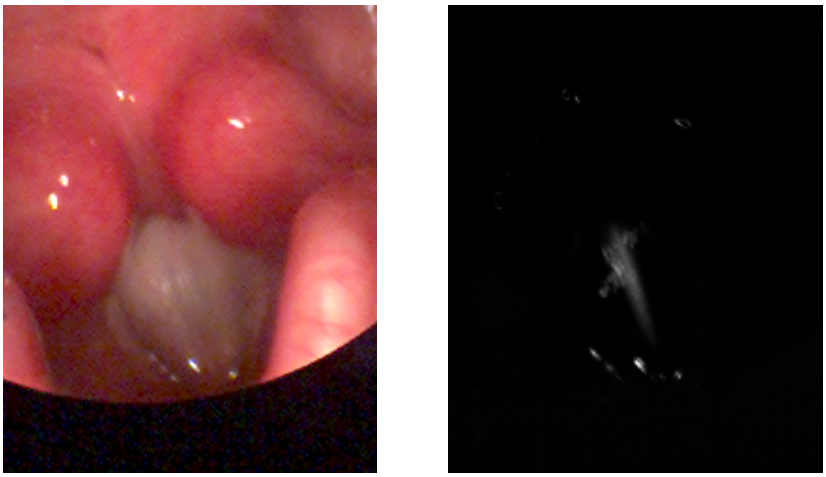


(a) Cuadro de ejemplo del video (b) Imagen de varianza

Figura 2.5: Ejemplo de imagen de varianza

La zona correspondiente a la glotis siempre tendrá varianzas distintas de cero ya que sus pixeles cambian de intensidad a lo largo del video. A partir de la imagen de varianza resultante es fácil extraer la región más iluminada y calcular una ROI. Sin embargo, este procedimiento como se ha descrito hasta ahora no sirve para todos los videos, ya que en algunos casos hay otras regiones aparte de la glotis que pueden tener varianzas incluso mayores, y por lo tanto al normalizar el resultado la glotis no quedará

bien iluminada. Generalmente esto es causado por reflejos de la luz del laringoscopio y es bastante común en los videos, como se muestra en la figura 2.6



(a) Cuadro de ejemplo del video (b) Imagen de varianza

Figura 2.6: En este caso la glotis no quedó iluminada en la imagen de varianza debido a un reflejo que causó que los pixeles afectados tuvieran más varianza que los pixeles glotales

Para solucionar esto se aplicaron dos restricciones al calcular la imagen de varianza. Primero, si el valor mínimo alcanzado por un pixel en los N cuadros multiplicado por 1.3 es mayor a la intensidad promedio del primer cuadro del video, ese pixel tendrá valor 0 en la imagen de varianza. Y segundo, si el valor máximo alcanzado por un pixel en los N cuadros es mayor a 200, también ese pixel tendrá valor 0 en la imagen de varianza. Es decir:

$$V(x,y) = \begin{cases} 0 & 1{,}3 \cdot Min(P(x,y)) > \bar{I} \\ 0 & Max(P(x,y)) > 200 \\ Var(P(x,y)) & e.o.c. \end{cases} \tag{2.1}$$

Donde $V(x,y)$ es la intensidad de un pixel en la imagen de varianza, $P(x,y)$ es el arreglo con las intensidades que toma dicho pixel en los N cuadros y $\bar{I}$ es la intensidad promedio del primer cuadro del video sin tomar en cuenta las áreas negras.Todos los umbrales son empíricos.

La idea de esto es encontrar aquellos pixeles afectados por reflejos y tirarlos a cero en la imagen de varianza. En la figura 2.7 se muestra una imagen de varianza calculada utilizando estas restricciones; note que ahora la glotis sí aparece iluminada.

Figura 2.7: Imagen de varianza calculada aplicando restricciones de valores mínimos y máximos a cada pixel. El video utilizado es el mismo que en la figure 2.6

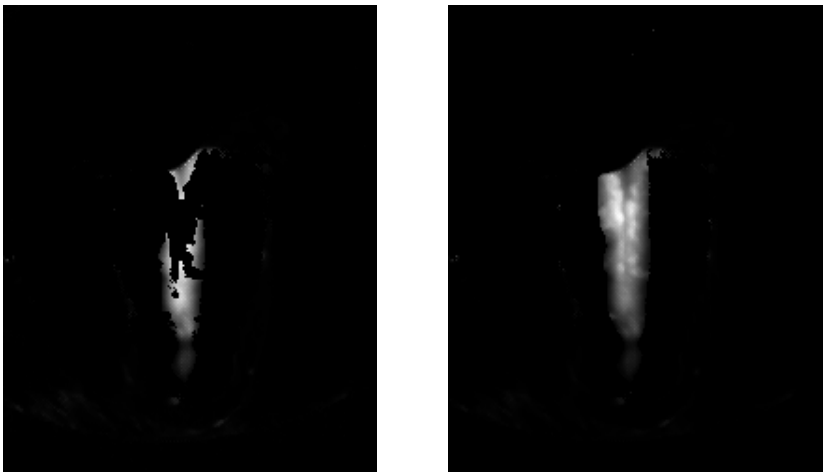
Se utilizó el primer cuadro del video para calcular la intensidad promedio porque se consideró que la intensidad promedio de los cuadros no varía demasiado entre distintos cuadros. Un detalle importante que no se ha mencionado es que esta intensidad promedio se calculó **sin tomar en cuenta las regiones negras en los costados del video**. Esto se hizo porque no todos los videos las tienen y afectan bastante la intensidad promedio de la imagen. En la sección 2.1 se explica cómo encontrar estas áreas negras.

Hasta aquí todo bien, pero todavía existen algunos casos donde la imagen de probabilidad no detecta bien la ubicación de la glotis; en particular cuando dentro del área abarcada por la glotis ocurren reflejos. Esto causa que los pixeles glotales que normalmente sí estarían iluminados tengan valor 0 debido a las restricciones impuestas, como se muestra en la figura 2.8.

(a) Cuadro de ejemplo del video (b) Imagen de varianza

Figura 2.8: En este caso la glotis no quedó completamente iluminada ya que reflexiones dentro del área glotal causaron que varios pixeles tuvieran valor cero. En la imagen (a) se pueden observar algunos de estos reflejos en el tejido que cubre la glotis.

Para solucionar esto, se calcularon dos versiones distintas de la imagen de varianza. Una es la que se acaba de explicar, es decir aplicando restricciones a los valores máximos y mínimos de intensidad de cada pixel. Para la segunda versión se aplicó sólo la restricción del valor mínimo. En la figura 2.9 se muestra el resultado.



(a) Imagen de varianza con ambas restricciones

(b) Imagen de varianza sólo con restricción de valor mínimo

Figura 2.9: Ambas versiones de la imagen de varianza. En la segunda versión la glotis queda completamente iluminada. El video utilizado es el mismo que para la figura 2.8.

Note que al aplicar sólo la restricción al valor mínimo se corrige el problema. Pero aplicar sólo esta restricción no es suficiente, ya que para algunos otros videos la segunda restricción es necesaria para filtrar correctamente las reflexiones. Por lo tanto lo que se hace es lo siguiente: Se aplica un umbral fijo de 30 a la primera versión de la imagen de varianza (calculada con ambas restricciones) seguido de una apertura morfológica para reducir el ruido. Luego se analizan todos los objetos detectados en la imagen binaria resultante y se guardará el objeto más grande que no toque la región negra en los costados de la imagen (ver sección 2.1) ni los bordes de la imagen. Este objeto corresponderá a la glotis en la mayoría de los videos, salvo en aquellos donde ocurran reflejos en la región glotal como en la figura 2.8. Luego se aplicará el mismo umbral y apertura morfológica a la segunda versión de la imagen de varianza (calculada sólo con la restricción al valor mínimo) y se buscará un objeto cuya intersección con el objeto obtenido a partir de la primera imagen de varianza sea distinta de cero. El resultado se ve en la figura 2.10.

(a) Objeto obtenido a partir de imagen de varianza calculada con ambas restricciones

(b) Objeto obtenido a partir de imagen de varianza calculada sólo con la restricción de valor mínimo

Figura 2.10: Objetos obtenidos aplicando un umbral y apertura morfológica a las imágenes de varianza; el de la imagen (b) claramente es el correcto. El video utilizado es el mismo que para la figura 2.9.

Una vez encontrados ambos objetos, se tomará como correcto el más grande entre los dos. En el caso de la figura 2.10 este sería el de la imagen (b). Como paso final, se

calculará la envoltura convexa del objeto detectado, y el área delimitada por esta envoltura será considerada como el objeto final. Esto fue necesario ya que existen videos donde porciones de la glotis nunca se cierran y por lo tanto tienen bajos valores de varianza y no quedan iluminadas, pero en general estas áreas se pueden cubrir calculando la envoltura convexa. Al objeto final detectado lo llamaremos *glotis de varianza* de aquí en adelante.

Una vez encontrada la glotis de varianza se procede a calcular la ROI, que se definió como aquellos puntos cuya distancia de Mahalanobis respecto de los puntos del objeto sea menor a 5.5. El resultado se muestra en la figura 2.11.

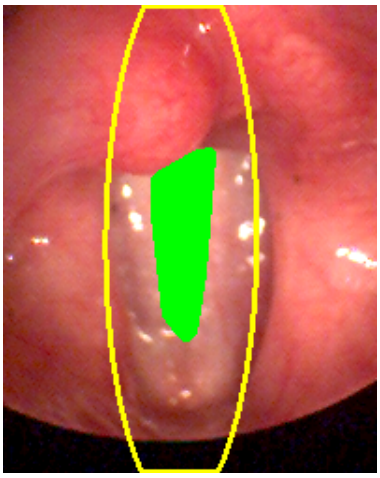


Figura 2.11: ROI en amarillo junto con la glotis de varianza en verde. El video utilizado es el mismo que para las últimas figuras.

La glotis de varianza junto con la ROI se utilizarán en la sección 1.4 del algoritmo, en particular durante la comparación con los descriptores de Fourier (sección 1.4.1). Antes de que se calcule el descriptor de Fourier de un objeto para compararlo con el set de entrenamiento, se calculará la intersección de ese objeto con la glotis de varianza. Si el objeto en cuestión no se encuentra completamente dentro de la ROI **o** si el área de la intersección es menor a un 60 % del tamaño del objeto, éste se descarta. La idea es eliminar aquellos objetos que no coincidan con la glotis de varianza. Esta restricción se suma a las que se describieron en la sección 2.1, que describe otras restricciones aplicadas en la misma parte del algoritmo.

También antes de aplicar el contorno activo como se describe en la sección 1.4.2 se

calculará una versión recortada de la imagen que consiste en el menor rectángulo que contiene a la ROI calculada. El contorno activo trabajará sobre esta versión recortada de la imagen, reduciendo el tiempo de cálculo y evitando posibles sobre-segmentaciones a otras regiones de la imagen. Una vez que converja el contorno activo, a las coordenadas del contorno resultante se les debe sumar un offset para que calcen con la imagen original.

Y finalmente, los umbrales utilizados en la comparación con descriptores de Fourier (sección 1.4.1) y comparación con GND (sección 1.4.3) se relajaron bastante; el primero aumentó de 0.32 a 6, mientras que el segundo bajó de 0.7 a 0.4. Esto significa que la importancia de la parte de machine learning en el algoritmo se ve reducida; en el algoritmo original la ubicación de la glotis se determinaba a través de machine learning haciendo comparaciones con el set de entrenamiento, pero ahora ese ya no es el caso. La ubicación de la glotis se encuentra a través del cálculo de la ROI con imagen de varianza, y la parte de machine learning queda como un filtro posterior para eliminar algunas falsas detecciones y ordenar los contornos detectados según su similitud al set de entrenamiento.

2.6. Versión para imágenes en escala de grises

Uno de los objetivos planteados para esta memoria es investigar la utilidad del uso del color del video en la segmentación de la glotis. Para cumplir este objetivo se desarrolló una versión alternativa del algoritmo que trabaja exclusivamente con videos compuestos por imágenes digitales en escala de grises.

La mayoría de las partes del algoritmo no cambian; se mantienen igual las secciones de cálculo y comparación de descriptores de Fourier, contornos activos, cálculo de ROI y eliminación de regiones no glotales. La estructura del algoritmo también sigue siendo la misma. En esta versión del algoritmo también se incluyeron todas las modificaciones ya descritas en este capítulo 2. Ninguna de ellas sufrió cambios tampoco.

Aquellas partes del algoritmo donde se hicieron cambios se detallan a continuación.

2.6.1. Pre-procesamiento

Antes de correr el algoritmo es necesario convertir los videos RGB a videos en escala de grises, tanto el video que se va a segmentar como los videos utilizados para el entrenamiento. Esta conversión se puede hacer de varias formas, pero las que se estudiaron en esta memoria fueron dos: La primera consiste en simplemente extraer uno de los tres canales de color del video que pueden ser el canal rojo, verde o azul, y trabajar con eso. La otra opción estudiada fue mezclar los tres canales del video para formar uno solo de la forma:

$$0{,}2989 \cdot R + 0{,}5870 \cdot G + 0{,}1140 \cdot B \tag{2.2}$$

Donde R es el canal rojo, G es el canal verde y B es el canal azul. Esta transformación es la misma que utiliza la recomendación UIT-R BT.601-7 sobre televisión digital, y la misma que utiliza el comando MATLAB *rgb2gray*.

El método se evaluó utilizando ambas conversiones, y los resultados se muestran de forma separada en la sección de resultados.

2.6.2. GND

Se requiere calcular GNDs en la fase de entrenamiento (sección 1.3.2) y cuando se evalúa un potencial contorno de glotis en la fase de reconocimiento (sección 1.4.3). Recordemos que para calcular el GND se toman 8 puntos base del contorno y para cada uno de ellos se calcula el vector de color medio ponderado por distancia de la forma:

$$\vec{V}_{mean,in} = \frac{\sum_{in} \omega(x_i, y_i) \cdot \vec{V}(x_i, y_i)}{\sum_{in} \omega(x_i, y_i)} \tag{2.3}$$

Donde $\vec{V}$ es el vector de color RGB del pixel y $\sum_{in}$ es una sumatoria sobre todos los puntos dentro de la glotis. Se hace el mismo cálculo para los puntos fuera de la glotis. Luego se calcula la norma de la diferencia entre los dos vectores, llamada *diferencia media de color local*:

$$DMCL = |\vec{V}_{mean,out} - \vec{V}_{mean,in}| \tag{2.4}$$

Luego de hacer esto para los 8 puntos base, el resultado es un vector de tamaño 1×8 que es el GND del contorno. Calcular el GND de esta forma ya no es posible con una imagen a escala de grises, por lo que hay que hacer adaptaciones. Primero, en vez del vector de color medio se calculará una intensidad de gris media ponderada por distancia de la forma:

$$I_{mean,in} = \frac{\sum_{in} \omega(x_i, y_i) \cdot I(x_i, y_i)}{\sum_{in} \omega(x_i, y_i)} \tag{2.5}$$

Y luego en vez de calcular la diferencia media de color local se calcula una *diferencia media de intensidad local*, que es el valor absoluto de la diferencia entre las dos intensidades medias:

$$DMIL = |I_{mean,out} - I_{mean,in}| \tag{2.6}$$

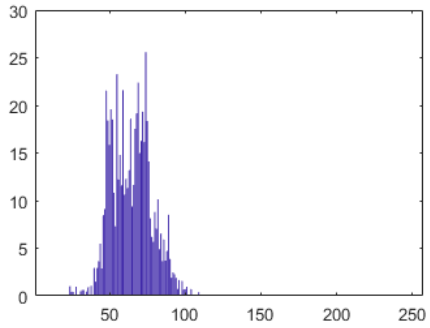
Luego de hacer esto para los 8 puntos base se obtiene el nuevo GND, del tamaño 1×8 que el original. La reducción de dimensiones con PCA y generación de una distribución de GND se hacen de la misma forma que en algoritmo original como se explicó en la sección 1.3.2.

2.6.3. Imagen de probabilidad

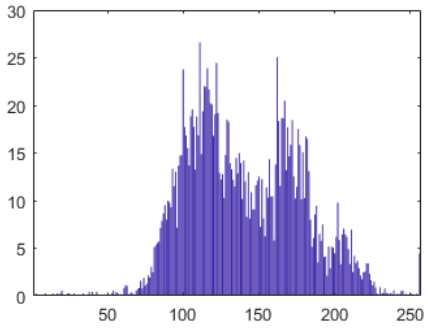
En la sección 1.5.2 se explica que se calcula una *imagen de probabilidad*, que representa la probabilidad de que cada pixel en el cuadro actual pertenezca a la glotis basada en sus propiedades de color. Se seleccionan 6 puntos de referencia sobre el contorno de la glotis del cuadro anterior y para cada uno de ellos se calculan dos histogramas tridimensionales ponderados por distancia para los valores rojo, verde y azul de los pixeles dentro de la ROI. El primer histograma corresponde a pixeles dentro de la glotis en el cuadro anterior, mientras que el segundo histograma corresponde a pixeles fuera de la

glotis en el cuadro anterior (pero dentro de la ROI). Luego los histogramas se suavizan para obtener distribuciones de color de los pixeles dentro y fuera de la glotis.

Ahora que las imágenes son a escala de gris no se pueden calcular histogramas tridimensionales de color. En vez de eso se calcularán histogramas unidimensionales de intensidad de gris como se muestra en la figura 2.12:



(a) Histograma de intensidad de gris de pixeles glotales



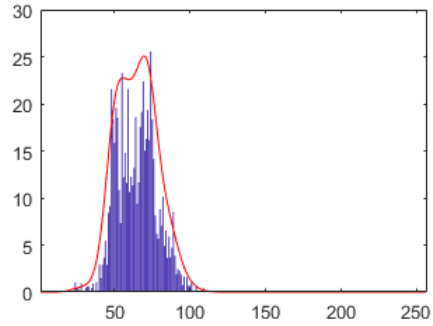
(b) Histograma de intensidad de gris de pixeles no glotales

Figura 2.12: Ejemplos de histogramas unidimensionales de intensidad de gris. Note que los pixeles glotales tienen en promedio menor intensidad de gris que los no glotales

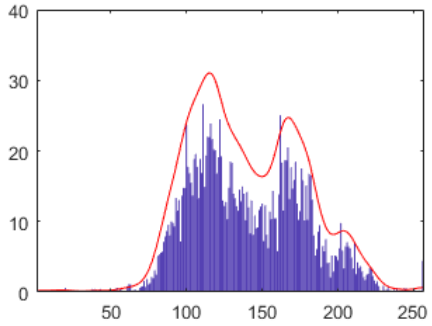
La ponderación por distancia es la misma que se detalló en la sección 1.5.2. Luego los histogramas serán suavizados de forma distinta a como se hizo anteriormente: se suavizará utilizando un kernel gaussiano. El procedimiento es el siguiente: Se inicializará el nuevo histograma suavizado en 0. Luego para cada bin (de 1 a 256) cuyo valor sea distinto de cero se sumará al histograma suavizado una función similar a una gaussiana centrada en el bin y ponderada por el valor del bin de la forma:

$$G(n) = H(i) \cdot e^{-\frac{(n-i)^2}{2\sigma^2}} \tag{2.7}$$

Donde i es el índice del bin, $H(i)$ es el valor del bin y σ es el ancho de banda del kernel y es igual a 5. Un ejemplo del resultado se muestra en la figura 2.13.



(a) Histograma suavizado de intensidad de gris de pixeles glotales



(b) Histograma suavizado de intensidad de gris de pixeles no glotales

Figura 2.13: Los mismos histogramas mostrados en la figura 2.12 pero con su versión suavizada dibujada en rojo encima

Estos histogramas se calculan para cada uno de los puntos de referencia del contorno. Esta vez ya que el tiempo de cálculo es mucho menor se eligieron 15 puntos de referencia en el contorno en vez de 6. Por la misma razón tampoco es necesario hacer cuantización de color. Habiendo calculado estos histogramas para todos los puntos de referencia del contorno se procede calcular las probabilidades a posteriori para los pixeles del cuadro actual. Para esto se utiliza la regla de Bayes:

$$P_{glotis} = P(glotis|I) \tag{2.8}$$

$$= \frac{P(I|glotis) \cdot P(glotis)}{P(I|glotis) \cdot P(glotis) + P(I|fondo) \cdot P(fondo)} \tag{2.9}$$

El procedimiento es el mismo, salvo que ahora $I(x,y)$ no es un vector si no un escalar que representa la intensidad de gris del pixel en cuestión. Utilizando estas probabilidades se calcula la imagen de probabilidad y el algoritmo puede continuar normalmente.

Capítulo 3

Evaluación cuantitativa de los algoritmos

Uno de los objetivos planteados al inicio de la memoria fue evaluar cuantitativamente el método y comparar su rendimiento con otros métodos existentes. Para poder evaluar un método objetivamente se requiere tener un punto de referencia con el cual comparar; en este caso se está evaluando la segmentación de la glotis en videos laringoscópicos, por lo que la referencia deben ser los contornos correctamente segmentados de la glotis en cada uno de los cuadros de los videos a segmentar. La forma más segura de obtener los contornos de la glotis es que un experto médico segmente la glotis a mano en cada uno de los cuadros, pero como se explicó en la introducción, la segmentación manual de la glotis en HSV es inviable ya que son demasiados cuadros.

Para ahorrar tiempo y trabajo se eligió la siguiente metodología: Primero, no se segmentarán todos lo videos sino que se escogerá un pequeño conjunto de ellos para ser usado como set de evaluación; estos videos serán excluidos de la fase de entrenamiento. Y segundo, no se segmentará el video entero sino que sólo 20 cuadros de cada uno distribuidos en distintas partes de distintos ciclos de apertura de la glotis, siempre en cuadros donde la glotis se encuentre abierta. Se encontró que evaluar estos 20 cuadros es más que suficiente para obtener una buena aproximación de cuán correcta es la segmentación producida por el algoritmo en todo el video. Se eligieron 10 videos para formar el set de evaluación, lo que da un total de 200 cuadros a segmentar a mano, un número perfectamente manejable. Todos los contornos segmentados fueron revisados por un experto médico para asegurarse de que fueran correctos.

Para hacer la comparación entre la referencia y el contorno a evaluar se utilizarán dos estadísticas: el *coeficiente Dice* [11] y el error de área. El coeficiente Dice se utiliza para medir la similitud entre dos muestras y para esta aplicación se define de la siguiente forma:

$$Dice = \frac{2 \cdot N(R_m \cap R_s)}{N(R_m) + N(R_s)} \tag{3.1}$$

Donde R_m es la región encerrada por el contorno correcto segmentado manualmente, R_s es la región encerrada por el contorno entregado por el algoritmo que se quiere evaluar y $N()$ representa el número de pixeles que contiene la región dentro de los paréntesis. Cuando los contornos son exactamente iguales el coeficiente será 1, si los contornos no se intersectan el coeficiente será 0, y si se intersectan parcialmente será un número intermedio entre 0 y 1. Mientras más alto sea el coeficiente, mayor será la similitud entre las regiones encerradas por los contornos.

El error de área es simplemente una comparación entre las áreas de la regiones encerradas por ambos contornos de la forma:

$$A_E = \frac{|A_m - A_s|}{A_m + A_s} \tag{3.2}$$

Donde A_m es el área de la región encerrada por el contorno correcto segmentado manualmente (número de pixeles) y A_s es el área de la región encerrada por el contorno a evaluar. Este indicador generalmente se comporta de forma inversa al coeficiente Dice; una segmentación perfecta arrojará un error de área 0, mientras que una segmentación errónea probablemente entregará valores bajos. Sin embargo este indicador no toma en cuenta la intersección de las áreas y por lo tanto es posible que un contorno erróneo con un área similar al correcto obtenga valores altos, aunque sea improbable. Este indicador es menos robusto que el coeficiente Dice, pero se incluye de todas formas como complemento.

Tomando en cuenta todo lo anterior, para evaluar la segmentación de un video se calculará el coeficiente Dice y el error de área de los contornos segmentados correspondientes a los 20 cuadros para los cuales se tenga disponible el contorno correcto. Se calculará el promedio, la mediana y la desviación estándar de los 20 resultados para cada indicador.

También es parte de los objetivos comparar el rendimiento del algoritmo implementado con el de otros algoritmos existentes. Para esto, como se mencionó en la sección de objetivos, se implementará el paper [3]: *"GlottalImageExplorer–An open source tool*

for glottis segmentation in endoscopic high-speed videos of the vocal folds" descrito brevemente en la sección 1.2 de la introducción. Se segmentarán los mismos videos del set de evaluación con este algoritmo y se calcularán los coeficientes Dice y el error de área de los contornos segmentados respecto de los contornos correctos.

Parte IV

Resultados, Discusión y Conclusiones

Evaluación de algoritmo original y mejoras

Primero se evaluaron tres versiones distintas del algoritmo: La primera corresponde al algoritmo original tal cual se describe en el paper [14], y aparece como "*Original*" en las tablas. La segunda versión incluye las mejoras descritas entre las secciones 2.1 y 2.4 de la parte de desarrollo del tema, en particular: Cambios en la comparación con descriptores de Fourier, cambios en la comparación con GND, resolución de colisiones y criterios de eliminación de regiones no-glotales. Aparece como "*Mod*" en las tablas. La tercera versión contiene todas las modificaciones propuestas incluyendo el cálculo de ROI inicial con imagen de varianza, y aparece como "*ROI*" en las tablas. Además, en línea con los objetivos de la memoria, se evalúa también el algoritmo propuesto por el paper [3] para comparar su rendimiento con el algoritmo desarrollado; aparece como "*GIE*" en las tablas. En esta evaluación, para el algoritmo GIE las imágenes en escala de gris fueron obtenidas aplicando la transformación descrita en la ecuación 2.2 a los tres canales de color.

La tabla 1 muestra el promedio ($\bar{x}$), mediana (Med) y desviación estándar (s) de los coeficientes Dice y errores de área (AE) calculados para cada uno de los videos. La primera columna de la tabla contiene los nombres de los videos; la nomenclatura usada es la siguiente:

- *F*, de *female*, indica que el video es de una paciente.
- *M*, de *male*, indica que el video es de un paciente.
- *N* indica que el paciente no está afectado por patologías.
- *P* y *D* indican que el paciente está afectado por patologías.
- *naso* indica que el video fue grabado durante una nasolaringoscopía utilizando un laringoscopio flexible. La calidad de estos videos generalmente es mala. Los videos que no están marcados con *naso* se grabaron con un laringoscopio rígido.
- *pre*, *lombard* y *adapt* indican que los videos se grabaron durante experimentos realizados para estudiar el efecto Lombard, los cuales consisten en grabar la voz del paciente mientras éste escucha un ruido ambiental fuerte ("*lombard*"), antes de que lo escuche ("*pre*") y después de que lo escuche ("*adapt*").

Tabla 1: Coeficientes Dice y errores de área del algoritmo original, su versión modificada, su versión con cálculo de ROI y del algoritmo GIE.

		Original		Mod		ROI		GIE	
		Dice	AE	Dice	AE	Dice	AE	Dice	AE
FN003	$\bar{x}$	0.8992	0.0905	0.8983	0.0893	0.9213	0.0541	0.9227	0.0388
	Med	0.9277	0.0654	0.9207	0.0631	0.9312	0.0431	0.9320	0.0249
	s	0.0640	0.0626	0.0689	0.0714	0.0466	0.0476	0.0365	0.0357
FN007	$\bar{x}$	0.8465	0.1520	0.8912	0.0970	0.9142	0.0738	0.8011	0.1926
	Med	0.9106	0.0974	0.9348	0.0535	0.9475	0.0471	0.9434	0.0530
	s	0.1250	0.1258	0.0938	0.0988	0.0806	0.0833	0.3458	0.3485
FP007	$\bar{x}$	0.8926	0.1665	0.7994	0.1947	0.7994	0.1947	0.9037	0.0859
	Med	0.8270	0.1708	0.8009	0.1991	0.8009	0.1991	0.9056	0.0849
	s	0.0511	0.0557	0.0536	0.0634	0.0536	0.0634	0.0215	0.0279
FP016	$\bar{x}$	0.8246	0.1456	0.7829	0.1950	0.8216	0.1463	0.7849	0.1689
	Med	0.8458	0.0883	0.8553	0.1024	0.8716	0.0857	0.8955	0.0779
	s	0.1221	0.1370	0.2212	0.2330	0.1297	0.1396	0.2807	0.2886
FN003 (naso)	$\bar{x}$	0	1	0	1	0.6741	0.2596	0.7797	0.1664
	Med	0	1	0	1	0.7761	0.1360	0.8354	0.1193
	s	0	0	0	0	0.2991	0.3335	0.2034	0.2257
FP005 (naso)	$\bar{x}$	0	1	0.1559	0.8297	0.7395	0.1878	0.8267	0.1140
	Med	0	1	0	1	0.7699	0.1480	0.8792	0.0707
	s	0	0	0.3224	0.3534	0.2071	0.2269	0.1209	0.1229
FP011 (naso)	$\bar{x}$	0	1	0	1	0.7046	0.2626	0.5845	0.3784
	Med	0	1	0	1	0.8671	0.1164	0.8114	0.1416
	s	0	0	0	0	0.3272	0.3404	0.3936	0.4195
FD003 (pre)	$\bar{x}$	0	0.4511	0.7120	0.2744	0.7210	0.2534	0.6905	0.2748
	Med	0	0.5192	0.8348	0.1517	0.8359	0.1235	0.8549	0.1193
	s	0	0.1868	0.2892	0.2992	0.2659	0.2757	0.3573	0.3743
FN003 (lombard)	$\bar{x}$	0.1220	0.8692	0.8860	0.0804	0.8833	0.0877	0.8924	0.0811
	Med	0	1	0.9353	0.0299	0.9215	0.0496	0.8952	0.0784
	s	0.3013	0.3209	0.1551	0.1603	0.1490	0.1560	0.0270	0.0343
MN003 (adapt)	$\bar{x}$	0.7620	0.0372	0.7113	0.2478	0.7650	0.1629	0.8870	0.0598
	Med	0.7602	0.0295	0.8745	0.0696	0.8674	0.0570	0.8979	0.0599
	s	0.1067	0.0323	0.3669	0.3885	0.2761	0.2934	0.0505	0.0369

Antes de comparar los resultados se deben hacer algunas observaciones respecto a cómo se comporta el promedio, la mediana y la desviación estándar en función de los resultados obtenidos. Lo que más llama la atención es que en aquellos casos donde la desviación estándar es alta (mayor a 0.2 aproximadamente) se observa una gran diferencia entre el promedio y la mediana, esta última indicando mejores resultados que el promedio. La explicación radica en que en los cuadros donde el algoritmo no detecta un contorno pero sí hay uno en el set de evaluación, el coeficiente Dice resultante es 0 (y el error de área es 1). Si esto ocurre sólo para unos pocos cuadros de la secuencia mientras que los demás están bien segmentados, el promedio de los indicadores bajará bastante, pero la mediana se mantendrá. Generalmente esto ocurre para cuadros que se encuentran hacia el final o comienzo del ciclo glotal donde el área de la glotis es pequeña y más difícil de detectar, o en cuadros pertenecientes a ciclos incompletos al comienzo o al final del video. Por esta razón, el indicador más confiable que mejor refleja el desempeño del algoritmo es la *Mediana*, ya que da una mejor idea de cómo resulta la segmentación en un cuadro cualquiera.

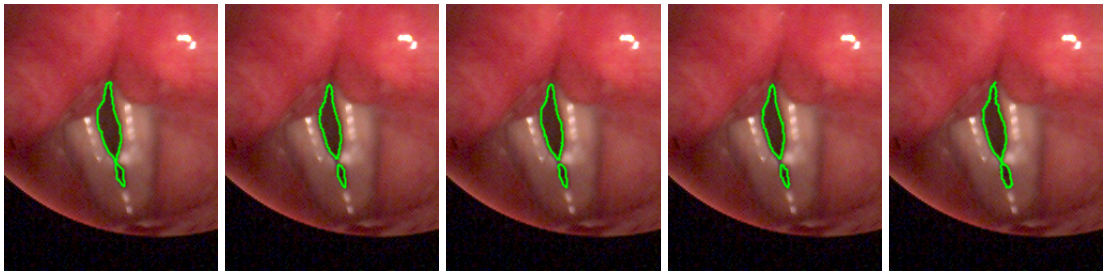
Ahora sí respecto de los resultados obtenidos, en promedio el algoritmo original sin ninguna mejora es el que presenta los peores resultados, ya que sólo funciona bien al segmentar los primeros 4 videos (FN003, FN007, FP007 y FP016). Falla completamente en los videos naso y del experimento lombard con la excepción de MN003 (adapt), pero incluso en ese caso los indicadores están por debajo de las demás versiones. La versión *Mod* del algoritmo que incluye algunas mejoras tiene resultados un poco mejores que el original. La mejora más notoria se da en los videos FD003 (pre) y FN003 (lombard) que logró segmentar correctamente a diferencia del algoritmo original. Sin embargo sigue fallando en los videos naso.

Al agregar el cálculo de ROI con imagen de varianza se observan aún más mejoras: a diferencia de las versiones Original y Mod del algoritmo, la versión ROI funciona con los videos naso y por lo tanto logra segmentar todo el set de evaluación. Sin embargo en algunos videos la segmentación no es perfecta ya que los coeficientes resultantes no son tan altos, como en el caso de FP005 (naso) cuya mediana de coeficientes Dice es 0.7820 lo cual indica algunos errores de segmentación. En comparación, el algoritmo GIE obtiene resultados bastante similares en general, siendo mejor en algunos videos como FP005(naso) y FP007, y peor en otros como FP011 (naso). En promedio se podría decir que el algoritmo GIE presenta resultados un poco mejores.

En cuanto a diferencias entre los videos, claramente los videos más complicados son los naso, ya que las versiones Original y Mod fallaron al segmentarlos y la versión ROI y el algoritmo GIE obtienen resultados peores en comparación a los demás videos. Estos videos son de calidad inferior a los demás y tienen problemas de ruido e iluminación, por lo que estos resultados son esperables. Por otro lado, los videos normales grabados con laringoscopios rígidos son los más fáciles (FN003, FN007, FP007 y FP016). Dentro de este subconjunto los videos patológicos llevan a resultados ligeramente peores que los normales en todos los algoritmos evaluados.

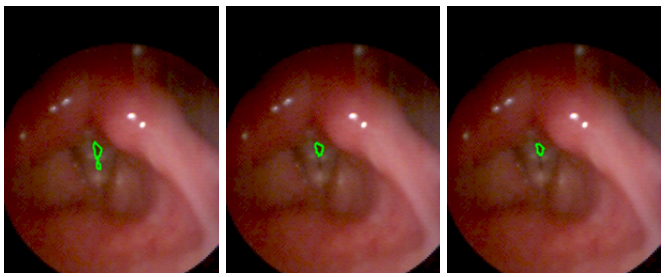
Sobre los indicadores utilizados, en general coinciden uno con el otro; un coeficiente Dice alto generalmente va acompañado de un error de área bajo. Pero existe una excepción en la evaluación del video FD003 (pre) con el algoritmo Original: en este caso el coeficiente Dice es cero mientras que el error de área es distinto de 1. La causa de esto es que el algoritmo segmentó una región errónea que no se intersecta con la región glotal, como se ve en la figura 3 (b) más adelante. Si tuviéramos sólo disponible el error de área como indicador, pensaríamos que la segmentación no estuvo completamente errónea, pero ese no es el caso. Debido a casos como este, el indicador más robusto entre los dos es el coeficiente Dice ya que toma en cuenta la ubicación de las regiones dentro de la imagen. El error de área no aporta ninguna información extra en comparación. Por lo tanto en las demás evaluaciones que se presenten se utilizará sólo el coeficiente Dice.

A continuación se muestra un reporte gráfico de la segmentación entregada por todas las versiones del algoritmo para algunos cuadros seleccionados de algunos videos, junto con el coeficiente Dice obtenido por cada algoritmo en el cuadro. También se muestra el contorno correcto como referencia.



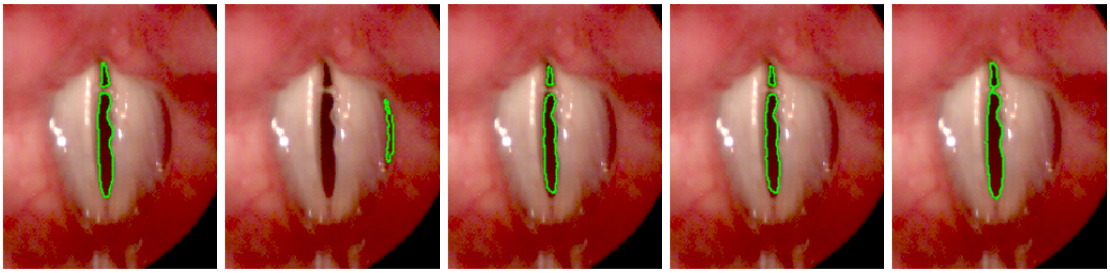
(a) Correcto (b) Original Dice: 0.9244 (c) Mod Dice: 0.9428 (d) ROI Dice: 0.9255 (e) GIE Dice: 0.9394

Figura 1: Cuadro 31 del video FN003. En este cuadro todos los videos obtuvieron un coeficiente alto lo cual es bueno, pero hay un detalle: Todos los algoritmos excepto el GIE separan el contorno detectado en dos partes, mientras que en la segmentación correcta las dos partes están unidas, formando un solo contorno.



(a) Correcto (b) ROI Dice: 0.7182 (c) GIE Dice: 0.6905

Figura 2: Cuadro 31 del video FN003 (naso). En este cuadro (y de hecho en todo el video) las versiones Original y Mod del algoritmo no lograron segmentar ningún contorno. La versión ROI y el algoritmo GIE sí entregaron un contorno, pero éstos no son completamente correctos ya que no cubren completamente la glotis. Note también que en este caso la calidad del video es menor y la glotis se ve mucho más pequeña.



(a) Correcto (b) Original Dice: 0 (c) Mod Dice: 0.8931 (d) ROI Dice: 0.9087 (e) GIE Dice: 0.9378

Figura 3: Cuadro 1 del video FD003 (pre). En este cuadro la mayoría de los algoritmos obtienen buenos resultados excepto el original que segmenta una región errónea. Esto causa que su coeficiente Dice sea cero, pero el error de área sea distinto de cero.

También se hizo una evaluación cualitativa del rendimiento del algoritmo utilizando una metodología de validación cruzada: uno de los videos es elegido para evaluar y todos los demás son utilizados para entrenar el algoritmo. Luego este proceso se repite para cada uno de los videos disponibles. Ya que en este caso no se tienen los contornos correctos para cada video, la evaluación se basa en qué tan bien se observa visualmente que el contorno entregado por el algoritmo se ajusta a la glotis. Los contornos entregados por el algoritmo se clasificarán en una de cuatro categorías dependiendo de qué tan bueno sea el resultado: la primera categoría (C1) corresponde a los contornos que resultaron ser correctos. La segunda (C2) corresponde a contornos que resultaron bien pero que tienen alguna falla menor como bordes un poco irregulares o sobre-segmentaciones pequeñas. La tercera categoría (C3) corresponde a contornos que presentan fallas significativas como sobre-segmentaciones hacia regiones adyacentes o que el contorno no cubra completamente la glotis. La última categoría (C4) corresponde a casos donde el contorno encierra una región completamente errónea que no corresponde a la glotis o la segmentación falló y no se entregó ningún contorno.

Es importante recalcar que los resultados presentados aquí son subjetivos y no están revisados por expertos médicos, por lo que deben analizarse con cuidado. La tabla 2 muestra cuántos resultados caen en cada categoría para cada algoritmo, con los videos separados en normales, naso y lombard.

Tabla 2: Resultados cualitativos de los algoritmos desarrollados

	Original				Mod				ROI				GIE			
	C1	C2	C3	C4	C1	C2	C3	C4	C1	C2	C3	C4	C1	C2	C3	C4
Videos normales	12	8	6	3	15	5	7	2	15	6	7	1	17	12	0	0
Videos naso	1	1	2	23	1	4	1	21	16	5	3	3	19	5	2	1
Videos lombard	10	2	8	5	12	5	6	2	14	2	7	2	18	4	2	1

Estos resultados confirman algunas observaciones hechas respecto de la tabla 1: La versión Mod presenta mejoras en los videos normales y lombard, pero sigue fallando en los videos naso, mientras que la versión ROI logra segmentar la mayoría de los videos naso correctamente. Sin embargo, también se observa que hay una cantidad significativa de videos para los cuales la versión ROI falla (categorías C3 y C4), indicando que el algoritmo no es perfecto. Respecto a la versión GIE, ahora se nota claramente que entrega mejores resultados que la versión ROI.

Evaluación de versión para imágenes en escala de gris

Aquí se muestran los resultados de la evaluación de la versión para imágenes en escala de gris del algoritmo, la cual se subdivide en 4 ”sub-versiones”, dependiendo de cómo se hayan obtenido las imágenes a escala de gris: extrayendo el canal rojo (“*R ch.*”), canal verde (“*G ch.*”), canal azul (“*B ch.*”) o aplicando la transformación descrita en la ecuación 2.2 a los tres canales (“*rgb2gray*”). El algoritmo GIE aquí también se separó en 4 sub-versiones de la misma forma; note que la versión rgb2gray del algoritmo GIE es la misma que se utilizó en la tabla 1. Los resultados de la versión ROI del algoritmo también se muestran para comparar.

Tabla 3: Coeficientes Dice de la versión con cálculo de ROI, versiones para imágenes en escala de gris y del algoritmo GIE.

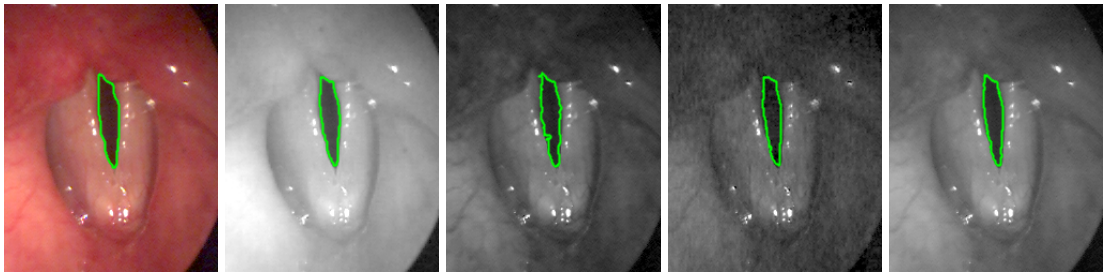
		ROI	Escala de grises				GIE			
			R ch.	G ch.	B ch.	rgb2gray	R ch.	G ch.	B ch.	rgb2gray
FN003	$\bar{x}$	0.9213	0.8603	0.5951	0.5167	0.8900	0.8878	0.8393	0.8123	0.9227
	Med	0.9312	0.9111	0.8014	0.5988	0.9258	0.8887	0.8676	0.9002	0.9320
	s	0.0466	0.1028	0.3574	0.3817	0.0838	0.0354	0.1256	0.2798	0.0365
FN007	$\bar{x}$	0.9142	0.8891	0.8801	0.8053	0.8997	0.9266	0.8861	0.8791	0.8011
	Med	0.9475	0.9365	0.9167	0.8792	0.9390	0.9362	0.9441	0.9478	0.9434
	s	0.0806	0.1001	0.0812	0.2210	0.0926	0.0301	0.2110	0.2176	0.3458
FP007	$\bar{x}$	0.7994	0	0.8247	0.7867	0.8467	0.9219	0.8899	0.6307	0.9037
	Med	0.8009	0	0.8238	0.7822	0.8695	0.9272	0.8912	0.8855	0.9056
	s	0.0536	0	0.0400	0.0648	0.0685	0.0299	0.0311	0.4245	0.0215
FP016	$\bar{x}$	0.8216	0.7328	0.6803	0.6818	0.7261	0.8013	0.5766	0.6644	0.7849
	Med	0.8716	0.8142	0.7404	0.7326	0.7905	0.8965	0.8546	0.8842	0.8955
	s	0.1297	0.2423	0.2525	0.2097	0.2152	0.2770	0.4158	0.3851	0.2807
FN003 (naso)	$\bar{x}$	0.6741	0.4161	0	0	0.4876	0.7971	0.1485	0.6445	0.7797
	Med	0.7761	0.6142	0	0	0.6913	0.8456	0	0.7440	0.8354
	s	0.2991	0.3920	0	0	0.3879	0.1969	0.3063	0.2565	0.2034
FP005 (naso)	$\bar{x}$	0.7395	0.6055	0	0.7010	0.6515	0.7813	0	0.6333	0.8267
	Med	0.7699	0.6977	0	0.7239	0.6731	0.8271	0	0.7602	0.8792
	s	0.2071	0.2744	0	0.1513	0.1177	0.1222	0	0.3157	0.1209
FP011 (naso)	$\bar{x}$	0.7046	0.5943	0.6186	0.4803	0.6817	0.8149	0.5170	0	0.5845
	Med	0.8671	0.8185	0.7161	0.5799	0.8176	0.8214	0.7473	0	0.8114
	s	0.3272	0.4015	0.2717	0.3144	0.2817	0.0506	0.3914	0	0.3936
FD003 (pre)	$\bar{x}$	0.7210	0.7518	0.5120	0.5553	0.6250	0.6908	0.4862	0.7056	0.6905
	Med	0.8359	0.8884	0.5482	0.6310	0.8227	0.8650	0.8172	0.8745	0.8549
	s	0.2659	0.2665	0.3694	0.3190	0.3647	0.3585	0.4524	0.3663	0.3573
FN003 (lombard)	$\bar{x}$	0.8833	0.8691	0	0	0.7851	0.8766	0.4410	0.4638	0.8924
	Med	0.9215	0.9188	0	0	0.8942	0.9161	0.4253	0.4485	0.8952
	s	0.1490	0.1750	0	0	0.2789	0.1394	0.4526	0.4760	0.0270
MN003 (adapt)	$\bar{x}$	0.7650	0.8297	0	0	0.5311	0.9306	0.3752	0.6637	0.8870
	Med	0.8674	0.8710	0	0	0.7008	0.9434	0.3128	0.8575	0.8979
	s	0.2671	0.1997	0	0	0.4113	0.0486	0.3882	0.3959	0.0505

De las 4 sub-versiones del algoritmo para imágenes en escala de gris, las que entregan los mejores resultados son la *R ch.* y *rgb2gray*, teniendo esta última resultados un poco más consistentes ya que la versión R ch. falla en el video FP007. Las versiones *G ch.* y *B ch.* entregan resultados peores en todos los videos en comparación a las otras dos versiones, con la excepción del video FP007. Para el algoritmo GIE también ocu-

rre lo mismo; las versiones R ch. y rgb2gray entregan los mejores resultados mientras que las versiones G ch. y B ch. resultan peor. Esto nos lleva a pensar que la mayor parte de la información se encuentra en el canal rojo, ya que al ignorarlo los resultados empeoran significativamente.

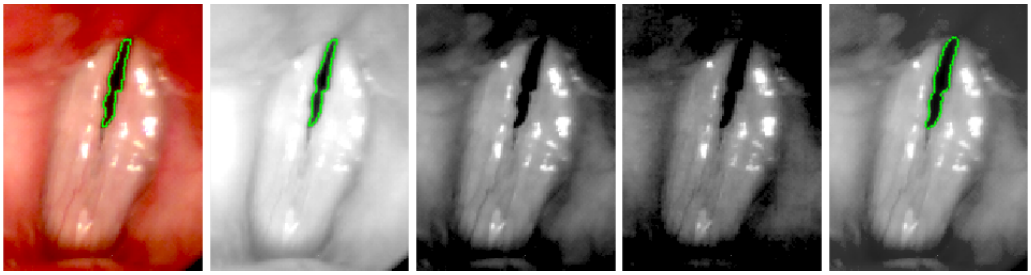
Se observa también que en promedio los coeficientes Dice de las versiones R ch. y rgb2gray son más bajos que la versión ROI del algoritmo, que es la versión en la que se basa la adaptación para imágenes en escala de grises, particularmente en los videos naso. Esto nos lleva a pensar que al eliminar el color de los videos y del algoritmo se está perdiendo información relevante y empeoran los resultados de la segmentación.

A continuación se muestra un reporte gráfico de la segmentación entregada por las versiones para videos en escala de grises del algoritmo, similar a como se hizo antes. En este caso los cuadros que se muestran para cada algoritmo están convertidos a escala de grises de acuerdo a cada algoritmo (i.e. para la versión R ch. se muestra el canal rojo del cuadro).



(a) Correcto | (b) R ch. Dice: 0.9388 | (c) G ch. Dice: 0.9370 | (d) B ch. Dice: 0.9054 | (e) rgb2gray Dice: 0.9375

Figura 4: Cuadro 1 del video FN007. Todas las versiones del algoritmo obtienen buenos resultados en este cuadro, aunque se observan algunas imperfecciones en el caso de G ch. que no alcanzan a verse reflejadas en el coeficiente Dice. Note también que los canales verde y azul del cuadro se ven más oscuros y con menor calidad que los demás.



(a) Correcto (b) R ch. Dice: 0.9598 (c) G ch. Dice: 0 (d) B ch. Dice: 0 (e) rgb2gray Dice: 0.8908

Figura 5: Cuadro 5 del video FN003 (lombard). En este caso las versiones G ch. y B ch. no logran encontrar el contorno. Note que en esos casos no se alcanza a ver el borde superior de la glotis debido a que la imagen es muy oscura, lo cual probablemente causa que el algoritmo no funcione.

Igual que para la evaluación de las modificaciones al algoritmo original, se hizo una evaluación cualitativa de las versiones para imágenes en escala de grises del algoritmo. Las categorías utilizadas son la mismas que para la tabla 2.

Tabla 4: Resultados cualitativos de los algoritmos desarrollados para imágenes en escala de grises

	R ch.				G ch.				B ch.				rgb2gray			
	C1	C2	C3	C4	C1	C2	C3	C4	C1	C2	C3	C4	C1	C2	C3	C4
Videos normales	5	9	6	9	2	6	13	8	7	4	13	5	10	9	9	1
Videos naso	6	6	1	14	1	4	7	15	0	9	10	8	4	6	15	2
Videos lombard	9	1	9	6	5	4	5	11	2	6	10	7	10	6	7	2

Estos resultados también confirman las observaciones hechas previamente, pero nuevamente con algunas salvedades. Efectivamente al trabajar con los canales azul y verde se obtienen los peores resultados y por lo tanto se podría decir que esos canales son los que contienen menos información. Pero en este caso los resultados obtenidos al trabajar con el canal rojo son peores en promedio que al trabajar con los canales combinados en un solo video en escala de grises (rgb2gray). Esto nos lleva a pensar que el canal rojo efectivamente es el que contiene más información, pero al ignorar los otros dos canales también se está perdiendo una cantidad importante de información.

Conclusiones

Sobre el algoritmo desarrollado

Se logró implementar el algoritmo propuesto en el paper [14] para la segmentación automática de la glotis. Sin embargo los resultados de la segmentación con este algoritmo no son particularmente buenos, ya que no logra segmentar videos grabados con laringoscopios flexibles (naso), debido a una alta sensibilidad al ruido y a problemas de iluminación en los videos. Aplicando las mejoras propuestas en esta memoria al algoritmo se logra que segmente la mayoría de los videos y mejore significativamente su rendimiento, pero se queda un poco corto del estado del arte representado por el algoritmo GIE [3]. La principal ventaja que tiene el algoritmo desarrollado sobre el algoritmo GIE es que es automático, mientras que el algoritmo GIE requiere de intervención del usuario para funcionar. La modificación que causa la mayor mejora en los resultados de la segmentación es el cálculo de ROI con imagen de varianza, ya que permite identificar con seguridad la región glotal al inicio del algoritmo de forma bastante insensible al ruido y cambios de iluminación, aprovechando la información entregada por las variaciones de las imágenes a lo largo del video.

Sobre la evaluación cuantitativa de los algoritmos

Para la evaluación de algoritmos teniendo los resultados correctos para comparar, el coeficiente Dice es un indicador efectivo. El error de área no es un indicador robusto ya que no toma en cuenta la ubicación espacial de las regiones y puede llevar a conclusiones erróneas.

Sobre el uso del color en el algoritmo

Al trabajar con videos en escala de grises y quitar todo el procesamiento de color del algoritmo los resultados si bien no son malos, son peores que al trabajar con videos a color. Por lo tanto se concluye que para este algoritmo la información de color del

video sí es importante y no debiera descartarse. De forma más general se puede concluir que no debiera descartarse inmediatamente la información de color de los videos para segmentar la glotis en videos laringoscópicos, ya que potencialmente puede ayudar en la segmentación, y no la empeorará. Esto concuerda con nuestra hipótesis inicial de que el color es importante ya que para los clínicos no es lo mismo ver una imagen en blanco y negro de la glotis que una a color.

Por otro lado, de los tres canales de color el que contiene más información en videos laringoscópicos es el rojo. Al trabajar con imágenes en escala de grises obtenidas a partir del canal rojo los resultados empeoran pero no tanto como cuando se ignora el canal; al hacer esto los resultados son significativamente peores. Los canales verde y azul contienen menos información, lo cual se puede apreciar visualmente al ver estos canales por separado. Pero como se mencionó anteriormente, no debieran eliminarse ya que siguen teniendo algo de información y el algoritmo empeora si no se consideran.

Sobre el uso de técnicas de machine learning

Al agregar el cálculo de ROI con imagen de varianza al inicio del algoritmo los resultados mejoraron significativamente. El hacer esto implica quitarle importancia a la parte de machine learning del algoritmo, la cual deja de ser el principal método para encontrar la ubicación de la glotis. Por lo tanto se concluye que de la forma que está implementada en este algoritmo, y con la cantidad de datos de entrenamiento que se tienen disponibles, la parte de machine learning no funciona particularmente bien. Por sí sola no es suficientemente robusta para encontrar con seguridad la ubicación de la glotis. Una causa probable son las distintas formas que puede tomar la glotis, ya que ésta puede estar parcialmente tapada, puede ser mas grande o pequeña dependiendo de qué tan lejos esté de la cámara, y puede estar afectada por patologías como nódulos que alteran su forma. También las condiciones de iluminación varían significativamente de video a video. Por lo tanto es muy difícil generalizar y lograr que el algoritmo aprenda cómo se ve una glotis típica. Se necesitan muchos datos y muchas muestras distintas para lograr eso, lo cual en este caso es difícil ya que se tiene una cantidad limitada de pacientes y videos. Todo esto probablemente explique la ausencia de técnicas de machine learning en la mayoría de los papers revisados en la sección de estado del arte. Sin embargo, esto no significa que se debiera dejar de investigar aplicaciones de

esta técnica para la segmentación de la glotis en estos videos; el algoritmo original sí logra funcionar utilizando esta técnica, sólo que no logra el mejor rendimiento. Con más datos disponibles, o aplicando machine learning tal vez de una forma distinta o en otra parte del algoritmo sí es posible que se obtengan resultados mejores.

Trabajo futuro

El algoritmo desarrollado todavía no alcanza el nivel humano de precisión en la segmentación, particularmente en videos grabados con laringoscopios flexibles. El estado del arte representado por el algoritmo GIE tampoco llega a este nivel, por lo que todavía hay trabajo por hacer.

Un caso donde el algoritmo desarrollado falla es al procesar videos donde exista un puente mucoso en la glotis que la separe en dos regiones distintas. Esto imposibilita la correcta inicialización del contorno y por lo tanto siempre llevará a errores en la segmentación. Estos casos no fueron tratados durante esta memoria y quedan pendientes.

Todavía se puede investigar más sobre la aplicación de técnicas de machine learning para la segmentación de la glotis en videos laringoscópicos. Se pueden realizar pruebas con datasets de entrenamiento distintos o más grandes, o aplicando otras técnicas de machine learning de formas distintas. Como se observó en la sección de revisión del estado del arte, hay pocos papers donde se ha tratado este tema y por lo tanto aún queda investigación por hacer.

Respecto a la evaluación cuantitativa de los algoritmos, es verdad que se logró hacer una evaluación numérica del rendimiento de cada versión del algoritmo, pero esos números sólo son válidos al trabajar con el dataset de evaluación que se utilizó aquí y sólo se evaluó un algoritmo distinto (el algoritmo GIE) como comparación. Existen muchos otros algoritmos propuestos para los cuales no se tiene una medida objetiva de su rendimiento en comparación a los algoritmos implementados en esta memoria. La única forma de lograr eso es implementar cada uno de ellos y evaluarlos utilizando el mismo dataset de evaluación, lo cual sería una tarea muy grande. La solución ideal a este problema es que exista un dataset público junto con una metodología estándar para medir el rendimiento de un algoritmo sobre el dataset, de tal forma que cualquier

persona pueda desarrollar un algoritmo de segmentación de la glotis y evaluarlo sobre este dataset, y que los resultados de dicha evaluación sean directamente comparables con los de otros algoritmos.

Referencias

[1] Ikram E Abdou y William K Pratt. "Quantitative design and evaluation of enhancement/thresholding edge detectors". En: *Proceedings of the IEEE* 67.5 (1979), págs. 753-763.

[2] Gustavo Andrade-Miranda y col. "An automatic method to detect and track the glottal gap from high speed videoendoscopic images". En: *Biomedical engineering online* 14.1 (2015), pág. 100.

[3] Peter Birkholz. "GlottalImageExplorer–An open source tool for glottis segmentation in endoscopic high-speed videos of the vocal folds". En: *Studientexte zur Sprachkommunikation: Elektronische Sprachsignalverarbeitung* (2016), págs. 39-44.

[4] Peter Birkholz. *GlottalImageExplorer download - VocalTractLab*. 2015. URL: `http://www.vocaltractlab.de/index.php?page=glottalimageexplorer-download` (visitado 30-07-2018).

[5] Juan J Cerrolaza y col. "Full-Automatic Glottis Segmentation With Active Shape Models." En: *Models and Analysis of Vocal Emissions for Biomedical Applications*. Firenze University Press, 2011, págs. 35-38.

[6] Antonin Chambolle y Thomas Pock. "A first-order primal-dual algorithm for convex problems with applications to imaging". En: *Journal of mathematical imaging and vision* 40.1 (2011), págs. 120-145.

[7] Tony F Chan y Luminita A Vese. "Active contours without edges". En: *IEEE Transactions on image processing* 10.2 (2001), págs. 266-277.

[8] Timothy F Cootes y col. "Active shape models-their training and application". En: *Computer vision and image understanding* 61.1 (1995), págs. 38-59.

[9] DD Deliyski, Szymon Cieciwa y Tomasz Zielinski. "Fast and robust endoscopic motion estimation in high-speed laryngoscopy". En: *7th International Conference: Advances in Quantitative Laryngology, Voice and Speech Research*. Vol. 7. 8. 2006, págs. 1-12.

[10] Jonathan Demeyer y col. "Glottis segmentation with a high-speed glottography: a fully automatic method". En: *3rd Adv. Voice Funct. Assess. Int. Workshop*. 2009.

[11] Lee R Dice. "Measures of the amount of ecologic association between species". En: *Ecology* 26.3 (1945), págs. 297-302.

[12] Andre Folkers y Hanan Samet. "Content-based image retrieval using Fourier descriptors on a logo database". En: *Pattern Recognition, 2002. Proceedings. 16th International Conference on*. Vol. 3. IEEE. 2002, págs. 521-524.

[13] Chris A Glasbey. "An analysis of histogram-based thresholding algorithms". En: *CVGIP: Graphical models and image processing* 55.6 (1993), págs. 532-537.

[14] Oliver Gloger y col. "Fully automated glottis segmentation in endoscopic videos using local color and shape features of glottal regions". En: *IEEE Transactions on Biomedical Engineering* 62.3 (2015), págs. 795-806.

[15] Rafael C Gonzalez. *Digital image processing*. Prentice hall, 2016.

[16] Sevasti-Zoi Karakozoglou y col. "Automatic glottal segmentation using local-based active contours and application to glottovibrography". En: *Speech Communication* 54.5 (2012), págs. 641-654.

[17] Michael Kass, Andrew Witkin y Demetri Terzopoulos. "Snakes: Active contour models". En: *International journal of computer vision* 1.4 (1988), págs. 321-331.

[18] Tony Lindeberg. "Feature detection with automatic scale selection". En: *International journal of computer vision* 30.2 (1998), págs. 79-116.

[19] Jörg Lohscheller y col. "Clinically evaluated procedure for the reconstruction of vocal fold vibrations from endoscopic digital high-speed videos". En: *Medical image analysis* 11.4 (2007), págs. 400-413.

[20] Brian Moriarty. *Perlenspiel — API — Colors*. URL: `http://users.wpi.edu/~bmoriarty/ps/colors.html` (visitado 31-07-2018).

[21] Mark Polak, Hong Zhang y Minghong Pi. "An evaluation metric for image segmentation of multiple objects". En: *Image and Vision Computing* 27.8 (2009), págs. 1223-1227.

[22] Hanan Samet y Markku Tamminen. "Efficient component labeling of images of arbitrary dimension represented by linear bintrees". En: *IEEE Transactions on Pattern Analysis & Machine Intelligence* 4 (1988), págs. 579-586.

[23] Fabian Schenk y col. "Automatic high-speed video glottis segmentation using salient regions and 3D geodesic active contours (2015)". En: *Annals of the British Machine Vision Association* (2015).

[24] James A Sethian y col. "Level set methods and fast marching methods". En: *Journal of Computing and Information Technology* 11.1 (2003), págs. 1-2.

[25] Day-Fann Shen y Ming-Tsong Huang. "A watershed-based image segmentation using JND property". En: *Acoustics, Speech, and Signal Processing, 2003. Proceedings.(ICASSP'03). 2003 IEEE International Conference on*. Vol. 3. IEEE. 2003, págs. III-377.

[26] Federico M Sukno y Alejandro F Frangi. "Reliability estimation for statistical shape models". En: *IEEE Transactions on Image Processing* 17.12 (2008), págs. 2442-2455.

[27] Markus Unger y col. "TVSeg-Interactive Total Variation Based Image Segmentation." En: *BMVC*. Vol. 31. Citeseer. 2008, págs. 44-46.

[28] Li Wang. *Active contours driven by local Gaussian distribution fitting energy - File Exchange - MATLAB Central*. 2009. URL: `https://la.mathworks.com/matlabcentral/fileexchange/38637-active-contours-driven-by-local-gaussian-distribution-fitting-energy` (visitado 28-06-2018).

[29] Li Wang y col. "Active contours driven by local Gaussian distribution fitting energy". En: *Signal Processing* 89.12 (2009), págs. 2435-2447.

[30] Yue Wu. *Chan Vese Active Contours without edges - File Exchange - MATLAB Central*. 2009. URL: `https://la.mathworks.com/matlabcentral/fileexchange/23445-chan-vese-active-contours-without-edges` (visitado 01-07-2018).

[31] Yuling Yan y col. "Snake based automatic tracing of vocal-fold motion from high-speed digital images". En: *Acoustics, Speech and Signal Processing (ICASSP), 2012 IEEE International Conference on*. IEEE. 2012, págs. 593-596.

[32] Jianming Zhang y Stan Sclaroff. "Saliency detection: A boolean map approach". En: *Computer Vision (ICCV), 2013 IEEE International Conference on*. IEEE. 2013, págs. 153-160.

[33] Karel Zuiderveld. "Contrast limited adaptive histogram equalization". En: *Graphics gems* (1994), págs. 474-485.

Anexo A

Active contours driven by local Gaussian distribution fitting energy (2009) [29]

En este anexo se explica la implementación de contorno activo propuesta en [29] y que se utilizó en la sección 1.4.2 de la parte de desarrollo del tema.

Se propone un contorno activo implícito basado en distribuciones de intensidad locales. El primer paso es caracterizar esta distribución de intensidades locales. Para cada punto x en el dominio de la imagen Ω se considera una vecindad circular con un pequeño radio ρ definida como $\mathscr{O}_x \triangleq \{y : |x-y| \leq \rho\}$. Sea $\{\Omega_i\}_{i=1}^N$ un conjunto de regiones disjuntas tal que $\Omega = \cup_{i=1}^N \Omega_i$, $\Omega_i \cap \Omega_j = \emptyset$, $\forall i \neq j$, donde N es el número de regiones. Estas regiones producen una producen una partición de la vecindad $\mathscr{O}_x$ de la forma: $\{\Omega_i \cap \mathscr{O}_x\}_{i=1}^N$. Ahora se considera una segmentación de esta región circular $\mathscr{O}_x$ basada en la *probabilidad máxima a posteriori* (MAP). Sea $p(y \in \Omega_i \cap \mathscr{O}_x \,|\, I(y))$ la probabilidad a posteriori de las subregiones $\Omega_i \cap \mathscr{O}_x$ dado el valor de intensidad de gris $I(y)$ del punto 'y' de la vecindad. Por la regla de Bayes:

$$p(y \in \Omega_i \cap \mathscr{O}_x \,|\, I(y)) = \frac{p(I(y) \,|\, y \in \Omega_i \cap \mathscr{O}_x) p(y \in \Omega_i \cap \mathscr{O}_x)}{p(I(y))} \tag{1}$$

Donde $p(I(y) \,|\, y \in \Omega_i \cap \mathscr{O}_x)$, denotado por $p_{i,x}(I(y))$, es la densidad de probabilidad i.e. la distribución de intensidad de gris en la región $\Omega_i \cap \mathscr{O}_x$. $p(y \in \Omega_i \cap \mathscr{O}_x)$ es la probabilidad a priori de de la partición $\Omega_i \cap \mathscr{O}_x$ entre todas las particiones, y se puede ignorar ya que a priori todas las particiones son igualmente probables. $p(I(y))$ es la probabilidad a priori del valor gris $I(y)$ la cual es independiente de la región y por lo tanto también se ignora.

Asumiendo que los pixeles de cada región son independientes, la MAP se logra cuando el producto de todos los $p_{i,x}(I(y))$ sobre las regiones $\mathscr{O}_x$ se maximiza:

$$\prod_{i=1}^{N} \prod_{y \in \Omega_i \cap \mathscr{O}_x} p_{i,x}(I(y)) \tag{2}$$

Es decir buscamos las particiones Ω_i que maximiza esta función.

Aplicando un logaritmo esta maximización se convierte en la minimización de la siguiente energía:

$$E_x^{LGDF} = \sum_{i=1}^{N} \int_{\Omega_i \cap \mathscr{O}_x} -\log p_{i,x}(I(y))dy \tag{3}$$

Las densidades de probabilidad se modelan a través de una gaussiana donde la media y varianza son parámetros que varían en el espacio para permitir la segmentación de regiones de intensidad no homogénea:

$$p_{i,x}(I(y)) = \frac{1}{\sqrt{2\pi}\sigma_i(x)} \exp(-\frac{(u_i(x) - I(y))^2}{2\sigma_i(x)^2}) \tag{4}$$

Donde $u_i(x)$ y $\sigma_i(x)$ son la media y desviación estándar local, respectivamente.

Agregando una función de peso a la ecuación 3 se define la siguiente función objetivo:

$$E_x^{LGDF} = \sum_{i=1}^{N} \int_{\Omega_i \cap \mathscr{O}_x} -\omega(x-y) \log p_{i,x}(I(y))dy \tag{5}$$

Donde $\omega(x-y)$ es una función de peso no negativa tal que $\omega(x-y) = 0$ para $|x-y| > p$ y $\int_{\mathscr{O}} \omega(x-y)dy = 1$. (Recuerde que x e y son puntos en el espacio). Para la función ω se elige una gaussiana truncada:

$$\omega(d) = \begin{cases} \frac{1}{d} \exp(-\frac{|d|^2}{2\sigma^2}) & |d| \leq \rho \\ 0 & |d| > \rho \end{cases} \tag{6}$$

Donde a es una constante tal que $\int \omega(d) = 1$. Con esto se puede rescribir la función objetivo como:

$$E_x^{LGDF} = \sum_{i=1}^{N} \int_{\Omega_i} -\omega(x-y) \log p_{i,x}(I(y)) dy \tag{7}$$

El objetivo final es minimizar E_x^{LGDF} para todos los puntos x en el dominio de la imagen Ω lo cual lleva a definir la siguiente función de energía con una doble integral:

$$E^{LGDF} = \int_{\Omega} E_x^{LGDF} dx = \int_{\Omega} (\sum_{i=1}^{N} \int_{\Omega_i} -\omega(x-y) \log p_{i,x}(I(y)) dy) dx \tag{8}$$

Se asume que el dominio de la imagen se puede separar en dos regiones: primer plano y fondo denotados por Ω_1 y Ω_2 respectivamente. Estas regiones se pueden representar como las regiones fuera y dentro del conjunto de nivel cero de ϕ, i.i. $\Omega_1 = \{\phi > 0\}$ y $\Omega_2 = \{\phi < 0\}$. Usando la función Heaviside H la energía de la ecuación 7 en términos de ϕ, u_i y σ_i^2:

$$E_x^{LGDF}(\phi, u_1(x), u_2(x), \sigma_1^2(x), \sigma_2^2(x)) \tag{9}$$

$$= -\int \omega(x-y) \log p_{1,x}(I(y)) M_1(\phi(y)) dy - \int \omega(x-y) \log p_{2,x}(I(y)) M_2(\phi(y)) dy \tag{10}$$

donde $M_1(\phi(y)) = H(\phi(y))$ y $M_2(\phi(y)) = 1 - H(\phi(y))$. Luego la energía de la ecuación 8 se puede reescribir como:

$$E^{LGDF}(\phi, u_1, u_2, \sigma_1^2, \sigma_2^2) \tag{11}$$

$$= \int_{\Omega} E_x^{LGDF}(\phi, u_1(x), u_2(x), \sigma_1^2(x), \sigma_2^2(x)) dx \tag{12}$$

Para un cálculo más preciso de la evolución del conjunto de nivel se penaliza su desviación de una función de distancia caracterizada por la siguiente energía:

$$\mathscr{P}(\phi) = \int \frac{1}{2} (|\nabla \phi(x)| - 1)^2 dx \tag{13}$$

Y además se debe regularizar el conjunto de nivel cero penlizando su largo para obtener un contorno suave durante la evolución:

$$\mathscr{L}(\phi) = \int |\nabla H(\phi(x))dx \tag{14}$$

Luego la función de energía completa es:

$$\mathscr{F}(\phi, u_1, u_2, \sigma_1^2, \sigma_2^2) = E^{LGDF}(\phi, u_1, u_2, \sigma_1^2, \sigma_2^2) + v\mathscr{L}(\phi) + \mu\mathscr{P}(\phi) \tag{15}$$

Donde $v, \mu > 0$ son constantes de peso. La función Heaviside H en la práctica se aproxima por una función suave definida como:

$$H_\varepsilon(x) = \frac{1}{2}[1 + \frac{2}{\pi}\arctan(\frac{x}{\varepsilon})] \tag{16}$$

Y su derivada es:

$$\delta_\varepsilon(x) = H'_\varepsilon(x) = \frac{1}{\pi}\frac{\varepsilon}{\varepsilon^2 + x^2} \tag{17}$$

Y finalmente la función de energía es aproximada por:

$$\mathscr{F}_\varepsilon(\phi, u_1, u_2, \sigma_1^2, \sigma_2^2) = E_\varepsilon^{LGDF}(\phi, u_1, u_2, \sigma_1^2, \sigma_2^2) + v\mathscr{L}_\varepsilon(\phi) + \mu\mathscr{P}(\phi) \tag{18}$$

Esta función se minimiza mediante *gradient descent*. Se puede demostrar que los valores de u_i y σ_i^2 que minimizan a $\mathscr{F}$ para un ϕ dado son:

$$u_i(x) = \frac{\int \omega(y-x)I(y)M_{i,\varepsilon}(\phi(y))dy}{\int \omega(y-x)M_{i,\varepsilon}(\phi(y))dy} \tag{19}$$

$$\sigma_i(x)^2 = \frac{\int \omega(y-x)(u_i(x) - I(y))^2 M_{i,\varepsilon}(\phi(y))dy}{\int \omega(y-x)M_{i,\varepsilon}(\phi(y))dy} \tag{20}$$

La minimización de $\mathscr{F}_\varepsilon$ con respecto a ϕ se logra resolviendo la siguiente ecuación de *gradient descent flow*:

$$\frac{\partial \phi}{\partial t} = -\delta_\varepsilon(\phi)(e_1 - e_2) + v\delta_\varepsilon(\phi) div(\frac{\nabla \phi}{|\nabla \phi|}) + \mu(\nabla^2 \phi - div(\frac{\nabla \phi}{|\nabla \phi|})) \tag{21}$$

donde:

$$e_1(x) = \int_\Omega \omega(y-x)[\log(\sigma_1(y)) + \frac{(u_1(y) - I(x))^2}{2\sigma_1(y)^2}]dy \tag{22}$$

$$e_2(x) = \int_\Omega \omega(y-x)[\log(\sigma_2(y)) + \frac{(u_2(y) - I(x))^2}{2\sigma_2(y)^2}]dy \tag{23}$$

Finalmente la implementación del método es la siguiente:

1) Inicializar la función de conjunto de nivel ϕ

2) Actualizar $u_i(x)$ y $\sigma_i(x)^2$ con las ecuaciones 19 y 20.

3) Actualizar la función conjunto de nivel ϕ con la ecuación 21.

4) Volver al paso 2 hasta que se cumpla el criterio de convergencia

Un criterio de convergencia simple puede ser que se estabilice el área encerrada por el contorno.

Una implementación de este contorno activo en MATLAB se encuentra públicamente disponible en [28] y se utilizó directamente en el desarrollo de esta memoria.

Anexo B

Active Contours Without Edges (2001) [7]

En este anexo se explica la implementación del contorno activo propuesta en [7] y que se utilizó en la sección 1.5.3 de la parte de desarrollo del tema.

Este método es la minimización de una función de energía. Para una curva cualquiera C se define el siguiente término de "fitting":

$$F_1(C)+F_2(C)=\int_{inside(C)}|u_0(x,y)-c_1|^2dxdy+\int_{outside(C)}|u_0(x,y)-c_2|^2dxdy \quad (24)$$

donde $u_0(x,y)$ es el valor de intensidad de la imagen en la coordenada (x,y), y las constantes c_1, c_2 son las intensidades promedio de u_0 dentro y fuera de C, respectivamente. El caso más simple posible es donde la imagen u_0 está formada por dos regiones de intensidad constante u_0^1 y u_0^2 separadas por una frontera C_0. En ese caso es fácil ver que la curva que minimiza la función es $C=C_0$ ya que $F_1(C_0)+F_2(C_0)=0$. En este modelo de contorno activo se minimizará el término presentado junto con un término de regularización basado en el largo de la curva. La función de energía a minimizar es:

$$F(c_1,c_2,C) \quad (25)$$

$$=\mu\cdot Largo(C)+\int_{inside(C)}|u_0(x,y)-c_1|^2dxdy+\int_{outside(C)}|u_0(x,y)-c_2|^2dxdy \quad (26)$$

La curva C será representada por el conjunto de nivel cero de una función Lipschitz $\phi:\Omega\to\mathbb{R}$ tal que:

$$\begin{cases} C=\{(x,y)\in\Omega:\phi(x,y)=0\}, \\ inside(C)=\{(x,y)\in\Omega:\phi(x,y)>0\}, \\ outside(C)=\{(x,y)\in\Omega:\phi(x,y)<0\} \end{cases} \quad (27)$$

Utilizando la función Heaviside H y su derivada δ_0 se pueden expresar los términos de F de la siguiente forma:

$$Largo(C) = \int_\Omega |\nabla H(\phi(x,y))| dx\,dy \tag{28}$$

$$= \int_\Omega \delta_0(\phi(x,y)) |\nabla \phi(x,y)| dx\,dy \tag{29}$$

$$\int_{\phi>0} |u_0(x,y) - c_1|^2 dx\,dy = \int_\Omega |u_0(x,y) - c_1|^2 H(\phi(x,y)) dx\,dy \tag{30}$$

$$\int_{\phi<0} |u_0(x,y) - c_2|^2 dx\,dy = \int_\Omega |u_0(x,y) - c_2|^2 (1 - H(\phi(x,y))) dx\,dy \tag{31}$$

Y luego la energía F se expresa como:

$$F(c_1,c_2,\phi) = \mu \int_\Omega \delta(\phi(x,y)) |\nabla \phi(x,y)| dx\,dy \tag{32}$$

$$+ \int_\Omega |u_0(x,y) - c_1|^2 H(\phi(x,y)) dx\,dy \tag{33}$$

$$+ \int_\Omega |u_0(x,y) - c_2|^2 (1 - H(\phi(x,y))) dx\,dy \tag{34}$$

La función Heaviside H se aproxima por la siguiente función

$$H_\varepsilon(x) = \frac{1}{2}[1 + \frac{2}{\pi} \arctan(\frac{x}{\varepsilon})] \tag{35}$$

Dejando c_1 y c_2 fijos y minimizando F con respecto a ϕ se deduce la ecuación de Euler-Lagrange para ϕ. La ecuación se resuelve por *gradient descent* parametrizado por un tiempo artificial t:

$$\frac{\partial \phi}{\partial t} = \delta_\varepsilon(\phi)[\mu div(\frac{\nabla \phi}{|\nabla \phi|}) - (u_0 - c_1)^2 + (u_0 - c_2)^2] = 0 \tag{36}$$

Finalmente los pasos del algoritmo son:

1) Inicializar ϕ con algún valor ϕ_0

2) Calcular $c_!$ y c_2

3) Resolver la ecuación diferencial 36 para obtener ϕ^{n+1}

4) Revisas si la solución es estacionaria. Si no, volver al paso 2.

Un criterio de convergencia simple puede ser que se estabilice el área encerrada por el contorno.

Una implementación de este contorno activo en MATLAB se encuentra públicamente disponible en [30] y se utilizó directamente en el desarrollo de esta memoria.