BENEMÉRITA
UNIVERSIDAD AUTÓNOMA DE PUEBLA

INSTITUTO NACIONAL DE ASTROFÍSICA, ÓPTICA Y ELECTRÓNICA

FACULTAD DE CIENCIAS DE LA COMPUTACIÓN

Localización y segmentación de palabras en documentos manuscritos antiguos y modernos para el sistema

PRISCUS

TESIS

PARA OBTENER EL TITULO DE:

LICENCIADO EN INGENIERÍA EN CIENCIAS DE LA COMPUTACIÓN

PRESENTA
HAZZIEL PIMENTEL CRISÓSTOMO

ASESOR 1
DR. ARTURO OLVERA LÓPEZ, BUAP

ASESORA 2
DRA. MARÍA DEL PILAR GÓMEZ GIL, INAOE

TONANTZINTLA, PUE OCTUBRE 2015

Resumen

En la actualidad existen gran cantidad de reconocedores de textos que se basan en diferentes técnicas de segmentación, sin embargo, aún no existe un software que obtenga el 100% de segmentación total de palabras. Esta situación ha motivado a investigar diferentes métodos para atacar este problema, debido a que computacionalmente es un problema del tipo NP. Existen dos grandes clasificaciones para el reconocimiento de texto: 1) En línea, que consiste en hacer un reconocimientos de las palabras de texto en tiempo real, y 2) fuera de línea, cuyo procedimiento se diferencia del anterior por no realizarse en tiempo real, por el contrario, lleva un proceso de digitalización previa del documento.

En la presente investigación se desarrolló un sistema de segmentación de palabras manuscritas, específicamente aplicado a documentos antiguos. La base de datos con la que se trabajó contiene imágenes de documentos manuscritos del presidente norteamericano George Washington que datan de 1741 a 1799. Asimismo, se incluyeron unas imágenes de escritura manuscrita contemporánea, así como una carta de la primera dama de los Estados Unidos, Abigail Adams del año de 1783. Se propone un método con dos etapas: pre-procesamiento y segmentación. La etapa de pre- procesamiento consiste en el mejoramiento de la calidad de la imagen, y la reducción de ruido a través del uso de filtros, los cuales son: escala de grises, basado en mediana y el método de Bernsen, este último usado para la binarización de la imagen. La etapa de segmentación consiste principalmente en tres pasos: 1) aplicación de la técnica de Componentes Conectados a toda la imagen, 2) segmentación de línea sobre los objetos extraídos del paso 1 que sean de un tamaño mayor al promedio de los demás objetos, y 3) a cada línea segmentada del paso anterior se le aplica nuevamente la técnica de componentes conectados para obtener finalmente todas las posibles palabras del documento. El segmentador PRISCUS "*Word Segmentation*" obtuvo un 92% de eficiencia en el mejor caso, comparado con el sistema ISAE que obtuvo en su mejor caso el 57% de eficiencia.

Índice general

Índice general I
Agradecimientos III
Índice de figuras 1
Índice de tablas 3
1. Introducción 5
1.1 Descripción del problema 5
1.2 Objetivos generales y específicos 6
1.2.1 Objetivo general 6
1.2.2 Objetivos específicos 6
1.3 Alcance 6
1.4 Limitaciones 7
1.5 Resultados obtenidos 7
2. Marco teórico 9
2.1 Reconocimiento de texto manuscrito 9
2.2 Pre-procesamiento 10
2.2.1 Escala de grises 10
2.2.2 Binarización 11
2.3 Segmentación 12
2.4 Trabajos relacionados 13
3. Descripción del sistema 17
3.1 Arquitectura general del sistema 18
3.1.1 Documento escaneado 19
3.1.2 Pre-procesamiento 19
3.1.3 Segmentación 21
3.2 Descripción del Software utilizado 24
3.3 Descripción de las funciones principales – Toolbox 25
4. Experimentos y resultados 31
4.1 Evaluación de resultados 31
4.2 Error de segmentación 31
4.3 Resultados ..... **¡Error! Marcador no definido.**
4.3.1 Experimentos realizados 32

4.3.2 Análisis de resultados 40
5. Conclusiones y trabajo a futuro 43
5.1 Conclusiones 43
5.2 Trabajo a futuro 45
I. Imágenes de Documentos 47
II. Manual de usuario 63
Bibliografía……..71

Agradecimientos

Primeramente agradezco a la Benemérita Universidad Autónoma de Puebla (BUAP) por haberme permitido ser parte de ella, así mismo doy gracias al Instituto Nacional de Astrofísica, Óptica y Electrónica (INAOE) por brindarme su apoyo para poder realizar el presente trabajo investigación.

Debo agradecer de manera muy especial a mi asesora de tesis la Dra. María del Pilar Gómez Gil por aceptarme para realizar esta tesis bajo su supervisión, por brindarme su apoyo y confianza, y por haberme facilitado los medios suficientes para llevar a cabo todas las actividades propuestas durante el desarrollo del proyecto.

A mi asesor de tesis, el Dr. José Arturo Olvera López, por haberme brindado la oportunidad de recurrir a su capacidad y conocimiento científico, y por haberme guiado durante el desarrollo de toda la tesis.

Agradezco a mis sinodales el Dr. Juan Manuel González Calleros y la Dra. Josefina Guerrero García, por su todo su apoyo y por haberme compartido su amplio conocimiento, No cabe duda que debo destacar que son unos excelentes investigadores y más que eso, unos grandes amigos, sin olvidar enfatizar su gran paciencia, disponibilidad y generosidad para compartir sus experiencias, así como su amplio conocimiento en el área.

Finalmente, el agradecimiento más profundo y especial es para mis padres Ángel Pimentel Rosas y Ana Luisa Crisóstomo Cruz por todo su apoyo, cariño y comprensión que me han brindado a lo largo de mi vida, debido que gracias a ellos soy la persona que soy hoy en día; a Jessica Nayeli López Espejel debo agradecerle sinceramente su apoyo, que ha creído en mí en todo momento y que pese a todas las circunstancias siempre ha estado a mi lado. Por último, doy gracias a Dios por haberme permitido conocer tan maravillosas personas, los quiero mucho y gracias por todo.

Índice de figuras

Figura 2.1 Escala de grises. (a) Imagen original. (b) Imagen en escala de grises. 11
Figura 2.2 Binarización. (a) Imagen en escala de grises. (b) Imagen binarizada. 12
Figura 3.1 Diagrama de bloques del sistema 18
Figura 3.2 Matriz de una imagen digital. 19
Figura 3.3 Código escala de grises. 20
Figura 3.4 Código filtro mediana. 21
Figura 3.5 Código para obtener los filtros de Bernsen y Niblack. 21
Figura 3.6 Componentes conectados. (a) 4-conectados. (b) 8-conectados. 22
Figura 3.7 Validación de un componente 22
Figura 3.8 Componentes conectados unidos 23
Figura 3.9 Segmentación mediante líneas de texto usando histograma. 23
Figura 3.10 Segmentación de componentes conectados por línea de texto. 23
Figura 3.11 Dibujado de componentes conectados (CC). 24
Figura 3.12 Diagrama de clases. 25
Figura 3.13 (a) Imagen antes de bwareaopen. (b) Imagen después de bwareaopen 26
Figura 3.14 Ejemplo de la función bwlabel 27
Figura 4.1 Segmentación del experimento 1. 33
Figura 4.2 Segmentación del experimento 2 (versión 1) 34
Figura 4.3 Implementación de la heurística: (a) antes. (b) después. 35
Figura 4.4 Segmentación del experimento 2 (versión 2). 36
Figura 4.5 Segmentación del experimento 2 (versión 3). 38
Figura I.1 Carta a Kennedy. 47
Figura I.2 Hoja de libreta. 48
Figura I.3 Carta de Abigail Smith Adams 49
Figura I.4 Carta de George R.J. 50
Figura I.5 IAM Handwriting Database 51
Figura I.6 IAM Handwriting Database 52
Figura I.7 Carta de George Washington 53
Figura I.8 Carta de George Washington 54
Figura I.9 Carta de George Washington 55

Figura I.10 Carta de George Washington 56
Figura I.11 Carta de George Washington 57
Figura I.12 Carta de George Washington 58
Figura I.13 Carta de George Washington 59
Figura I.14 Carta de George Washington 60
Figura I.15 Carta de George Washington 61
Figura II.1 Ventana principal de *PRISCUS*-Segmentador 64
Figura II.2 Menú archivo 65
Figura II.3 Menú pasos 66
Figura II.4 (a) Filtrado en proceso. (b) Filtrado terminado 67
Figura II.5 (a) Segmentación en proceso. (b) Segmentación terminada. 67
Figura II.6 Ver cada segmento generado 68
Figura II.7 Botón ver 69

Índice de tablas

Tabla 3.1 Clases del sistema 17
Tabla 3.2 Propiedades de la función regionprops() 28
Tabla 4.1 Filtros utilizados en las pruebas 32
Tabla 4.2 Resultados del experimento 1 33
Tabla 4.3 Resultados del experimento 2 (versión 1) 35
Tabla 4.4 Resultados del experimento 2 (versión 2) 36
Tabla 4.5 Resultados del experimento 2 (versión 3) 39
Tabla 4.6 Resultados del experimento 3 40
Tabla 4.7 Resultados los experimentos 40

Capítulo 1

Introducción

1.1 Descripción del problema

El problema a resolver con este proyecto consiste en localizar y segmentar de forma automática las palabras de documentos previamente digitalizados, que contienen manuscritos antiguos. El reconocimiento de textos manuscritos se refiere a un conjunto de técnicas cuyo objetivo es la reconstrucción de texto de un documento, a partir de una imagen, sin la necesidad de un experto. El reconocimiento de texto manuscrito es una parte muy importante para la transcripción de documentos antiguos y para su preservación; no obstante, en la actualidad sigue siendo una tarea complicada de llevarse a cabo, debido principalmente al tipo de papel y sus condiciones dañadas por el tiempo y la complejidad y variedad de estilos de escritura presentes en dichos documentos. El problema fundamental asociado a la localización de texto, como en otros problemas de localización en una imagen, está relacionado con la identificación de las características de la zona deseada que permitirán distinguirla de otras zonas que no son de interés. Con respecto a imágenes que contienen texto, se debe identificar qué características espaciales, de textura o de color permitirán conseguir máxima eficiencia.

En los últimos años se han desarrollado varios métodos para la segmentación de palabras manuscritas, pero no todos presentan un alto rendimiento; esto se debe a las limitaciones existentes por ser un problema no resuelto en su totalidad actualmente. Una buena segmentación depende de la calidad del documento original y de los métodos utilizados para la limpieza del documento lo cual permitirá prepararlo para su posterior análisis; evidentemente también depende de un buen método para la localización de las palabras.

A fin de cooperar en el avance de la tecnología en este campo, se presenta en este trabajo un método para la segmentación de documentos manuscritos y antiguos, llamado "*PRISCUS Word Segmentation".* Este sistema forma parte de un proyecto mayor llamado *PRISCUS*[1] (que significa "antiguo" en latín) y que está realizándose actualmente en el Instituto Nacional de Astrofísica, Óptica y Electrónica, por las coordinaciones de Electrónica y Computación. El objetivo final de *PRISCUS* es construir un software inteligente e integral para la transcripción automática de documentos antiguos.

1.2 Objetivos generales y específicos

En los siguientes apartados se presenta la descripción explícita del objetivo general y objetivos específicos del trabajo de investigación PRISCUS "*Word Segmentation".*

1.2.1 Objetivo general

- Desarrollar un sistema de binarización y segmentación, capaz de localizar palabras en una imagen digital obtenida de un texto manuscrito.

1.2.2 Objetivos específicos

- Seleccionar e implementar algoritmos adecuados para limpieza de las imágenes involucradas en este proyecto, considerando complejidad y eficiencia.
- Seleccionar e Implementar algoritmos para la localización y segmentación de palabras inmersas en las imágenes previamente pre-procesadas.

1.3 Alcance

El proyecto es una de las etapas más importante en el reconocimiento de texto manuscrito desarrollándose en el proyecto PRISCUS. En este caso, el proyecto se centra en localizar y segmentar las palabras situadas en imágenes digitales, cuyas palabras segmentadas serán utilizadas para su posterior reconocimiento.

[1] Pagina del proyecto: http://ccc.inaoep.mx/~pgomez/PRISCUS/

1.4 Limitaciones

El proyecto está limitado a localizar y segmentar palabras inmersas en un documento antiguo previamente digitalizado, en donde las palabras deben estar suficientemente separadas para que puedan ser segmentadas, aunque el sistema es robusto ante pequeñas superposiciones de algunos rasgos de las palabras. Otra limitación para una correcta segmentación es el ruido existente en el documento, porque eso puede ocasionar que las palabras no sean localizadas de manera correcta. Aun así, el sistema cuenta con filtros que eliminan parte de este ruido.

1.5 Resultados obtenidos

Los resultados obtenidos del método propuesto fueron satisfactorios, puesto que en el mejor caso se obtuvo el 92% de eficiencia sobre la base de datos utilizada y descrita en el capítulo 3. Los experimentos ejecutados en este trabajo mostraron una reducción en el error en el número de palabras incorrectas localizadas, en comparación con versiones anteriores del método propuesto.

El presente trabajo de investigación se divide en 5 capítulos. El capítulo II describe los trabajos relacionados que se han investigado previamente, y el marco teórico detalla las bases teóricas fundamentales para el desarrollo de PRISCUS "*Word Segmentation*". El capítulo III presenta la descripción del sistema y de los algoritmos que se implementaron; el capítulo IV muestra los resultados obtenidos de todos los experimentos que fueron realizados. Finalmente, el capítulo V puntualiza las conclusiones y propone el trabajo que puede realizarse para mejorar este proyecto.

Capítulo 2

2 Marco teórico

En este capítulo se presentan algunos conceptos fundamentales y trabajos relacionados al presente trabajo de investigación.

2.1 Reconocimiento de texto manuscrito

El reconocimiento de textos manuscritos se refiere a la transformación de un conjunto de símbolos impresos en su transcripción digital. Es muy importante hoy en día la digitalización de varios de estos manuscritos, debido que existen gran cantidad de documentos en papel, que deben ser interpretados en forma digital. Tal es el caso de los manuscritos antiguos, que fueron creados por personajes trascendentes de la historia, cuya preservación es importante.

En la actualidad existen diversos problemas a los que se enfrentan los sistemas de reconocimiento de textos manuscritos, por ejemplo, imperfecciones en el papel del documento (orificios, papel amarillo, manchas, etc.). Las imperfecciones ocasionan que los algoritmos existentes no puedan localizar y/o reconocer las palabras del documento con una eficiencia del 100%. Por otro lado, los estilos de escritura manuscrita son extremadamente diversos, por lo que resulta de mucha dificultad para los sistemas saber identificar las palabras. Muchos sistemas de reconocimiento se enfrentan con el problema de la confusión en la distinción de espacios entre letras con respecto al espacio entre palabras, por ello es muy importante la utilización de un buen método de segmentación para saber identificar la parte del texto que corresponde a una palabra, para su posterior reconocimiento.

Antes de abordar el problema de segmentación, es importante recalcar que existen dos tipos de reconocimiento de texto manuscrito: en línea y fuera de línea (Toselli, 2004). El reconocimiento en “línea”, consiste en transcribir el texto de una imagen en tiempo real, operación comúnmente realizada en dispositivos móviles, tales como tablas digitalizadoras o celulares. Por otro lado, el reconocimiento de texto “fuera de línea” es una tarea de gran dificultad, donde la entrada es un conjunto de imágenes escaneadas disponibles de documentos. Una de las principales dificultades del reconocimiento fuera de línea consiste en encontrar el área de escritura de la imagen, por lo que el pre-procesamiento y extracción de características adquieren un papel muy importante (Luna Pérez, 2011).

En este trabajo, nos enfocamos en el problema de segmentación sobre un sistema reconocedor de documentos conocido como PRISCUS (Gómez Gil & Cuevas Farfán, 2008). En este sistema se identifican entre otras actividades, dos etapas interesantes: el pre-procesamiento de la imagen y la segmentación de la palabra. A continuación se describen cada una de estas etapas (Luna Pérez, 2011).

2.2 Pre-procesamiento

El pre-procesamiento digital de imágenes reúne a un conjunto de técnicas con el objetivo de manipular y mejorar imágenes preparándolas para la extracción de la información de manera automática. El pre-procesamiento pretende reducir imperfecciones como son el tipo de papel y la variación de la escritura; además busca reducir el ruido que pudiera tener una imagen, mediante la aplicación de filtros.

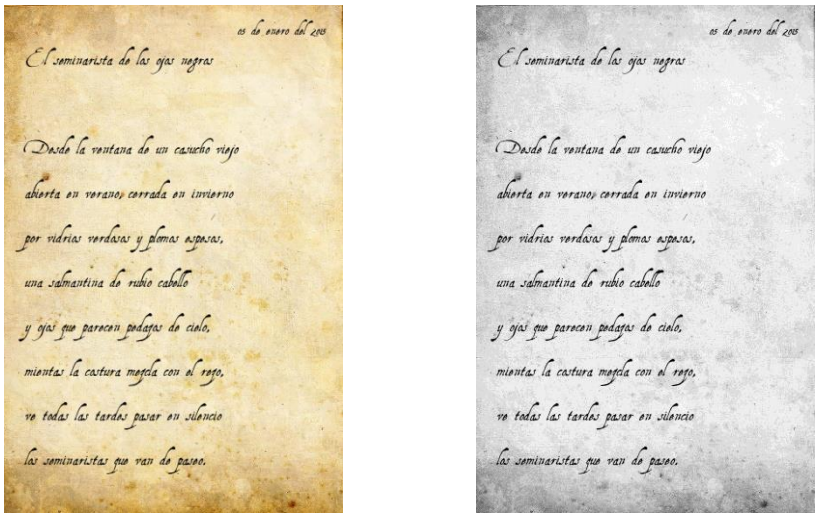
2.2.1 Escala de grises

Una imagen a escala de grises es una matriz cuyos valores han sido escalados para representar un determinado número de intervalos conformados en el rango de [0,255] (Cuevas, Zaldívar, & Pérez, 2010). La transformación de una imagen RGB (por sus siglas en inglés Red, Green y Blue) en una imagen en

escala de grises se puede obtener promediando las 3 bandas R, G y B de cada pixel de la imagen original., como se establece en la ecuación 2.1.

$$gris = \frac{R + G + B}{3} \tag{2.1}$$

En la figura 2.1 se ejemplifica el uso del filtro de escala de grises a partir de una imagen RGB.

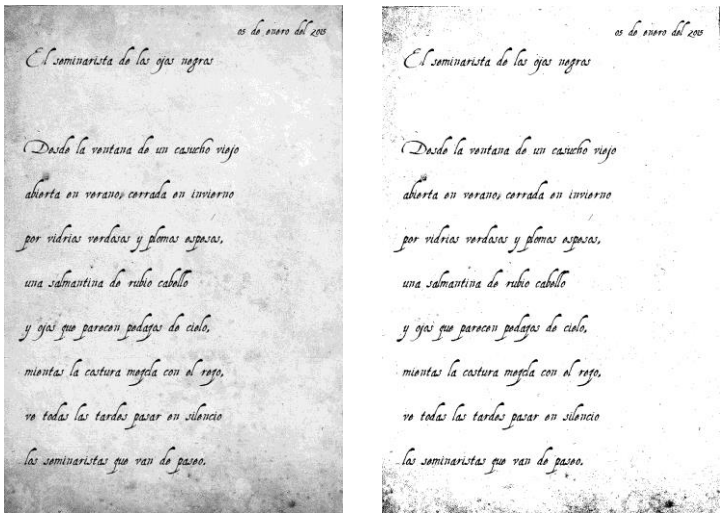


(a) ***(b)***

Figura 2.1 Escala de grises. (a) Imagen original. (b) Imagen en escala de grises.

2.2.2 Binarización

La binarización o umbralización es una técnica de segmentación ampliamente usada; se emplea cuando puede establecerse una diferencia entre los objetos a extraer y el fondo de la escena de una imagen. Existen diversas técnicas desarrolladas a lo largo de los años; la umbralización en dos niveles de una imagen, es una técnica basada en la búsqueda de un umbral óptimo, que permita distinguir en una imagen los objetos del fondo (Universidad Nacional de Quilmes , 2005). El objetivo de la binarización es el convertir una imagen en escala de grises a una nueva imagen binaria, representa como una matriz que contiene solo unos y ceros. (Cuevas, Zaldívar, & Pérez, 2010). La figura 2.2 presenta un ejemplo de binarización de un documento antiguo.



(a) (b)

Figura2.2 Binarización. (a) Imagen en escala de grises. (b) Imagen binarizada.

2.3 Segmentación

La segmentación es un paso importante para la transcripción de un documento digital; su objetivo es fragmentar una imagen en varias partes a fin de localizar objetos en ésta. Sin embargo, a diferencia de documentos impresos por computadoras, la segmentación de los documentos manuscritos se considera todavía un problema difícil, debido a las separaciones irregulares entre las palabras y las variaciones en la forma de escribir de cada persona (Jewoong Ryu, 2015).

Los sistemas de reconocimiento utilizan el proceso de segmentación de varias maneras, ya que pueden buscar segmentar frases completas o palabras y sobre éstas realizar un reconocimiento global, o segmentar cada letra o segmento de palabra, y con esta información identificar a la palabra correspondiente (Toselli, 2004).

2.4 Trabajos relacionados

Existen una gran cantidad de trabajos que han atacado el problema de reconocimiento de escritura manuscrita, así como el de segmentación. La investigación actual incluye estudios en sistemas de reconocimiento de todo tipo: en línea, fuera de línea, con segmentación y sin segmentación. Enseguida se describen algunos trabajos, tanto del área de reconocimiento como de segmentación, que se considera un aspecto de gran relevancia.

Romero (Romero Gómez, 2010) presenta un sistema fuera de línea de reconocimiento de texto manuscrito (HTR por sus siglas en íngles: Handwriting Text Recognition). En dicho sistema, primero se realiza la etapa de pre-procesamiento, donde se da lugar a los siguientes pasos: eliminación de fondo y reducción de ruido, corrección de sesgo (skew), extracción de línea, corrección de la pendiente (slope), corrección de inclinación (slant) y normalización del tamaño. Posteriormente se lleva a cabo la extracción de características, para identificar las palabras en la línea extraída. Por último se realiza el reconocimiento de las palabras segmentadas mediante un Modelo Oculto de Markov (HMM). El sistema HTR, a diferencia de otros, presenta una ventaja en la localización, al identificar la línea completa para su posterior análisis, aprovechando la forma de escritura de los seres humanos al escribir manuscritos (escritura horizontal). Una de las desventajas de este sistema se encuentra en que únicamente localiza líneas de texto; algunas imperfecciones pueden tomarse como líneas de texto, o bien, algunas palabras alejadas del resto del texto podrían no ser localizadas.

Iqbal (Khalid Iqbal, 2014) propone el uso de una red bayesiana entrenada con un algoritmo conocido como K2. Este método utiliza características geométricas sobre texto obtenido usando el método de regiones extremales máximamente estables (MSERs). En la etapa 1 se realiza la extracción de regiones de caracteres candidatos, usando el algoritmo MSER. En la etapa 2 se realiza la localización de texto directamente en la imagen, usando características geométricas. En la etapa 3 se entrena la red bayesiana usando el algoritmo K2. Durante la etapa 4, un clasificador de texto es usado para clasificar el resto de las

regiones candidatas válidas. La evaluación del método propuesto tiene un rendimiento de f–mediciones de 72.44%, quedando en cuarto lugar al compararse con otros métodos, donde el máximo resultado obtenido fue de 75.89%. La ventaja del método propuesto por Iqbal se ve reflejada en la implementación de una serie de pasos, para la localización de palabras en una imagen compleja. Una de las posibles desventajas de este método está en la extracción de características, ésto debido a que algunas palabras podrían no ser tomadas como regiones candidatas. La omisión de características importantes (palabras y caracteres), durante la etapa de extracción de regiones candidatas reduce la tasa de rendimiento.

Senior (Senior & Robinson, 1998) describe un sistema fuera de línea de Reconocimiento Manuscrito de escritura cursiva, utilizando la base de datos llamada “Lancaster-Oslo/Bergen corpus” (LOB). Primero se realiza el esquema del sistema, que consiste en escanear imágenes e implementar un algoritmo de segmentación buscando espacios entre líneas y palabras. Después se realiza la normalización, una herramienta que elimina algunas variaciones en la imagen que no afecten la identidad de la palabra. Posteriormente se realiza la representación, expresando la información más sobresaliente de la imagen permitiendo su tratamiento para el sistema. Enseguida se lleva a cabo la etapa de reconocimiento, donde se intenta deducir la identidad de la palabra mediante una red neuronal, obteniendo como salida los posibles caracteres de la palabra. A continuación un Modelo Oculto de Markov (HMM) se encarga de tomar decisiones para identificar el significado de la palabra. El resultado obtenido fue de 87% de reconocimiento en pruebas de vocabulario abierto. El sistema de reconocimiento propuesto por Senior ofrece una buena tasa de reconocimiento al incluir en su sistema una red neuronal, para deducir la identidad de la palabra y un modelo oculto de Markov para deducir el significado de la palabra, dos elementos que otorgan al sistema excelentes resultados.

Fernández y colaboradores (Fernández Mota, Lladós, & Fornés, 2014) proponen un método de segmentación de líneas en documentos manuscritos históricos. El método consta de dos etapas: localización y segmentación de la

línea. En la etapa de localización se lleva a cabo la binarización, eliminación de ruido y márgenes y localización de la línea de texto. En la etapa de segmentación de líneas, primero se obtiene el esqueleto de la imagen, después se realiza una construcción de grafos y se agregan bordes virtuales. A continuación se busca el camino entre los grafos y por último se verifica la consistencia. Los resultados obtenidos, al ser probados en documentos escritos por George Washington (Rath & Manmatha, 2007) fueron del 92.70%.

Manmatha y Rothfer (Manmatha & Rothfeder, 2005) presentan una técnica para la segmentación automática de palabras en documentos históricos del presidente George Washington. Primero se limpia la imagen y se eliminan los márgenes. Después se aplica un algoritmo para localizar las líneas de texto en el documento mediante una proyección (histograma horizontal y/o vertical). A continuación cada línea de texto segmentada se aplica un filtro Laplaciano anisotrópico en varias escalas. Este procedimiento produce manchas que corresponden a caracteres y palabras. Las marcas recuperadas son encerradas en cajas rectangulares para la recuperación de la palabra. En un post-procesamiento se eliminan las cajas de tamaño inusual que no corresponden a palabras. En pruebas realizadas con 100 documentos, se obtuvo una tasa de error del 17%. La ventaja de la técnica propuesta por estos autores parece estar en el aprovechamiento de la forma de escritura de muchos países (escritura horizontal) para la extracción de las líneas de texto.

Fernández (Fernández Mota, Lladós, & Fornés, 2014) propone el método de Otsu para limpiar una imagen sin ruido mediante la binarización de imágenes de documentos. A fin de determinar el método más adecuado para binarizar una imagen, se realizaron pruebas con varios métodos. El método de Nickblack generó ruido en la imagen; el método de Sauvola generó pérdida de información y adelgazó los caracteres del documento; el método de Bersen obtuvo mejores resultados con respecto binarización de la imagen, pero con un alto costo a nivel computacional.

Louloudis (G. Louloudis, 2008) presenta un método para la detección de líneas en un documento manuscrito. Para esto, se realiza una búsqueda sobre el grafo que se define por los componentes conectados como vértices y las distancias entre ellos, determinando las áreas del documento que son propensos a ser líneas de texto. Posteriormente, el método corta el flujo mínimo y máximo para dividir las áreas de texto que aparecen a abarcar más de una línea de texto, teniendo una tasa de detección del 95.8%.

Capítulo 3

3 Descripción del sistema

En este capítulo se presenta una breve descripción de la arquitectura del sistema PRISCUS – *Word Segmentation* para la localización y segmentación de palabras en documentos manuscritos antiguos. La Tabla 3.1 muestra las clases que componen el sistema; de igual forma se detallan las etapas que conforman el método propuesto para la segmentación.

Tabla 3.1 Clases del sistema

Clases	Descripción
Averagefilter.m	Realiza el promedio de una matriz A. Cada pixel de salida contiene el valor promedio de la vecindad de MxN correspondiente a la matriz de imagen de entrada.
bernsen.m	Efectúa una umbralización local de una matriz bidimensional de una imagen con el método de Bernsen.(Sección 3.1.2)
niblack.m	Realiza la umbralización local de una matriz bidimensional de una imagen mediante el método de niblack.(Sección 3.1.2)
maxfilt2.m	Obtiene el valor máximo en la vecindad de MxN alrededor de cada pixel de entrada de la imagen.
minfilt2.m	La función devuelve un valor mínimo en la vecindad de MxN alrededor de cada pixel de la imagen.
vanherk.m	Realiza el filtrado máx./min de un vector usando un filtro de longitud N. Esta función utiliza el algoritmo de Van Herk(Mathworks, s.f.), realiza los cálculos de los valores máx/min de cada elemento en solamente 3 pasos.

segmentaLinea.m	Realiza la segmentación por línea de cada imagen de entrada mediante el histograma.
histograma.m	Función que devuelve un vector correspondiente al histograma de la imagen.
main.m	Clase principal donde se puede localizar el código de cada etapa del sistema.

3.1 Arquitectura general del sistema

En la Figura 3.1 se mencionan las etapas del sistema: adquisición del documento mediante digitalización, pre-procesamiento y segmentación.

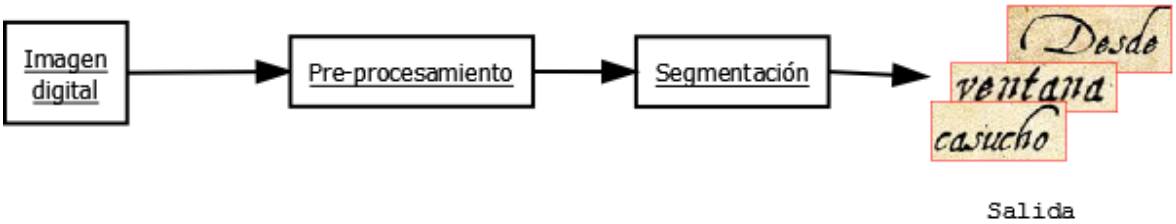


Figura 3.1 Diagrama de bloques del sistema

Este proceso se lleva a cabo utilizando las clases que se describen en la tabla 3.1. El pre-procesamiento de imagen es ejecutado por la clase main.m , en la cual se utilizan unas funciones del toolbox de “Image Porcessing”, así mismo se hace también el llamado a la función de bernsen.m y niblack.m . La segmentación es iniciada por la clase main.m, la cual llama a la función segmentaLinea.m, para mayor detalle en la figura 3.12 se muestra el diagrama de clases.

3.1.1 Documento escaneado

La digitalización de los documentos manuscritos antiguos se realiza mediante un escáner, sin embargo, en nuestro sistema se utilizó una base de datos con imágenes de documentos manuscritos de George Washington que datan desde 1741 a 1799 (Billington, 2015).

3.1.2 Pre-procesamiento

El pre-procesamiento utilizado por el método propuesto busca reducir la mayor cantidad de ruido posible, sin embargo, el pre-procesamiento también tiende a fallar y esto se debe al tipo de papel, estilo de escritura, entre otros factores. El pre-procesamiento en nuestra investigación consta de los siguientes pasos:

a) Escala de grises: Una imagen en escala de grises se obtiene con el objetivo de utilizar dicha imagen para realizar posteriormente la binarización. En la Figura 3.2 se muestra la representación matricial de una imagen, cuyos valores están entre [0, 255] (Cuevas, Zaldívar, & Pérez, 2010), en donde a cada valor x_{mn} se le aplica la ecuación 3.1.

$$I = \begin{bmatrix} x_{11} & x_{12} & \cdots & x_{1n} \\ x_{21} & x_{22} & \cdots & x_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ x_{m1} & x_{m2} & \cdots & x_{mn} \end{bmatrix}$$

Figura 3.2 Matriz de una imagen digital.

$$\boldsymbol{I(x,y) = \frac{R+G+B}{3}} \qquad (3.1)$$

Para realizar la etapa de pre-procesamiento, primero se verifica que la imagen se encuentre en RGB. Esta verificación se realiza para convertir una imagen a escala de grises mediante la función rgb2gray () de la toolbox "*Image processing*" de Matlab. La función recibe como parámetro de entrada una imagen a color y devuelve una nueva imagen en escala de grises.

```
% Convertir imagen RGB a escala de grises
if size(imagenFiltrada,3) == 3
imagenFiltrada=rgb2gray(imagenOriginal);
end
```

Figura 3.3 Código escala de grises.

b) Eliminación de ruido: el método utilizado para eliminar el ruido es el filtro de la Mediana. Este filtro es un método de suavizado que consiste en visitar cada pixel de la imagen y reemplazarlo por la mediana de los pixeles vecinos. La mediana se calcula ordenando los valores de los pixeles vecinos y seleccionando el que queda en medio (Cuevas, Zaldívar, & Pérez, 2010), definido de la siguiente forma:

Considérese un conjunto de valores $x_1, x_2, \ldots, x_n$ que representan una muestra ordenada en forma creciente o decreciente. La mediana se obtiene para valores pares e impares, cuándo n es impar se obtiene el valor central de M_e definida en la ecuación 3.2 para n par (Cuevas, Zaldívar, & Pérez, 2010)

$$\boldsymbol{M_e = x_{\frac{n+1}{2}}} \tag{3.2}$$

Si n es par se obtiene el valor aritmético de los dos valores centrales (ecuación 3.3 para n impar).

$$\boldsymbol{M_e = \frac{x_{\frac{n}{2}} + x_{\frac{n+1}{2}}}{2}} \tag{3.3}$$

Una vez obtenida la imagen en escala de grises se utilizará un método con la finalidad de reducir el ruido en la imagen, mediante el filtro de la mediana de la función medfilt2() de la librería de "*Image processing*" de Matlab. Esta función recibe como parámetro de entrada una imagen, y devuelve una nueva imagen con el ruido reducido. La figura 3.4 ejemplifica el uso de la función medfilt2().

```
%Eliminar ruido mediante el filtro Mediana
imagenFiltrada = medfilt2(imagenFiltrada);
```

Figura 3.4 Código filtro mediana.

c) Binarización: Es una técnica muy importante en el reconocimiento de texto manuscrito, mediante la cual se obtiene una imagen que contiene solo dos valores (0,1), a partir de una imagen en escala de grises. En este paso se optó por utilizar los métodos de umbralización local de Bernsen y Niblack, ambos basados en un operador umbral. El operador umbral consiste en cambiar el valor de intensidad de cada pixel de la imagen, si el valor de intensidad es igual o mayor al umbral, el nuevo valor de intensidad para ese pixel I(x, y) es el máximo, de lo contrario, es el mínimo; de esta forma se obtiene una imagen (Nieto Bonilla, 2010).

Para desarrollar nuestro sistema, se utilizaron los dos métodos mencionados en el párrafo anterior; (ver código en la Figura 3.5). Estos métodos son los que producen mejores resultados de binarización, sin embargo, el método de Niblack además de realizar la umbralización también ayuda a reducir el ruido (Cortés Osorio, Chaves Osorio, & Mendoza Vargas, 2012).

```
% Binarización
imagenFiltrada = bernsen(imagenFiltrada);
imagenFiltrada = niblack(imagenFiltrada);
```

Figura 3.5 Código para obtener los filtros de Bernsen y Niblack.

d) Segmentación

El método propuesto de segmentación se basa en el uso de “Componentes Conectados” (CC) para localizar las palabras. Este método consta de los siguientes pasos:

1) Se localizan todos los componentes conectados. Se entiende por componente conectado a un conjunto de pixeles que se encuentran adyacentes al

pixel en cuestión; dicha vecindad puede ser de 4 u 8 pixeles conectados en su entorno, como se muestra en la figura 3.6.

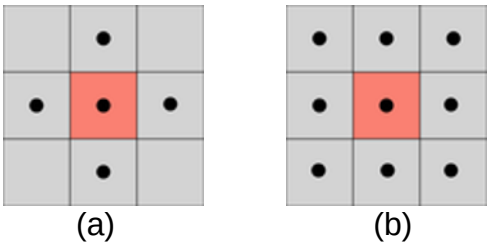


Figura 3.6 Componentes conectados. (a) 4-conectados. (b) 8-conectados.

2) Para cada componente, se valida que los pixeles que lo conforman no correspondan a una palabra incompleta. Para esto, se realiza un recorrido sobre 16 pixeles desde el punto p1 a p4 (ver figura 3.7) de forma vertical descendente. Posteriormente se van recorriendo las columnas de los puntos p1 a p4 hasta llegar a los puntos p2 y p3, validando que no existan pixeles de color negro. En caso de existir pixeles negros, se actualizan los valores obtenidos correspondientes a las coordenadas y tamaño de los objetos localizados en la primera búsqueda de CC.

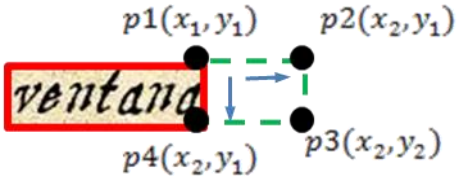


Figura 3.7 Validación de un componente

3) Se obtiene el valor promedio del tamaño de todos los componentes localizados. Se seleccionan los componentes conectados de mayor tamaño que la media y se mandan a la función segmentaLinea () con la finalidad de tratar aquellas palabras que estén unidas con otras, como por ejemplo, las que se muestran en la figura 3.8.

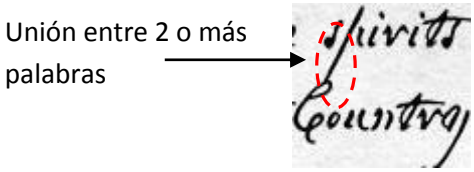


Figura 3.8 Componentes conectados unidos

4) En la función segmentaLinea(), primero se obtiene el histograma de cada uno de los componentes de mayor valor que la media, esto con la finalidad de realizar una segmentación de líneas de texto, como se muestra en la figura 3.9. Posteriormente, se actualizan los valores correspondientes a las coordenadas y tamaño de cada componente que se trató mediante la segmentación por líneas. La función devuelve todos aquellos valores que fueron modificados durante la segmentación.

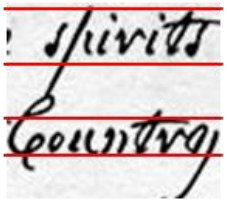


Figura 3.9 Segmentación mediante líneas de texto usando histograma.

5) A cada línea de texto segmentada en el paso anterior, se le aplica el método de componentes conectados, descrito en la sección 3.13, para dividir cada posible palabra que pudiera existir dentro de la línea de texto (ver figura 3.10).

Figura 3.10 Segmentación de componentes conectados por línea de texto.

6) Finalmente, se dibuja un rectángulo alrededor de cada componente conectado. El dibujado se hace con los valores obtenidos durante la validación del

paso anterior. Un ejemplo del uso de componentes conectados se muestra en la figura 3.11.

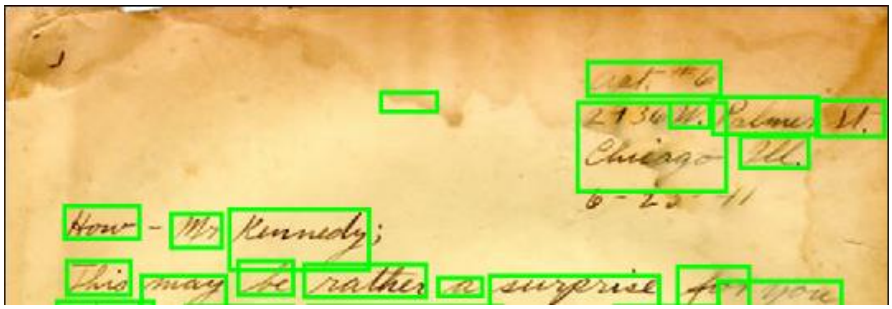


Figura 3.11 Dibujado de componentes conectados (CC).

3.2 Descripción del Software utilizado

"PRISCUS- *Word segmentation*" es un sistema diseñado para localizar y segmentar palabras en textos manuscritos antiguos, mediante la técnica de Componentes Conectados. La figura 3.12 muestra el diagrama de clases del software, el cual se realizó en Matlab, utilizando las librerías de "*Image Processing*". Enseguida se explican cada uno de los componentes del software.

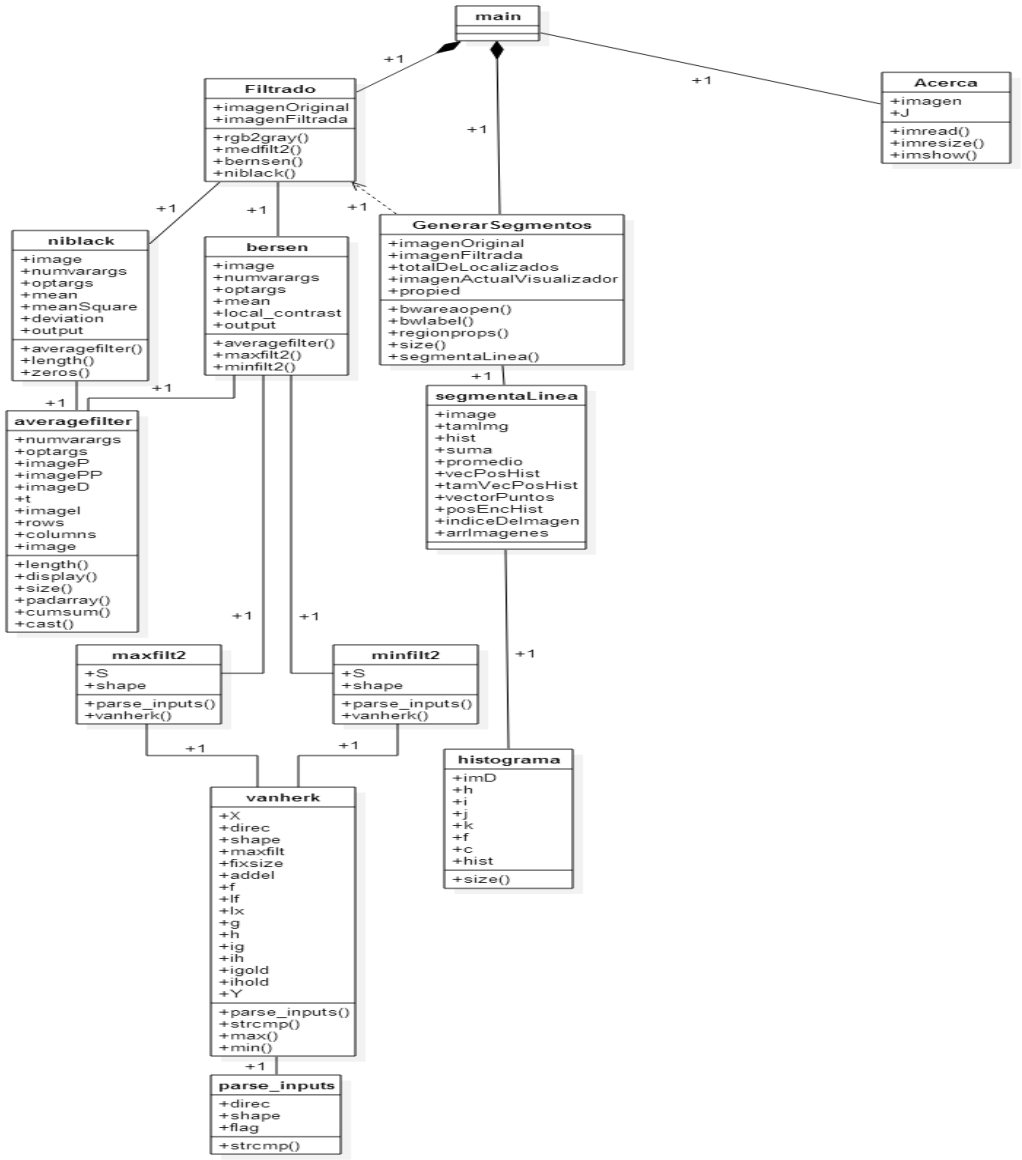


Figura 3.12 Diagrama de clases del proyecto PRISCUS ”Word Segmentation”.

3.3 Descripción de las funciones principales – Toolbox

Las funciones principales utilizadas en el sistema “PRISCUS- Word Segmentation” se describen a continuación. Esta información es descrita con detalle en (Gómez Gil & Cuevas Farfán, 2008)

Bwareaopen()

Esta función, que forma parte de la librería “Image Processing”, elimina de una imagen binaria BW (por sus siglas en inglés Black-White) todos los componentes conectados (objetos) que tienen menos de P pixeles. La sintaxis de la función es: bwareaopen(BW, P). Los valores de entrada de la función son: una imagen binaria BW y un número de P pixeles, donde todos aquellos objetos menores a P pixeles serán removidos. La salida de la función es una imagen que ya no contiene los componentes conectados que se identificaron anteriormente.

Por ejemplo, la figura 3.13 (a) muestra una imagen de la letra “ i ” donde se desean remover componentes; la figura 3.13 (b) muestra una imagen después de ser aplicada la función, donde ya se han eliminado todos objetos que contenían menos de P pixeles, y P=30.

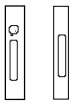


Figura 3.13 (a) Imagen antes de bwareaopen. (b) Imagen después de bwareaopen

Bwlabel()

La función bwlabel() etiqueta todos los pixeles de una imagen, identificando a que componente pertenecen. Los componentes conectados (objetos) en una imagen se etiquetan de la siguiente manera: el 0 se utiliza para etiquetar el fondo, mientras el resto de las etiquetas identifican objetos (1, 2, 3,…, n).

La sintaxis de la función es: [L Ne]=bwlabel(BW,n), donde los valores que regresa son:

1) una matriz L del mismo tamaño que la imagen binaria BW; dicha matriz contiene las etiquetas para los componentes conectados en BW

2) el número total de elementos conectados que fueron encontrados, denotado por la variable Ne (por sus siglas en inglés *Number-elements*).

A continuación se ejemplifica el uso de la función bwlabel(), donde la imagen de entrada es la letra “C” con tamaño 8x11, aplicando el etiquetado de los componentes conectados, se obtiene como resultado una matriz L, donde los 0’s son el fondo y los 1’s son los pixeles correspondientes al carácter.

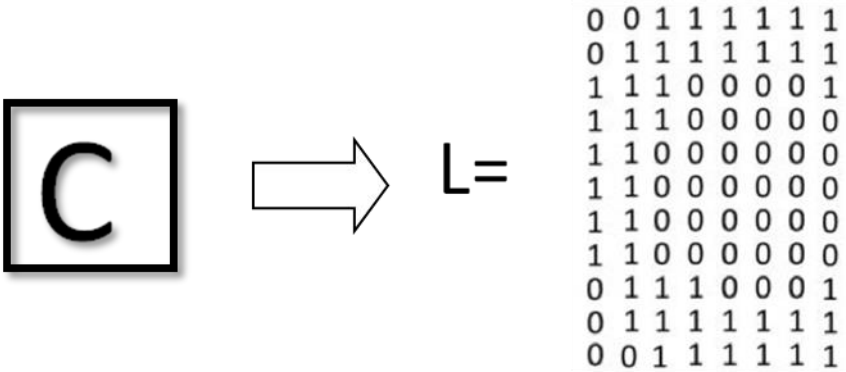


Figura 3.14 Ejemplo de la función bwlabel

Regionprops()

La función regionprops() permite obtener diversas medidas de un objeto, por ejemplo, se puede obtener el ángulo de inclinación, el área (devuelve un escalar con el número real de pixeles del objeto), orientación y perímetro. Los valores que regresa la función se obtienen a partir de la selección de diferentes propiedades (ver tabla 3.2) de un componente conectado en la imagen binaria BW.

La sintaxis de la función es: propied = regionprops(BW, 'BoundingBox'), que recibe como parámetros de entrada: una imagen binaria BW y la propiedad *'BoundingBox'*, misma que se mencionan en la tabla 3.2. Como resultado se obtienen las medidas según la propiedad establecida.

Tabla 3.2 Propiedades de la función regionprops()

Propiedades	Descripción
Área	Devuelve un escalar con el número real de pixeles en una región determinada.
BoundingBox	Devuelve el rectángulo más pequeño que contiene una región especifica, esta propiedad es muy usada por la función de regionprops para obtener las dimensiones de los componentes conectados.
Centroid	Devuelve un vector un vector de 1 por Q que especifica el centro de masa de la región. El primer elemento del vector es la coordenada horizontal (coordenada x) del centroide. El segundo elemento es la coordenada vertical (coordenada y), mientras que los demás elementos son las dimensiones del centroide.
ConvexArea	Devuelve una matriz de p por 2 que especifica el polígono convexo más pequeño que puede contener la región.
ConvexImage	Devuelve una imagen binaria que especifica la envolvente convexa, es decir, un conjunto de puntos x de dimensión n como la intersección de todos los conjuntos convexos que contienen a x.
Eccentricity	Devuelve un escalar que especifica la excentricidad de la elipse de una región. La excentricidad es la relación de distancia entre los focos de la elipse y su longitud del eje principal.
EquivDiameter	Devuelve un escalar que especifica el

	diámetro de un círculo con la misma área que la región. Calculado con la siguiente expresión: $\sqrt{4 * \frac{Área}{pi}}$
EulerNumber	Devuelve un escalar que especifica el número de objetos en la región menos el número de agujeros en estos objetos.
Extent	Devuelve un escalar que especifica la proporción de pixeles en la región de un cuadro total de delimitación. Calculado como el área dividida por el área del cuadrado delimitador.
Extrema	Devuelve una matriz de 8x2 que especifica los puntos extremos en una región (esquinas).
FilledArea	Devuelve un escalar que especifica el número de pixeles en la imagen de llenado.
FilledImage	Devuelve una imagen binaria con todos los hoyos rellenos.
Image	Devuelve una imagen binaria, donde los pixeles corresponden a la región y los otros pixeles están apagados.
MajorAxisLength	Devuelve un escalar que especifica la longitud (en pixeles) del eje mayor de la elipse.
MinorAxisLength	Devuelve un escalar que especifica la longitud (en pixeles) del eje menor de la elipse.
Orientation	Devuelve el ángulo entre eje x y el eje y de la elipse. El valor devuelto esta en grados, que van

	desde -90 hasta 90 grados.
Perimeter	Devuelve un valor que especifica la distancia alrededor del límite de la región que se esta trabajando.
PixelIdxList	Devuelve un vector de p-elementos que contiene los índices lineales de los pixeles de la región.
PixelList	Devuelve una matriz de p por Q que especifica la ubicación de los pixeles de la región de interés.
Solidity	Devuelve el porcentaje de los pixeles en la envolvente convexa de la región. Calculado como área/areaConvexa.

Capítulo 4

Experimentos y resultados

En este capítulo se presentan los experimentos realizados con el algoritmo de segmentación en sus diversas versiones, así como también se presentan los resultados obtenidos durante dichos experimentos.

4.1 Evaluación de resultados

Para evaluar el rendimiento del método propuesto, se utilizó una métrica correspondiente al porcentaje de datos correctos y se calcula dividiendo el número de palabras correctas entre el total de palabras del documento, como se específica en la ecuación 4.1 utilizada para medir la tasa de rendimiento.

$$Tasa\ de\ rendimiento = \frac{palabras\ correctas}{total\ de\ palabras\ de\ la\ imagen} \tag{4.1}$$

4.2 Error de segmentación

Para evaluar la tasa de error del método propuesto, se utilizó una métrica que corresponde al porcentaje de datos incorrectos, y es calculada dividiendo el número de palabras incorrectas entre el total de palabras del documento (ecuación 4.2 utilizada para medir la tasa de error).

$$Error = \frac{palabras\ incorrectas}{total\ de\ palabras\ de\ la\ imagen} \tag{4.2}$$

4.3 Resultados

En esta sección se presentan los experimentos que se realizaron para probar el método propuesto, los resultados obtenidos y las medidas con las que fueron evaluados.

4.3.1 Experimentos realizados

Se llevaron a cabo 3 experimentos distintos, en los cuales se realizaron pruebas con 15 imágenes diferentes (Apéndice I). El propósito de realizar estos experimentos es comprobar el desempeño del método propuesto en comparación con otros métodos de segmentación.

Pre-procesamiento

Durante la etapa de pre-procesamiento se proponen 4 distintos métodos utilizando los filtros mencionados en la tabla 4.1. Los filtros implementados fueron utilizados para eliminar ruido y limpiar los documentos de algunas imperfecciones, por ejemplo, documentos mojados o deteriorados por el tiempo.

Tabla 4.1 Filtros utilizados en las pruebas

Filtros		Abreviatura
Mediana	-	F1
Bernsen	-	F2
Niblack	-	F3
Sauvola	-	F4

Experimento 1

En la etapa de pre-procesamiento de este experimento se hizo uso de los filtros F2, F3 y F4. Ejecutados en dicho orden. El método de segmentación utilizado fue mediante componente conectado. En el experimento 1 los pasos que se siguieron se describen a continuación:

a) Primero se realizó el pre-procesamiento para mejorar la imagen de ruido y poder ser utilizada por el método de segmentación.
b) Se empleó la técnica de componente conectado para buscar todos los objetos dentro de la imagen.
c) Se segmentaron directamente sin ningún tipo de validación todos los objetos que fueron encontrados en el paso anterior.
d) Se dibujan con rectángulos los objetos localizados (Figura 4.1).

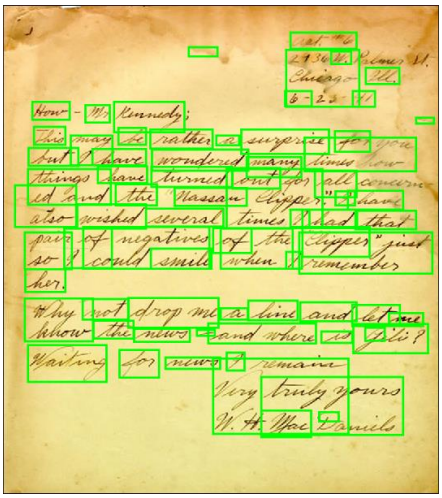


Figura 4.1 Segmentación del experimento 1.

Los resultados obtenidos se pueden observar en la tabla 4.2, donde se puede visualizar que el mejor resultado obtenido lo tuvo la figura del apéndice I.1 (d) con un 57% de eficiencia.

Tabla 4.2 Resultados del experimento 1

Método de Segmentación: Componente Conectado						
No. Experimento	Figura 3.15	Correctos	Incorrectos	Total	Total palabras	Eficiencia
1	(a)	45	27	72	92	49%
	(b)	7	202	209	140	5%
	(c)	15	5	20	158	9%
	(d)	42	14	56	74	**57%**
					PROMEDIO	**30%**

Experimento 2 (Versión 1)

Los filtros utilizados para el pre-procesamiento durante el experimento fueron el filtro F1, F2, F3 y F4, ejecutados en dicho orden. El método de segmentación utilizado fue mediante componentes conectados. En el experimento 2, los pasos que se siguieron se describen a continuación:

a) Primero se realizó el pre-procesamiento para mejorar y limpiar la imagen de ruido.
b) Se utilizó la técnica de componente conectado para buscar todos los objetos dentro de la imagen.
c) Se segmentaron directamente sin ningún tipo de validación todos los objetos que fueron encontrados en el paso anterior.
d) Los objetos localizados se dibujan con rectángulos. (Figura 4.2).

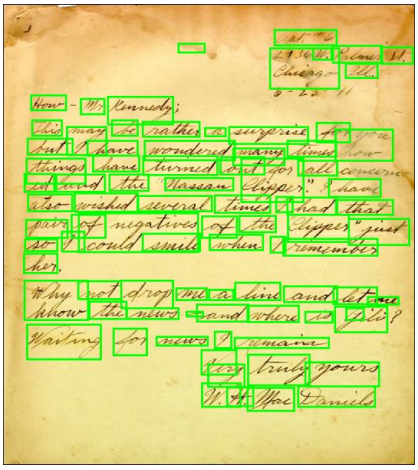


Figura 4.2 Segmentación del experimento 2 (versión 1)

La ventaja que nos ofrece el experimento 2 con respecto al experimento 1 ocurre durante la etapa de pre-procesamiento. Esta mejora del experimento 2 se debe a la aplicación del filtro de la mediana que se utilizó para reducir el ruido de la imagen. Con este experimento se obtuvieron mejores resultados en todas las pruebas realizadas (Tabla 4.3).

Tabla 4.3 Resultados del experimento 2 (versión 1)

Método de Segmentación: Componente Conectado						
No. Experimento	Figura 3.15	Correctos	Incorrectos	Total	Total palabras	Eficiencia
2-a	(a)	69	16	85	92	75%
	(b)	40	418	458	140	29%
	(c)	57	70	127	158	36%
	(d)	57	10	67	74	**77%**
					PROMEDIO	**54%**

Experimento 2 (Versión 2)

Los filtros utilizados para el pre-procesamiento durante el experimento fueron los filtros F1, F2, F3 y F4, los mismos que en experimento 2 versión 1. El método de segmentación utilizado fue mediante componentes conectados en conjunto con una nueva heurística que valida que una palabra no esté incompleta (detalles en sección 3.1.3). En el experimento 2 (versión 2), los pasos que se realizaron fueron los siguientes:

a) Primero se realizó el pre-procesamiento y se aplicó el filtro de la mediana para reducir el ruido de la imagen.
b) Se utilizó la técnica de componente conectado para buscar todos los objetos dentro de la imagen.
c) Se realizó una validación de que cada palabra localizada no se encontrara incompleta, tomando en cuenta el número de espacios en blanco entre objeto y objeto (Figura 4.3). Esta heurística está explicada con detalle en la sección (poner aquí la sección donde la explicas)
d) Se segmentaron todos los objetos que fueron encontrados, de acuerdo a lo descrito en los pasos anteriores.
e) Se dibujan con rectángulos los objetos localizados (Figura 4.4).

(a) (b)

Figura 4.3 Implementación de la heurística: (a) antes. (b) después.

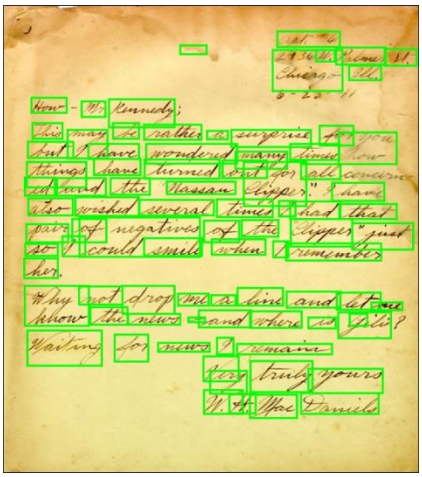


Figura 4.4 Segmentación del experimento 2 (versión 2).

La mejora realizada en esta versión del experimento se ve reflejada en el incremento del número de palabras correctas, debido a que en pruebas anteriores algunas palabras se encontraban incompletas. Los resultados obtenidos se incrementaron principalmente en la imagen de la figura I.1 (a) ubicada en el apéndice I, teniendo como mejor tasa de rendimiento del 78% (tabla 4.4).

Tabla 4.4 Resultados del experimento 2 (versión 2)

Método de Segmentación: Componente Conectado, incluyendo heurística de espacios						
No. Experimento	Figura 3.15	Correctos	Incorrectos	Total	Total palabras	Eficiencia
2-b	(a)	89	77	166	92	**78%**
	(b)	20	543	563	140	14%
	(c)	64	101	165	158	41%
	(d)	58	19	77	74	**78%**
	(e)	24	249	273	98	24%
	(f)	30	253	283	99	30%
	(g)	101	60	161	323	31%
	(h)	124	53	179	332	37%
					PROMEDIO	**44%**

Experimento 2 (Versión 3)

Los filtros utilizados para el pre-procesamiento durante el experimento fueron los filtros F1, F2 y F3. El método de segmentación utilizado fue mediante componentes conectados, con la mejora de la heurística que valida que una palabra no esté unida con otra palabra (detalles en sección 3.1.3). En el experimento 2 (versión 3), los pasos que se realizaron fueron los siguientes:

a) Primero se realizó el pre-procesamiento, se aplicó el filtro de la mediana para reducir el ruido de la imagen (denotada por la abreviatura F1, tabla 4.1).
b) Se utilizó la técnica de componente conectado para buscar todos los objetos dentro de la imagen.
c) Se realizó una validación de que cada palabra localizada no se encontrara incompleta tomando en cuenta el número de espacios en blanco entre objeto y objeto (figura 4.3).
d) Se realizó la validación de que las palabras no estuvieran unidas unas con otras (figura 3.8), para lo que se realizó una segmentación mediante la detección de líneas de texto (figura 3.9).
e) A cada una de las líneas de texto se les aplicó segmentación mediante componentes conectados, debido a que existe la posibilidad de que la línea de texto segmentada anteriormente contenga más de una palabra.
f) Se segmentaron todos los objetos que fueron encontrados en los pasos anteriores.
g) Se dibujan con rectángulos los objetos localizados (figura 4.5).

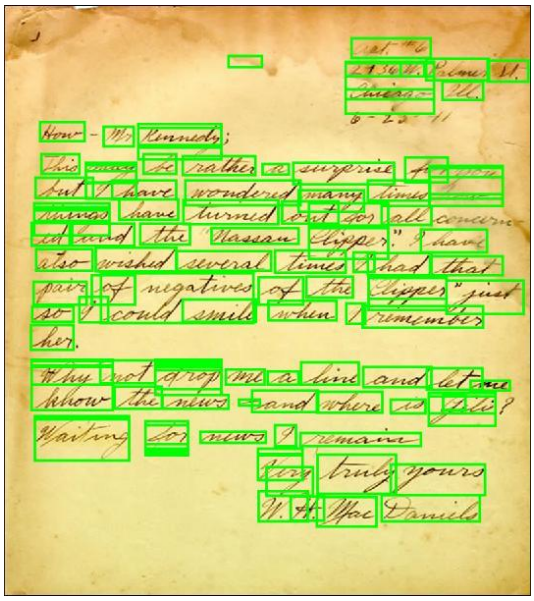


Figura 4.5 Segmentación del experimento 2 (versión 3).

En los resultados obtenidos para esta versión del experimento, se observa que el número palabras incorrectas se ve reducido considerablemente, y el número de palabras correctas se mantiene en el margen con una leve reducción de su eficiencia. La reducción del ruido (palabras incorrectas) es favorable para el trabajo a futuro del reconocimiento de texto, debido que al procesar todos los objetos encontrados, la mayoría serán palabras correctas. Los mejores resultados de la versión 3 se obtuvieron usando la imagen de la figura I.1 (a) ubicada en el apéndice I, teniendo una tasa de rendimiento del 92% (tabla 4.5).

Tabla 4.5 Resultados del experimento 2 (versión 3)

Método de Segmentación: Componente Conectado, incluyendo heurística de espacios –Mejorado						
No. Experimento	Figura 3.15	Correctos	Incorrectos	Total	Total palabras	Eficiencia
2-c	(a)	85	18	103	92	**92%**
	(b)	20	564	584	140	14%
	(c)	69	66	135	158	44%
	(d)	57	12	69	74	77%
	(e)	20	357	377	98	20%
	(f)	18	345	363	99	18%
	(g)	76	91	167	323	24%
	(h)	122	51	173	332	37%
	(i)	97	35	132	179	54%
	(j)	115	66	181	182	63%
	(k)	116	61	177	204	57%
	(l)	116	39	155	206	56%
	(m)	168	43	211	207	81%
	(n)	168	33	201	259	65%
	(ñ)	110	45	155	169	65%
					PROMEDIO	**51%**

Experimento 3

Los filtros utilizados para el pre-procesamiento durante el experimento fueron los filtros F1, F2, F3 y F4. El método de segmentación utilizado fue una combinación de la técnica de detección de líneas de texto y componentes conectados. En el experimento 3, los pasos que se realizaron fueron los siguientes:

a) Primero se realizó el pre-procesamiento, aplicando todos los filtros mencionados en la tabla 4.1.
b) Se aplicó la técnica de detección de líneas de texto.
c) Cada línea de texto fue tratada con la técnica de componentes conectados para la extracción de todas las palabras localizadas dentro de cada línea.
d) Se segmentan todos los objetos que fueron encontrados del paso anterior.

Los resultados obtenidos se pueden observar en la tabla 4.6, donde se puede visualizar que el mejor resultado obtenido lo tuvo la figura I.1 (a) ubicada en el apéndice I con una tasa de rendimiento del 88%.

Tabla 4.6 Resultados del experimento 3

Método de Segmentación: Por línea						
No. Experimento	Figura 3.15	Correctos	Incorrectos	Total	Total palabras	Eficiencia
3	(a)	81	82	163	92	**88%**
	(b)	19	671	690	140	14%
	(c)	70	125	195	158	44%
	(d)	33	66	99	74	45%
					PROMEDIO	**48%**

4.3.2 Análisis de resultados

De los 3 experimentos realizados con el propósito de poder compararlos y obtener cuál de estos experimentos fue el que consiguió mejores resultado, se obtuvo lo siguiente:

Tabla 4.7 Resultados los experimentos

No. Experimento	Eficiencia
1	57%
2 (Versión 1)	77%
2 (Versión 2)	78%
2 (Versión 3)	92%
3	88%

En base a la tabla 4.7 de los resultados obtenidos, se recomienda utilizar el algoritmo del experimento 2 (Versión 3), debido que en este experimento se destaca un algoritmo de segmentación más eficiente, los resultado alcanzados durante el experimento 2 superan los obtenidos en el trabajo de investigación realizado por Senior (Senior & Robinson, 1998) en el cuál se obtuvo un 87%.

Sin embargo, comparando el algoritmo desarrollado contra el algoritmo implementado por Fernández y colaboradores (Fernández Mota, Lladós, & Fornés, 2014), ambos algoritmos utilizaron imágenes de documentos del presidente George Washington, se obtuvo que el algoritmo de Fernández fue mejor con una eficiencia de del 92.70, mientras que el algoritmo de segmentación desarrollado en PRISCUS fue de 81%.

Capítulo 5

Conclusiones y trabajo a futuro

5.1 Conclusiones

En esta investigación se propuso y se implementó un método para la segmentación de palabras, en documentos manuscritos antiguos, utilizando la base de datos de la biblioteca del Congreso de Estados Unidos con imágenes de documentos de George Washington que datan de 1741- 1799, así como en otros documentos. La propuesta se basa en una combinación de la técnica de Componentes Conectados y segmentación por línea de texto.

La razón por la que se utilizaron los Componentes Conectados es porque es una técnica para encontrar todos los objetos que son candidatos a ser palabras, es decir, su funcionamiento se basa en encontrar objetos en donde sus pixeles internos estén unidos y de esta manera puedan agruparse. La segmentación de línea se utilizó, basándose en la idea propuesta por Manmatha y Rothfer (Manmatha & Rothfeder, 2005) motivados en el estilo de escritura de las personas, es decir, existen culturas y regiones que escriben de forma horizontal, o bien, de forma vertical.

El método propuesto para la segmentación de palabras utiliza tres etapas diferentes: 1) aplicación de la técnica de Componentes Conectados a toda la imagen, 2) segmentación de línea a los objetos extraídos del paso 1 que sean de un tamaño mayor que el promedio de los demás objetos extraídos, 3) a cada línea segmentada del paso anterior, se le aplica nuevamente la técnica de componentes conectados para obtener finalmente todas las posibles palabras del documento. Esta combinación de pasos mostró mejorar la segmentación de palabras en comparación con la utilización de estas técnicas de manera separada. La técnica se utilizó usando 3 experimentos con imágenes de diferentes características, obteniéndose como mejores eficiencias de cada experimento respectivamente

57% para el experimento 1. En el experimento 2 se realizaron 3 versiones: en la versión 1 se obtuvo el 77%, en la versión 2 se logró el 78% mientras que en la versión 3 se obtuvo un 92% de eficiencia. En el experimento 3 se obtuvo una eficiencia de 88%.

El presente trabajo aporta una heurística novedosa para atacar el problema de la segmentación de palabras en los documentos manuscritos antiguos. Una ventaja importante del algoritmo propuesto, que fue implementado en el proyecto priscus PRISCUS (Gómez Gil & Cuevas Farfán, 2008) y que se nombró "*Word Segmentation*", es que no genera muchos objetos que no corresponden a palabras del texto. Esto permite que, una vez conectada esta parte de segmentación en PRISCUS, la etapa de reconocimiento de este último será más eficiente.

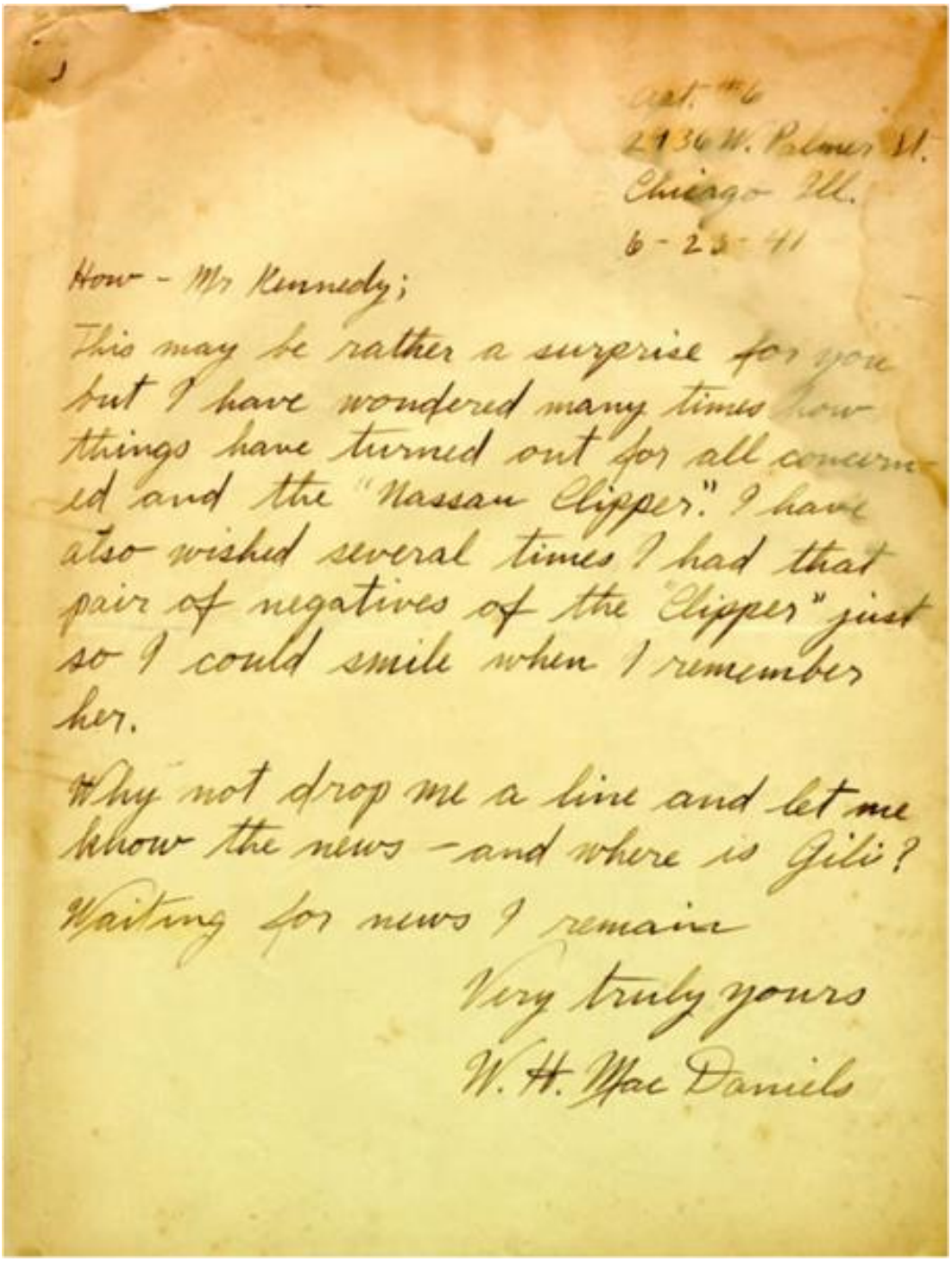
5.2 Trabajo a futuro

Existen varias técnicas involucradas relacionadas con el método propuesto en esta investigación. Esto implica que se puede seguir indagando y mejorando muchas de estas técnicas para lograr mejores resultados que los obtenidos actualmente. Al usar el método de segmentación de esta investigación, se notó que en algunas ocasiones aparecen palabras cortadas, debido a que el espacio entre cada una de ellas es levemente mayor a 16 pixeles. Entonces, una posible mejorar al método propuesto podría conseguirse al implementar una heurística que, de manera automática, detecte si los “n” pixeles que se encuentran entre carácter y carácter o entre palabra y palabra, corresponden a espacios en blanco. Asimismo, puede analizarse automáticamente si existe mucho espacio entre carácter y carácter; esto evitaría lo que sucede con el algoritmo actualmente que da como resultado una palabra cortada.

En este trabajo se llegó hasta la etapa de segmentación, sin embargo, como trabajo a futuro también se considera la siguiente etapa, es decir, el reconocimiento de las palabras segmentadas. En el reconocimiento se pudiesen implementar algoritmos cuyo objetivo sea determinar la ubicación de la palabra reconocida dentro del documento digitalizado, a través de la localización de cada componente conectado. Esto se debe a que en la etapa de segmentación las palabras ubicadas se encuentran en desorden.

Apéndice I

I. Imágenes de Documentos



Apt. #6
2136 W. Palmer St.
Chicago Ill.
6-23-41

How - Mr Kennedy;

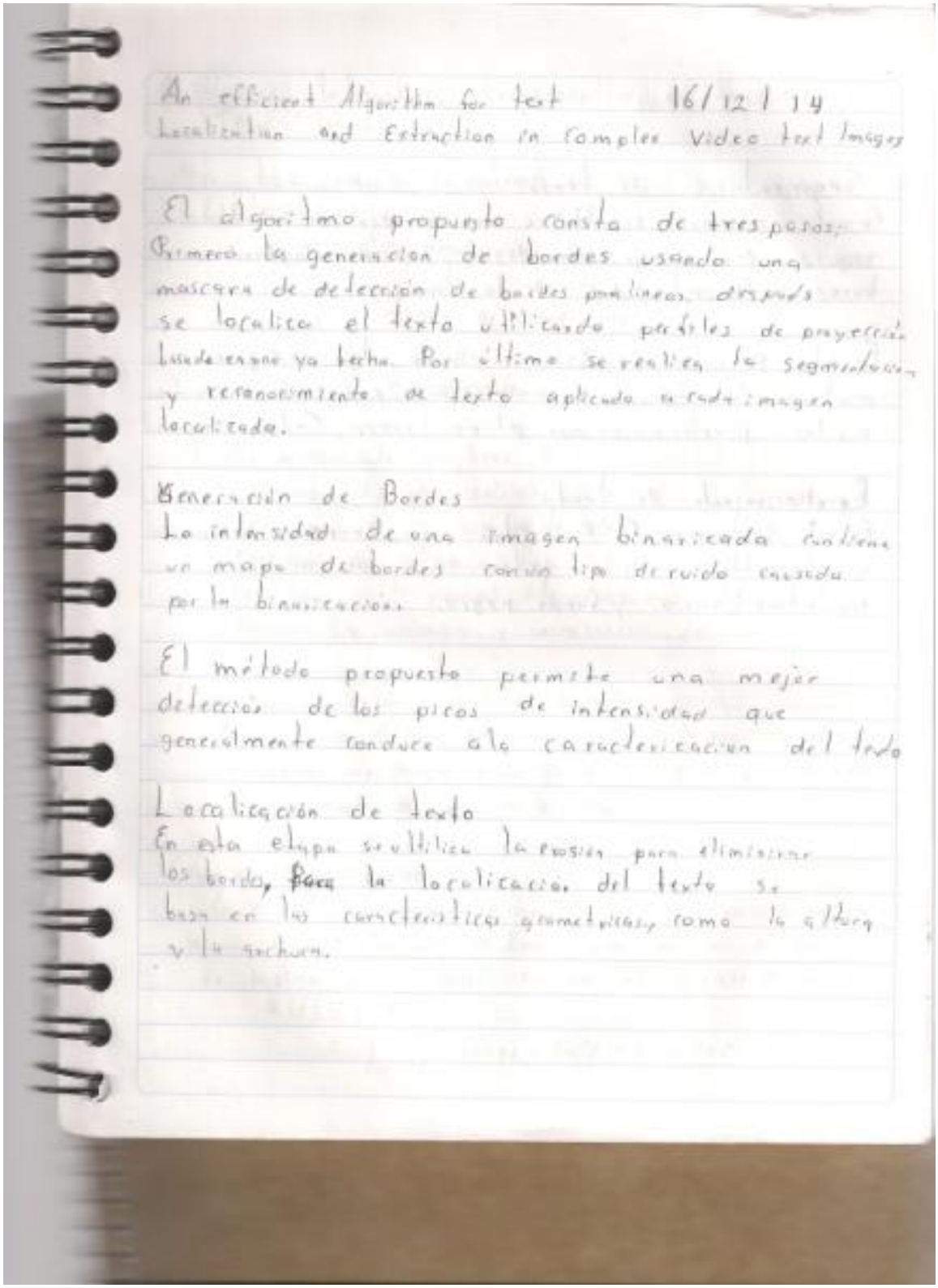
This may be rather a surprise for you but I have wondered many times how things have turned out for all concerned and the "Nassau Clipper." I have also wished several times I had that pair of negatives of the "Clipper" just so I could smile when I remember her.

Why not drop me a line and let me know the news - and where is Gili?

Waiting for news I remain

Very truly yours
W. H. Mac Daniels

Figura I. 1 Carta a Kennedy



An efficient Algorithm for text 16/12/14
Localization and Extraction in Complex Video text Images

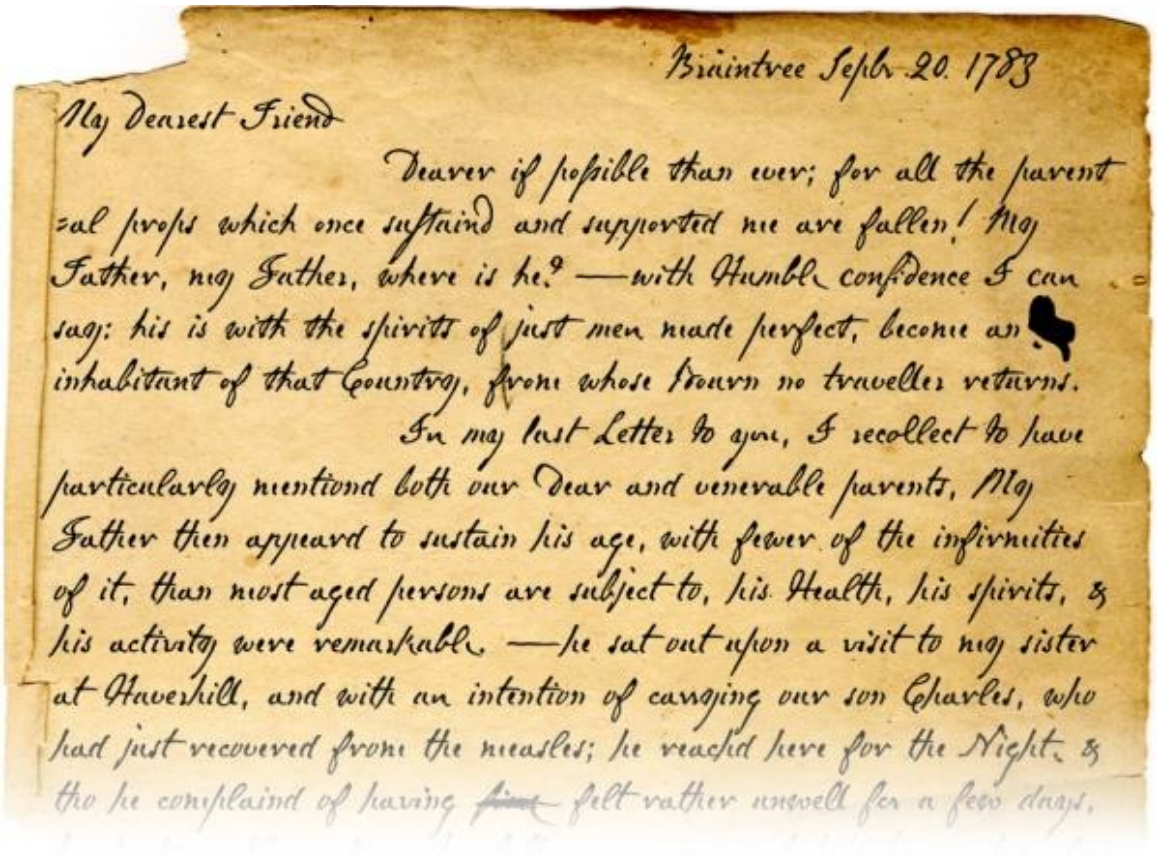
El algoritmo propuesto consta de tres pasos: Primero la generación de bordes usando una mascara de detección de bordes puntiagudos, después se localiza el texto utilizando perfiles de proyección usado en uno ya hecho. Por último se realiza la segmentación y reconocimiento de texto aplicado a cada imagen localizada.

Generación de Bordes
La intensidad de una imagen binarizada contiene un mapa de bordes con un tipo de ruido causado por la binarización.

El método propuesto permite una mejor detección de los picos de intensidad que generalmente conduce a la caracterización del texto

Localización de texto
En esta etapa se utiliza la erosión para eliminar los bordes, ~~para~~ la localización del texto se basa en las características geométricas, como la altura y la anchura.

Figura I. 2 Hoja de libreta.



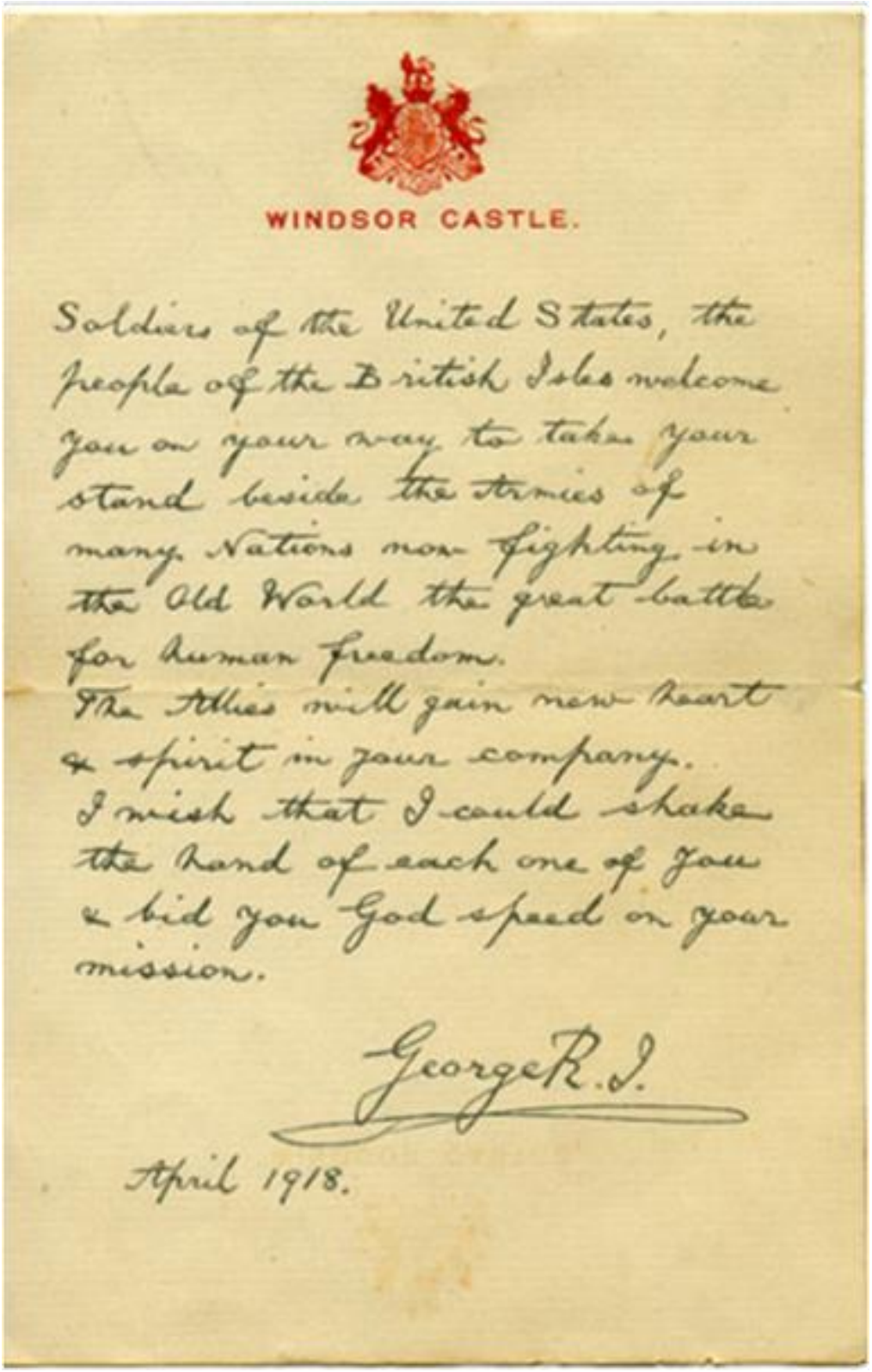
Braintree Sept. 20. 1783

My Dearest Friend

Dearer if possible than ever; for all the parent-al props which once sustaind and supported me are fallen! My Father, my Father, where is he? —with Humble confidence I can say: his is with the spirits of just men made perfect, become an inhabitant of that Country, from whose bourn no traveller returns.

In my last Letter to you, I recollect to have particularly mentiond both our Dear and venerable parents, My Father then appeard to sustain his age, with fewer of the infirmities of it, than most aged persons are subject to, his Health, his spirits, & his activity were remarkable —he sat out upon a visit to my sister at Haverhill, and with an intention of carrying our son Charles, who had just recovered from the measles; he reachd here for the Night, & tho he complaind of having ~~[illegible]~~ felt rather unwell for a few days,

Figura l. 3 Carta de Abigail Smith Adams



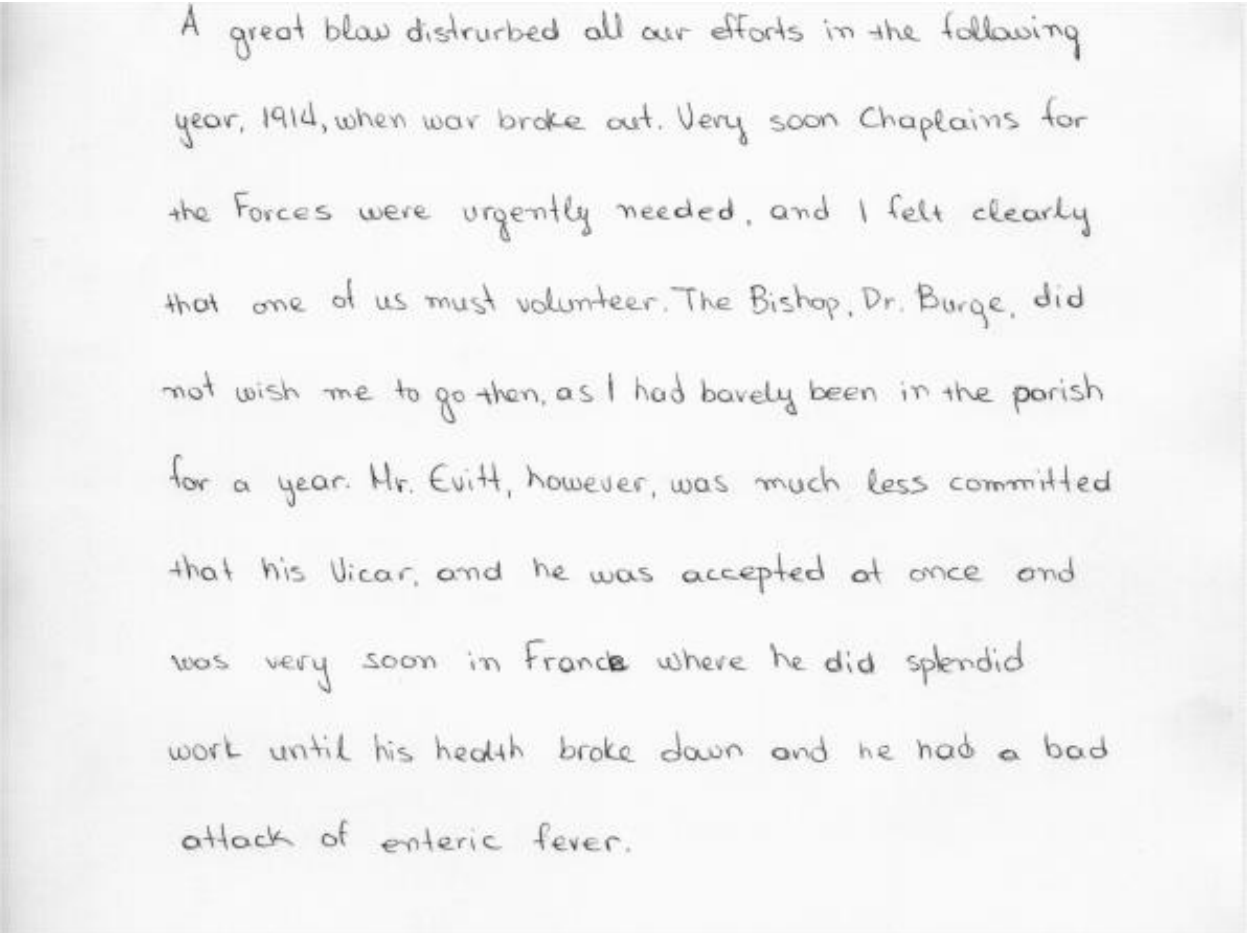
WINDSOR CASTLE.

Soldiers of the United States, the people of the British Isles welcome you on your way to take your stand beside the armies of many Nations now fighting in the Old World the great battle for human freedom.
The Allies will gain new heart & spirit in your company.
I wish that I could shake the hand of each one of you & bid you God speed on your mission.

George R.I.

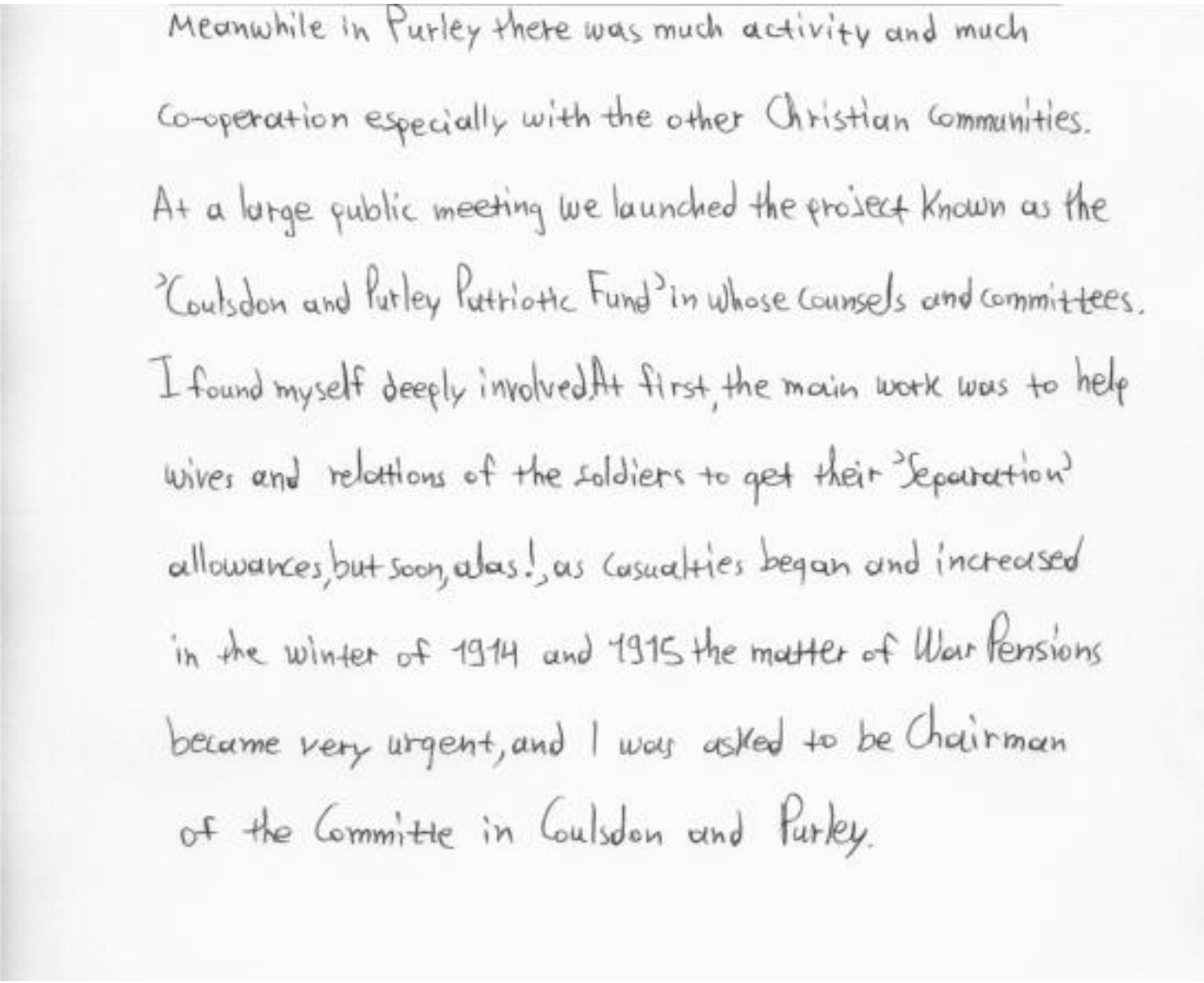
April 1918.

Figura l. 4 Carta de George R.J.



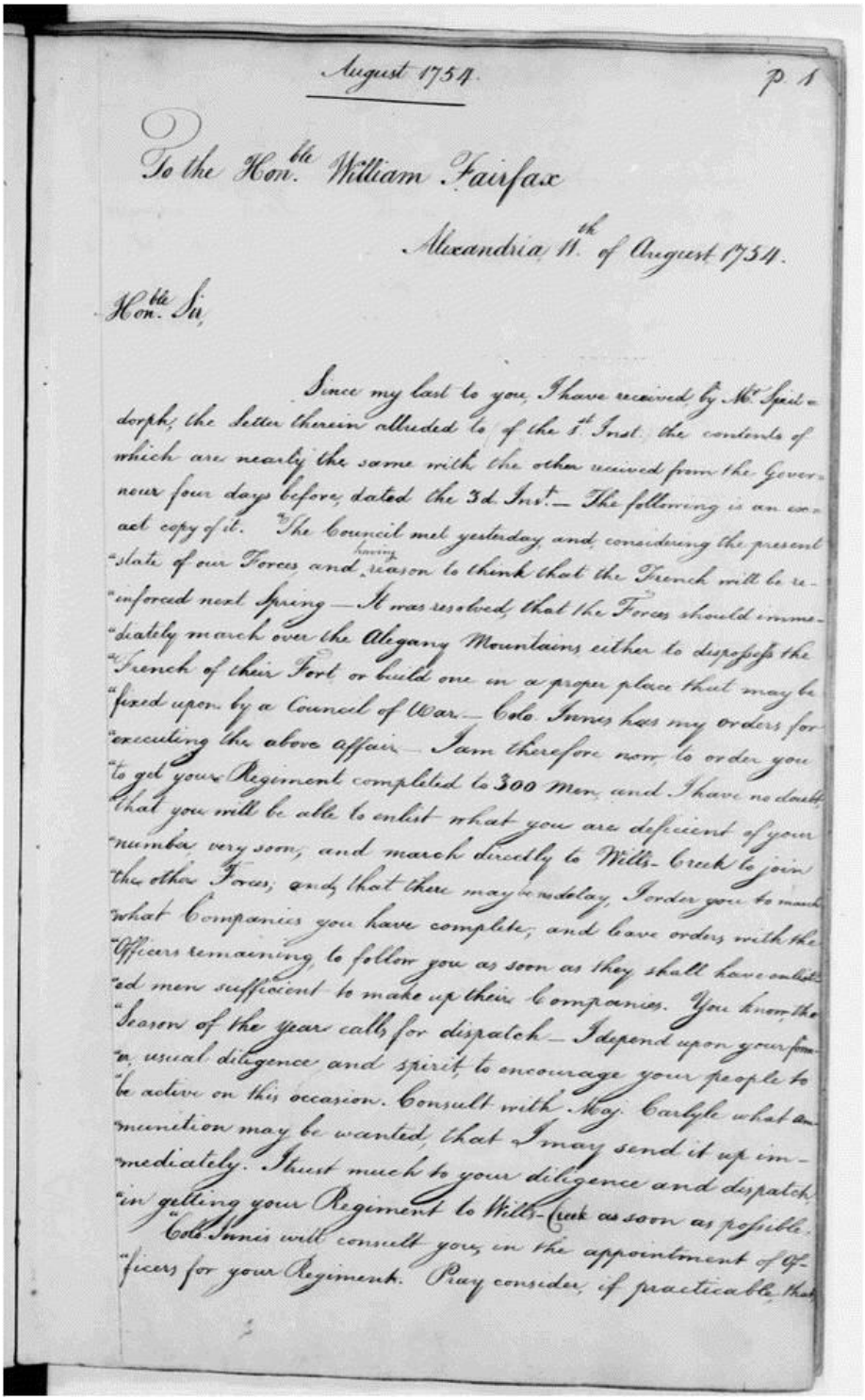
A great blow distrurbed all our efforts in the following year, 1914, when war broke out. Very soon Chaplains for the Forces were urgently needed, and I felt clearly that one of us must volunteer. The Bishop, Dr. Burge, did not wish me to go then, as I had barely been in the parish for a year. Mr. Evitt, however, was much less committed that his Vicar, and he was accepted at once and was very soon in France where he did splendid work until his health broke down and he had a bad attack of enteric fever.

Figura I. 5 IAM Handwriting Database



Meanwhile in Purley there was much activity and much
Co-operation especially with the other Christian Communities.
At a large public meeting we launched the project Known as the
'Coulsdon and Purley Patriotic Fund' in whose Counsels and committees.
I found myself deeply involved. At first, the main work was to help
wives and relations of the soldiers to get their 'Separation'
allowances, but soon, alas!, as Casualties began and increased
in the winter of 1914 and 1915 the matter of War Pensions
became very urgent, and I was asked to be Chairman
of the Committe in Coulsdon and Purley.

Figura I. 6 IAM Handwriting Database



August 1754. p. 1

To the Hon.ble William Fairfax

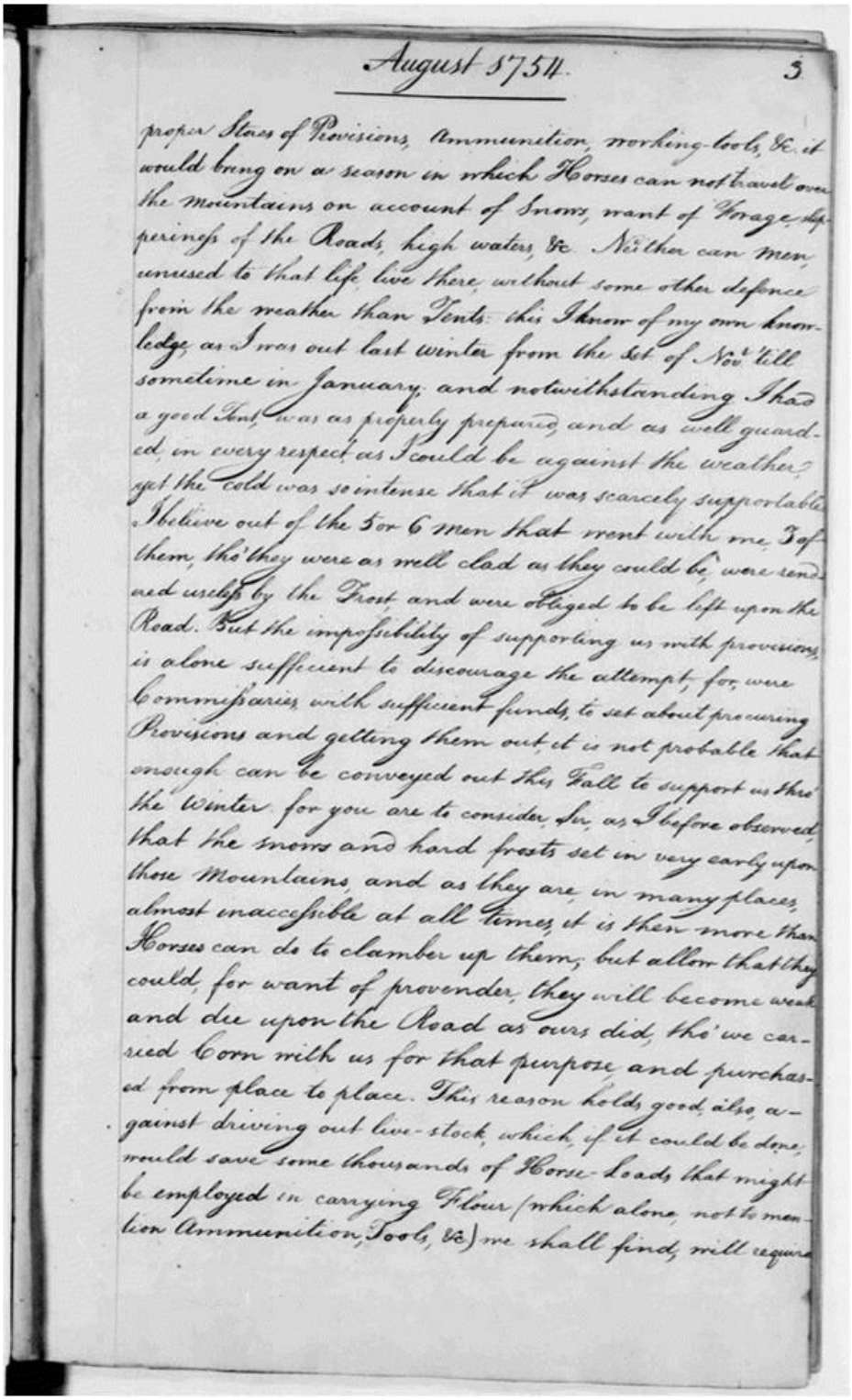
Alexandria 11th of August 1754.

Hon.ble Sir,

Since my last to you, I have received by Mr. Speidorph, the letter therein alluded to of the 5th Inst. the contents of which are nearly the same with the other received from the Governour four days before, dated the 3d Inst. — The following is an exact copy of it. "The Council met yesterday and considering the present "state of our Forces and having reason to think that the French will be re-"inforced next Spring — It was resolved, that the Forces should imme-"diately march over the Allegany Mountains either to dispossess the "French of their Fort or build one in a proper place that may be "fixed upon by a Council of War — Colo. Innes has my orders for "executing the above affair — I am therefore now to order you "to get your Regiment completed to 300 Men, and I have no doubt, "that you will be able to enlist what you are deficient of your "numbr very soon, and march directly to Wills-Creek to join "the other Forces, and, that there may be no delay, I order you to march "what Companies you have complete, and leave orders with the "Officers remaining to follow you as soon as they shall have enlist-"ed men sufficient to make up their Companies. You know the "Season of the year calls for dispatch — I depend upon your fur-"ther usual diligence and spirit to encourage your people to "be active on this occasion. Consult with Mr. Carlyle what am-"munition may be wanted, that I may send it up im-"mediately. I trust much to your diligence and dispatch, "in getting your Regiment to Wills-Creek as soon as possible.

"Colo. Innes will consult you in the appointment of Of-"ficers for your Regiment. Pray consider, if practicable, that

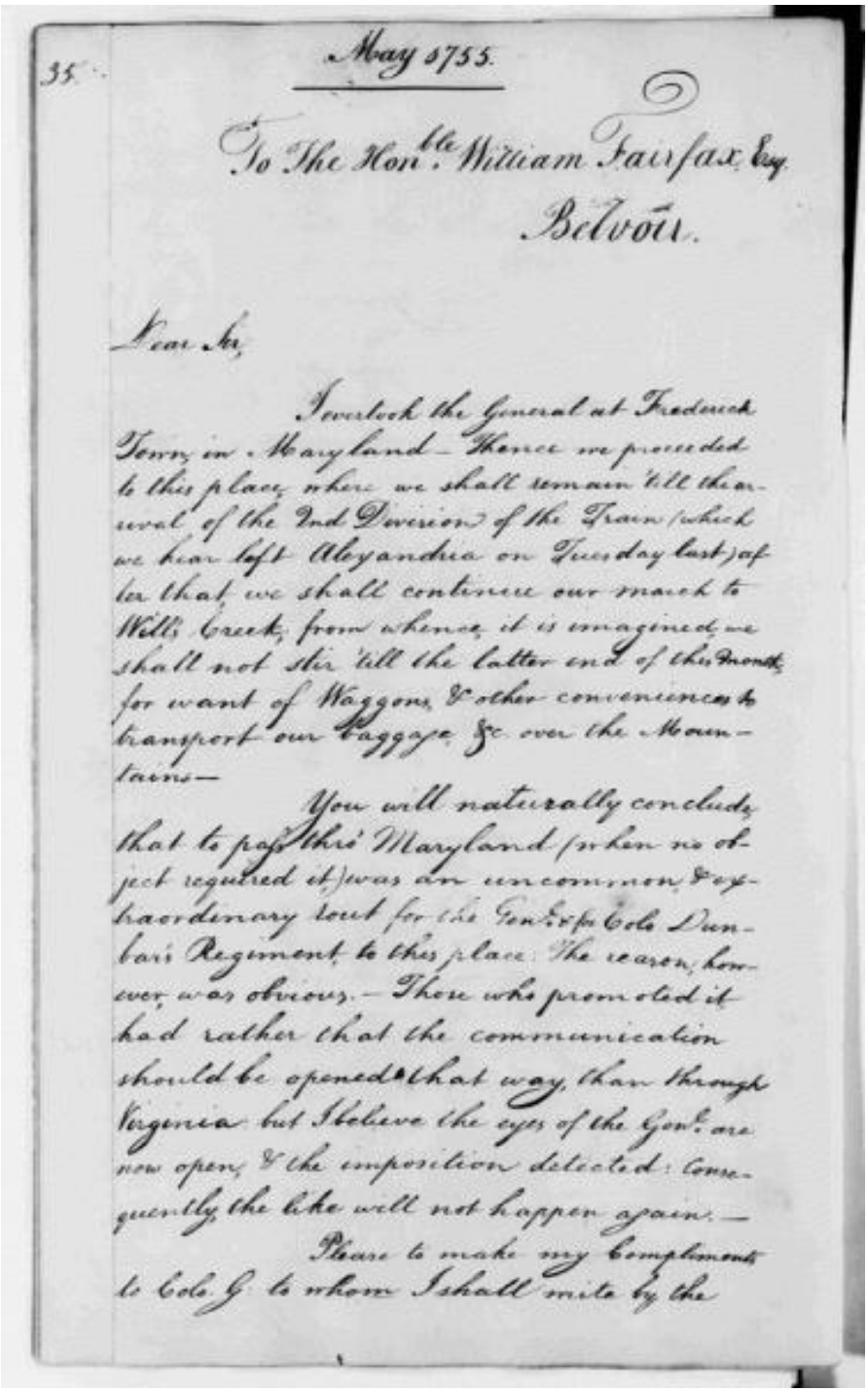
Figura I. 7 Carta de George Washington



August 1754. 3

proper Stores of Provisions, Ammunition, working tools, &c. it would bring on a season in which Horses can not travel over the Mountains on account of Snows, want of Forage, Slipperiness of the Roads, high waters, &c. Neither can men unused to that life live there, without some other defence from the weather than Tents: this I know of my own knowledge, as I was out last winter from the 1st of Nov.r till sometime in January, and notwithstanding I had a good Tent, was as properly prepared, and as well guarded, in every respect, as I could be against the weather, yet the cold was so intense that it was scarcely supportable. I believe out of the 5 or 6 men that went with me, 3 of them, tho' they were as well clad as they could be, were rendered useless by the Frost, and were obliged to be left upon the Road. But the impossibility of supporting us with provisions, is alone sufficient to discourage the attempt, for, were Commissaries, with sufficient funds, to set about procuring Provisions and getting them out it is not probable that enough can be conveyed out this Fall to support us thro' the Winter for you are to consider, Sir, as I before observed, that the snows and hard frosts set in very early upon those Mountains, and as they are, in many places, almost inaccessible at all times, it is then more than Horses can do to clamber up them; but allow that they could, for want of provender, they will become weak and die upon the Road as ours did, tho' we carried Corn with us for that purpose and purchased from place to place. This reason holds good, also, against driving out live-stock, which, if it could be done, would save some thousands of Horse-loads that might be employed in carrying Flour (which alone, not to mention Ammunition, Tools, &c) we shall find, will require

Figura I. 8 Carta de George Washington



35.

May 1755.

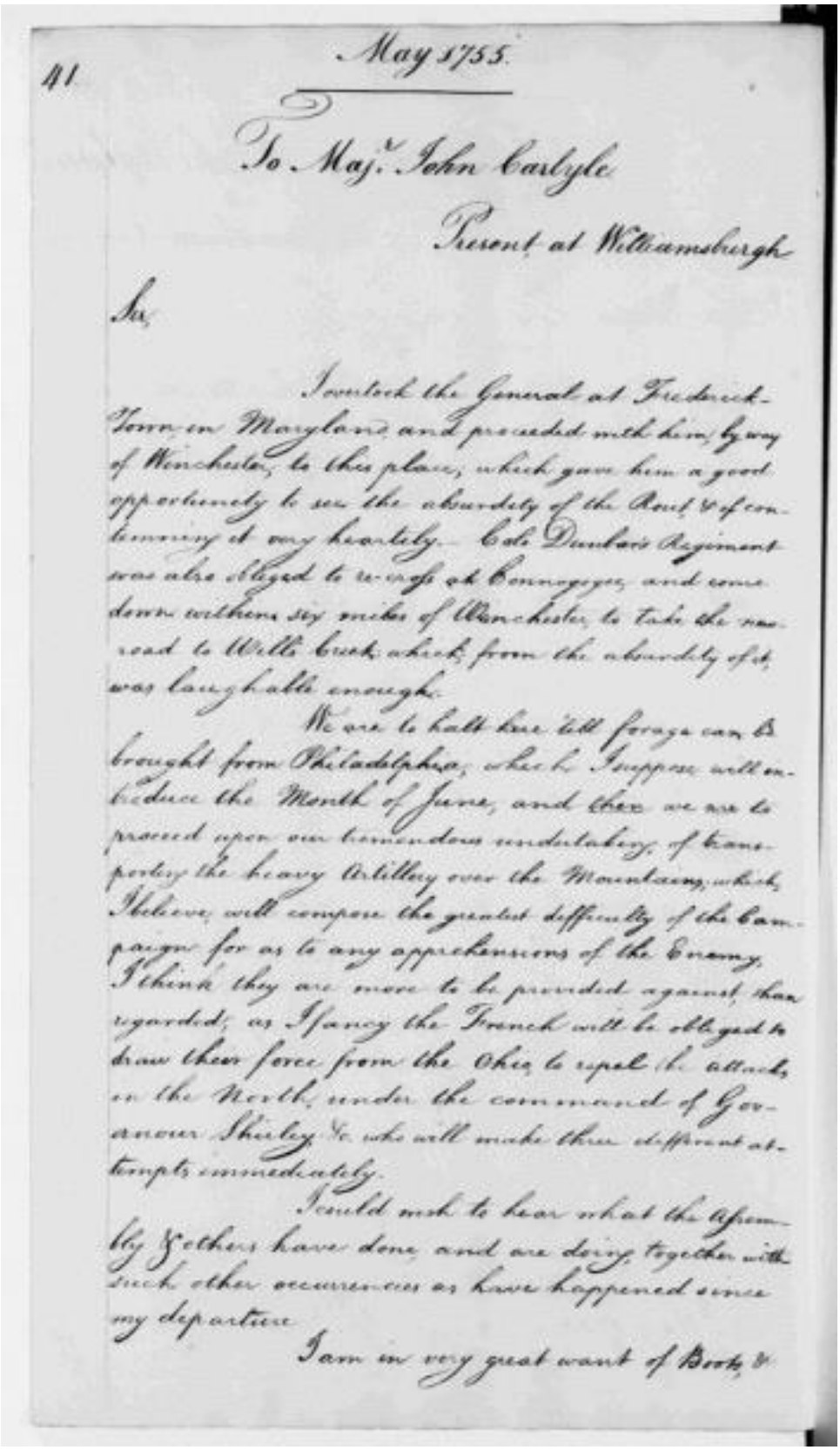
To The Hon.ble William Fairfax Esq.
Belvoir.

Dear Sir,

I overtook the General at Frederick Town in Maryland — Thence we proceeded to this place, where we shall remain till the arrival of the 2nd Division of the Train (which we hear left Alexandria on Tuesday last) after that we shall continue our march to Wills Creek, from whence it is imagined we shall not stir till the latter end of this month, for want of Waggons, & other conveniences to transport our baggage, &c. over the Mountains —

You will naturally conclude that to pass thro' Maryland (when no object required it) was an uncommon & extraordinary route for the Gen.l & Col. Dunbar's Regiment, to this place. The reason, however, was obvious. — Those who promoted it had rather that the communication should be opened that way, than through Virginia but I believe the eyes of the Gov.t are now open, & the imposition detected: Consequently the like will not happen again. —

Please to make my compliments to Col. G. to whom I shall write by the



41

May 1755.

To Maj.r John Carlyle

Present at Williamsburgh

Sir,

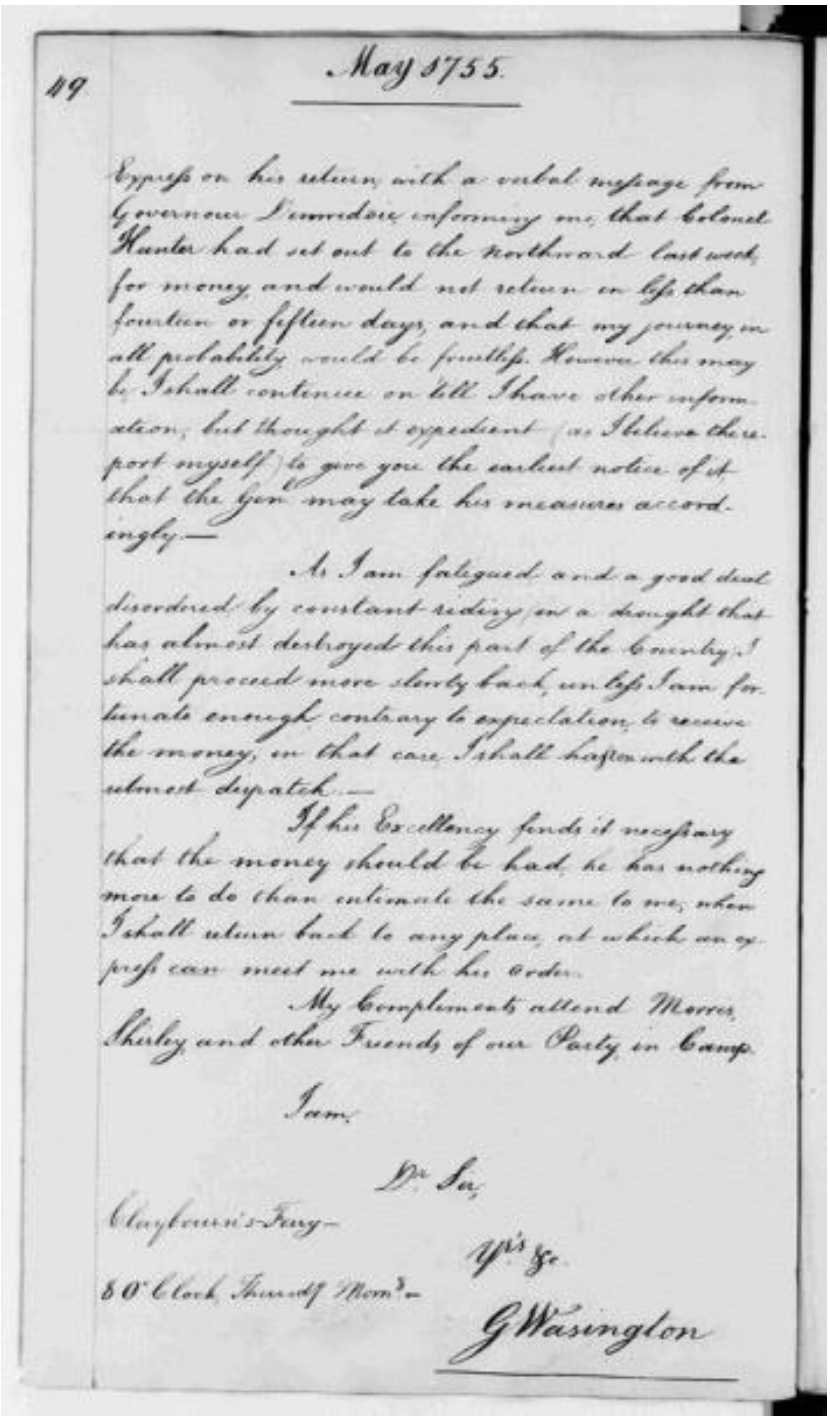
I overtook the General at Fredrick-Town in Maryland, and proceeded with him, by way of Winchester, to this place, which gave him a good opportunity to see the absurdity of the Road, & of condemning it very heartily. Colo Dunbar's Regiment was also obliged to wind off at Connogogue, and come down withem six miles of Winchester, to take the new road to Wills Creek, which, from the absurdity of it, was laughable enough.

We are to halt here till forage can be brought from Philadelphia, which I suppose will inducce the Month of June, and then we are to proceed upon our tremendous undertaking, of transporting the heavy Artillery over the Mountains, which, I believe, will compose the greatest difficulty of the Campaign for as to any apprehensions of the Enemy, I think they are more to be provided against, than regarded; as I fancy the French will be obliged to draw their force from the Ohio, to repel the attack, in the North, under the command of Governour Shirley &c. who will make three different attempts immediately.

I could wish to hear what the Assembly & others have done, and are doing, together with such other occurrences as have happened since my departure.

I am in very great want of Boots &

Figura I. 10 Carta de George Washington



49 May 1755.

Express on his return with a verbal message from Governour Dinwiddie informing me, that Colonel Hunter had set out to the northward last week for money, and would not return in less than fourteen or fifteen days, and that my journey in all probability would be fruitless. However this may be, I shall continue on till I have other information; but thought it expedient (as I believe the report myself) to give you the earliest notice of it that the Gen may take his measures accordingly.—

As I am fatigued and a good deal disordered by constant riding in a drought that has almost destroyed this part of the Country, I shall proceed more slowly back, unless I am fortunate enough contrary to expectation, to receive the money, in that case I shall hasten with the utmost dispatch.—

If his Excellency finds it necessary that the money should be had, he has nothing more to do than intimate the same to me, when I shall return back to any place, at which an express can meet me with his Order.

My Compliments attend Morris, Shirley and other Friends of our Party in Camp.

I am,

Dr Sir,

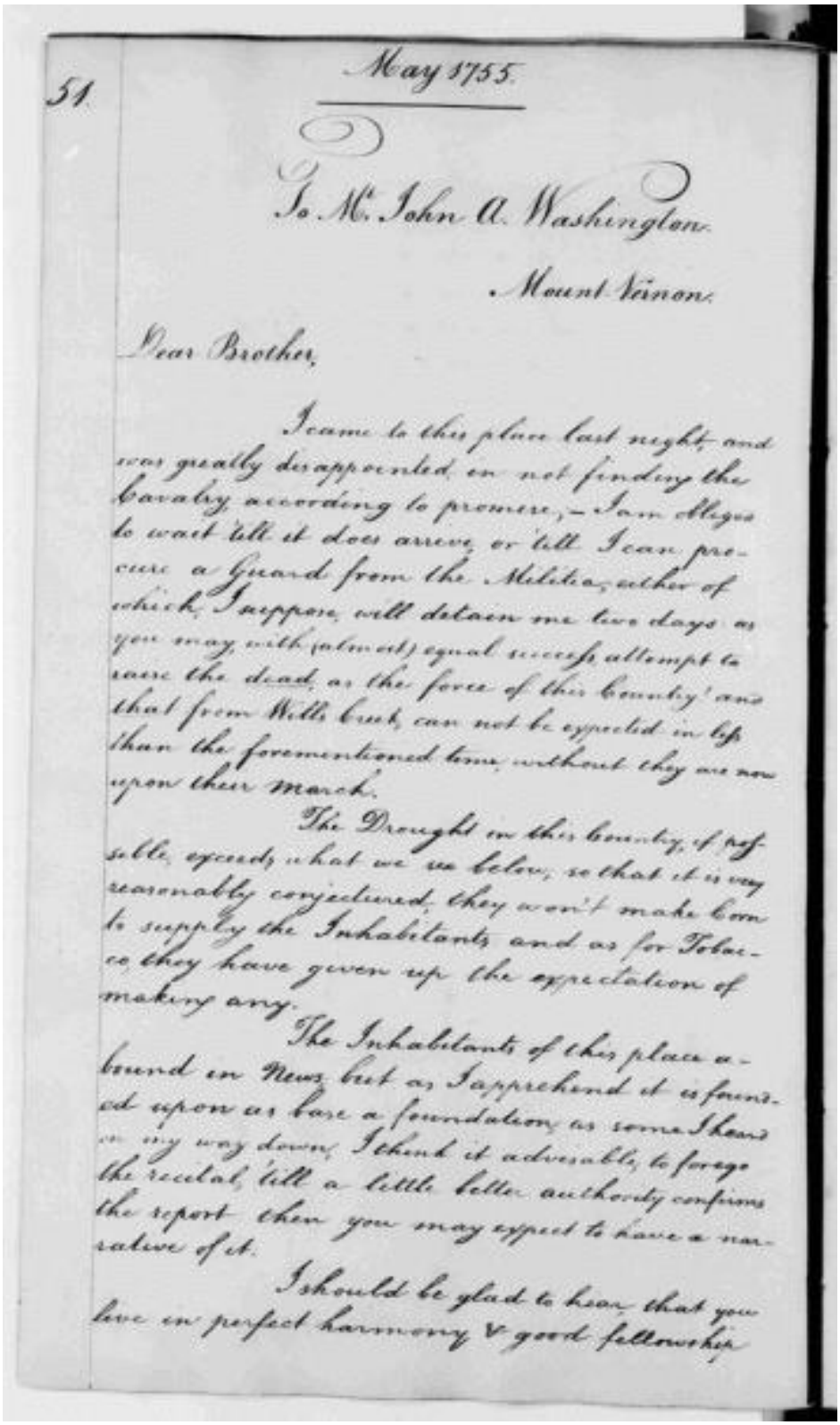
Yrs &c.

G Washington

Claytown's-Fury—

8 O Clock Thursdy Morng—

Figura I. 11 Carta de George Washington



51.

May 1755.

To Mr. John A. Washington.

Mount Vernon.

Dear Brother,

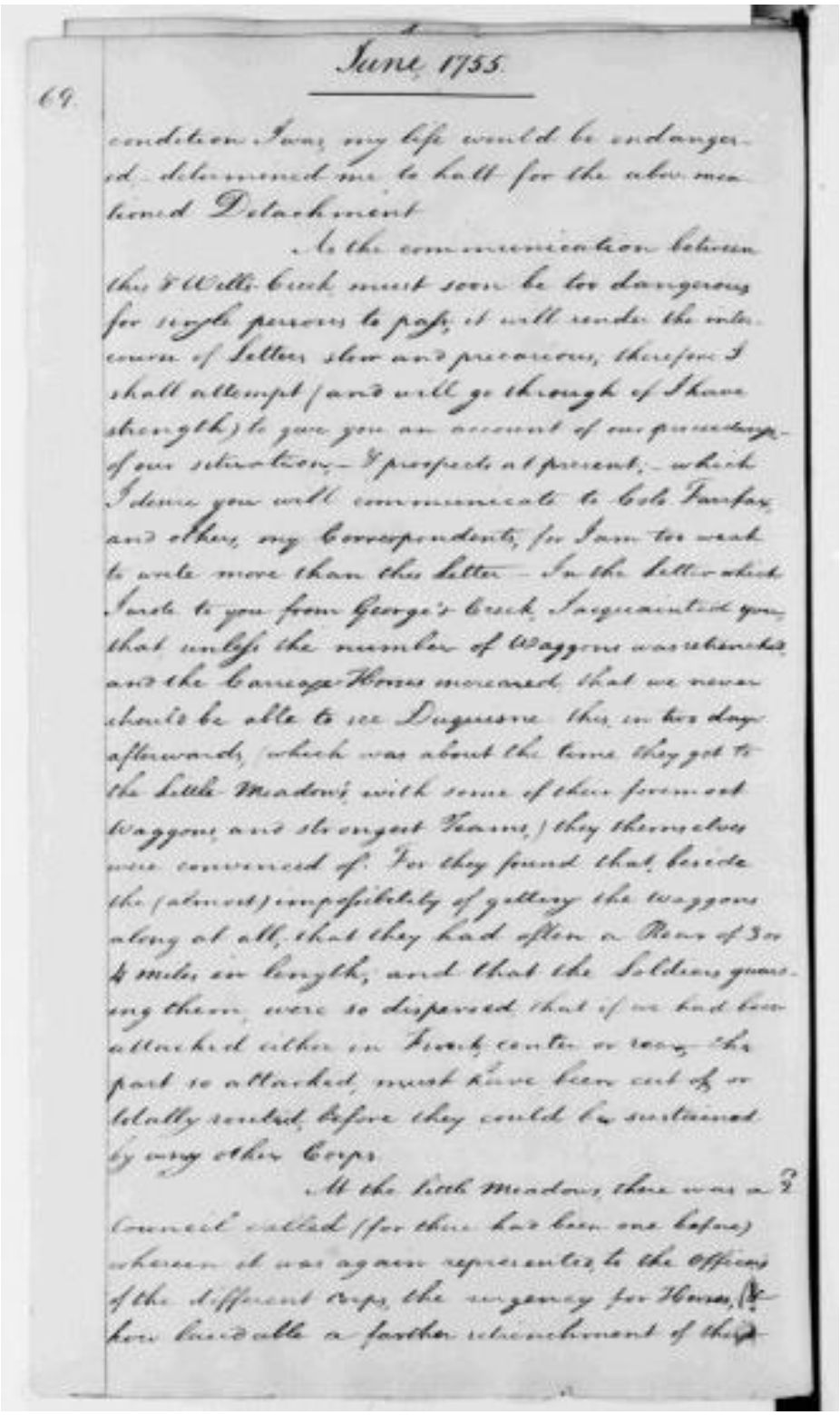
I came to this place last night, and was greatly disappointed on not finding the Cavalry according to promise, — I am obliged to wait till it does arrive, or till I can procure a Guard from the Militia, either of which, I suppose, will detain me two days as you may with (almost) equal success attempt to raise the dead, as the force of this County; and that from Wills Creek, can not be expected in less than the forementioned time, without they are now upon their March.

The Drought in this County, if possible, exceeds what we use below, so that it is very reasonably conjectured, they won't make Corn to supply the Inhabitants and as for Tobacco, they have given up the expectation of making any.

The Inhabitants of this place abound in News, but as I apprehend it is founded upon as base a foundation, as some I heard in my way down, I think it advisable, to forego the recital, till a little better authority confirms the report then you may expect to have a narrative of it.

I should be glad to hear that you live in perfect harmony & good fellowship

Figura I. 12 Carta de George Washington



June 1755.

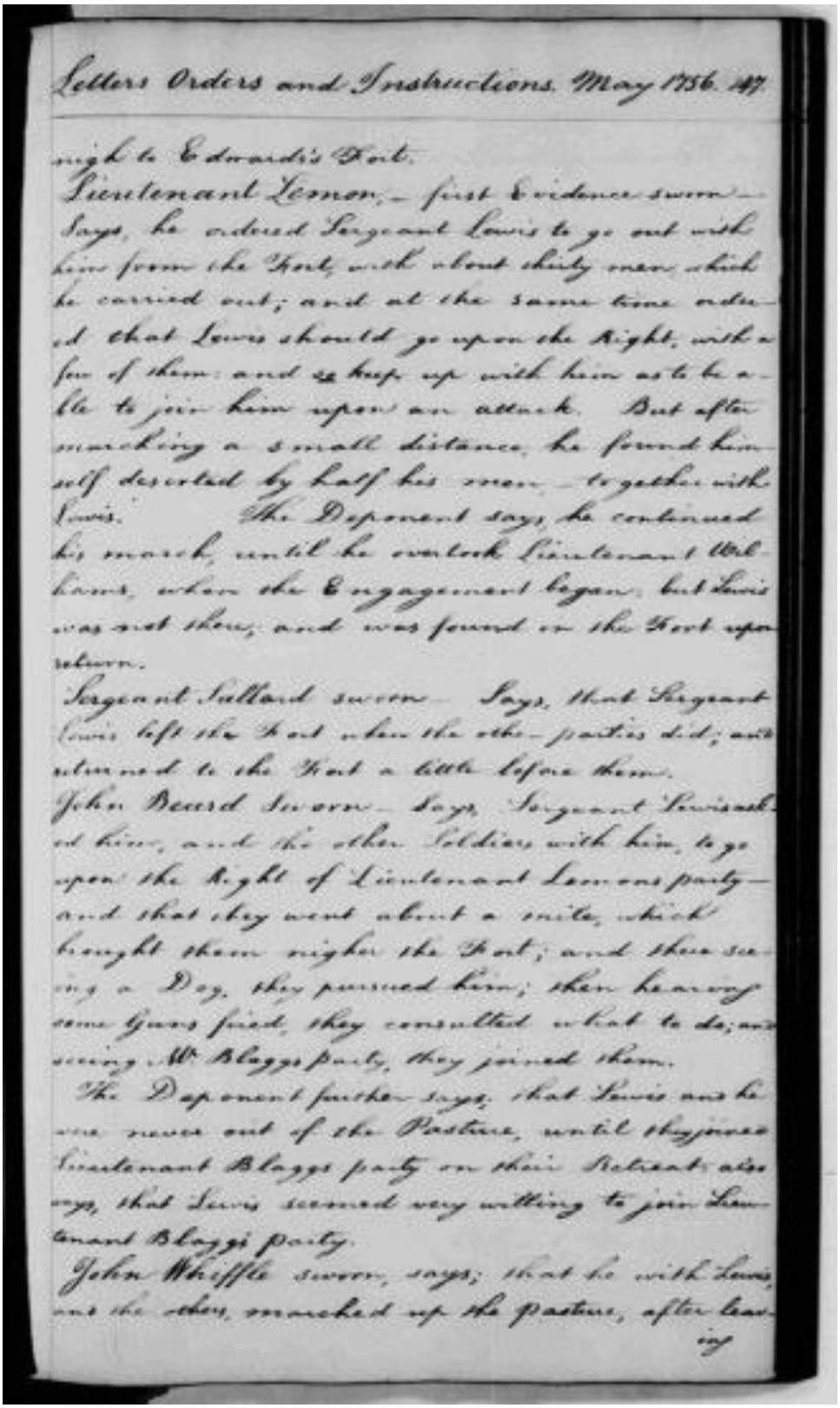
69.

condition I was, my life would be endangered, determined me to halt for the above mentioned Detachment

As the communication between this & Wills-Creek must soon be too dangerous for single persons to pass, it will render the intercourse of Letters slow and precarious, therefore I shall attempt (and will go through if I have strength) to give you an account of our proceedings, of our situation, & prospects at present, which I desire you will communicate to Colo Fairfax, and others, my Correspondents, for I am too weak to write more than this letter. In the letter which I wrote to you from George's Creek, I acquainted you that unless the number of Waggons was encreased, and the Carriage-Horses increased, that we never should be able to see Duquesne. This in two days afterwards, (which was about the time they got to the Little Meadows) with some of their fresh set Waggons, and strongest Teams, they themselves were convinced of. For they found that, beside the (almost) impossibility of getting the waggons along at all, that they had often a Rear of 3 or 4 miles in length, and that the Soldiers guarding them were so dispersed that if we had been attacked either in Front, center or rear, the part so attacked, must have been cut off or totally routed, before they could be sustained by any other Corps

At the Little Meadows, there was a Council called (for there has been one before) wherein it was again represented, to the Officers of the different Corps, the urgency for Horses, & how laudable a further retrenchment of the

Figura I. 13 Carta de George Washington



Letters Orders and Instructions. May 1756. 47

nigh to Edwards's Fort.

Lieutenant Lemon, – first Evidence sworn – Says, he ordered Sergeant Lewis to go out with him from the Fort, with about thirty men which he carried out; and at the same time ordered that Lewis should go upon the Right, with a few of them: and keep up with him as to be able to join him upon an attack. But after marching a small distance, he found himself deserted by half his men – together with Lewis. The Deponent says, he continued his march, until he overtook Lieutenant Williams, when the Engagement began, but Lewis was not there, and was found in the Fort upon return.

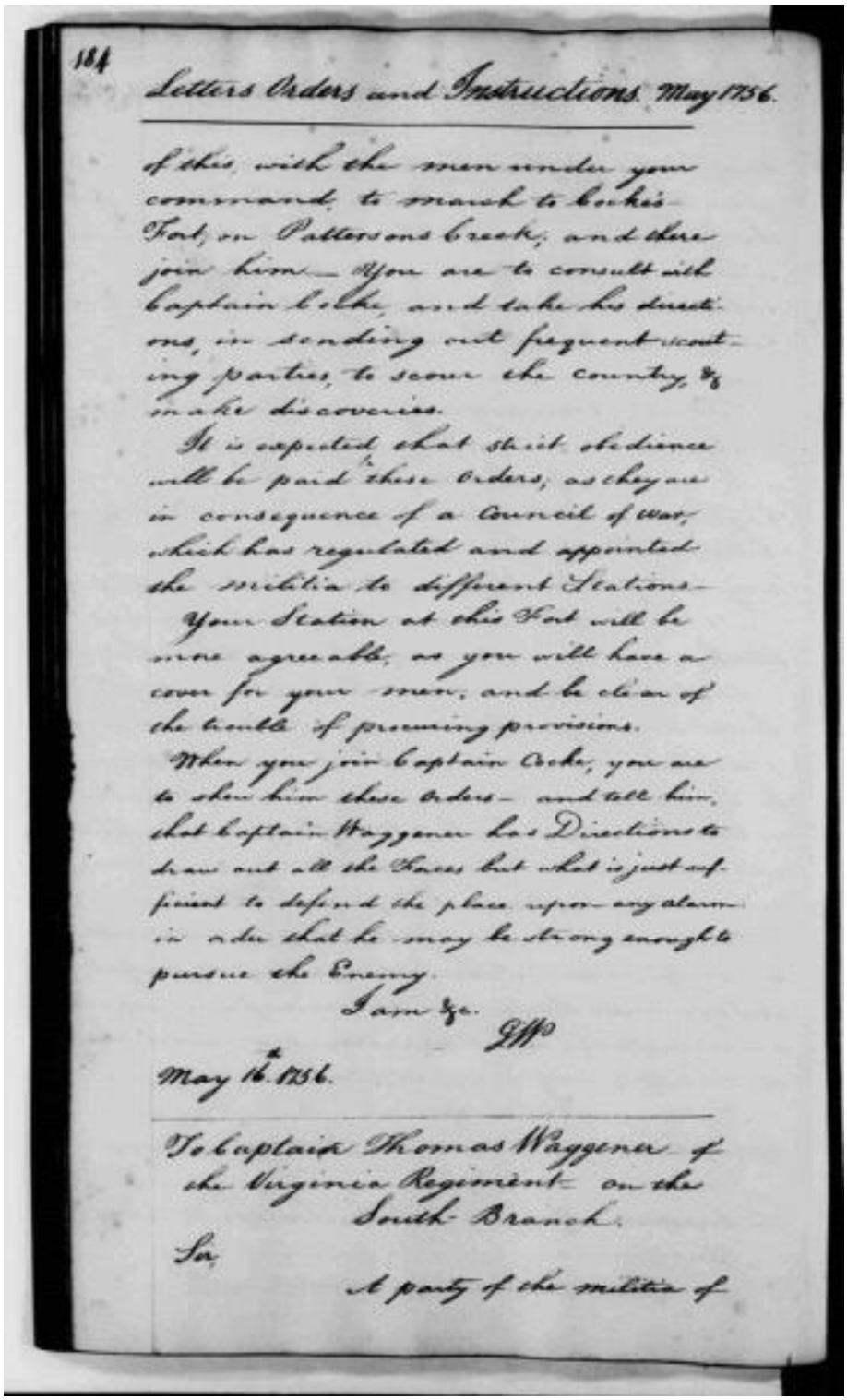
Sergeant Sullard sworn – Says, that Sergeant Lewis left the Fort when the other parties did; and returned to the Fort a little before them.

John Beard Sworn – Says, Sergeant Lewis called him, and the other Soldiers with him, to go upon the Right of Lieutenant Lemons party – and that they went about a mile, which brought them nigher the Fort; and there seeing a Dog, they pursued him; then hearing some Guns fired, they consulted what to do; and seeing Mr Blaggs party, they joined them.

The Deponent further says, that Lewis and he were never out of the Pasture, until they joined Lieutenant Blaggs party on their Retreat; also says, that Lewis seemed very willing to join Lieutenant Blaggs party.

John Whiffle sworn, says; that he with Lewis and the other, marched up the pasture, after leaving

Figura l. 14 Figura l. 14 Carta de George Washington



184 Letters Orders and Instructions May 1756.

of this, with the men under your command, to march to Cocke's Fort, on Pattersons Creek, and there join him — You are to consult with Captain Cocke, and take his directions, in sending out frequent scouting parties, to scour the country, & make discoveries.

It is expected that strict obedience will be paid to these Orders, as they are in consequence of a Council of War, which has regulated and appointed the Militia to different Stations —

Your Station at this Fort will be more agreeable, as you will have a cover for your men, and be clear of the trouble of procuring provisions.

When you join Captain Cocke, you are to show him these Orders — and tell him, that Captain Waggener has Directions to draw out all the Men but what is just sufficient to defend the place upon any alarm in order that he may be strong enough to pursue the Enemy.

I am &c.

GW

May 16th 1756.

To Captain Thomas Waggener of the Virginia Regiment on the South Branch.

Sir,

A party of the militia of

Figura I. 15 Carta de George Washington

Apéndice II

II. Manual de usuario

Este apéndice describe los pasos necesarios para utilizar el sistema de segmentación de palabras de PRISCUS[2]. El sistema se ejecuta bajo Matlab, versión 2014-b, con la librería "*Image processing toolbox*". Los programas necesarios para ejecutarse se encuentran en el disco de instalación del sistema.

La ventana principal se muestra en la Figura II.1 en donde se observa, en primera instancia, el nombre del sistema "PRISCUS- Segmentador". En la parte inferior al título se encuentran los pasos necesarios para realizar la segmentación, indicados a través de botones, que realizan las acciones de filtrado, generación de segmentos y visualización de los segmentos. Junto a los botones se encuentran dos áreas de despliegue, donde aparecerán la imagen original del documento a segmentar y la imagen con las palabras localizadas, marcadas con rectángulos verdes. En la parte inferior de la ventana principal se muestra una ventana, donde podrán verse, uno por uno, los segmentos generados. En las siguientes líneas se describe con mayor detalle cada uno de los elementos de la interfaz gráfica y su uso.

[2] Una descripción general de este sistema de reconocimiento de letra manuscrita y antigua se encuentra en: http://ccc.inaoep.mx/~pgomez/PRISCUS/

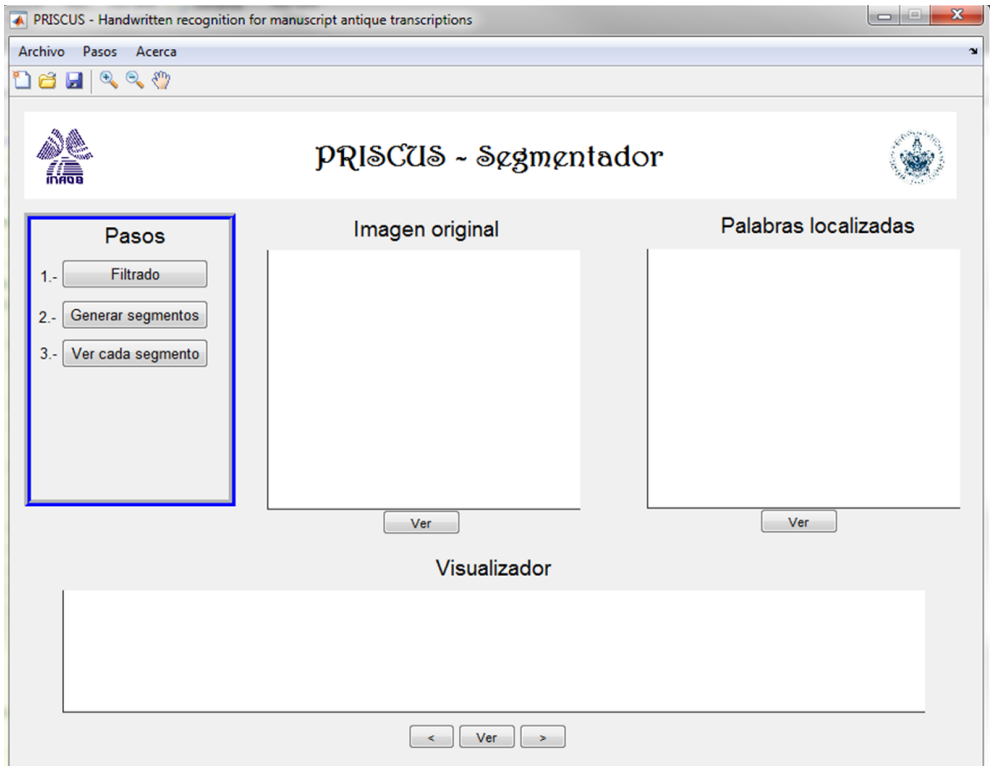


Figura II. 1 Ventana principal de *PRISCUS*-Segmentador

Menús

El menú “archivo” (Figura II.2) despliega cuatro opciones posibles a realizar:

- **Nuevo**. Limpia toda la interfaz de cualquier imagen o proceso realizado anteriormente para dar paso a la segmentación de una nueva imagen.
- **Abrir imagen**. En esta opción, el usuario tiene la posibilidad de elegir la imagen que él/ella desee, para localizar y segmentar las palabras.
- **Guardar como**. Solicita el nombre de un directorio y en él se guarda la imagen original y todos los segmentos generados, uno en cada archivo diferente.
- **Salir**. Esta opción permite salir de la aplicación.

Es importante hacer notar que, en cada una de las ventanas del sistema donde aparecen imágenes, se pueden utilizar los botones de ampliación, reducción y traslación de la imagen, que se encuentran en la parte superior de la interfaz. En esa misma sección se encuentran tres menús colgantes, titulados “Archivo”,

“pasos” y “Acerca”. Como en otros sistemas, la opción “Acerca” despliega la información de autoría y derechos de autor de este software.

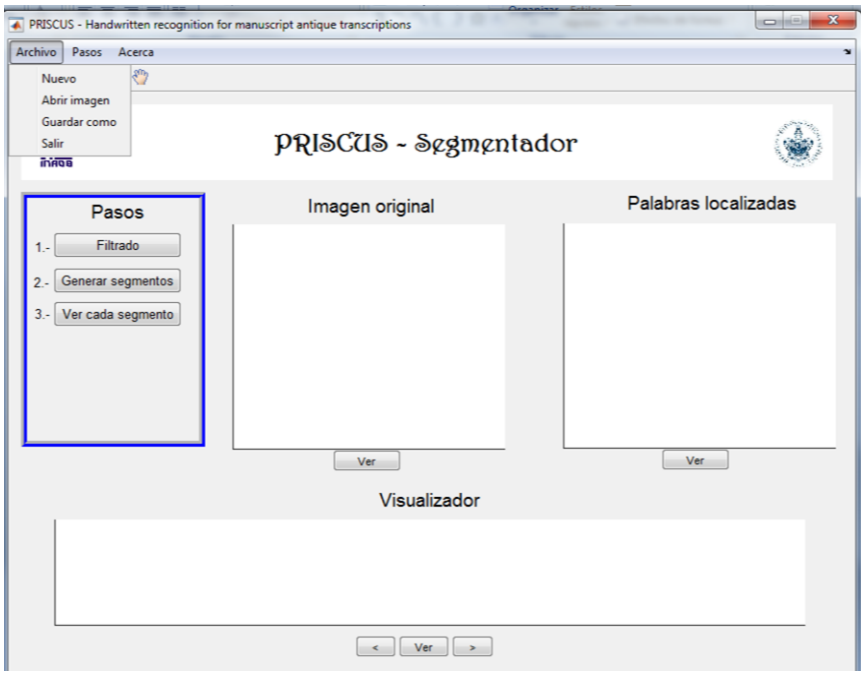


Figura II. 2 Menú archivo

El menú Pasos (Figura II.3) despliega tres opciones. Es importante mencionar, que las mismas acciones que se realizan en el menú, para comodidad del usuario también se encuentran en el cuadro con contorno azul identificado como **Pasos.**

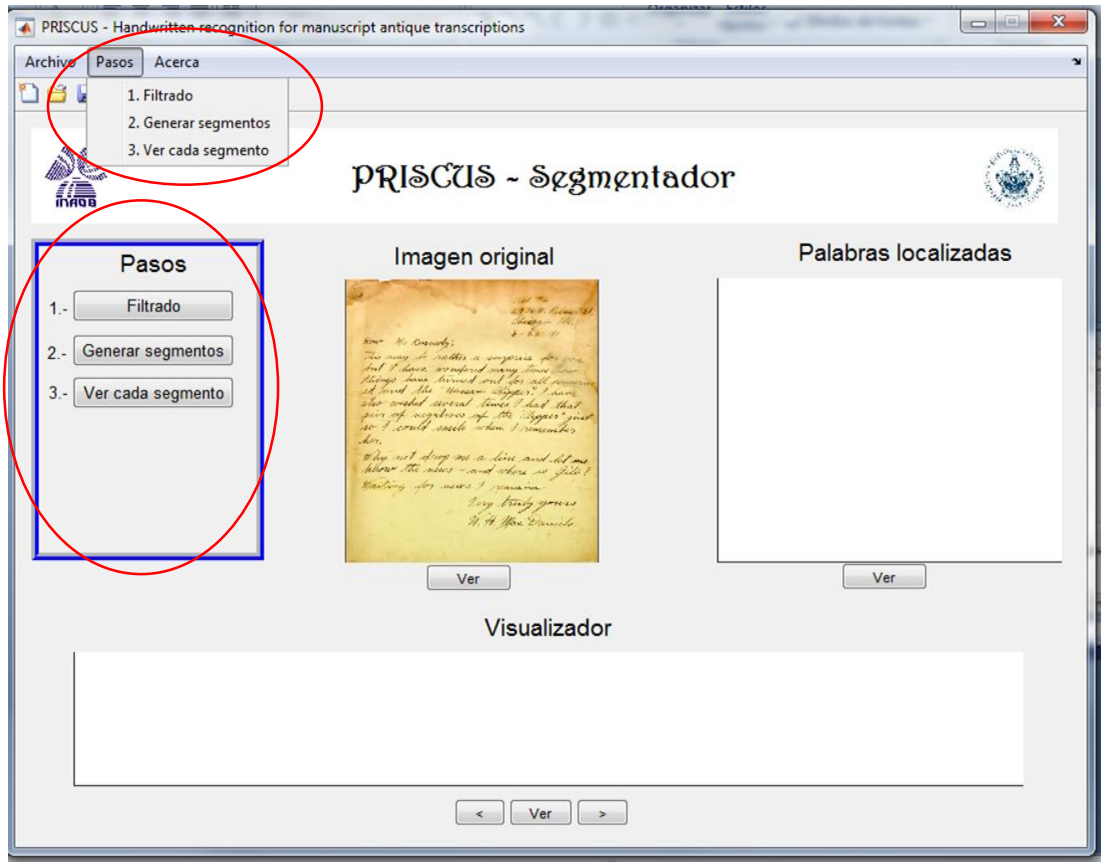
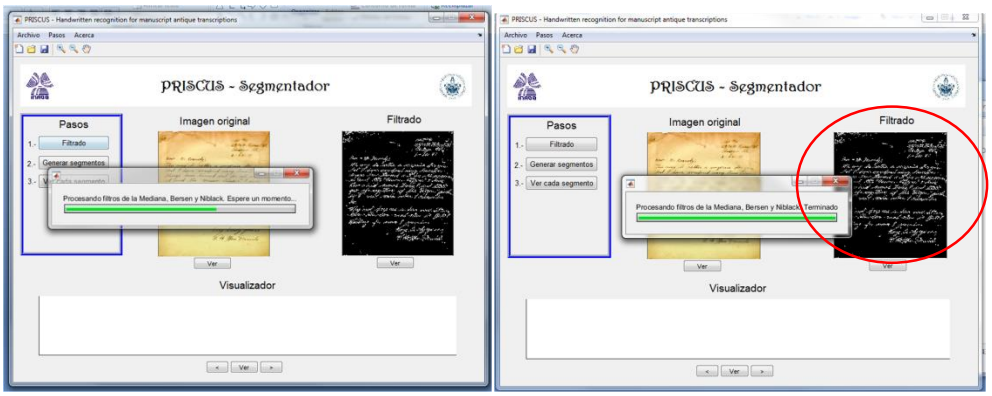


Figura II. 3 Menú pasos

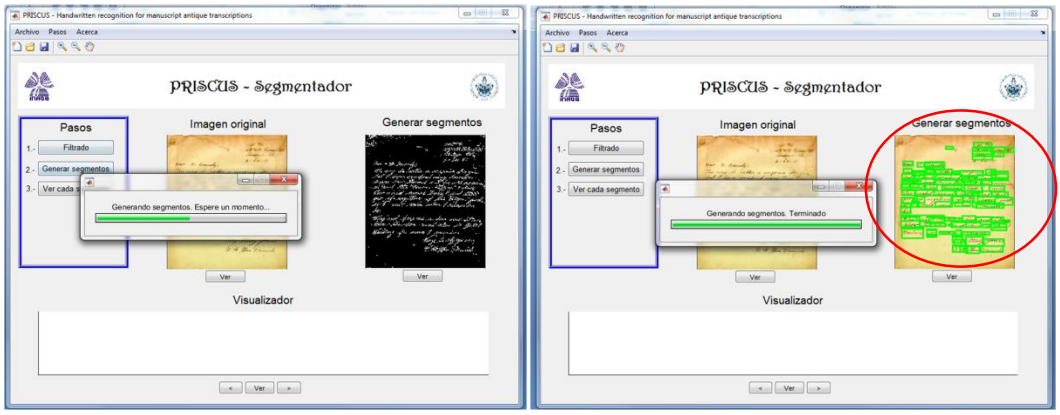
1. **Filtrado.** Realiza el pre- procesamiento de una imagen cargada previamente. Este paso es necesario a fin de limpiar la imagen de artefactos que puedan entorpecer el proceso de segmentación. En la figura II.4 (a) se observa el progreso del filtrado conforme se va realizando; en la figura II.4 (b) se muestra el filtrado terminado en su totalidad, así como su respectiva imagen, la cual se encuentra marcada por un círculo rojo en la figura.



(a) (b)

Figura II. 4 (a) Filtrado en proceso. (b) Filtrado terminado

2. **Generar segmentos.** Esta opción realiza la segmentación de las palabras de la imagen original cargada en el sistema. En la figura II.5 (a) se observa el progreso de la segmentación conforme se va realizando, y en la figura II.5 (b) se muestra la segmentación terminada, así como su respectiva imagen. Nótese que los segmentos están marcados con rectángulos verdes.



(a) (b)

Figura II. 5 (a) Segmentación en proceso. (b) Segmentación terminada.

3. **Ver cada segmento**. Al seleccionar esta opción, en la ventana titulada "visualizador" se mostrarán se muestran cada una de las imágenes de las palabras segmentadas. (Figura II.6). Las flechas dibujadas en los botones bajo el visualizador permiten avanzar o retroceder en cada uno de los segmentos visualizados.

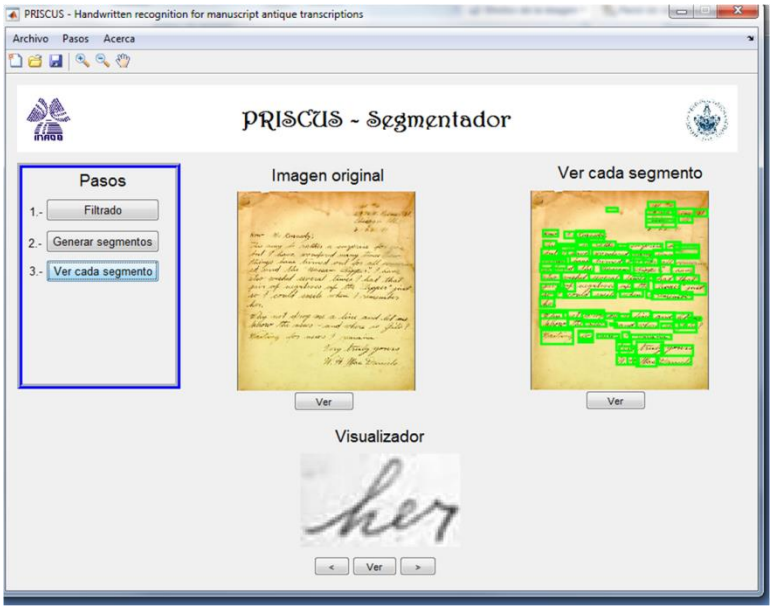


Figura II. 6 Ver cada segmento generado

Es importante mencionar que en la parte central inferior de cada área donde se visualiza la imagen se encuentra un botón con la leyenda "Ver", cuyo objetivo es ampliar la imagen correspondiente, para que el usuario tenga mejor visibilidad de ella. Un ejemplo de una imagen ampliada se muestra en la figura II.7.

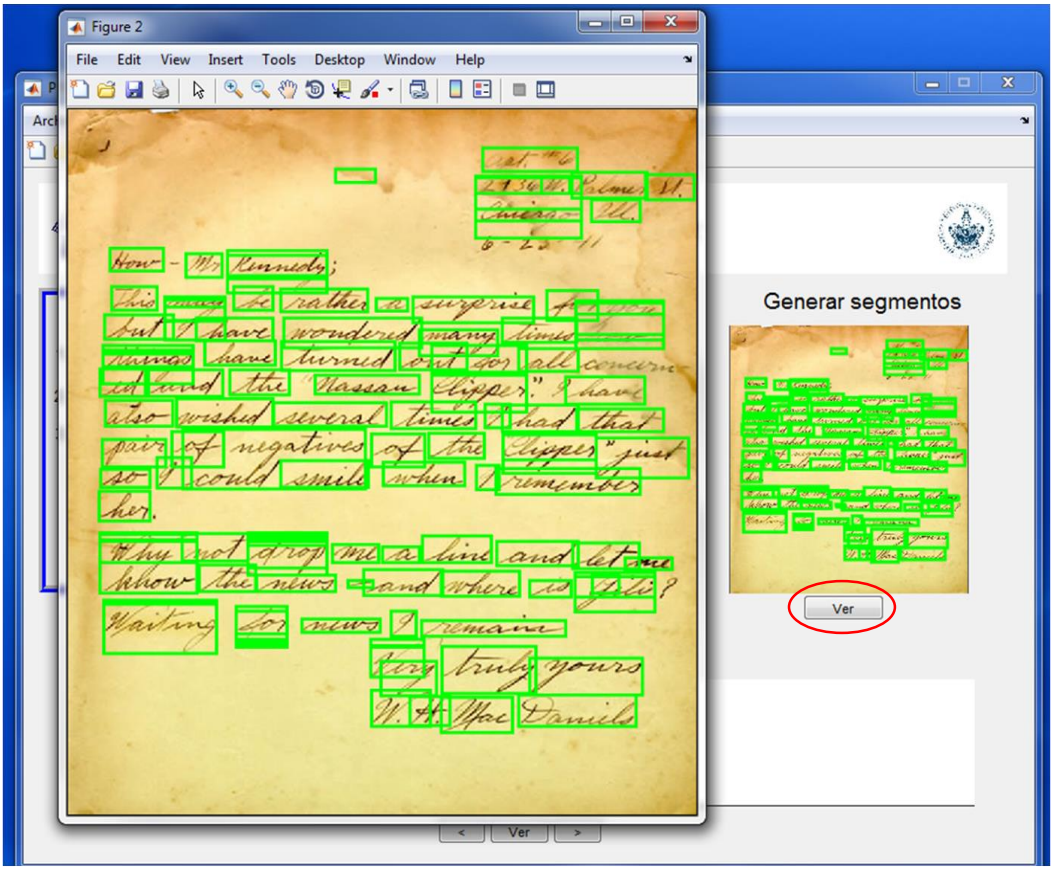


Figura II. 7 Botón ver

Cualquier duda o comentario que se tenga al respecto de este sistema, puede enviarse un correo a: pgomez@acm.org.

Bibliografía

Billington, J. H. (14 de 04 de 2015). *Library of congress*. Obtenido de http://memory.loc.gov/ammem/gwhtml/gwhome.html

Cortés Osorio, J. A., Chaves Osorio, J. A., & Mendoza Vargas, J. A. (2012). Comparación cualitativa y cuantitativa de las técnicas básicas de umbralización local para el procesamiento digital de imágenes. *Scientia et Technica Año XVII*, 236-241.

Cuevas, E., Zaldívar, D., & Pérez, M. (2010). *Procesamiento digital de imagenes con MATLAB y Simulink* (Primera ed.). D.F., México: Alfaomega.

Fernández Mota, D., Lladós, J., & Fornés, A. (2014). *A graph-based approach for segmenting touching lines in historical handwritten documents.* Berlin: Springer.

G. Louloudis, ∗. B. (2008 йил December). Text line detection in handwritten documents. *Pattern Recognition, 41*, 3758 -- 3772.

Gómez Gil, P., & Cuevas Farfán, E. (2008 йил Noviembre). PRISCUS: Reconocedor Óptico de Caracteres Manuscritos y Antiguos. *Memorias del 9o. encuentro de investigación. Instituto Nacional de Astrofísica, Óptica y*, 147-150.

Jewoong Ryu, H. I. (2015). Word Segmentation Method for Handwritten Documents based on Structured Learning. *IEEE SIGNAL PROCESSING LETTERS*, 1161-1165.

Khalid Iqbal, X.-C. Y. (2014). Bayesian Network Scores Based Text Localization in Scene Images. *International Joint Conference on Neural Networks (IJCNN)* (pp. pp.2218,2225). Beijing, China: IEEE.

Luna Pérez, R. (2011). *Reconocimiento fuera de línea de palabras manuscritas, basado en un Mapa Auto-Organizado.* Puebla: Instituto Nacional de Astrofísica Óptica y Electrónica.

Manmatha, R., & Rothfeder, J. L. (2005). A Scale Space Approach for Automatically Segmenting Words from Historical Handwritten Documents. *IEEE TRANSACTIONS ON PATTERN ANALYSIS AND MACHINE INTELLIGENCE*, 1212-1225.

Nieto Bonilla, D. (15 de Enero de 2010). Tesis: Sistema de reconocimiento de kanjis japoneses basado en procesamiento digital de imágenes aplicado a dispositivos móviles. 12-13. Cholula, Puebla, México: Universidad de las Américas Puebla.

Rath, T. M., & Manmatha, R. (2007). Word Spotting for Historical Documents. 1-40.

Romero Gómez, V. (2010 йил 3-Septiembre). Thesis: Multimodal Interactive Transcription of Handwritten. Valencia, España: Universidad Politecnica.

Senior, A., & Robinson, A. (1998). An Off-Line Cursive Handwriting Recognition System. *IEEE*, 309-321.

Toselli, A. H. (2004). *Reconocimiento de Texto Manuscrito Continuo.* Valencia: UNIVERSIDAD POLITÉCNICA DE VALENCIA.

Universidad Nacional de Quilmes . (2005 йил Octubre). Segmentación por Umbralización. *Visión Artificial* . Buenos Aires.