Capítulo 3. Características de la Base de Datos y del Sistema ANSSYD

Capítulo 3 Características de la Base de Datos y del Sistema ANSSYD

3.1 Construcción de la base de datos

En el proyecto anterior se construyó una base de datos que contiene las imágenes de palabras provenientes de los telegramas escritos por el Gral. Porfirio Díaz. Dichas imágenes estuvieron sujetas a un proceso de limpieza en el cual se eliminaban manchas o pixeles que no formaban parte de las palabras. Posteriormente se llevó a cabo un proceso de normalización para remover algunas variaciones de tamaño e inclinación de las de las mismas y sólo después de lo anterior se prosiguió con la fase de segmentación. A continuación se explicarán dichos procesos brevemente.

En primera instancia se realizó una selección de 25 copias fotostáticas de los microfilms de los manuscritos proporcionados por la Biblioteca de la Universidad de las Américas. Dichos documentos fueron digitalizados en formato *pict* (Picture) a través de un Scanner Agfa utilizando el paquete FotoLook v. 2.09.3. Debido al deterioro de los originales fue necesario llevar a cabo una limpieza de cada imagen utilizando Adobe Photoshop v. 5.0 para finalmente guardar cada documento en formato gif (Grafical Interchange File).

El siguiente paso fue la selección de palabras para la construcción de la base de datos. Se escogieron aquellas palabras cuyas letras fuesen lo más legible posible y que representaran a la mayoría de los caracteres del alfabeto. Una vez seleccionadas se

extrajeron manualmente de los documentos digitalizados y limpiados. Cada imagen se guardó en formato *gif* conteniendo únicamente a una palabra aislada. Este conjunto de imágenes constituyeron parte de la base de datos que se usó en el proyecto anterior y que será nuevamente utilizada en el presente [Linares & Spínola, 2000].

3.2 Rasgos característicos de la escritura de Porfirio Díaz.

Muchos de los caracteres escritos por el General varían de una palabra a otra dependiendo de los caracteres que los anteceden o si se encuentran al final o al inicio de la palabra. Como se muestra en la Figura 3.1 se pude observar fácilmente el cambio en la forma de las letras contenidas incluso en la misma palabra.

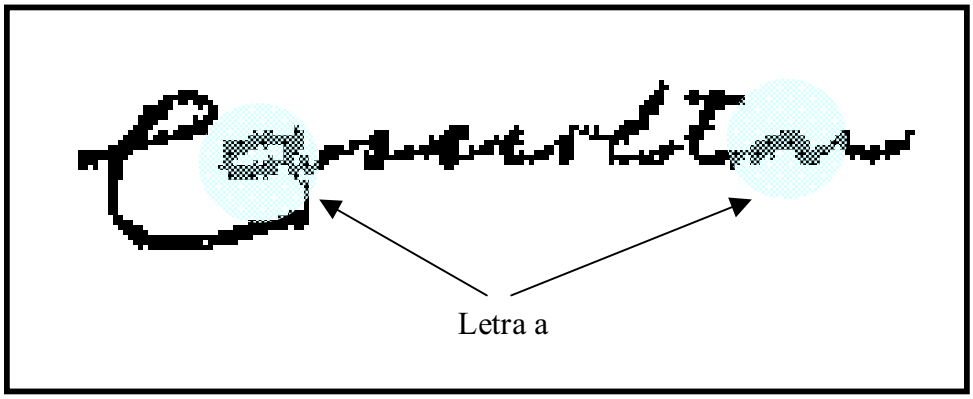
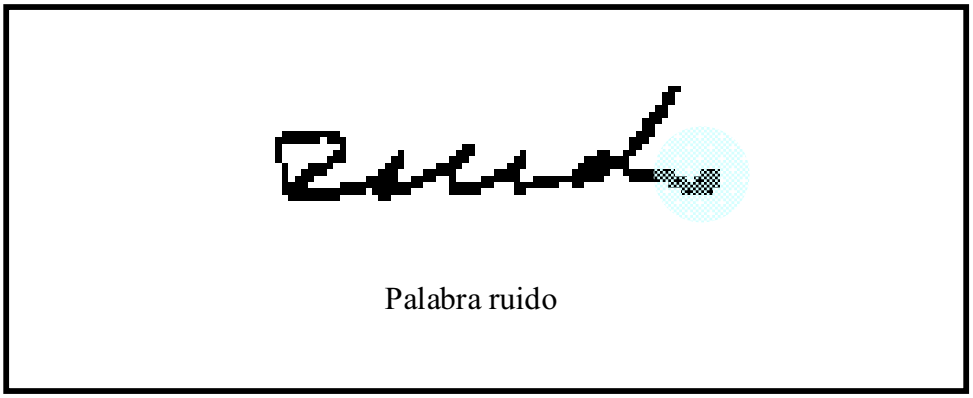


Figura 3.1 Variación de las letras dentro de una misma palabra

Una característica de este tipo de letra es el que las ligaduras (segmentos de unión entre una letra y otra) se pueden confundir fácilmente con parte del segmento que constituye algunas vocales o consonantes como se puede ver en la figura 3.2.



Palabra ruido

Figura 3.2 Caracteres que pueden ser confundidos como ligaduras

Es notable también el encontrar constantemente palabras incompletas, ya que al momento de escribirlas, el último caracter de la palabra no se llegó a formar completamente al grado de que llega a desaparecer o se transforma en un punto como en el caso de la a,e,o,x. Esto se observa claramente en la Figura 3.3

Palabra indıgena

Figura 3.3 Ejemplo de palabras con letras incompletas

Una caracterıstica más es que el cırculo que normalmente constituyen a las letras q,p,g y b no se encuentran totalmente definidos derivando en una sola línea vertical con un muy ligero abultamiento en la parte media.

Las letras *i,l* y *e* son muy similares y la única diferencia entre ambas son el tamaño (que no varía de manera importante) , el ancho o el punto que constituye a la *i*.

3.3 Análisis del algoritmo de segmentación de palabras del Shell Neuronal ANNSYD

El algoritmo de segmentación de palabras que utiliza el sistema ANNSYD se denomina "Algoritmo de segmentación usando histogramas de densidad horizontal". El primer paso de este método es el convertir a la imagen que contiene la palabra en una matriz binaria de unos y ceros. Sus dimensiones son de *m * n* donde m y n representan el tamaño en pixeles de la imagen, es decir cada pixel de la imagen está representado por una celda en la matriz. Cada celda puede tener el valor de 1 o 0, si el pixel representado por la celda es negro se le asigna un uno y si es blanco se le asigna un cero. Este proceso no afecta en lo absoluto al archivo [Kussul & Kasatkina, 1999].

Una vez obtenida la matriz de puntos se determina un histograma de ésta. Este se genera contando el número de unos de forma vertical que existe en la matriz. El resultado de cada columna es representado en una barra de el mismo número de pixeles que acumuló, teniendo como resultado un histograma . Un ejemplo de este tipo de histograma se presenta en la Figura 7.4 que posteriormente será explicado a detalle.

El histograma sirve como entrada para generar uno nuevo pero basado en un umbral. Lo que se hace es sumar la densidad de cada una de las columnas de la matriz. Luego se divide la densidad total entre el ancho de la palabra, es decir entre su valor m. Este valor se multiplica por 0.8 ya que segun se probo genera mejores resultados para la segmentacion. Ahora, el numero de unos de cada columna de la matriz es comparado con este umbral medio y si lo rebasa, la densidad de la columna es de valor uno y si no lo hace se le asigna un cero.

De esta manera se puede determinar claramente los puntos que posiblemente constituyen a un caracter (columna con valor 1) y los que constituyen a una ligadura (columna con valor 0). Para refinar este algoritmo se tomo la decision de que si el ancho de la supuesta letra no era mayor a tres columnas de la matriz se le concatenaba al segmento anterior y de esta manera se disminuyeron los puntos de segmentacion erroneos.

3.4 Pruebas sobre el sistema ANSSYD

Una vez analizado el algoritmo, se probo cada una de las 67 palabras seleccionadas por el trabajo anterior para obtener un porcentaje de exito general con la intencion de que este sirva como punto de referencia para la comparacion con los nuevos algoritmos a implementar. El procedimiento fue el siguiente:

- Se probaron las 67 palabras en el algoritmo de segmentacion de densidad horizontal
- Ademas de las palabras sin correccion tambien se probaron las palabras normalizadas

- Se observo si el numero de puntos de segmentacion eran los correctos
- Se verifico que a pesar de que el numero de puntos de segmentacion fuese el correcto, que cada uno se localizara en la posicion adecuada.

En la Tabla 3.1 se presentan los resultados obtenidos para cada una de las palabras seleccionadas. La figura se interpreta de la siguiente forma. Las palabras sin corregir son aquellas palabras sin estar sujetas al proceso de correccion de la inclinacion y las palabras corregidas son aquellas cuya inclinacion ha sido modificada a traves del editor de imagenes. El numero de errores es el numero de puntos de segmentacion que no fueron colocados (en caso de que el valor sea negativo) o los puntos que fueron colocados de mas, mejor conocido como sobresegmentacion (en caso de que el valor sea positivo). La columna de exito significa si la palabra fue segmentada con el numero de puntos de segmentacion correctos y colocados en la posicion adecuada (0 significa fallo, uno significa exito).

Palabras sin correccion de inclinacion	No. de errores	Exito	Palabras corregidas en inclinacion	No. de errores	Exito
acaso	2	0	acaso	0	1
acuerdo	2	0	acuerdo	0	1
aesas	0	1	aesas	0	1
aesegob	1	0	aesegob	1	0
ahumada	2	0	ahumada	0	0
algunos	-1	0	algunos	1	0
anexos	4	0	anexos	3	0
asi	-1	0	asi	0	1
asuntoque	2	0	asuntoque	0	0
aud	1	0	aud	0	1
autoridades	0	0	autoridades	0	0
carmelita	1	0	carmelita	3	0
civil	-1	0	civil	0	1

comunicacion	1	0	comunicacion	1	0
con	2	0	con	2	0
Palabras sin correccion de inclinacion	No. de errores	Exito	Palabras corregidas en inclinacion	No. de errores	Exito
cordialmente	1	0	cordialmente	1	0
corresponda	-3	0	corresponda	-1	0
cualesla	-1	0	cualesla	1	0
cuando	2	0	cuando	0	1
culpables	-2	0	culpables	2	0
davila	-1	0	davila	0	0
de	0	1	de	0	1
del	-1	0	del	0	0
del2	1	0	del2	3	1
descanso	0	1	descanso	0	1
destacamento	0	0	destacamento	0	0
desu	1	0	desu	2	0
digna	1	0	digna	2	0
dirija	1	0	dirija	0	1
el	0	1	el	0	1
enfermo	1	0	enfermo	6	0
ent ´	1	0	ent ´	2	0
estomago	2	0	estomago	3	0
gobernador	-3	0	gobernador	0	0
haciendo	1	0	haciendo	1	0
hecho	0	0	hecho	2	1
inconscientes	0	0	inconscientes	0	0
indıgena	2	0	indıgena	0	0
indıgenas	0	0	indıgenas	0	0
intrigado	-1	0	intrigado	1	0
jesus	1	0	jesus	2	0
juez	1	0	juez	1	0
libertad	-1	0	libertad	1	0
llamala	2	0	llamala	2	0
luis	-1	0	luis	0	1
menos	1	0	menos	1	0
mi	2	0	mi	2	0
mucho	3	0	mucho	5	0
orden	3	0	orden	0	1
parte	-1	0	parte	1	0
polıtico	-2	0	polıtico	-2	0
por	0	0	por	2	0
puedo	2	0	puedo	1	0
quedan	3	0	quedan	3	0
quien	1	0	quien	1	0
respecto	-1	0	respecto	0	1

ruido	2	0	ruido	0	1
se	1	0	se	0	1
Palabras sin correccion de inclinacion	No. de errores	Exito	Palabras corregidas en inclinacion	No. de errores	Exito
sean	1	0	sean	0	1
secretaria	-1	0	secretaria	0	1
seleccione	-1	0	seleccione	-1	0
serefiere	3	0	serefiere	0	1
solo	0	1	solo	0	1
suplen	1	0	suplen	2	0
suqueja	1	0	suqueja	1	0
transmitirlo	-2	0	transmitirlo	-2	0
venla	3	0	venla	4	0
verdadero	4	0	verdadero	4	0
Totales	86	5	Totales	68	22
Porcentaje de exito= 5.81% Porcentaje de error= 94.18%			Porcentaje de exito=32.35% Porcentaje de error= 67.64%		

Tabla 3.1 Descripcion del proceso de segmentacion

Como se puede observar el proceso de normalizacion mejoro de 5.81% de exito en la segmentacion a un 32.35%. Sin embargo existen casos en que la normalizacion incremento el numero de errores en la segmentacion de solo algunas palabras.

Como se observa en la Figura 3.4 existen algunas letras con las cuales el algoritmo de segmentacion resulta poco eficiente. En este ejemplo observamos que la letra *v* no posee la densidad necesaria para ser considerada como una sola letra y debido a que los extremos verticales de esta letra son practicamente los unicos que presentan densidad, es entonces segmentada por el algoritmo como dos letras separadas, que simulan dos *i*'s. Por otra parte tambien podemos ver que existen algunas ligaduras que fueron consideradas letras. Esto se debio a que dichas ligaduras presentan una pendiente lo suficientemente elevada como para generar una densidad vertical que rebasa el umbral preestablecido.

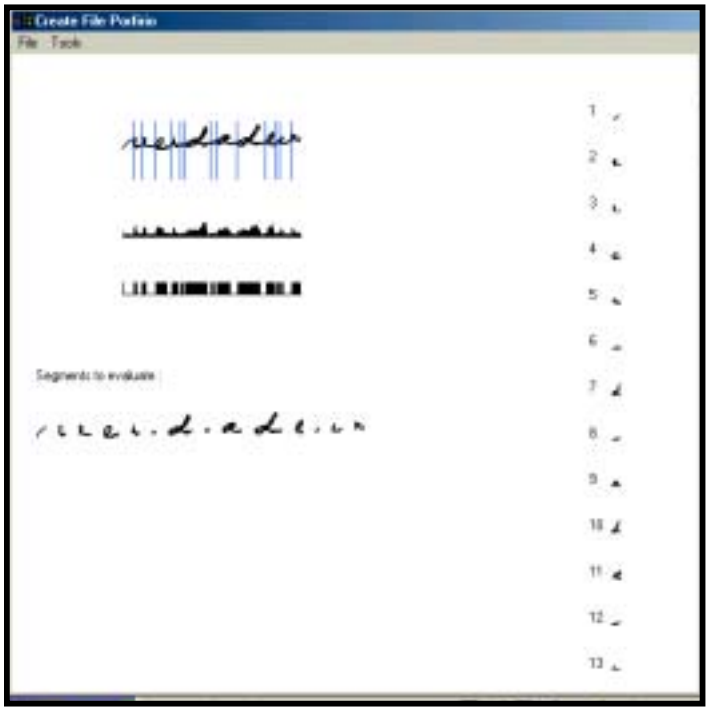


Figura 3.4 Segmentación de la palabra *verdadero*

Otros problemas que presentan en este algoritmo es que si una letra fue escrita con menor intensidad puede provocar una baja densidad ocasionando una sobre segmentación sobre una misma letra, como es el caso de la "a" en la Figura 3.5. También esta imagen nos permite darnos cuenta de que la *v* es nuevamente confundida con dos caracteres separados. Así mismo hay que destacar que las letras e y l no presentaron problema alguno. Como posteriormente se mostrará este par de letras resultan ser unas de las más consistentes ya que siempre generan un alto nivel de densidad y en muy raras ocasiones son confundidas como segmentos separados.

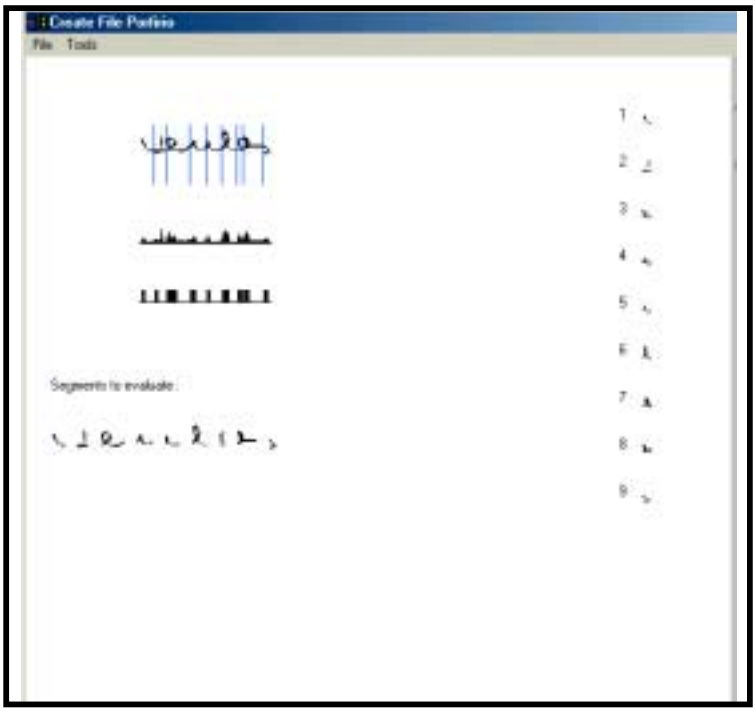


Figura 3.5 Segmentación de la palabra *venla*

A pesar de los grandes fallos que presenta este algoritmo existen palabras que son segmentadas correctamente ya que se encuentran conformadas por caracteres que sobrepasan el umbral preestablecido. En la figura 3.6 se muestra la palabra "sean", la cual es segmentada correctamente ya que la s,e,a y n se encuentran perfectamente definidas y sus ligaduras no presentan mucha inclinación.

Figura 3.6 Separacion de la palabra *sean*

Otro ejemplo de palabra correctamente segmentada es la presentada en la figura 3.7 ya que existe una ausencia de letras como la u,v,w,m y n.

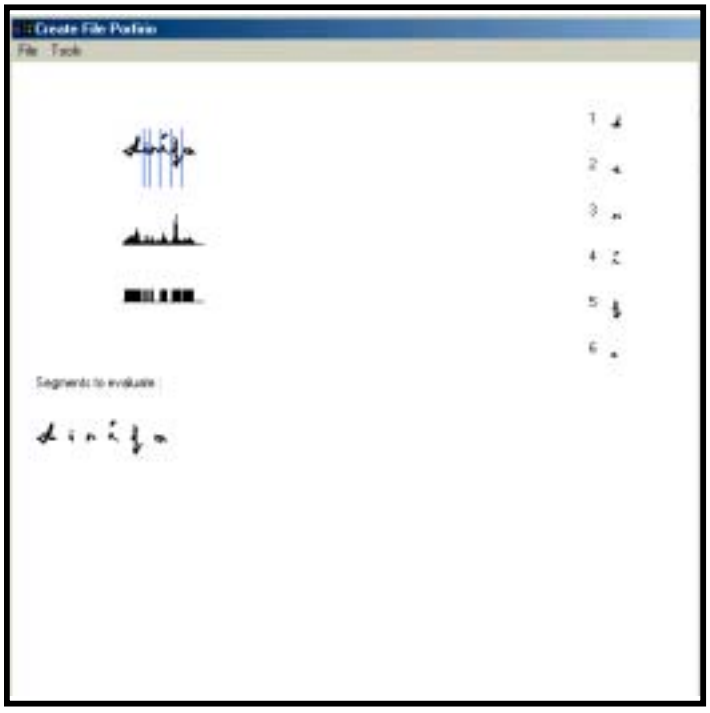


Figura 3.7 Segmentación de la palabra *dirija*

Un ejemplo de los extremos en los que puede caer este algoritmo son palabras constituidas por letras r, m y n como es el caso de la palabra "*enfermo*" donde se generan una cantidad importante de errores ya que por si misma la palabra posee una baja densidad total, entonces al sacar el umbral este resulta muy bajo y se generan muchas columnas de longitud muy pequeña. Esto se refleja en la sobresegmentación final en la Figura 3.8.

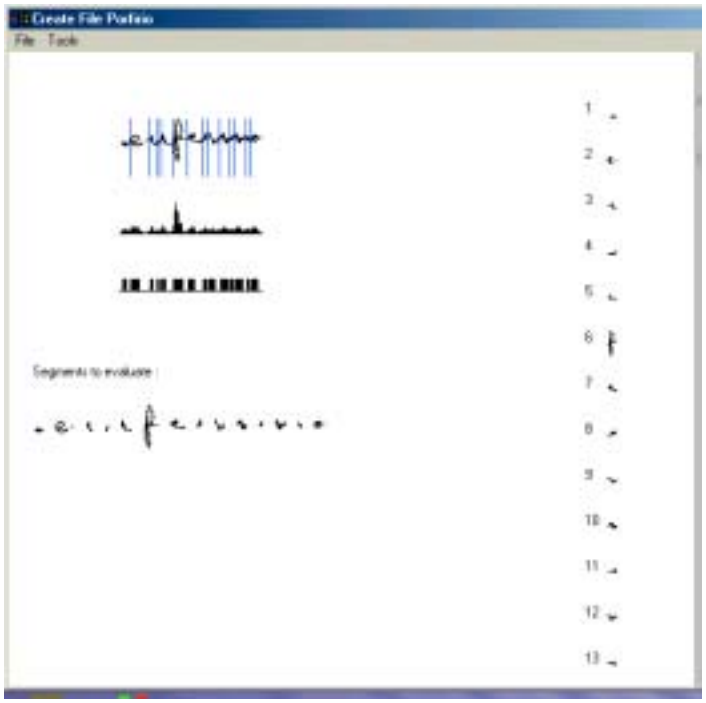


Figura 3.8 Segmentación de la palabra *enfermo*

El análisis de la escritura manuscrita fue necesario para tener una visión más completa de la complejidad del problema (la segmentación). Como se pudo observar en la figuras presentadas, existe una gran variabilidad en las imágenes que son recibidas como entrada, es por ello que en el siguiente capítulo se explicará la necesidad de aplicar técnicas de normalización para obtener imágenes estandarizadas.