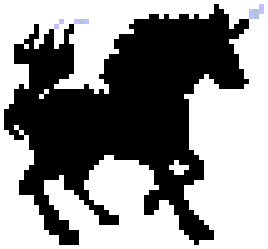
Proyecto de fin de carrera



Captura de datos en expedientes judiciales

Integrantes:
Esteban Bordón
Andrea Mallada
Ignacio Vignolo

Tutor: Guillermo Calderón
Cotutor: Mathías Etcheverry

Montevideo
2012

Resumen de trabajo

Este documento presenta el proyecto de grado “Captura de datos en expedientes judiciales” desarrollado en el Instituto de Computación de la Facultad de Ingeniería de la Universidad de la República. Dicho trabajo se enmarca en el proyecto AJPROJUMI llevado a cabo conjuntamente por el Poder Judicial y la UdelaR con el financiamiento de la Unión Europea, cuyo objetivo principal es preservar los expedientes provenientes de la Justicia Militar en los años de gobierno dictatorial.

El proyecto propone el estudio y relevamiento de las herramientas de Reconocimiento Óptico de caracteres (**OCR**) disponibles hoy en día, utilizadas para la extracción de texto a partir de archivos de imágenes. Dicho estudio se realizó a partir de un subconjunto de los expedientes, proporcionados por el Poder Judicial. Por otra parte se realizó un estudio de herramientas para el pre procesamiento de imágenes, factor que resultó fundamental para obtener mejores resultados al momento de aplicar la herramienta OCR seleccionada. Luego de tener la representación en texto de los expedientes se realizó el procesamiento de estos para encontrar nombres de personas y obtener información relevante a partir ciertas palabras claves utilizadas en las búsquedas proporcionadas al usuario.
Como siguiente paso se utiliza una aplicación que procesa de forma automática esa información y carga una base de datos la cual será accedida a través de una aplicación web también desarrollada para este proyecto. Dicha aplicación web permite examinar los datos cargados a través del ingreso de nombres de personas hallando aproximaciones de éstos con cierto grado de error, además de permitir el acceso a las imágenes de los expedientes que contienen dichos nombres.

En el informe se documentan las pruebas realizadas para la elección de las herramientas de OCR y de pre procesamiento de imágenes. También se documentan los cambios realizados a la herramienta de procesamiento de lenguaje natural y el diseño de la solución presentada en la aplicación web para búsquedas en los expedientes judiciales.

El trabajo se enmarca dentro de la necesidad de contribuir a la preservación de los documentos en papel de forma de poseer un acceso electrónico a los expedientes de la dictadura y realizar búsquedas sobre ellos con una mayor rapidez evitando así realizarlo de forma manual y menos práctica.

Tabla de contenido

Capítulo 1 - Introducción

AJPROJUMI es un proyecto llevado a cabo conjuntamente por el Poder Judicial y la UDELAR, financiado por la Unión Europea. El mismo consiste en construir una base de datos como producto de la digitalización de 3001 expedientes que provienen de la Justicia Militar en los años de gobierno dictatorial (1972-1985) que involucran a 10.134 personas [Ref: AJPROJUMI121031]. Los expedientes fueron restaurados por la Escuela Universitaria de Bibliotecología y Ciencias Afines para luego ser escaneados, finalmente se almacenaron sus imágenes para ser accedidas posteriormente con varios fines (investigación, aplicación de indemnizaciones, etc.).
A su vez, un equipo de la facultad de ingeniería se encarga de realizar una base de datos con la información relevante de cada expediente de acuerdo con los requerimientos recabados en la cual se ingresarán datos de forma manual.
Una de las líneas de investigación del proyecto AJPROJUMI es estudiar la viabilidad de utilizar técnicas de Reconocimiento Óptico de Caracteres para extraer información de los expedientes y así facilitar el alta de los datos.

En este capítulo se presenta una descripción global del proyecto, así como también su motivación, objetivos, resultados esperados y la organización del documento.

1.1 Motivación

La principal motivación de este proyecto es promover la preservación de los documentos existentes sobre civiles sometidos a la Justicia Militar durante el proceso cívico-militar.
Con el paso del tiempo los expedientes físicos han sufrido un importante desgaste por lo que digitalizarlos es una manera de preservarlos y evitar el deterioro de los mismos debido a la consulta permanente.

Por otro lado existe una fuerte necesidad de obtener información deseada de forma ágil de manera de ahorrar tiempo valioso. Para ello en este proyecto se propone la implementación de una aplicación web que proporcione diferentes tipos de búsqueda para recabar información relevante de los expedientes.

Previo a la implementación de esta aplicación web se debe realizar el estudio de las herramientas OCR disponibles en el mercado así como también la realización de un programa que procese los archivos digitalizados para recabar la información necesaria.

1.2 Objetivos

Los principales objetivos de este proyecto son los siguientes:

- Relevar y estudiar las herramientas de reconocimiento óptico de caracteres (OCR) existentes en el mercado.
- Estudiar técnicas de análisis de texto para la obtención de información relevante utilizada posteriormente por una aplicación web.
- Crear una aplicación web para que el usuario final pueda realizar las búsquedas necesarias sobre los expedientes.

1.3 Resultados esperados

Los resultados esperados del proyecto son:

- Documentación del estado del arte en cuanto a las distintas herramientas de reconocimiento óptico de caracteres existentes en el mercado.
- Documentación de las técnicas de análisis de texto utilizadas y software implementado para tal motivo.
- Aplicación web que permita extraer información relevante de los expedientes cargados en ella.

1.4 Organización del documento

El presente informe consta de seis capítulos diferenciados cuyo contenido se esboza a continuación.

El capítulo 1 es una **introducción inicial al problema** en el cual se detallan las características del problema a resolver.

El capítulo 2 consta de una serie de **Definiciones y características generales** en donde se realiza una breve introducción a los conceptos importantes mencionados en varias secciones del informe.

El capítulo 3 es un **estudio del estado del arte** en donde se hace un recorrido por las principales herramientas tanto de pre procesamiento de imágenes como de procesamiento OCR y procesamiento del lenguaje natural.

En el capítulo 4 se presenta la **elección de las herramientas** para cada una de las etapas mencionadas anteriormente. Además se evalúan las herramientas siguiendo los criterios de evaluación definidos y se realiza la comparación de las mismas.

En el capítulo 5 se describe el **diseño y desarrollo de la solución** propuesta en donde se presentan tanto los cambios realizados en las herramientas seleccionadas como el desarrollo de la aplicación que realiza la búsqueda de personas.

En el capítulo 6 se presentan los **resultados obtenidos** y las **conclusiones**, dando lugar al **trabajo a futuro** a partir de los resultados alcanzados en este trabajo.

El documento también cuenta con siete anexos, los cuales se describen a continuación:

En el **anexo 1** se presenta un conjunto de pruebas realizadas que comprueban la necesidad que existe de aplicar pre procesamiento a las imágenes previo a su digitalización.

En los **anexos 2 y 3** se muestra un conjunto de pruebas realizadas para encontrar cotas inferiores y superiores de los parámetros del script LocalTresh y las pruebas realizadas para alcanzar la mejor configuración de las herramientas de pre procesamiento LocalTresh y ScanTailor.

En el **anexo 4** se detallan los módulos de Freeling utilizados para el post procesamiento. Por otra parte, en el **anexo 5** se especifican los pasos de instalación que se deben seguir para cada una de las herramientas utilizadas en el desarrollo de la solución y la configuración a utilizar por la aplicación web desarrollada.

En el **anexo 6** se presentan los detalles de la interfaz de administración del framework web Django. Por último en el **anexo 7** se puede encontrar el análisis de la estructura de un expediente modelo donde se muestran a su vez las palabras claves identificadas en cada sección.

1.5 Confidencialidad

Debido a la confidencialidad de los documentos utilizados para la realización de este proyecto, se ocultaron los nombres que aparecen en las imágenes de muestra.

Capítulo 2 - Conceptos importantes

En este capítulo se presentan las definiciones de los conceptos relevantes sobre procesamiento OCR, métodos de pre procesamiento, análisis morfológico y extracción de información en archivos de texto, además de otros conceptos al momento de analizar este proyecto. Estos conceptos forman parte de la base teórica necesaria para entender el dominio del problema atacado.

2.1 OCR

El Reconocimiento Óptico de Caracteres (OCR), es un proceso que busca a partir de una imagen identificar automáticamente símbolos o caracteres pertenecientes a un determinado alfabeto de manera de almacenar éstos en forma de datos. Como consecuencia se puede interactuar con los datos mediante un programa de edición de texto o similar.

2.2 Pre procesamiento

En este informe, se entiende por pre procesamiento a los procesos a realizar sobre una imagen previo a la ejecución de los procesos OCR. Algunos conceptos importantes de la etapa de pre procesamiento son:

Binarización: es una herramienta del procesamiento de imágenes en la cual se transforma una imagen a dos colores: blanco y negro.

Umbral (threshold): valor utilizado en el proceso de binarización para decidir si el sector analizado toma color blanco (por debajo del umbral) o negro (por encima del umbral).

2.3 Post Procesamiento

Etapa posterior al procesamiento OCR, es la etapa donde se analizan los archivos de texto generados y se carga la base de datos.
Algunos conceptos manejados en esta etapa son:

Procesamiento del lenguaje natural: conjunto de métodos y técnicas eficientes desde un punto de vista computacional para la comprensión y generación de lenguaje natural. Técnicas y herramientas de PLN son utilizadas para obtener información relevante de los expedientes.

Tokenización: Segmentación del texto en unidades independientes. Estas unidades en general están dadas por las palabras y los signos de puntuación.

Capítulo 3 - Estado del arte

La principal motivación de este proyecto es promover la preservación de los documentos existentes sobre civiles sometidos a la Justicia Militar durante el proceso cívico-militar.
Con el paso del tiempo los expedientes físicos han sufrido un importante desgaste por lo que digitalizarlos es una manera de preservarlos y evitar el deterioro de los mismos debido a la consulta permanente.

El primer objetivo del proyecto es el estudio y relevamiento de técnicas de OCR para la transformación de archivos de imágenes a texto.
En este capítulo se muestran los resultados de una revisión de los papers y artículos más destacados sobre el tema así como el análisis de las herramientas mejor calificadas por la comunidad de desarrolladores y usuarios de OCR.
Previo al estudio de dichas herramientas se debe destacar que la investigación está planteada de manera exploratoria y no orientada a resultados, por lo que se tratará de encontrar la herramienta que mejor se adapte al contexto del proyecto, sin tener a priori, la garantía de que se comporte de la manera deseada.

En los últimos años la digitalización de la información (textos, imágenes, sonido, etc.) ha devenido un punto de interés para la sociedad. En el caso concreto de los textos, existen y se generan continuamente grandes cantidades de información escrita, tipográfica o manuscrita en todo tipo de soportes. En este contexto, poder automatizar la introducción de caracteres evitando la entrada por teclado, implica un importante ahorro de recursos humanos y un aumento de la productividad, al mismo tiempo que se mantiene, o hasta se mejora, la calidad de muchos servicios.

Existen muchos proyectos de digitalización de libros, como Google Books [Ref.: GoogleBooks], PRImA (Pattern Recognition and Image Analysis Research) [Ref.:PRImA] o Gutemberg [Ref.: Gutemberg]. Sin embargo el mayor reto es la digitalización de libros de hace más de 30 años donde la tinta comienza a desgastarse o las páginas toman un color amarillento debido al paso de los años, en estos casos las herramientas OCR no tienen un 100% de efectividad y se necesita de la supervisión humana para obtener el resultado esperado. El proyecto Recaptcha [Ref.:Recaptcha] utiliza el poder de internet, ayudando a Google Books a digitalizar palabras que no se pudieron reconocer por medio de procesos automáticos.

3.1 Esquema OCR

El Reconocimiento Óptico de Caracteres (OCR), es un proceso que busca a partir de una imagen identificar automáticamente símbolos o caracteres de un determinado alfabeto para almacenarlos en forma de datos con los que se podrá interactuar mediante un programa de edición de texto o similar.
Si bien OCR está enfocado a la digitalización de textos, el reconocimiento de una palabra en una imagen está dada por la concatenación del reconocimiento de cada carácter individual [Ref.: Tao], por lo que una palabra bien reconocida depende de cada uno de los caracteres de la misma.

Figura 1 - Software OCR. Tomado de [Athento]

La principal ventaja de los sistemas OCR [Ref.: Ventajas y desventajas OCR] es la capacidad de buscar y obtener información a partir de imágenes y documentos escaneados.
Por otro lado existen muchas dificultades en el reconocimiento óptico de caracteres ya que las imágenes reales a digitalizar suelen no ser perfectas, por lo tanto se encuentran varios problemas de diversos tipos, como ruido que puede ser introducido por el escáner, la distancia entre caracteres, si no es uniforme puede producir errores de reconocimiento. La conexión de dos o más caracteres por píxeles comunes también puede producir varios errores.
Otras posibles fuentes de dificultad se deben al parecido entre ciertas letras y/o dígitos (U-V, C-L, a-d, n-h, I-1, Z-2, S-5, G-6, etc.), que incluso en algunos casos pueden escribirse igual (O-0, l-1, etc.) o casi indistinguiblemente de sus correspondientes en mayúsculas (O-o, K-k, C-c, etc.), lo que a menudo obliga a recurrir al contexto para diferenciarlas.

En el procesamiento de OCR están presentes cuatro pasos básicos para lograr el reconocimiento de caracteres: **binarización**, **fragmentación**, **adelgazamiento** y **comparación de patrones**. Desde la sección 3.1.1 hasta la 3.1.4 se describe cada uno de estos pasos.

3.1.1 Binarización

Se puede considerar la característica más importante a tener en cuenta en una imagen de entrada para un motor OCR. La binarización es una herramienta del procesamiento de imágenes en la cual se deja una imagen en dos colores: blanco y negro. Para hacer esto se debe primero transformar a escala de grises, con esto se gana que cada pixel pase de tener una tupla de tres valores (niveles de rojo, verde y azul) a un entero entre 0 y 255.
El proceso continúa fijando un valor de umbral entre 0 y 255, convirtiendo a 255 todos los valores superiores, y a 0 los menores a dicho umbral.
La calidad de la imagen resultado depende del valor de umbral seleccionado, por lo que para llegar al óptimo se debe variar este valor y observar cuál se adapta mejor a cada imagen.
También existen otros métodos de binarización local, donde se toma solamente una región del documento y se deja en blanco o negro dependiendo del entorno, estas técnicas se conocen como técnicas de Umbral Adaptativo y son utilizadas dentro del proyecto debido a que se adaptan mejor al caso.

3.1.1.1 Umbral Adaptativo

El umbral adaptativo (Adaptive Threshold) [Ref.: Adaptive Threshold] típicamente toma una imagen como entrada y, en su implementación más simple, la salida es una imagen binaria representando la segmentación. Para cada pixel en la imagen se calcula el umbral, si el valor del pixel se encuentra por debajo, el pixel es fijado con el color de fondo, de otro modo se fija el valor de primer plano.
Existen dos enfoques principales para encontrar el umbral: “Chow and Kaneko” [Ref: Chow and Kaneko] y "Local Threshold". El supuesto detrás de los dos métodos es que las regiones más pequeñas de una imagen son más propensas a tener la iluminación aproximadamente uniforme, por lo tanto se puede obtener un umbral más adecuado.
Chow y Kaneko divide una imagen en una serie de sub-imágenes superpuestas y luego encuentra el umbral óptimo para cada sub-imagen mediante la investigación de su histograma[1]. El umbral para cada píxel se encuentra mediante la interpolación de los resultados de las sub-imágenes. El inconveniente de este método es que es caro computacionalmente, por lo tanto, no es apropiado para aplicaciones en tiempo real.
Una alternativa es buscar el umbral local (“Local Threshold”) estadísticamente examinando la intensidad de los vecinos cercanos de cada pixel. Funciones sencillas y rápidas son la media de la distribución de intensidad local,

La media: T = *media*
La mediana: *T = mediana*

[1] Un **histograma** es una representación gráfica de una variable en forma de barras, donde la superficie de cada barra es proporcional a la frecuencia de los valores representados. En el eje vertical se representan las frecuencias, y en el eje horizontal los valores de las variables, normalmente señalando las marcas de clase, es decir, la mitad del intervalo en el que están agrupados los datos.

La media de los valores máximo y mínimo: $T = (max + min)/2$

El tamaño del vecindario debe ser lo suficientemente grande para cubrir suficientes píxeles del frente y del fondo, de lo contrario se elegirá un umbral malo. Por otro lado, elegir regiones muy grandes puede violar la presunción de iluminación aproximadamente uniforme. Este método es menos intensivo computacionalmente que Chow and Kaneko y produce buenos resultados para algunas aplicaciones.

3.1.2 Segmentación de la imagen

Este proceso, también conocido como *fragmentación*, es el más costoso y necesario para el posterior reconocimiento de caracteres. La segmentación de una imagen implica la detección mediante procedimientos de etiquetado determinista o estocástico de los contornos o regiones de la imagen, basándose en la información de intensidad o información espacial.
Permite la descomposición de un texto en diferentes entidades lógicas, que han de ser suficientemente invariables y suficientemente significativas para su reconocimiento.

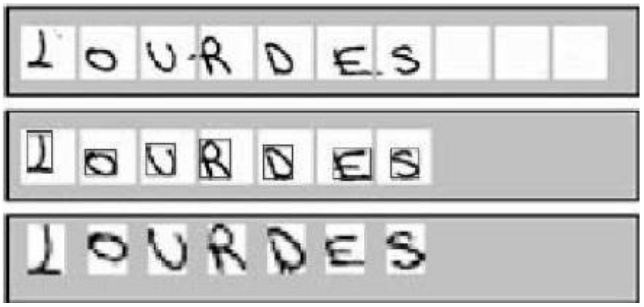


Figura 2 - Segmentación de la imagen

No existe un método genérico para llevar a cabo esta segmentación de la imagen que sea lo suficientemente eficaz para el análisis de un texto. Aunque, las técnicas más utilizadas son variaciones de los métodos basados en proyecciones lineales.
Una de las técnicas más clásicas y simples para imágenes de niveles de grises consiste en la determinación de los modos o agrupamientos (“clusters”) a partir del histograma, de tal forma que permitan una clasificación o umbralización de los pixeles en regiones homogéneas.

3.1.3 Adelgazamiento de las componentes

Una vez aisladas las componentes conexas de la imagen, se les aplica un proceso de adelgazamiento a cada una de ellas. Este procedimiento consiste en ir borrando

sucesivamente los puntos de los contornos de cada componente de forma que se conserve su tipología.
La eliminación de los puntos ha de seguir un esquema de barridos sucesivos para que la imagen continúe teniendo las mismas proporciones que la original y así conseguir que no quede deformada.

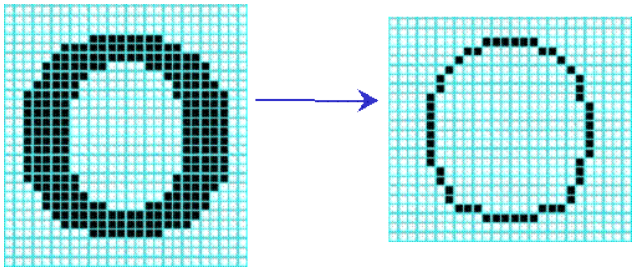


Figura 3 - Adelgazamiento de componentes

Se debe hacer un barrido en paralelo, es decir, señalar los píxeles borrables para eliminarlos todos a la vez. Este proceso se lleva a cabo para hacer posible la clasificación y reconocimiento, simplificando la forma de las componentes.

3.1.4 Comparación con patrones

En esta etapa se comparan los caracteres obtenidos anteriormente con unos teóricos (patrones) almacenados en una base de datos. El buen funcionamiento del OCR se basa en gran medida en una buena definición de esta etapa. Existen diferentes métodos para llevar a cabo la comparación. Uno de ellos es el Método de Proyección en el cual se obtienen proyecciones verticales y horizontales del carácter por reconocer y se comparan con el alfabeto de caracteres posibles hasta encontrar la máxima coincidencia.
Existen otros métodos como por ejemplo: Métodos geométricos o estadísticos, Métodos estructurales, Métodos Neuro-miméticos, Métodos Markovianos o Métodos de Zadeh. El análisis de estos métodos queda fuera del alcance de este proyecto.

3.2 Análisis de herramientas

En este capítulo se busca analizar las herramientas existentes para las distintas etapas del proceso de digitalización de imágenes. En la etapa de estado del arte se realizó un relevamiento de herramientas para las etapas de pre procesamiento, procesamiento OCR y post procesamiento y en esta etapa se realiza el análisis de las mismas orientado a las imágenes que se buscan digitalizar en este proyecto.

3.2.1 Pre procesamiento

De las cuatro etapas clásicas de un esquema OCR, la de binarización pertenece a la etapa de pre procesamiento debido a que es una tarea previa a la ejecución del motor OCR en sí mismo.
En esta sub sección se analizan algunas herramientas que realizan esta tarea. Si bien muchas herramientas OCR incluyen etapa de binarización, se detectó que al aplicar binarización externa a la herramienta se obtienen mejores resultados. En el Anexo 1 se pueden ver las diferencias para Tesseract, herramienta elegida para el procesamiento OCR.

3.2.1.1 Aletheia

Aletheia [Ref.: Aletheia] ha sido desarrollado en el contexto de IMPACT (IMProoving Acess to Text) [Ref.: IMPACT], un proyecto fundado por la Unión Europea que busca mejorar las tecnologías para la digitalización masiva de documentos históricos que comprende bibliotecas y universidades en todo Europa.
El principal objetivo de este sistema es la eficiencia, la precisión, la flexibilidad y usabilidad en gran escala.

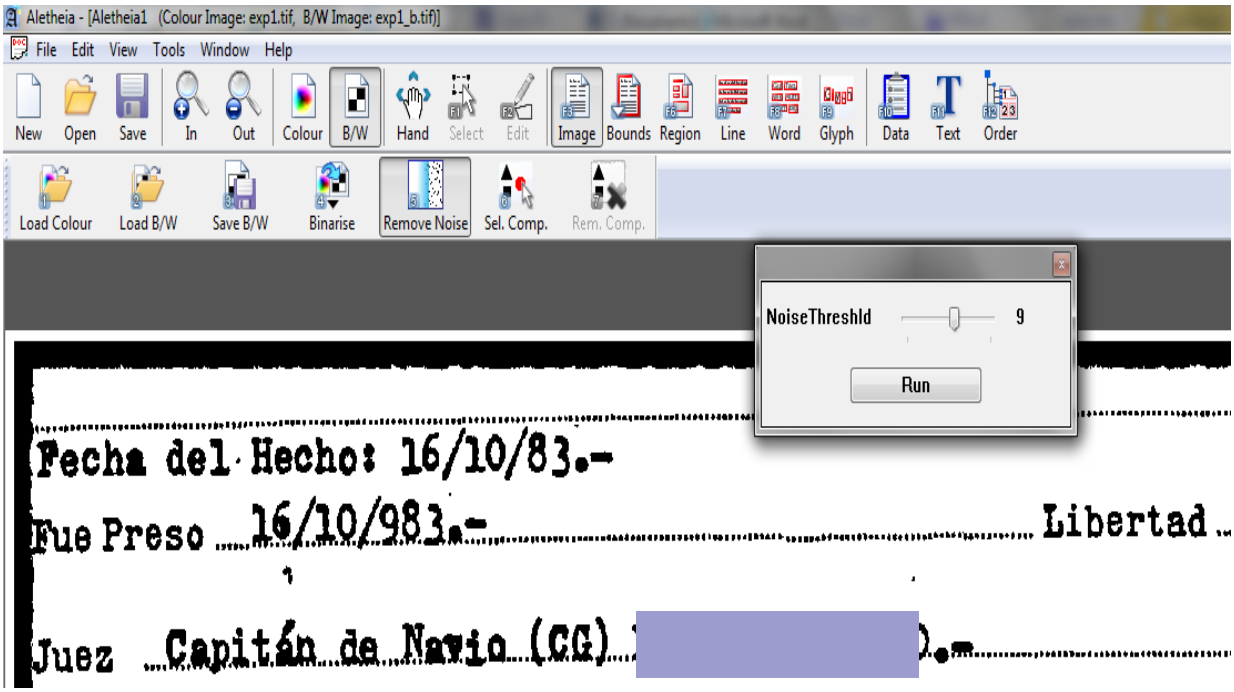


Figura 4 - Aletheia

Aletheia posee herramientas de binarización, segmentación, selección de componentes y reducción de ruido.
La herramienta de binarización posee tres métodos para esa tarea, estos son:

- Umbralización global: Dado un valor de umbral, este se utiliza para binarizar todo el documento.
- Umbral auto calculado: Igual que el anterior salvo que el valor del umbral es auto calculado.
- Umbral adaptativo: Dado un tamaño de la ventana se calcula el umbral en zonas de ese tamaño.

Si bien la herramienta de binarización es muy simple y efectiva, Aletheia necesita de la interacción humana para poder realizar las tareas, objetivo que no es deseado para este proyecto.

3.2.1.2 ImageMagick

ImageMagick [Ref.: ImageMagick] es una suite de software para crear, editar, componer, o convertir imágenes de mapa de bits. Puede leer y escribir imágenes en una variedad de formatos (más de 100) incluyendo GIF, JPEG, PDF, PNG, Postscript, SVG y TIFF. ImageMagick se puede utilizar para cambiar el tamaño, convertir el formato de una imagen, girar, reflejar, rotar, distorsionar, eliminar ruido, cortar y transformar las imágenes, ajustar colores de la imagen, modificar el tinte, aplicar diversos efectos especiales, etc. Es una herramienta de software libre cuya funcionalidad es utilizada típicamente a través de la línea de comandos.
Para este proyecto en particular se analizó un script perteneciente a la suite de ImageMagick llamado **localtresh** [Ref.: Localtresh] el cual permite binarizar imágenes utilizando un enfoque de umbral adaptativo, donde para cada ventana la ubicación del pixel central se compara con alguna medida de:

- promedio o,
- una combinación del promedio y la desviación estándar o,
- el promedio de la desviación estándar absoluta dentro de la ventana.

Si el píxel central es más grande que esta medida por un valor de sesgo, entonces el píxel central se hace blanco, de lo contrario se hace negro.

3.2.1.3 Scan Tailor

Scan Tailor [Ref.: ScanTailor] es una herramienta interactiva para el pos procesamiento de páginas escaneadas. Lleva a cabo operaciones tales como la división de páginas, corrección de inclinación, añadir/eliminar bordes, entre otras.

Aunque se trata de una herramienta interactiva, Scan Tailor posee un modo de ejecución mediante línea de comandos en donde se pueden indicar los parámetros para procesar automáticamente las imágenes.
Scan Tailor es software libre y fue desarrollado tanto para Windows como para Linux.

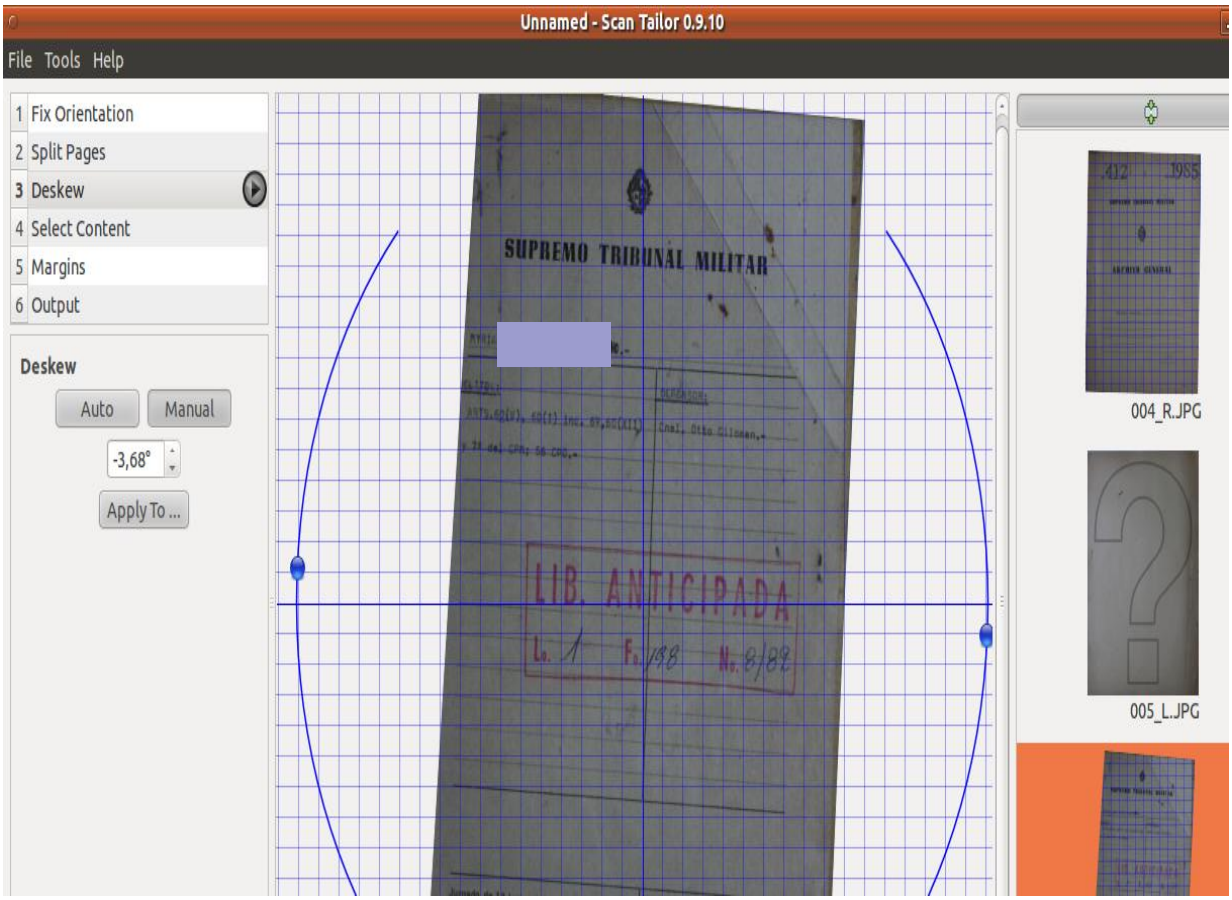


Figura 5 - Scan Tailor

3.2.2 Procesamiento OCR

Existen muchas herramientas para el procesamiento OCR, las cuales ofrecen diversas características según el tipo de imagen de entrada, realizan algunas correcciones y generan la salida correspondiente. En esta sección se mostrará el estudio de las herramientas más populares de OCR y se realizará la comparación de las mismas tomando como dominio recortes de los expedientes digitalizados por AJPROJUMI.

3.2.2.1 Omnipage

OmniPage 16 [Ref.: Omnipage] es un software OCR pago desarrollado por NUANCE [Ref.: Nuance]. Esta herramienta permite convertir documentos impresos, archivos PDF e imágenes, entre otros, en archivos electrónicos con capacidad de modificación, búsqueda, distribución y archivado. Este software, además, posee funcionalidades que permiten realizar un pre procesamiento de la imagen a ser convertida tales como la eliminación de ruido, la modificación del contraste, la modificación de la resolución, etc.
Entre sus principales características se destaca la conversión en 123 idiomas, el reconocimiento de diccionarios financieros, jurídicos y de especialidades médicas y el procesamiento automatizado de grandes lotes de archivos a partir de una carpeta origen. Posee varios núcleos de procesamiento paralelo, integración de herramientas de gestión de documentos y automatización en recogida de datos de formularios.

Esta herramienta se encuentra disponible para sistemas Operativos Windows.

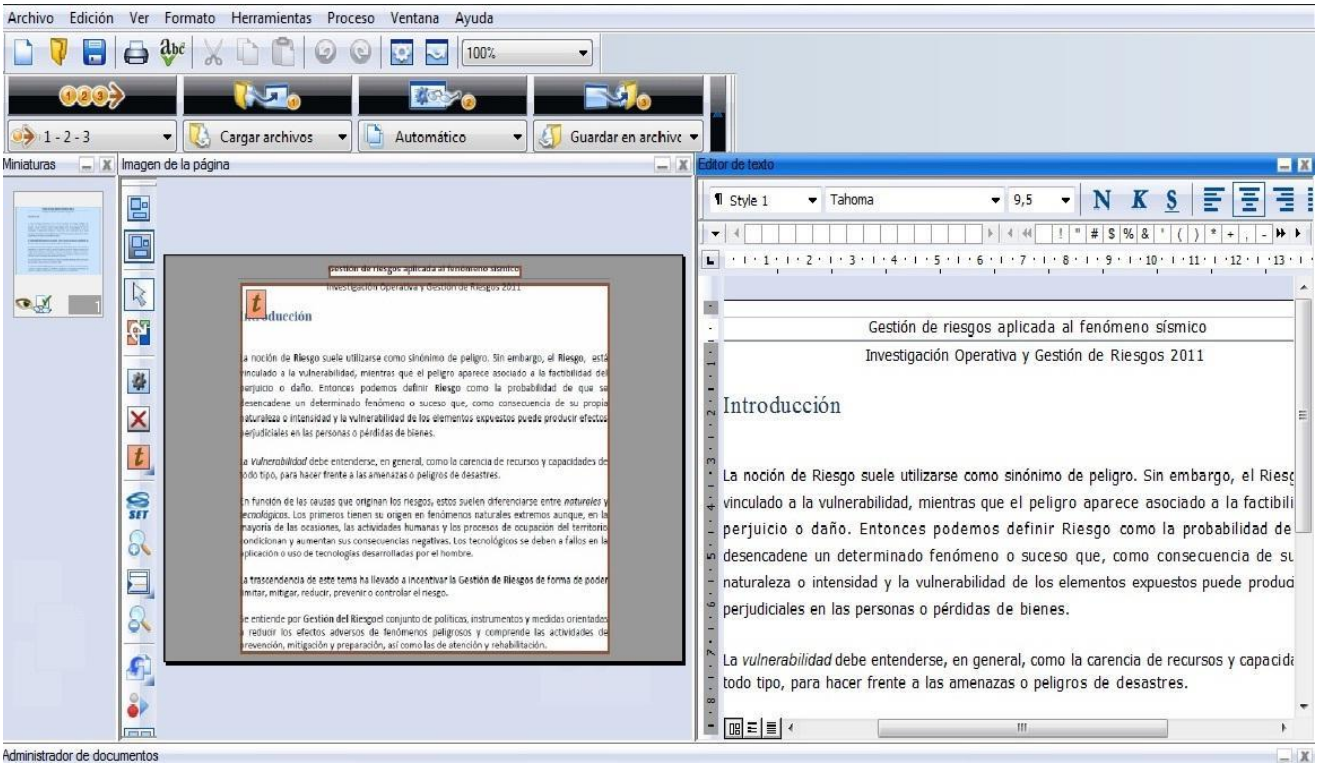


Figura 6 - Herramienta OmniPage

3.2.2.2 ABBYY FineReader

ABBYY FineReader 10 [Ref: ABBYY] es una herramienta de OCR desarrollada por ABBYY. FineReader fue diseñado como una aplicación de nivel profesional para convertir imágenes escaneadas, fotografías de documentos y archivos PDF a formatos editables y de búsqueda, tales como Microsoft Word, Microsoft Excel, Microsoft PowerPoint, RTF, HTML, PDF, CSV y archivos de texto.
ABBYY FineReader le permite desbloquear el contenido de archivos PDF e imágenes y convertirlos en información manejable y accesible.

Entre las principales características de este software se encuentran el reconocimiento de texto en 186 idiomas, conservación de casi todos los elementos de diseño del documento (gráficos, diagramas, imágenes, tablas, etc.), creación de diccionarios personalizados, funcionalidades de pre procesamiento de las imágenes a ser convertidas y procesamiento de documentos con múltiples páginas que se encuentra dividido para su ejecución en paralelo, lo que permite aumentar la velocidad de procesamiento.

Esta herramienta se encuentra disponible para sistemas Operativos Windows.

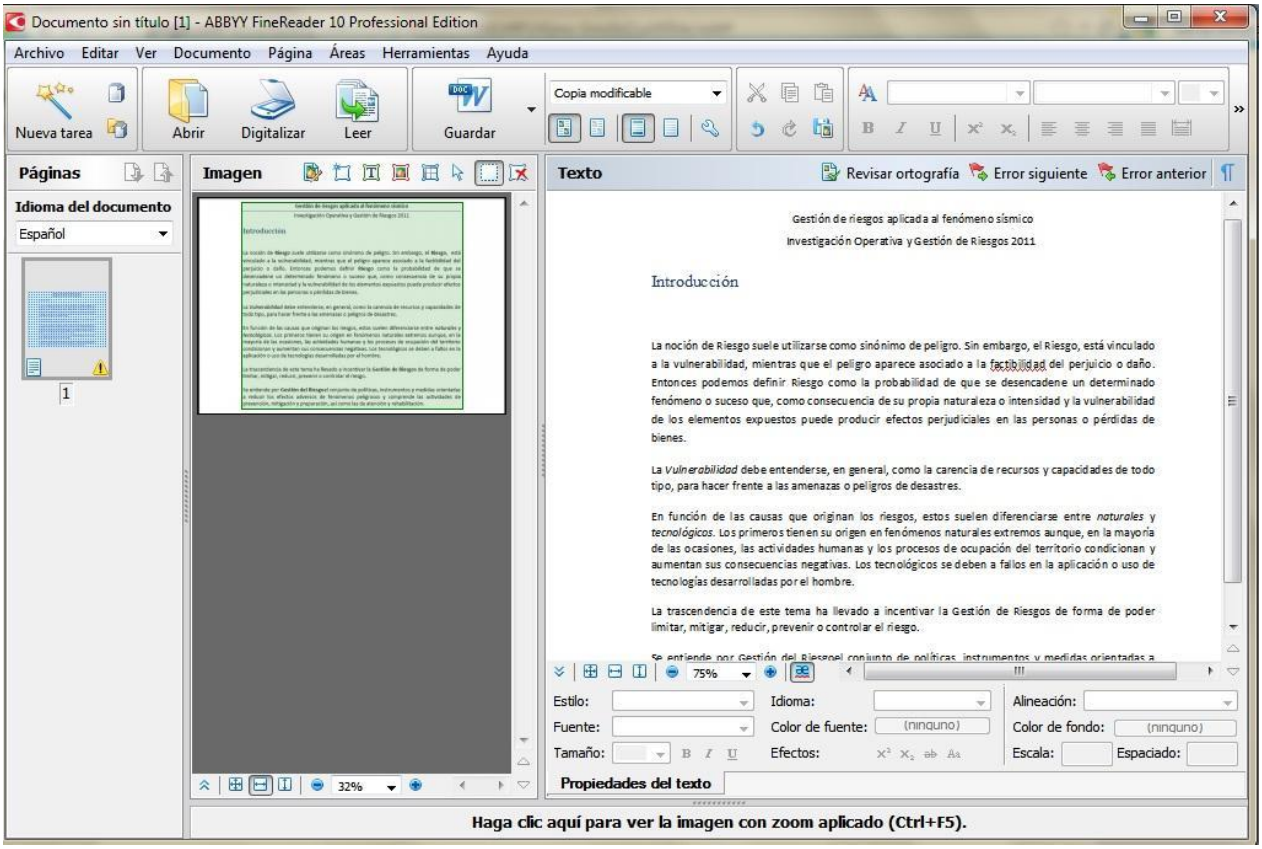


Figura 7 - Herramienta Abbyy FineReader

3.2.2.3 Presto! OCR

Presto! OCR 4 Pro [Ref.: Presto!] es un software de Reconocimiento Óptico de Caracteres desarrollado por NewSoft [Ref.: NewSoft]. Esta herramienta no sólo es capaz de leer documentos sino que conserva el formato del texto original así como el diseño de página, incluyendo las columnas, tablas y gráficos. Además, posee la funcionalidad de ejecución por lotes, es decir, Presto OCR! Pro puede escanear y reconocer varios documentos al mismo tiempo almacenando éstos para su posterior edición.
Las Principales características de este software son el reconocimiento en 53 idiomas, el entrenamiento de la herramienta en fuentes inusuales, símbolos y caracteres, la reducción de ruido en las imágenes así como el enderezamiento de las mismas.

Esta herramienta se encuentra disponible para sistemas Operativos Windows.

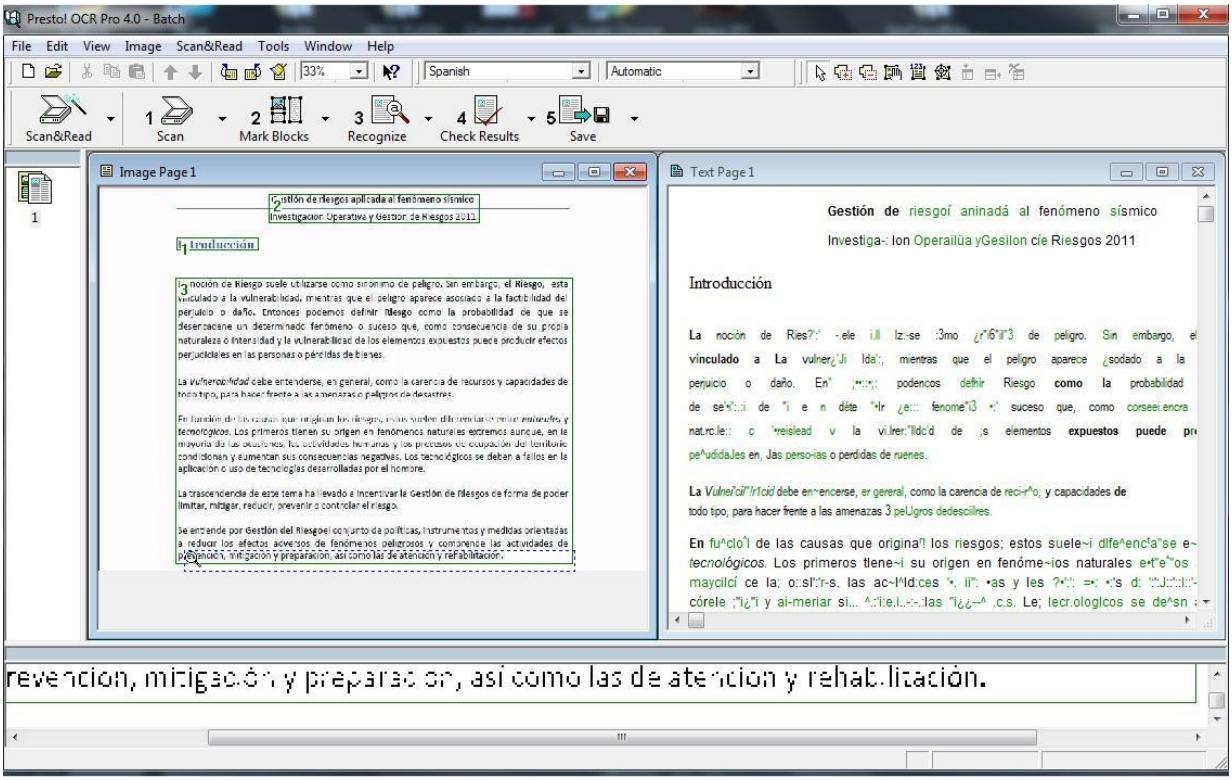


Figura 8 - Herramienta Presto! OCR

3.2.2.4 Readiris Pro

Readiris™ 11 Pro [Ref.: Readiris] es un software de OCR desarrollado por I.R.I.S. [Ref.: Iris] que permite digitalizar pequeños o grandes volúmenes de documentos escaneados y convertirlos en texto editable en una gran variedad de aplicaciones informáticas (Microsoft

Word, Microsoft Excel, PDF, HTML, etc.) con exactamente el mismo diseño que el documento original y sin necesidad de volver a escribir el texto.
Este software puede trabajar con documentos en papel escaneados, archivos PDF y archivos de imagen.

Entre las principales características de este software se pueden encontrar el reconocimiento de documentos multilingües e interfaz de usuario localizada a varios idiomas, el reconocimiento de documentos compuestos por varias páginas, herramientas para la mejora de imágenes, reconocimiento hasta en 118 idiomas así como el reconocimiento de letras manuscritas (ICR). Esta última característica se encuentra limitada al reconocimiento de números, letras mayúsculas separadas (de la A a la Z) y los siguientes signos de puntuación: "," (coma), "." (punto) y "-" (guión).
Su versión corporativa incluye reconocimiento por lotes de todas las imágenes previamente escaneadas que se encuentren dentro de una carpeta específica, donde los documentos reconocidos tendrán el mismo nombre que los archivos de imagen.

Esta herramienta se encuentra disponible para sistemas Operativos Windows y Mac OS X.

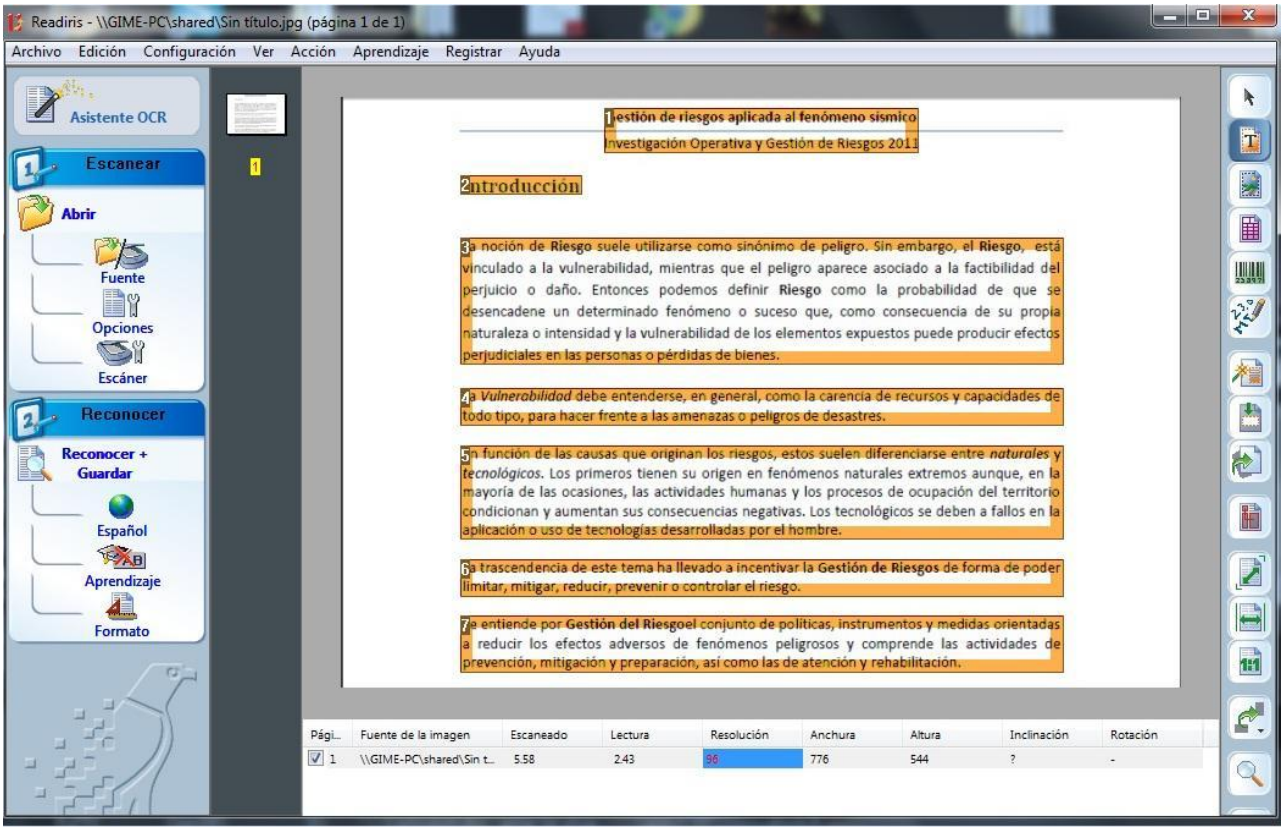


Figura 9 - Herramienta Readiris Pro

3.2.2.5 Maestro Recognition

Maestro Recognition Server 5.0 [Ref: Maestro] ha sido creado y diseñado para la fuerza industrial, la exploración del volumen empresarial y las necesidades de OCR. Maestro convierte todos los documentos escaneados en archivos PDF con posibilidad de realizar búsquedas en él. Los usuarios pueden configurar esta herramienta para obtener una mayor velocidad o precisión.
Una de las principales características de este software es la capacidad del procesamiento de imágenes agrupadas en carpetas para una mayor comodidad a la hora del procesamiento. Otras características son el procesamiento por lotes, la rapidez de procesamiento, el control de la salida generada en 10 formatos diferentes así como la entrada de 11 diferentes formatos de archivo incluyendo TIFF y PDF.

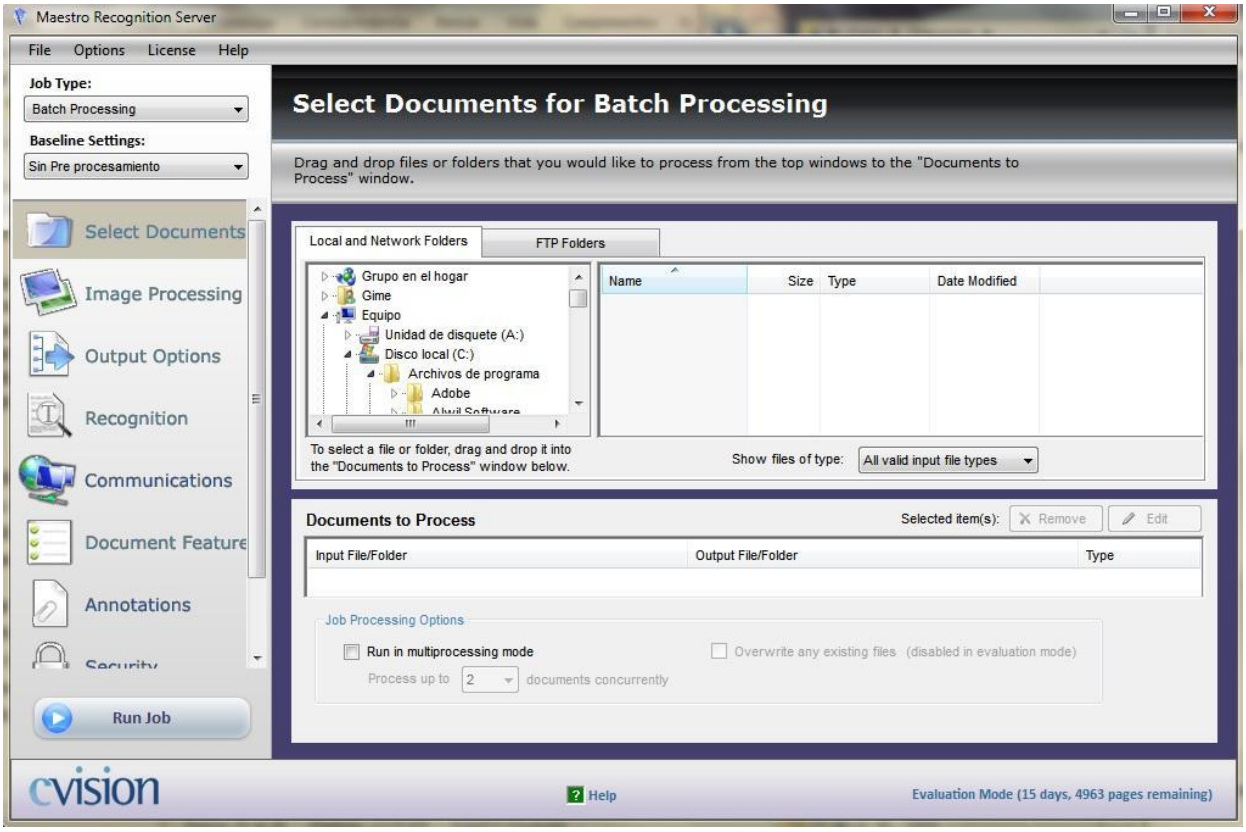


Figura 10 - Herramienta Maestro Recognition

3.2.2.5 Tesseract

Tesseract [Ref.: Tesseract] es un motor OCR libre. Fue desarrollado originalmente por Hewlett Packard como software propietario entre 1985 y 1995. Tras diez años sin ningún desarrollo, fue liberado como código abierto en el año 2005 por Hewlett Packard y la

Universidad de Nevada, Las Vegas. Tesseract es desarrollado actualmente por Google y distribuido bajo la licencia Apache, versión 2.0. Tesseract está considerado como uno de los motores OCR libres con mayor precisión disponible actualmente.

Tesseract es código abierto, multiplataforma, permite la ejecución por línea de comandos y brinda la posibilidad de entrenarlo para mejorar los resultados en un dominio particular, por ejemplo un tipo de letra especial [Ref.: TesseractEngine].

3.2.3 Post Procesamiento

Luego de realizar las etapas de binarización, segmentación, adelgazamiento de componentes y comparación de patrones, se deben implementar mecanismos para obtener información relevante en los textos generados. Para eso se estudió la herramienta Freeling, ya que además de ser de código abierto (LGPL), tiene un diccionario del español que cubre más del 90% de la lengua, el diccionario de más cobertura de uso totalmente libre.

3.2.3.1 Freeling

FreeLing [Ref.:Freeling] es una librería de código abierto utilizada para el procesamiento multilingüe automático, que proporciona una amplia gama de servicios de análisis lingüístico para diversos idiomas.
Ofrece a los desarrolladores de aplicaciones de Procesamiento de Lenguaje Natural funciones de análisis y anotación lingüística de textos, con la consiguiente reducción del coste de construcción de dichas aplicaciones.

La versión 3.0 provee varios módulos de análisis, cada uno de los cuales brinda distintas funcionalidades de procesamiento de texto, algunos ejemplos de estos módulos son:

- Módulo Identificador de lenguaje.
- Módulo de Tokenización.
- Módulo para detección de oraciones (Splitter).
- Módulo para detección de números.
- Módulo para detección de puntuación.
- Módulo para reconocimiento de entidades con nombre (named entities).
- Módulo para clasificación de entidades con nombre.

3.3 Problemas del OCR

El proceso básico que se lleva a cabo en el Reconocimiento Óptico de Caracteres es convertir el texto que aparece en una imagen en un archivo de texto que podrá ser editado y utilizado como tal por cualquier otro programa o aplicación que lo necesite.

Partiendo de una imagen perfecta, es decir, una imagen con sólo dos niveles de gris y sin ruido, el reconocimiento de estos caracteres se realizará básicamente comparándolos con unos patrones o plantillas que contienen todos los posibles caracteres. Ahora bien, las imágenes reales no son perfectas, por lo tanto el OCR se encuentra con varios problemas, a saber:

- El dispositivo que obtiene la imagen puede introducir niveles de grises al fondo que no pertenecen a la imagen original.
- La resolución de estos dispositivos puede introducir ruido en la imagen, afectando los píxeles que han de ser procesados.
- La distancia que separa a unos caracteres de otros, al no ser siempre la misma, puede producir errores de reconocimiento.
- La conexión de dos o más caracteres por píxeles comunes también puede producir errores.

Capítulo 4 - Elección de las herramientas

Luego de realizar el relevamiento de las herramientas más populares para la transformación de una imagen a texto editable, es necesario escoger aquellas que se adecuen mejor al contexto del proyecto. Es por esto que se realizaron pruebas de distintas herramientas en cada una de las grandes etapas del proceso: **pre procesamiento**, **OCR** y **post procesamiento**.
En este capítulo se describen las pruebas realizadas para la elección de la herramienta en cada etapa.

4.1 Herramienta de pre procesamiento

Los buenos resultados de un motor OCR dependen fuertemente de la calidad de la imagen que reciba como entrada, por lo que se deben realizar tareas de pre procesamiento en las imágenes para obtener resultados aceptables. A modo de ejemplo, un conjunto de tareas para el pre procesamiento puede ser: Pasar a escala de grises -> Binarizar -> Reducción de ruido -> Recortar bordes.
Debido al estado de deterioro que presentan las imágenes de los expedientes a ser manipulados, se decidió realizar un pre procesamiento de éstas para obtener de esta forma resultados aceptables.
Si bien muchas herramientas OCR incluyen etapa de binarización, se detectó que al aplicar binarización externa a la herramienta se obtienen mejores resultados. En el Anexo 1 se pueden ver las diferencias para Tesseract, herramienta elegida para el procesamiento OCR.
En esta sección se describen las pruebas realizadas para la etapa de pre procesamiento para la elección de la herramienta.

4.1.1 Herramientas comparadas

Si bien existen muchos programas para la corrección y transformación de imágenes, no todas poseen la posibilidad de trabajar por lotes y de invocarlas mediante otros procesos automáticos. Por esta razón es que la elección se realizará entre las herramientas ScanTailor y Localtresh.

Si bien Scan Tailor [Ref.: Scan Tailor] es una herramienta interactiva, posee un modo de ejecución mediante línea de comandos en donde se pueden indicar algunos parámetros para la transformación de la imagen.
En su ejecución por línea de comandos, Scan Tailor lleva a cabo operaciones de división de páginas, corrección de inclinación, recorte de bordes, reducción de ruido y detección de contenido.

Por su parte Localtresh [Ref.: **Localtresh**] es un script perteneciente a la suite de ImageMagick con el que se pueden realizar diversas operaciones para la mejora de una imagen. A modo de resumen este script binariza una imagen utilizando un enfoque de ventana móvil con umbral adaptativo. Los parámetros que recibe este script son el método de binarización, el radio de la ventana y el sesgo de umbralización.

4.1.2 Método de evaluación

Para la comparación de Localtresh vs. Scan Tailor se seleccionó un conjunto de imágenes representativas de los expedientes de AJPROJUMI. A partir de dicho conjunto se realizó el siguiente procedimiento:

- Ejecutar cada herramienta con su mejor configuración sobre todas las imágenes de la muestra.
- Ejecutar Tesseract para cada imagen resultado del punto anterior.
- Comparar los archivos de texto obtenidos por cada herramienta de forma de seleccionar aquella que obtenga mejores resultados.
- Asignar una unidad a la herramienta por cada archivo generado que tenga la mejor tasa de acierto.
- Elegir la herramienta que obtuvo más puntos.

La selección de las imágenes se realizó de forma de poder abarcar los distintos tipos de estas que fueron detectados en los expedientes; correspondiendo dichos tipos a la combinación de elementos tales como grado de inclinación y existencia o no de renglones en las imágenes.

4.1.3 Configuración

En el Apéndice 3 - Elección de parámetros óptimos se muestran las pruebas realizadas para llegar a la mejor combinación de parámetros en cada herramienta. En esta sección se presenta la comparación de ambas herramientas utilizando la mejor configuración encontrada para cada una.

La mejor configuración de ScanTailor fue:

Parámetro	Valor
Content-detection	Cautios
Color-mode	black_and_white
Despeckle	Aggressive
Layout	auto detect
Deskew	Auto

Tabla 1 - Mejor Configuración Scan Tailor

Mientras que los parámetros con los que Localtresh presentó mejores resultados fueron los siguientes:

Parámetro	Valor
Método	3
Radio	42
Sesgo	35
Negado	Yes

Tabla 2 - Mejor configuración LocalTresh

4.1.4 Resultado de las pruebas

En esta sección se muestran los resultados obtenidos al comparar las herramientas de pre procesamiento descritas en la sección 4.1.1 así como la conclusión obtenida.

Imagen	Características de la Imagen	Scan Tailor	Localtresh
009_LC	➢ Sin renglones ➢ Con inclinación	0	1
016_RC	➢ Con renglones ➢ Con gran inclinación	1	0
022_RC	➢ Sin renglones ➢ Con inclinación	1	0
039_L	➢ Con renglones ➢ Con pequeña inclinación	1	0
215_L	➢ Sin renglones ➢ Sin inclinación	0	1
338_R	➢ Sin renglones ➢ Con pequeña inclinación	1	0
ImgSRCI	➢ Sin renglones ➢ Con inclinación	0	1
ImgCRCI	➢ Con renglones ➢ Con gran inclinación	1	0
ImgSRSI	➢ Sin renglones ➢ Sin inclinación	1	1
ImgSRCMI	➢ Sin renglones ➢ Con pequeña inclinación	1	0
TOTAL		**7**	**4**

Tabla 3 - LocalTresh vs. Scan Tailor

Como se observa en la tabla 3 se obtuvieron mejores resultados con la herramienta ScanTailor ya que de 10 pruebas realizadas sobre imágenes seleccionadas a partir de distintos expedientes ésta superó a Localtresh en seis de ellas empatando en solo una, dejando también en evidencia que Localtresh no se comporta de la forma deseada frente a imágenes que contienen renglones.

Dado el resultado de las pruebas presentadas y teniendo en cuenta que existe un porcentaje alto de imágenes con renglones e inclinación en los expedientes se seleccionó a Scan Tailor como la herramienta de pre procesamiento a ser utilizada. Si bien ésta fue seleccionada en lugar de Localtresh, en la tabla 3 se puede apreciar que no hay una herramienta que tenga mejor resultado sobre todas las imágenes de la muestra.

4.2 Herramienta de OCR

En la sección 3.2.2 se describen las herramientas comerciales más destacadas para el procesamiento OCR. En este subcapítulo se realiza la comparación de las mismas y la elección de la herramienta más adecuada para el marco del proyecto de grado.

4.2.1 Herramientas comparadas

A la hora de elegir una herramienta de procesamiento OCR se debe analizar lo que el mercado ofrece y tener en cuenta cuáles se adaptan mejor al proyecto en el cual se va a utilizar.

Debido a que la cantidad de herramientas existentes en el mercado es bastante amplia se seleccionaron aquellas que cumplían con las necesidades que requería el proyecto teniendo en cuenta además las valoraciones que tenían éstas dentro del mercado [Ref.: Herramientas OCR].

Conjuntamente se realizó un estudio de las herramientas de OCR de software libre dando estas resultados no aceptables con la excepción del motor de OCR Tesseract.
Por las razones enumeradas anteriormente la elección se realizó entre las siguientes herramientas de OCR: *OmniPage*, *Abby FineReader*, *Presto! OCR*, *Readiris Pro*, *Maestro Recognition Server* y *Tesseract*. Estas Herramientas fueron descritas en la sección 3.2.2.

4.2.2 Método de evaluación

Para comparar las distintas herramientas OCR seleccionadas se eligió un conjunto de imágenes representativas de los expedientes de AJPROJUMI. Si bien la mayoría de estas herramientas tienen incorporada la etapa de binarización, se observó que se obtenían mejores resultados al realizar un pre procesamiento de las imágenes. Para realizar estas pruebas se optimizaron las imágenes con la aplicación **Scan Tailor**.

Para llevar a cabo la elección del software de OCR a utilizar se definió una lista de criterios ponderados de forma de poder puntuar cada una de las herramientas bajo estudio. Los criterios definidos fueron los siguientes:

Desempeño

A la hora de elegir un programa una de las cosas más importantes que se evalúa es si hace lo que debe hacer y de forma correcta. En este contexto particular lo que se evalúa es que la herramienta pueda convertir el contenido de una imagen a texto editable con la

mayor precisión posible. Dado que esta es una característica fundamental que debe tener el software elegido se la pondero con el valor 2.

Plataforma

Dado que AJPROJUMI es un proyecto desarrollado dentro de las instalaciones del Poder Judicial en las cuales los pc's cuentan con sistemas operativos Unix se decidió tener en cuenta dicha característica ponderando a ésta con el valor 1.8.

Formato del archivo de salida

Debido a que el proceso de transformación de imágenes a textos editables mediante una herramienta de OCR es solo una de las etapas que se deben cumplir para poder lograr la extracción de información relevante de los expedientes, se concluyó que el formato del archivo procesado juega un rol importante ya que se debe trabajar con su contenido en etapas posteriores. Se ponderó con el valor 2 a dicha característica.

Ejecución Automatizada

Análogamente al punto anterior se consideró importante que la etapa de conversión de imágenes a texto editable fuera automática (sin interacción humana) de forma de poder integrarla al proceso de extracción de datos. Se ponderó con el valor 2 a dicha característica.

Ejecución por lotes

La ejecución por lotes es una característica similar a la anterior con la salvedad de que se necesita la interacción humana para llevarla a cabo, es decir, esta propiedad permite la conversión de un gran volumen de imágenes (p. ej. imágenes alojadas en una carpeta del sistema) a través de una acción solicitada por un usuario. Dado que esta tarea es parcialmente automatizada se la pondero con el valor 1.5.

Software Pago

Por un tema de costos se decidió tener en cuenta a la hora de la elección de la herramienta OCR si ésta era paga o no. Por tal motivo se ponderó con 1.5 dicha característica.

Código Abierto

Debido a que el problema de la digitalización de archivos (imágenes, pdf's, etc.) en textos electrónicos editables que representen una copia fiel del original aún está lejos de resolverse en casos en los cuales los archivos a digitalizar son antiguos o están en mal

estado como los tratados en este proyecto; se decidió que sería interesante tener en cuenta esta característica para que en el caso que lo amerite se pueda explotar, por tanto se ponderó con el valor 1.3 a dicha propiedad.

El procedimiento llevado a cabo para realizar la comparación de las diferentes herramientas fue el siguiente:

- Pre procesamiento de las imágenes seleccionadas.
- Obtención de los archivos de texto producto de la aplicación de cada una de las herramientas de OCR sobre las imágenes de la muestra.
- Cálculo de la tasa de aciertos que tuvo cada herramienta en cada uno de los archivos obtenidos en el paso anterior. Luego se promedian dichos cálculos agrupados por herramienta de forma de obtener un promedio de la tasa de aciertos.
- Valoración de las herramientas según los criterios antes definidos.
- Elección de la herramienta según su puntuación.

4.2.3 Resultado de las pruebas

En esta sección se presenta una muestra de las pruebas realizadas en las diferentes herramientas OCR así como la comparación de los resultados y la conclusión alcanzada.

Análisis de desempeño

En este punto se mostrarán los resultados obtenidos al procesar las diferentes imágenes seleccionadas en cada una de las herramientas OCR así como una valoración de las mismas.

Imagen 1

PREGUNTADO:- Si ha estado detenido con anterioridad, en caso afirmativo fecha, causas.- -
CONTESTA:- Que fue detenido en el mes de febrero de 1974, permaneciendo detenido en el Penal de Punta Carretas hasta el mes de mayo del año 1975, habiendo sido procesado por la Justicia Militar por el delito de "Ataque a la Fuerza Moral de las FF.AA.", en razón de haber sido sorprendido distribuyendo volantes.- Que a la fecha se encuentra bajo el regimen de Libertad Vigilada, cumpliendo sus presentaciones men-- suales en la Guardia de Seguridad de la Dirección de Información e - Inteligencia.- -

Figura 11 - Fragmento correspondiente a la imagen 008_R del expediente A86-T1-N333

Herramienta	Caracteres reconocidos	Caracteres del documento	Tasa de Acierto
OmniPage	488	495	98.6%
Abbyy FineReader	488	495	98.6%
Presto! OCR Pro	428	495	86.5%
Readiris Pro	450	495	90.9%
Maestro Recognition server	361	495	72.9%
Tesseract	483	495	97.6%

Tabla 4 - Desempeño alcanzado por cada herramienta sobre la figura 11

Imagen 2

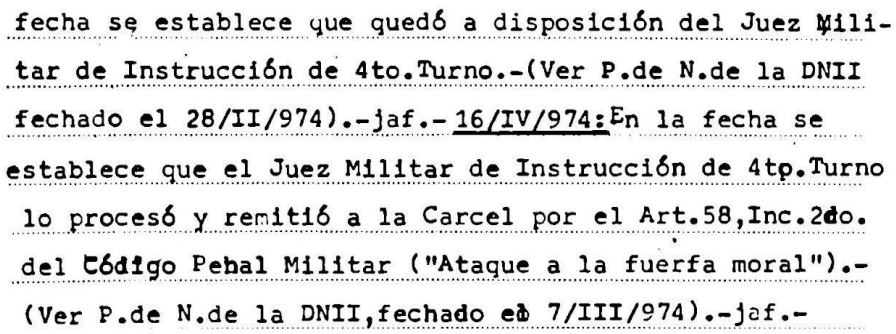
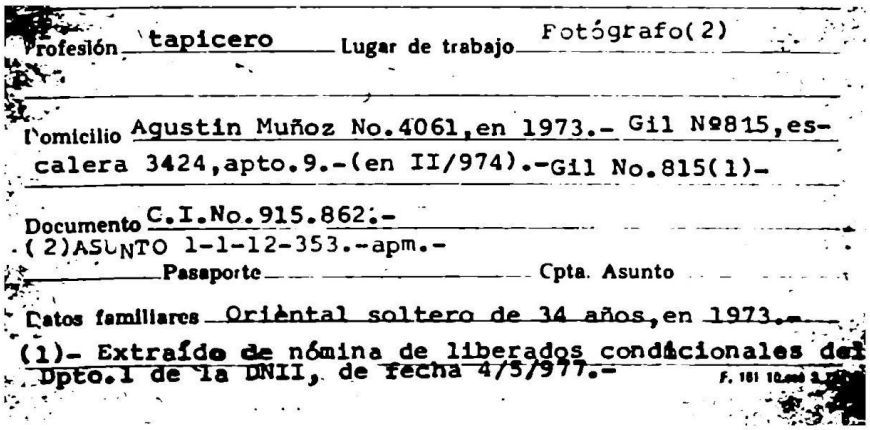

fecha se establece que quedó a disposición del Juez Militar de Instrucción de 4to.Turno.-(Ver P.de N.de la DNII fechado el 28/II/974).-jaf.- 16/IV/974: En la fecha se establece que el Juez Militar de Instrucción de 4tp.Turno lo procesó y remitió a la Carcel por el Art.58,Inc.2do. del Código Pebal Militar ("Ataque a la fuerfa moral").- (Ver P.de N.de la DNII,fechado en 7/III/974).-jaf.-

Figura 12 - Fragmento correspondiente a la imagen 016_RC del expediente A86-T1-N333

Herramienta	Caracteres reconocidos	Caracteres del documento	Tasa de Acierto
OmniPage	313	329	95.1%
Abbyy FineReader	265	329	80.5%
Presto! OCR Pro	310	329	94.2%
Readiris Pro	12	329	3.6%
Maestro Recognition server	308	329	93.6%
Tesseract	303	329	92.1%

Tabla 5 - Desempeño alcanzado por cada herramienta sobre la figura 12

Imagen 3


Profesión tapicero Lugar de trabajo Fotógrafo(2)

Domicilio Agustín Muñoz No.4061,en 1973.- Gil Nº815,es-
calera 3424,apto.9.-(en II/974).-Gil No.815(1)-

Documento C.I.No.915.862:-
(2)ASUNTO 1-1-12-353.-apm.-
Pasaporte Cpta. Asunto

Datos familiares Oriental soltero de 34 años,en 1973.-
(1)- Extraído de nómina de liberados condicionales del
Dpto.1 de la DNII, de fecha 4/5/977.-

Figura 13 - Fragmento correspondiente a la imagen 014_RC del expediente A86-T1-N333

Herramienta	Caracteres reconocidos	Caracteres del documento	Tasa de Acierto
OmniPage	263	317	83.0%
Abbyy FineReader	210	317	66.2%
Presto! OCR Pro	212	317	66.9%
Readiris Pro	161	317	50.8%
Maestro Recognition server	177	317	55.8%
Tesseract	196	317	61.8%

Tabla 6 - Desempeño alcanzado por cada herramienta sobre la figura 13

Imagen 4

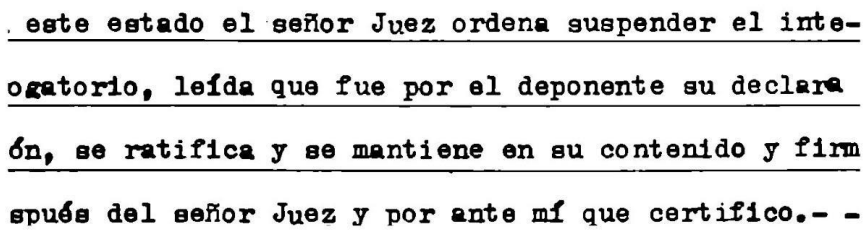
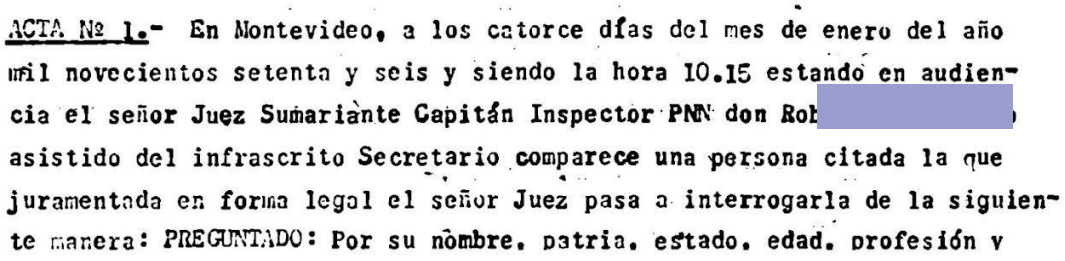

este estado el señor Juez ordena suspender el inte-
rogatorio, leída que fue por el deponente su declara
ción, se ratifica y se mantiene en su contenido y firm
spués del señor Juez y por ante mí que certifico.- -

Figura 14 - Fragmento correspondiente a la imagen 032_R del expediente A86-T1-N333

Herramienta	Caracteres reconocidos	Caracteres del documento	Tasa de Acierto
OmniPage	162	169	95.9%
Abbyy FineReader	157	169	92.9%
Presto! OCR Pro	159	169	94.1%
Readiris Pro	150	169	88.8%
Maestro Recognition server	118	169	69.8%
Tesseract	160	169	94.7%

Tabla 7 - Desempeño alcanzado por cada herramienta sobre la figura 14

Imagen 5



ACTA Nº 1.- En Montevideo, a los catorce días del mes de enero del año mil novecientos setenta y seis y siendo la hora 10.15 estando en audiencia el señor Juez Sumariante Capitán Inspector PNN don Rob asistido del infrascrito Secretario comparece una persona citada la que juramentada en forma legal el señor Juez pasa a interrogarla de la siguiente manera: PREGUNTADO: Por su nombre, patria, estado, edad, profesión y

Figura 15 - Fragmento correspondiente a la imagen 017_L del expediente A86-T1-N337

Herramienta	Caracteres reconocidos	Caracteres del documento	Tasa de Acierto
OmniPage	267	271	98.5%
Abbyy FineReader	266	271	98.2%
Presto! OCR Pro	263	271	97.0%
Readiris Pro	245	271	90.4%
Maestro Recognition server	254	271	93.7%
Tesseract	263	271	97.0%

Tabla 8 - Desempeño alcanzado por cada herramienta sobre la figura 15

Imagen 6

Para esta prueba en particular solo se consideró el texto que aparece en letra imprenta como la totalidad de caracteres del documento ya que no va a ser considerado en el cálculo de desempeño de cada herramienta. No obstante se decidió mostrar una imagen que contuviera letra manuscrita de forma de poder ilustrar la problemática que representa el reconocimiento de este tipo de letra (ICR).

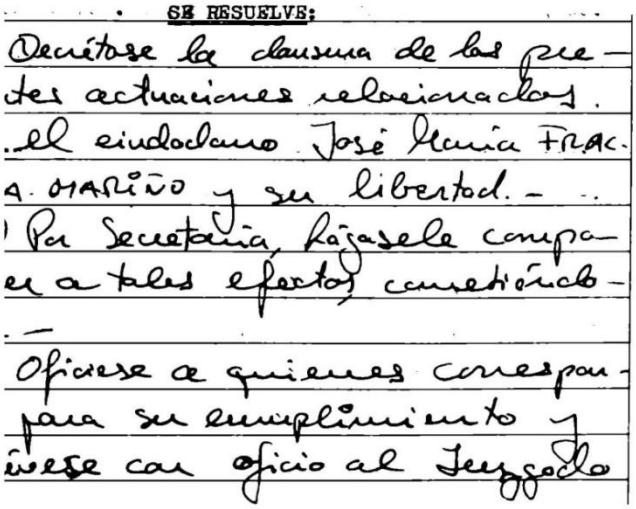

SE RESUELVE:
Decrétase la clausura de las pre-
tes actuaciones relacionadas.
.el ciudadano José Maria FRAC.
A. MARIÑO y su libertad. -
Por Secretaría, líbrasele compa-
en a tales efectos comunicado-
.-
Ofíciese a quienes corresponda
para su cumplimiento y
ívese con oficio al Juzgado

Figura 16 - Fragmento correspondiente a la imagen 042_R del expediente A86-T1-N333

Herramienta	Caracteres reconocidos	Caracteres del documento	Tasa de Acierto
OmniPage	0	11	00.0%
Abbyy FineReader	4	11	36.4%
Presto! OCR Pro	4	11	36.4%
Readiris Pro	0	11	00.0%
Maestro Recognition server	0	11	00.0%
Tesseract	8	11	72.7%

Tabla 9 - Desempeño alcanzado por cada herramienta sobre la figura 16

En base a las pruebas presentadas se realizó el cálculo del desempeño alcanzado por cada una de las herramientas de OCR seleccionadas, dando como resultado la tabla mostrada a continuación:

Herramienta	Desempeño	Valoración
OmniPage	**94.2%**	**0.942*2 = 1.884**
Abbyy FineReader	87.3%	0.873*2 = 1.746
Presto! OCR Pro	87.7%	0.877*2 = 1.754
Readiris Pro	64.9%	0.649*2 = 1.298
Maestro Recognition server	77.2%	0.772*2 = 1.544
Tesseract	88.6%	0.886*2 = 1.772

Tabla 10 - Desempeño promedio alcanzado por cada herramienta

La tabla10 muestra el desempeño promedio alcanzado por cada herramienta OCR seleccionada así como el puntaje obtenido. Dicho puntaje se calculó como el producto entre el factor de ponderación y el valor de desempeño de cada herramienta.

Análisis de la plataforma

Aquí se indicará si la herramienta OCR que se está analizando ofrece o no una versión de la misma compatible con sistemas operativos Unix. En caso que exista tal versión de la herramienta se le asignará el valor uno, de lo contrario, se asignará el valor cero. Dado que cada criterio definido fue ponderado, el puntaje final correspondiente a esta característica se calculará como el producto de los valores antes mencionados.

Herramienta	Plataforma Unix	Valoración
OmniPage	No	0*1.8 = 0
Abbyy FineReader	No	0*1.8 = 0
Presto! OCR Pro	No	0*1.8 = 0
Readiris Pro	No	0*1.8 = 0
Maestro Recognition server	No	0*1.8 = 0
Tesseract	**Si**	**1*1.8 = 1.8**

Tabla 11 - Análisis de la plataforma para cada herramienta

La tabla anterior muestra el puntaje obtenido por cada herramienta OCR seleccionada dependiendo de las versiones de sistema operativo que ofrezca cada una de ellas.

Análisis del formato del archivo de salida

En este punto se evaluará si la herramienta OCR que se está analizando ofrece como formato posible para el archivo procesado el de texto plano o texto sin formato. En caso que exista tal opción se le asignará a la herramienta el valor uno, en caso contrario, el valor cero. Dado que cada criterio definido fue ponderado, el puntaje final correspondiente a esta característica se calculará como el producto de los valores antes mencionados.

Herramienta	Formato de salida: Texto plano	Valoración
OmniPage	Si	1*2 = 2
Abbyy FineReader	Si	1*2 = 2
Presto! OCR Pro	Si	1*2 = 2
Readiris Pro	Si	1*2 = 2
Maestro Recognition server	Si	1*2 = 2
Tesseract	Si	1*2 = 2

Tabla 12 - Análisis del formato de salida ofrecido en cada herramienta

La tabla 12 muestra el puntaje obtenido por cada herramienta OCR seleccionada dependiendo de los formatos en los que pueden ser almacenados los archivos procesados.

Análisis de la ejecución automatizada y por lotes

Dado que estos criterios están fuertemente relacionados se optó por analizarlos de forma conjunta. La forma de evaluación elegida consistirá en valorar primero la propiedad con más peso, en este caso la ejecución automatizada, de forma tal que si la herramienta cumple con ésta no se evaluará la siguiente característica. En el caso en que la herramienta no fuera automatizable el puntaje de la misma estaría dado por el cumplimiento o no de la propiedad de ejecución por lotes.

Dado que cada criterio definido fue ponderado, el puntaje final se calculará como el producto entre el factor de ponderación y el valor obtenido por la herramienta según el cumplimento de las propiedades antes mencionadas.

Herramienta	Ejecución automatizada	Ejecución por lotes	Valoración
OmniPage	Si	---	1*2 = 2
Abbyy FineReader	Si	---	1*2 = 2
Presto! OCR Pro	No	Si	1*1.5 = 1.5
Readiris Pro	Si	---	1*2 = 2
Maestro Recognition server	No	Si	1*1.5 = 1.5
Tesseract	Si	---	1*2 = 2

Tabla 13 - Análisis de ejecución automatizada y por lotes de cada herramienta

La tabla anterior muestra el puntaje obtenido por cada herramienta de OCR seleccionada que cumpla con alguna de las propiedades antes mencionadas.

Análisis de software pago

Aquí se indicará si la herramienta OCR que se está analizando es paga o no. En caso que no lo sea se le asignará el valor uno, de lo contrario, se le asignará el valor cero. Debido a que cada criterio definido fue ponderado, el puntaje final correspondiente a esta característica se calculará como el producto de los valores antes mencionados.

Herramienta	Software Pago	Valoración
OmniPage	No	0*1.5 = 0
Abbyy FineReader	No	0*1.5 = 0
Presto! OCR Pro	No	0*1.5 = 0
Readiris Pro	No	0*1.5 = 0
Maestro Recognition server	No	0*1.5 = 0
Tesseract	**Si**	**1*1.5 = 1.5**

Tabla 14 - Análisis del licenciamiento de cada herramienta

La tabla 14 muestra el puntaje obtenido por cada herramienta OCR seleccionada dependiendo de si ésta es paga o no.

Análisis de código abierto.

En este último punto se valorará si el software OCR que se está analizando es una herramienta de código abierto. En caso que lo sea el valor que se le asignará a la misma será de uno, en caso contrario, será de cero. Dado que cada criterio definido fue ponderado, el puntaje final correspondiente a esta propiedad se calculará como el producto de los valores antes mencionados.

Herramienta	Código abierto	Valoración
OmniPage	No	0*1.3 = 0
Abbyy FineReader	No	0*1.3 = 0
Presto! OCR Pro	No	0*1.3 = 0
Readiris Pro	No	0*1.3 = 0
Maestro Recognition server	No	0*1.3 = 0
Tesseract	**Si**	**1*1.3 = 1.3**

Tabla 15 - Análisis de propiedad de cada herramienta

La tabla anterior muestra el puntaje obtenido por cada herramienta OCR seleccionada dependiendo de si esta es un software de código abierto.

Elección de la Herramienta OCR

En esta sección se agrupan las valoraciones obtenidas en los puntos anteriores de forma de poder seleccionar aquella herramienta que posea mayor puntaje.

Herramienta	Puntuación Final
OmniPage	5.884
Abbyy FineReader	5.746
Presto! OCR Pro	5.254
Readiris Pro	5.298
Maestro Recognition server	5.044
Tesseract	**10.372**

Tabla 16 - Comparativa de herramientas de OCR

En la tabla 16 se puede observar claramente que la herramienta de OCR que obtuvo mayor puntaje fue Tesseract, por tanto, se puede afirmar que ésta posee prácticamente todas las características definidas como importantes a la hora de la elección.

Capítulo 5 - Diseño de la solución

En éste capítulo se describe la solución desarrollada para la digitalización de los expedientes de AJPROJUMI a partir de las herramientas elegidas en el capítulo 4.

5.1 Descripción de la arquitectura

La solución propuesta se basa en 4 etapas a través de las cuales a partir de los documentos escaneados en formato JPG se carga información relevante de los mismos en una base de datos. Luego una aplicación web consulta dicha base para realizar búsquedas de interés.

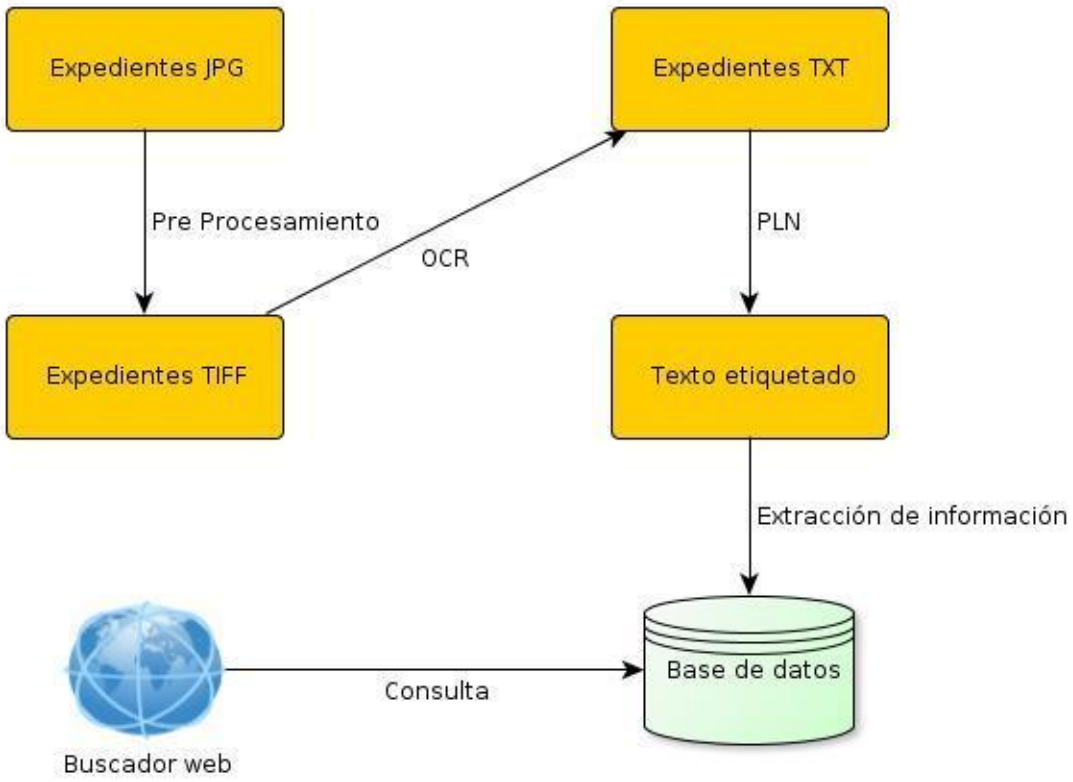


Figura 17 - Esquema de la solución

Los expedientes escaneados se almacenan bajo algún directorio del sistema de archivos, a su vez la estructura de ese directorio se replica en otro directorio del sistema de archivos al que se le aplica el pre procesamiento con la herramienta Scan Tailor. En este caso la extensión de los archivos es TIFF ya que es el formato de imagen mejor interpretado por el motor OCR utilizado.

Una vez generados los archivos TIFF se procede a ejecutar el motor OCR para cada una de las imágenes en la estructura, este proceso devuelve un archivo de texto (txt) para cada imagen procesada, los cuales tendrán el mismo nombre (excepto la extensión) que los archivos de imagen. En este caso el motor OCR utilizado es la herramienta Tesseract.
Luego se utiliza Freeling para realizar el procesamiento y etiquetado del texto generado, para esto se realizaron modificaciones de algunos módulos de esta herramienta de manera de poder asociar etiquetas a los nombres encontrados. Estas etiquetas se describen en la sección 5.4.2.
La última etapa de este proceso se encarga de poblar una base de datos con la información etiquetada generada por Freeling. Un proceso desarrollado en Python se encarga de recorrer cada archivo etiquetado y extraer la información de la base de datos para luego ser accedida a través de la aplicación web descrita en la sección 5.5.

5.2 Pre procesamiento

El objetivo de la etapa de pre procesamiento es mejorar la calidad de las imágenes de acuerdo a las características deseadas para obtener mejores resultados en la etapa de OCR.

Estas tareas se realizan con la herramienta Scan Tailor, seleccionada en la sección 4.1 Herramienta de pre procesamiento, y son invocadas mediante la función **binarize** del script de Python *procesar_imagen.py.*
Para cada imagen de cada expediente de la estructura se ejecuta la función binarize con el siguiente pseudocódigo.

```
binarize(infile, outdir):
      """
      Binariza la imagen "infile" y guarda la salida en "outdir".
      Realiza las correcciones necesarias, por ejemplo endereza la
imagen, quita ruido.
      """
      ejecutarScanTailor(infile)
      guardarImagen(outdir)
```

En el Anexo 3 se muestra el proceso de elección de los parámetros óptimos para las imágenes utilizadas para este proyecto, estos parámetros son:

Parámetro	Valor
Content-detection	cautios
Color-mode	black_and_white
Despeckle	aggressive
Layout	auto detect
Deskew	auto

Tabla 17 - Parámetros Scan Tailor

A modo ilustrativo, en la figura 18 se muestra el recorte de una página de un expediente en su versión original (JPG) junto con la imagen salida del proceso de pre procesamiento.

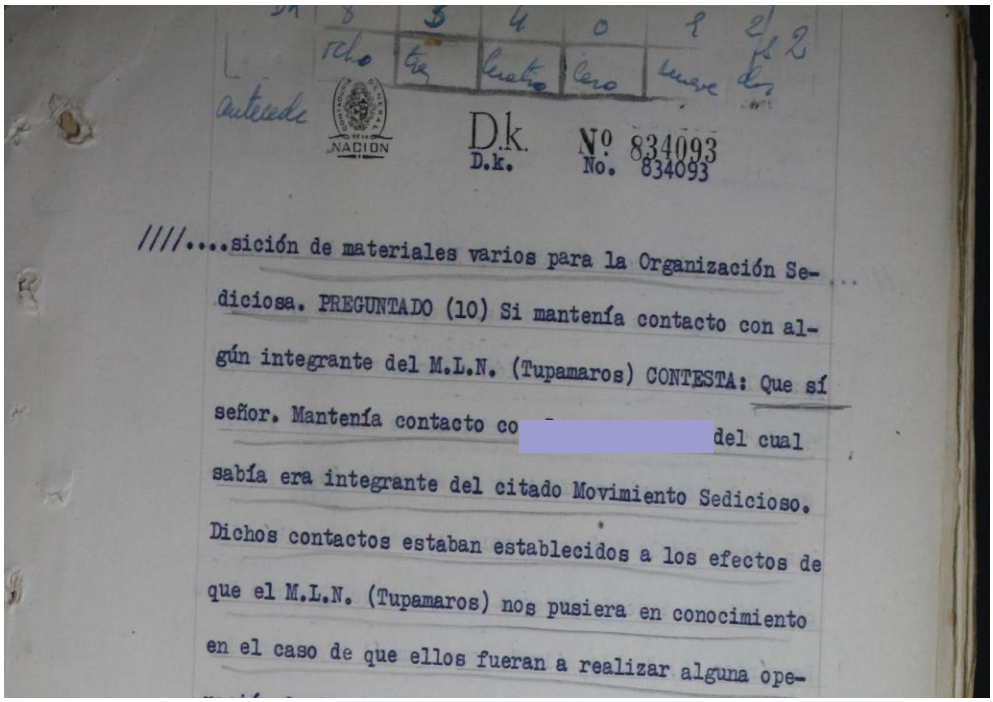
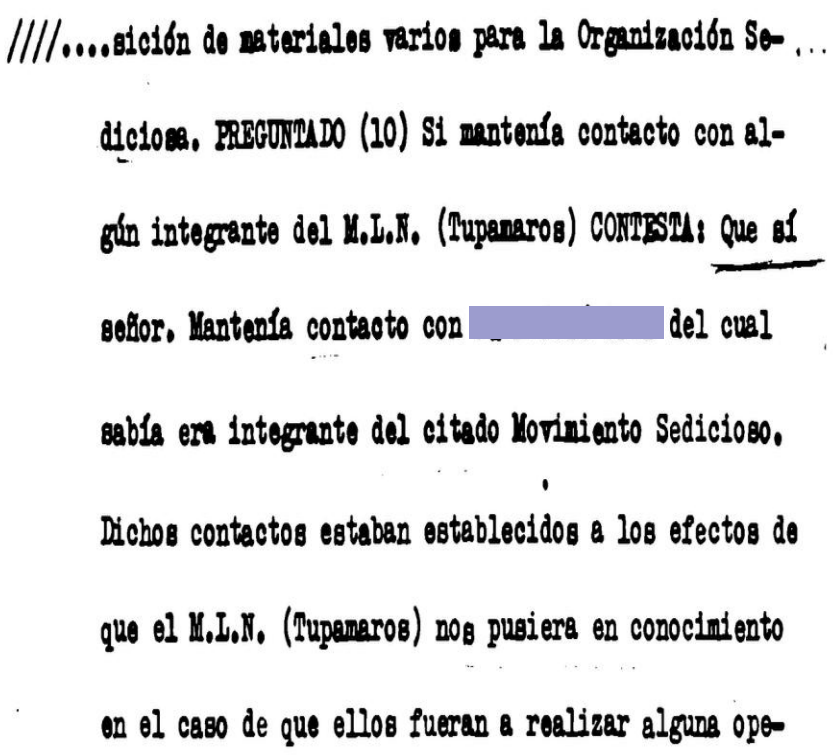


Figura 18 - Fragmento de la imagen 018_R del expediente A86-T3-N227-S-X397_87


////....sición de materiales varios para la Organización Se- ...

diciosa. PREGUNTADO (10) Si mantenía contacto con al-

gún integrante del M.L.N. (Tupamaros) CONTESTA: Que sí

señor. Mantenía contacto con [redacted] del cual

sabía era integrante del citado Movimiento Sedicioso.

Dichos contactos estaban establecidos a los efectos de

que el M.L.N. (Tupamaros) nos pusiera en conocimiento

en el caso de que ellos fueran a realizar alguna ope-

Figura 19 - Imagen original vs salida de Scan Tailor

Se puede observar que la selección de contenido recorta los márgenes de la imagen, los cuales tienen un efecto negativo en el proceso OCR. También se puede notar la eliminación del ruido, hay algunas marcas y subrayados en el archivo que también pueden afectar negativamente a los resultados del reconocimiento de caracteres.

En esta etapa se realiza una copia de la estructura de directorios donde se encuentran los archivos originales de manera de preservar la integridad de estos y no generar archivos extra en ese lugar. La salida del pre procesamiento generará archivos en esta nueva estructura, con el mismo nombre que los originales y con extensión TIFF.

5.3 Procesamiento OCR

Para esta etapa, el script *procesar_imagen.py* mencionado en la sección anterior provee un método llamado **ocr** donde se ejecuta Tesseract para cada una de las imágenes resultado del pre procesamiento.

```
ocr:
      para cada expediente
            para cada imagen
                  ejecutarTesseract(imagen)
                  guardarTexto
```

El archivo de texto correspondiente a la salida de la ejecución de Tesseract se guardará en el mismo directorio donde se encuentra la imagen (TIFF) procesada.

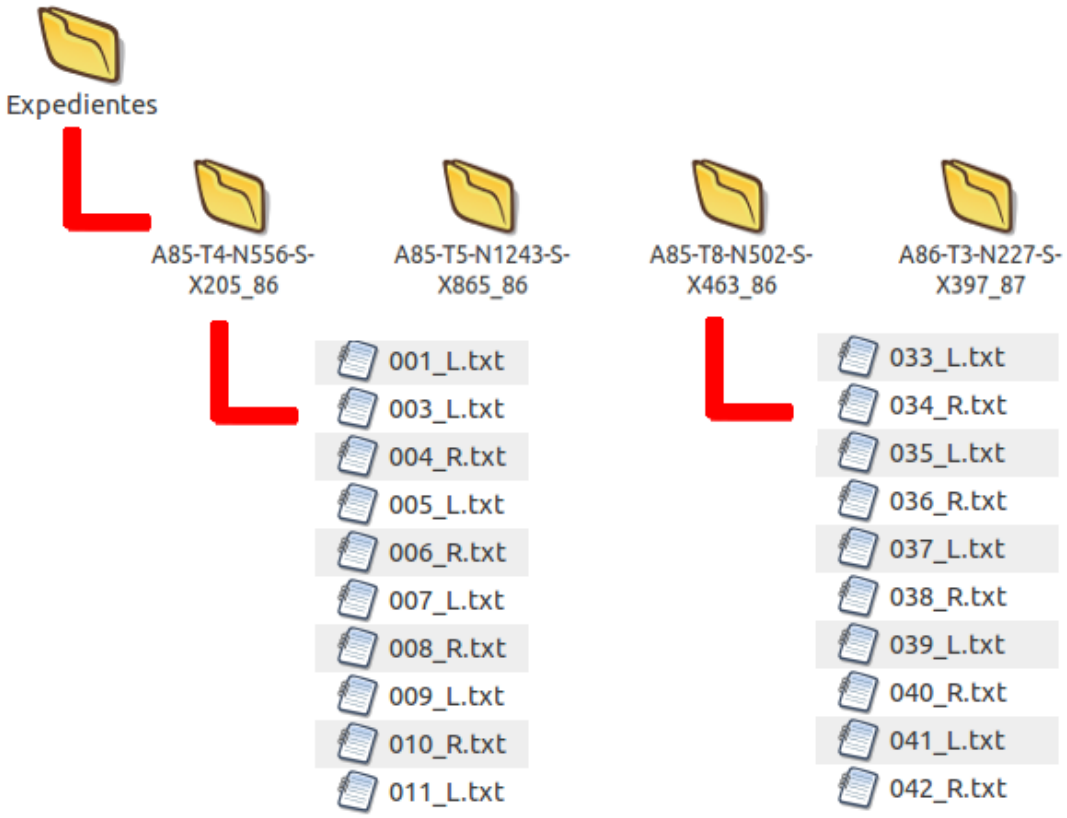


Figura 20 - Estructura de la salida del procesamiento de expedientes

La herramienta tesseract en su versión 3.0.2 recibe algunos parámetros para su ejecución, estos son el lenguaje donde naturalmente se elige el español (spa) y el parámetro psm (page segmentation mode), en el cual se le indica la forma de interpretar la imagen de entrada. Debido a la variedad de tipos de imágenes que se tienen, este parámetro indicará que la segmentación sea automática por lo que Tesseract intentará detectar el formato de la imagen.

5.4 Post Procesamiento

En las etapas anteriores se observó cómo a partir de las imágenes de los expedientes, se realizaron optimizaciones de las mismas utilizando técnicas de pre procesamiento de manera de poder aplicar de forma más eficaz las técnicas OCR, obteniendo así la representación en texto de todo el expediente.

El objetivo de esta etapa es analizar el texto resultado de aplicar OCR y extraer información que se considere relevante. La tarea principal se basa en obtener nombres de personas encontrados en los expedientes para luego hallar información relacionada a estas, como ser datos del expediente en donde se encuentra, si fue procesado, si se está hablando del encausado, de un juez militar, de un testigo, etc.
Además de la búsqueda de nombres, se realizaron búsquedas de fechas relevantes encontradas en los expedientes y de etiquetas particulares que resultaron de interés mostrar al usuario final.

Para poder realizar estas búsquedas se necesita tener la representación del texto a analizar en oraciones definidas y cada palabra de estas oraciones con su correspondiente etiqueta asignada. A continuación se mostrará el proceso realizado para cada uno de los archivos pertenecientes al expediente a analizar.

Para el procesamiento de los archivos de texto se utilizó Freeling, una de las herramientas más utilizadas para realizar análisis de lenguaje natural.
Freeling está compuesto por varios módulos de análisis, cada uno de los cuales brinda distintas funcionalidades de procesamiento de texto, algunos ejemplos de estos módulos son:

- Módulo Identificador de lenguaje.
- Módulo de Tokenización.
- Módulo para detección de sentencias (Splitter).
- Módulo para detección de números.
- Módulo para detección de puntuación.
- Módulo para reconocimiento de entidades con nombre (named entities).
- Módulo para clasificación de entidades con nombre.

En el Anexo 4 se detallan las características principales de dicha herramienta describiendo cada uno de los módulos utilizados.

5.4.1 Implementación del analizador (“ajprojumi analyzer”)

Para la implementación del analizador “ajprojumi analyzer” se utilizaron varios de los módulos de Freeling antes mencionados. A continuación se enumeran dichos módulos y se describen los detalles de la implementación del analizador:

- Tokenizer
- Splitter
- Analizador morfológico
- Tagger
- Reconocimiento de multipalabras
- Reconocimiento de entidades con nombre
- Clasificación de entidades con nombre.

El programa ajprojumiAnalyzer recibe como parámetro dos archivos, el archivo a procesar y el archivo en donde se desea guardar el resultado del proceso.

Ejemplo de ejecución:

```
$./ajprojumiAnalyzer archivoEntrada.txt archivoSalida.txt
```

A partir del archivo de entrada se procede a realizar la siguiente secuencia de pasos:

1 Ejecución de función tokenize del módulo Tokenizer (ver Anexo 4) para obtener la lista de palabras del texto.
2 Ejecución de función split del módulo Splitter (ver Anexo 4) para obtener lista de sentencias de texto.
3 Ejecución de función analyze del módulo Analizador morfológico que no realiza ningún procesamiento en particular sino que se utiliza para instanciar a otros submódulos utilizados(ver Anexo 4)
4 Ejecución de función analyze del módulo Tagger para etiquetar todas las palabras simples del texto (ver Anexo 4)
5 Ejecución de función analyze del módulo Reconocimiento de multipalabras para etiquetar todas las multipalabras definidas del texto (ver Anexo 4)

Luego de esta serie de pasos se tiene la lista de oraciones con cada palabra etiquetada, a partir de aquí se procede a realizar las búsquedas de nombres de personas, de fechas y de etiquetas particulares, los cuales se describen en las próximas tres secciones. La idea de estas búsquedas es extraer todas las oraciones que posean información relevante y escribirlas en un archivo para su posterior procesamiento.

5.4.2 Búsqueda de nombres de personas

Para realizar la búsqueda de nombres de personas se realizó previamente un estudio de la estructura de los expedientes (Ver Anexo 7), en él se identificaron palabras claves necesarias para recabar la información requerida.
A continuación se presenta una lista de las palabras detectadas que, en general, están asociadas a un nombre de persona:

- Juez militar
- Fiscal
- Encausado/s
- Ciudadano/s
- Testigo/s
- Procesado/s
- Persona
- Recluso

A cada una de estas palabras claves se le asigna una etiqueta especial a los efectos de poder ser interpretadas en futuras etapas. A modo de ejemplo, a “Juez Militar” se le asigna la etiqueta “JUEZMILITAR”, de esta forma cada palabra clave tendrá una única etiqueta relacionada.
Para lograr la asignación de etiquetas especiales se realizaron modificaciones sobre el diccionario provisto por Freeling, el cual contiene palabras simples, y sobre el archivo de configuración perteneciente al módulo de “Reconocimiento de multipalabras” que contiene las palabras claves compuestas. (Ver Anexo 5 – sección Freeling).

El archivo pasado como parámetro al constructor del módulo “Reconocimiento de multipalabras” contiene la lista de palabras compuestas a ser reconocidas por el analizador. Dicho archivo contiene una multipalabra por línea con el siguiente formato:

```
form lema1 pos1 lema2 pos2 … [ A | I ]
```

Cualquier número de pares lema-etiqueta puede ser asignado a la multipalabra, el módulo tagger seleccionará el más apropiado.
El último campo especifica si la multipalabra es ambigua o no (puede ser una multipalabra o no dependiendo del contexto). Un ejemplo del archivo de configuración es:

```
juez_militar_de_instruccion juez_militar_de_instruccion JUEZMILITAR
```

La lista completa de multipalabras agregadas se encuentra en el Anexo 5 - sección Freeling.

Luego de realizar estos cambios sobre los módulos se podrán identificar fácilmente las palabras que están relacionadas a nombres de personas. En la implementación del analizador al encontrar una de estas palabras en la sentencia, se procede a ejecutar los módulos de reconocimiento y clasificación de entidades con nombre solo sobre dicha sentencia, la cual es guardada en el archivo de salida tal como se muestra en la sección 5.4.5.

5.4.3 Búsqueda de fechas

Al igual que en la búsqueda de nombres se lograron identificar palabras clave relacionadas a fechas que resultaron de interés para su extracción. Estas son:

- Fue preso
- Detención
- Liberado
- Fecha de detención
- Fecha del hecho
- Fecha del procesamiento

La búsqueda de estas palabras se realiza de la misma manera que para las entidades con nombre, se procesa línea a línea el texto y si se encuentra una de estas palabras clave se procede a ejecutar el módulo de reconocimiento de fechas para la sentencia actual.
Dicha sentencia al contener una de las palabras clave se guarda en el archivo de salida para su posterior procesamiento.
Las fechas son etiquetadas por el módulo de reconocimiento de fechas con la marca “W”.

5.4.4 Búsqueda de etiquetas particulares

En el análisis de la estructura de los expedientes se identificaron palabras que, aunque no están directamente ligadas a un nombre, son claves ya que en el entorno de esta se puede obtener información relevante.

La lista de etiquetas particulares es:

- Datos filiatorios
- Datos patronímicos
- Datos dactiloscópicos
- Se resuelve
- Fallo

- Autos caratulados
- Caratulado
- Acta
- Sumario
- Sumario Instruido

Las secciones donde se encuentran dichas palabras poseen ciertas peculiaridades que hacen difícil extraer información. Por ejemplo, se observó que en todos los expedientes en la sección donde se encuentra la palabra clave “Se resuelve” el resto del contenido se encuentra en letra manuscrita, con lo cual OCR no retorna ningún texto legible.
Sin embargo se consideró interesante identificar las secciones con dichas palabras para que el usuario tenga acceso directo a información relevante como puede ser el fallo de un caso, simplificando así la búsqueda de dicha información.

5.4.5 Pseudocódigo de "ajprojumi analyzer"

```
//Inicialización de módulos de Freeling

list<word> listaPalabras = tokenize(texto);
list<sentence> listaSentencias = split(listaPalabras);

AnalizadorMorfologico->analyze(listaSentencias);
Tagger->analyze(listaSentencias);
ReconocimientoMultipalabras->analyze(listaSentencias);

Para cada sentencia "s" de lista de sentencias
    Para cada palabra "w" de "s"
        Si w.tag in [JUEZMILITAR, FISCAL, ENCAUSADO, ENCAUSADOS,
CIUDADANO, CIUDADANOS, TESTIGO, TESTIGOS, PROCESADO, PROCESADOS, PERSONA,
RECLUSO]
            RealizarBusquedaDeNE = true
        Sino, Si w.tag in [FUEPRESO, DETENCION, LIBERADO, FECHADETENCION,
FECHAHECHO, FECHAPROCESAMIENTO, LIBERTAD]
            RealizarBusquedaDeFecha = true
        Sino, Si w.tag in [DATOSFILIATORIOS, SERESUELVE,
DATOSPATRONIMICOS, DATOSDACTILOSCOPICOS, FALLO, AUTOSCARATULADOS,
CARATULADO, ACTA, SUMARIO, SUAMRIOINSTRUIDO]
            AgregarSentenciaASalida = true;
    Fin Para

    Si RealizarBusquedaDeNE
        ReconocimientoDeNE->analyze(s);
        ClasificacionDeNE->analyze(s);
        ImprimirSentenciaEnArchivo(s);
    Sino, Si RealizarBusquedaDeFecha
        ReconocimientoDeFechas->analyze(s);
        ImprimirSentenciaEnArchivo(s);
    Sino, Si AgregarSentenciaASalida
        ImprimirSentenciaEnArchivo(s);
Fin Para
```

Como se muestra en el pseudocódigo luego de que las sentencias son etiquetadas por los módulos "Tagger" y "Reconocimiento de multipalabras" se procede a recorrer cada una de

ellas para decidir si alguna de las palabras de la sentencia contiene alguno de los tags definidos.
En caso que se encuentre alguna de las etiquetas relacionadas a nombres de personas (RealizarBusquedaDeNE = true), se realizará una posterior búsqueda y clasificación de entidades dentro de la oración en la que se encontró dicha etiqueta. Para esto se utilizan los módulos NER (Named Entity Recognition) y NEC (Named Entity Clasification) de Freeling.
En caso que se encuentre alguna de las etiquetas relacionadas a fechas de interés (RealizarBusquedaDeFecha = true), se ejecutará la función analyze del módulo de reconocimiento de fechas para posteriormente extraer la fecha en cuestión.
Por último para el caso de las etiquetas especiales no es necesaria la ejecución de ningún módulo en especial ya que solo es necesario extraer la oración en donde se encuentra dicha etiqueta.

Si la sentencia que se está procesando cumple uno de los 3 casos recién mencionados esta se escribe en un archivo de salida para su posterior procesamiento en la aplicación web.
Dentro del archivo de salida cada línea contiene la información de una palabra del texto, mientras que la línea en blanco denota la separación entre sentencias. A continuación se muestra dicho formato:

```
<WORD form=”palabra1Sentencia1” lemma=”lema1Sentencia1” pos=”tag1”>
<WORD form=”palabra2Sentencia1” lemma=”lema2Sentencia1” pos=”tag2”>
<WORD form=”palabra3Sentencia1” lemma=”lema3Sentencia1” pos=”tag3”>

<WORD form=”palabra1Sentencia2” lemma=”lemma1Sentence2” pos=”tag4”>
<WORD form=”palabra2Sentencia2” lemma=”lemma2Sentence2” pos=”tag5”>
<WORD form=”palabra3Sentencia2” lemma=”lemma3Sentence2” pos=”tag6”>
<WORD form=”palabra4Sentencia2” lemma=”lemma4Sentence2” pos=”tag7”>

...
...
...
```

La búsqueda de estas palabras claves es un proceso sencillo ya que al tener todas las palabras etiquetadas, solo se necesita recorrer línea a línea el archivo y seleccionar las oraciones donde exista al menos una de estas palabras especiales.

5.4.6 Ejemplo de archivo de salida

La figura 21 muestra las oraciones que debería retornar el analizador para un determinado texto de entrada.

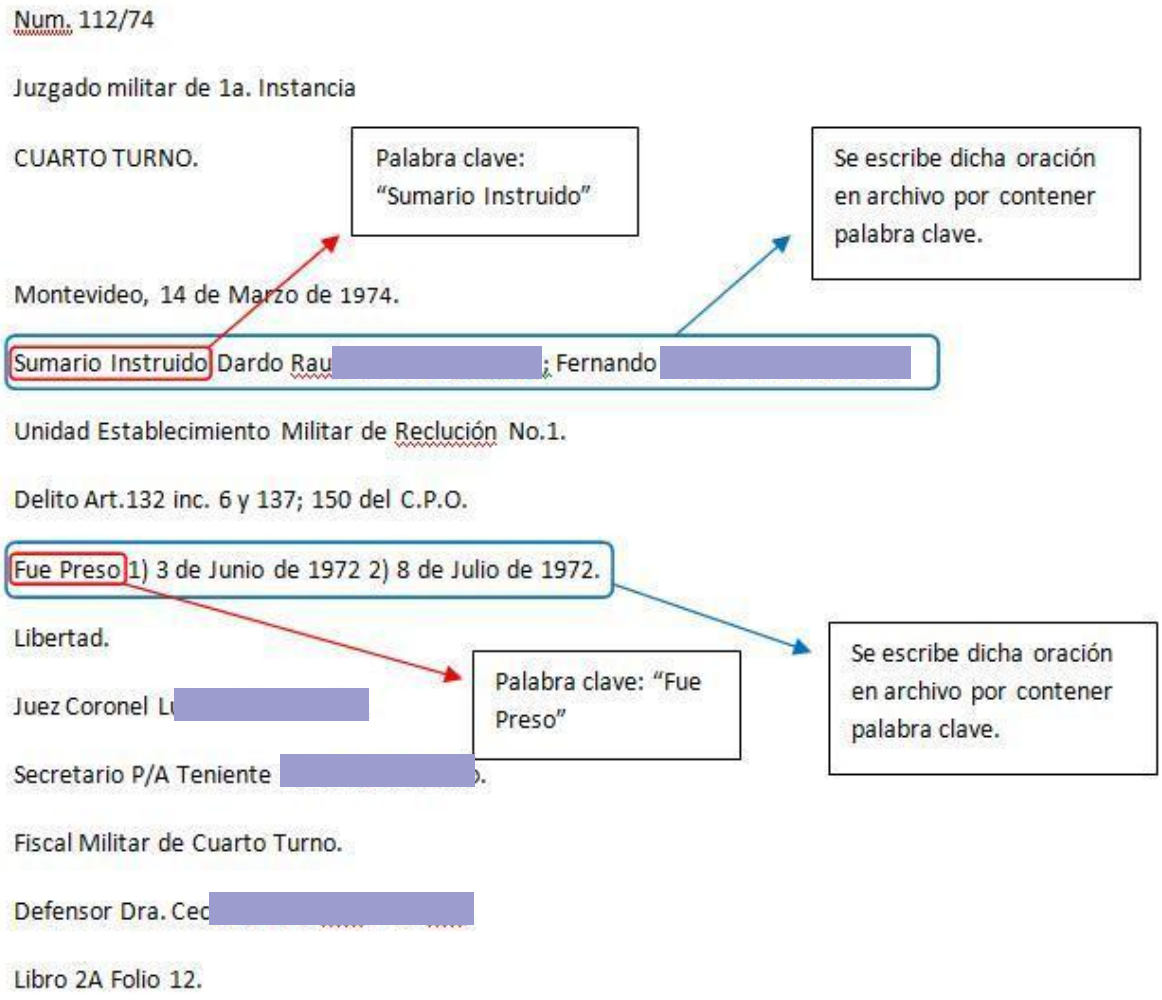


Figura 21 - Ejemplo de extracción de oraciones

El archivo de salida donde se etiquetan las oraciones relevantes para este ejemplo es:

```
<WORD form="Sumario_instruido" lemma="sumario_instruido" pos="SUMARIO">
<WORD form="Dardo_Raul_FE[redacted]A" lemma="dardo_raul_fe[redacted]" pos="NP00SP0">
<WORD form=";" lemma=";" pos="Fx">
<WORD form="Fernando_G[redacted]EIS" lemma="fernando_g[redacted]" pos="NP00SP0">
<WORD form="." lemma="." pos="Fp">

<WORD form="Fue_Preso" lemma="fue_preso" pos="FUEPRESO">
<WORD form="1" lemma="1" pos="Z">
<WORD form=")" lemma=")" pos="Fpt">
<WORD form="3_de_Junio_de_1972" lemma="[??:3/6/1972:??.??:??]" pos="W">
<WORD form="2" lemma="2" pos="Z">
<WORD form=")" lemma=")" pos="Fpt">
<WORD form="8_de_Julio_de_1972" lemma="[??:8/7/1972:??.??:??]" pos="W">
<WORD form="." lemma="." pos="Fp">
```

5.5 Buscador Web

La digitalización de imágenes es un proceso complejo que involucra varias etapas tales como la corrección previa, la aplicación de OCR y el post procesamiento utilizando técnicas de procesamiento del lenguaje natural (PLN). La salida de este proceso son archivos de texto, uno por cada archivo de imagen, con la información correspondiente al etiquetado de cada palabra reconocida.
Para poder hacer un verdadero uso de los datos generados es necesario crear una estructura que los maneje. Por tal motivo se crea la aplicación “ajprojumi web” que permite realizar búsquedas en la información obtenida durante los procesos previos. Esta aplicación se realizó utilizando el framework Python Django [Ref.:Django].

El dato más importante que se puede obtener de los archivos son los nombres de personas, sin este dato es difícil obtener información interesante. A partir del nombre se puede ampliar la información según el contexto, si la persona fue presa o procesada, indagada como testigo, si es juez, etc.

La siguiente figura muestra el modelo de la aplicación con las relaciones correspondientes:

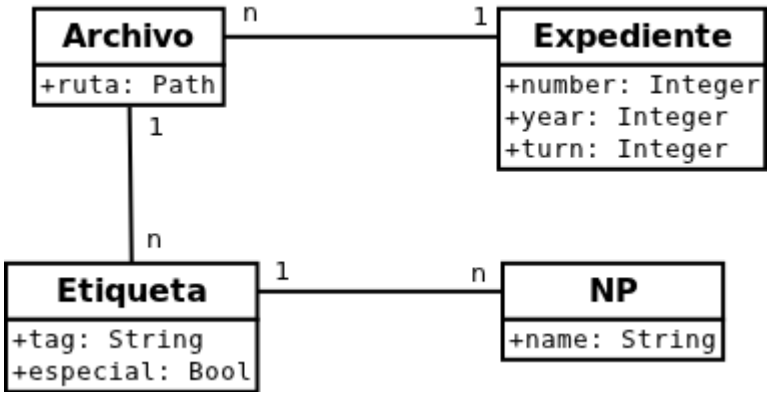


Figura 22 - Modelo de datos

El framework Django transforma este modelo en una estructura de base de datos utilizando el motor definido en la configuración del mismo, por lo que es transparente para el desarrollador. Para el prototipo se utilizó sqlite3.
El script *procesar_imagen.py* realiza una copia de la estructura de almacenamiento previamente definida para las imágenes, dentro de la misma se colocan los archivos de texto fruto de la salida de la aplicación de OCR. De los nombres de los directorios se obtiene la información de la clase Expediente, mientras que en cada archivo del mismo se encuentran los nombres de personas que se buscan. Un NP puede aparecer en más de un archivo.
Esta estructura tiene en su raíz un directorio por cada expediente escaneado, del cual a partir de su nombre se obtienen los datos del año, turno y número de expediente tal como se muestra en la siguiente imagen.

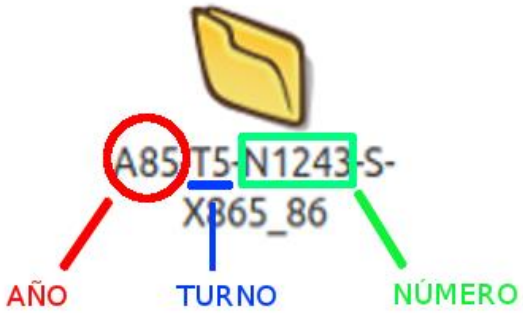


Figura 23 - Mapeo de datos de expedientes

Dentro de cada uno de estos directorios se guarda el resultado del OCR para cada expediente, el cual se utiliza para analizar y poblar la base de datos de la aplicación ajprojumi web.

5.5.1 Carga de datos

Mediante la página de carga se dan de alta todos los datos encontrados en los expedientes. En el formulario se ingresa la ruta al directorio raíz donde se encuentran todos los expedientes en modo texto, esta ruta es local al servidor donde se aloja la aplicación web.

Figura 24 - Carga de expedientes

Cuando el usuario hace clic en el botón “Procesar” se dispara la etapa de post procesamiento, para cada expediente en el directorio pasado como parámetro se ejecuta el “ajprojumi analyzer” seguido de algunas funciones que interpretan la salida de este analizador. El pseudo código del post procesamiento es el siguiente:

```
Para cada expediente en el directorio
      Obtener año, turno y número
      Para cada archivo del expediente
            Ejecutar ajprojumi analyzer
            Extraer información
```

Mientras que la extracción de la información se realiza de la siguiente manera:

```
Para cada oración
      Inicializar variables
      Para cada palabra
            Si etiqueta es “especial”
                  Crear Etiqueta(especial = True)
            Si etiqueta es “trascendente”
                  Crear Etiqueta()
            Si etiqueta es “NP00SP0" o "NP00V00”
                  Crear NP (Etiqueta)
```

La función de post procesamiento categoriza las etiquetas encontradas en tres grupos: **nombres, roles** y **especiales**. Esta categorización es necesaria ya que hay etiquetas que están relacionadas con personas y otras que referencian lugares del expediente donde se puede encontrar información relevante para quien realiza las búsquedas, como puede ser referente a sentencia, fallo o sumario entre otros. La lista completa de las etiquetas se referencia en la sección 5.4.4 Búsqueda de etiquetas particulares.

Las etiquetas “procesado”, “encausado”, “ciudadano” o “testigo” pueden referenciar tanto a una sola persona como a varias, por lo que al detectar alguna de éstas en plural el algoritmo seguirá buscando nombres y asociándolos a la etiqueta correspondiente hasta tanto no se encuentre una etiqueta o se llegue al final de la oración.

El resto de las etiquetas: "juez militar", "fiscal", "fue preso", "persona" y "recluso" referencian sólo a una persona por lo que el algoritmo luego de encontrar una de estas busca por un nombre para establecer la relación correspondiente.

5.5.2 Búsqueda

La idea del buscador es obtener información relacionada con un nombre dado, ya sean los expedientes en los que aparece, la causa en la cual esa persona está implicada y si fue procesada o no. La diferencia principal con lo que puede ser otro buscador de nombres son los datos de entrada, ya que provienen de mecanismos totalmente automáticos.
Los datos utilizados por el buscador son obtenidos mediante procesos automáticos para la digitalización de imágenes y los resultados obtenidos en su mayoría no son exactos, la calidad de las imágenes no es la mejor y cualquier tipo de ruido en un expediente puede causar que el texto resultado difiera del dato original. Estas causas generan la necesidad de encontrar un método más efectivo para los casos en que la diferencia entre el dato original y el resultado sea leve.

Para calcular qué tan parecidas son dos palabras se utiliza la *distancia de Levenshtein* [Ref.:Lev] que se define como la cantidad de operaciones necesarias para transformar una palabra en otra. Por operación se entiende una inserción, una eliminación o la sustitución de un carácter por otro.
El criterio utilizado establece que dos palabras (p1 y p2) son parecidas si cumplen:

- Si tamaño de p1 es menor que 3, p1 es parecido a p2 si distancia (p1, p2) = 0.
- Si tamaño de p1 entre 3 y 5, p1 es parecido a p2 si distancia (p1, p2) <= 1.
- Si tamaño de p1 entre 5 y 7, p1 es parecido a p2 si distancia (p1, p2) <= 2.
- Si tamaño de p1 es mayor que 7, p1 es parecido a p2 si distancia (p1, p2) <= 3.

Se toma este criterio para encontrar un equilibrio entre la inclusión de la mayor cantidad posible de palabras candidatas pero tratando de no incluir demasiados falsos positivos. Establecer margen de error en palabras de 3 letras o menos implicaría incluir muchos datos erróneos.
Este método si bien no corrige palabras debido a que no se tiene certeza cuál es la correcta, busca aproximaciones en la base de datos del nombre que se está buscando así como el expediente al cual pertenece.

Los nombres de personas están formados al menos por un nombre y un apellido, también pueden tener nombre y/o apellido compuesto. El usuario puede querer buscar solamente por el apellido por ejemplo y la consulta debería devolver las coincidencias. En este sentido no alcanza sólo con calcular la distancia de las palabras sino hay que establecer criterios sobre cada palabra de los nombres de personas registrados en la base de datos.
El buscador hará las búsquedas particionando el nombre buscado y los nombres registrados en la base de datos, en palabras individuales, las cuales se comparan una a una devolviendo el nombre completo si alguna de estas palabras coincide con las registradas.
Si bien es probable que este método devuelva nombres que no se buscan, también asegura que encuentre nombres que deberían aparecer pero debido al proceso de digitalización contienen pequeños errores.

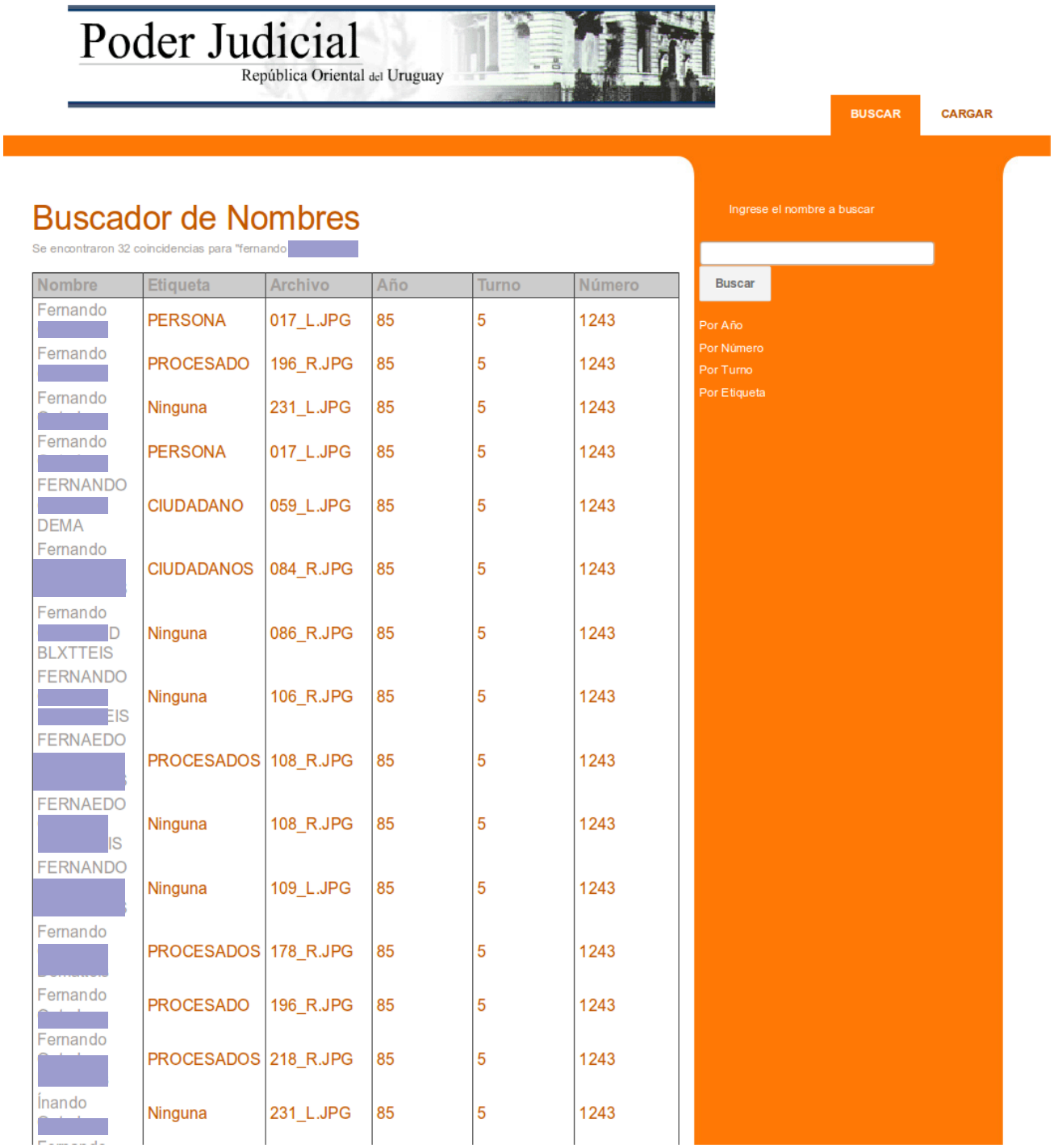


Figura 25 - Buscador de nombres

Para cada nombre encontrado el buscador devolverá la etiqueta relacionada, el archivo en el cual aparece y el expediente correspondiente dado por la combinación año-turno-número.

En la imagen se nota el texto “Ninguna” en el campo etiqueta, esto se da debido a que el analyzer en ocasiones no encuentra una etiqueta específica para un nombre, generalmente porque la salida del OCR devuelve la palabra distorsionada o directamente no la reconoce.
Si bien no se encuentra una etiqueta para algunos nombres, el sólo hecho de saber si un nombre aparece en un archivo es un indicio importante de información relevante en ese documento por lo que se decidió que existan nombre que no estén relacionados a alguna etiqueta.

5.5.3 Filtros

Para tener una mejor navegabilidad sobre los expedientes cargados en la base de datos se plantean algunos filtros a través de los cuales se puede llegar a un expediente deseado. Estos filtros son:

- Por año
- Por turno
- Por número
- Por etiqueta

Figura 26 - Filtros de búsqueda

En el caso de los filtros por año, número o turno el resultado final será uno o un conjunto de expedientes que cumplan con los criterios seleccionados mientras que el filtro por

etiquetas llevará a la lista de los nombres marcados con la misma. Estos filtros son globales por lo que abarcan todos los expedientes.

5.5.4 Validación

Para validar que la aplicación web muestra la información correcta, se realizó una comparación entre los resultados de una búsqueda en la aplicación web y la salida de Freeling para cuatro expedientes.

El procedimiento realizado fue el siguiente:

- Cargar expedientes desde la aplicación web.
- Realizar una búsqueda vacía la cual muestra todos los resultados guardados en la base de datos.
- Para cada nombre devuelto por la aplicación
    - Ejecutar ajprojumiAnalyzer para el archivo referenciado por el nombre
    - Confirmar que la etiqueta que muestra la aplicación web es la misma que se marca en el archivo de texto devuelto por ajprojumiAnalyzer

Al aplicar este procedimiento se verifica que todos los datos etiquetados por Freeling efectivamente se muestran en la aplicación web. Lo que no se puede demostrar con estas pruebas es que la aplicación web muestre todos los nombres que aparecen en los expedientes. De hecho eso no pasa ya que hay muchos casos donde Freeling no logra etiquetar información debido al estado de los expedientes.
Realizar pruebas con mayor cantidad de expedientes y obtener indicadores de precisión y recall quedan por fuera del alcance de este proyecto por lo que se recomienda incluir estos estudios en futuros trabajos relacionados.

5.6 Tiempos de procesamiento

Un aspecto a mejorar en esta solución son los tiempos de procesamiento. El trabajo de cada una de las etapas realizadas es costoso al punto de llegar a demorar en el entorno de tres horas para lograr procesar aproximadamente 600 páginas.
La medida de los tiempos de procesamiento se realizó en dos expedientes con 174 y 605 archivos de imagen respectivamente, los resultados obtenidos se pueden ver en la siguiente tabla:

Etapa \ Expediente	A85-T4-N556-S-X205_86 (174 imágenes)	A86-T3-N227-S-X397_87 (605 imágenes)
Pre procesamiento + OCR	12 min	138 min
Post Procesamiento	12 min	39 min
Total	24 min	177 min

Capítulo 6 - Conclusiones y Trabajo a Futuro

En este capítulo se exponen las conclusiones alcanzadas a lo largo del trabajo de forma de poder plasmar los logros obtenidos así como los problemas detectados y posible líneas a seguir en el futuro.
Debido a que cada vez es más importante que la información se encuentre disponible de forma rápida y que esta sea almacenada de modo que pueda conservarse incambiada es que el reconocimiento óptico de caracteres (OCR) ha jugado un papel muy importante estos últimos años. Pero la digitalización de documentos no es un proceso simple, más aún, cuando los documentos a ser digitalizados presentan un estado de deterioro importante o determinadas características que hacen que el proceso de digitalización falle o sea parcial.

6.1 Resultados obtenidos

Se realizó un estudio del estado del arte sobre las técnicas de procesamiento óptico de caracteres (OCR) así como un análisis de herramientas de pre procesamiento de imágenes y de extracción de información en texto digital. Para ello se instalaron, ejecutaron y compararon diversas aplicaciones de OCR eligiendo Tesseract como la indicada para este proyecto. Análogamente al análisis realizado para las herramientas de OCR, se llevó a cabo el estudio de software disponible de pre procesamiento de imágenes seleccionando a Scan Tailor como la herramienta a utilizar.
Por otra parte se logró la identificación de nombres e información relevante asociada a estos dentro de los archivos de texto generados utilizando la herramienta de procesamiento de lenguaje natural Freeling.
A partir del análisis realizado en las etapas anteriores se logró unificar estas en un prototipo que partiendo de los expedientes en formato de imagen logra poblar una base de datos de nombres de forma automática. A su vez, se implementó una aplicación web que permite realizar búsquedas en la base de datos generada por el proceso de digitalización.

6.2 Conclusiones

A lo largo del desarrollo de este proyecto se reforzó la idea de que la calidad de las imágenes a ser procesadas influye en gran medida sobre el resultado obtenido, por lo que tener buenos algoritmos de reconocimiento de caracteres no asegura un buen resultado. Detalles en las imágenes como anotaciones manuscritas, manchas, subrayados, renglones, sellos, líneas punteadas o superposición de caracteres provocan ruido que dificulta la recuperación de un alto porcentaje de palabras.

Se observó que por más que muchas de las aplicaciones de reconocimiento óptico de caracteres incluyen funcionalidades para el pre procesamiento de las imágenes, se obtienen mejores resultados cuando éste es realizado a través de herramientas externas. Sin embargo debido a la heterogeneidad de las imágenes en cuanto a su estado de deterioro y los diversos formatos de documentos, no hay una herramienta de pre procesamiento que sea mejor que el resto para todo el conjunto de imágenes pertenecientes a los expedientes del proyecto AJPROJUMI.

La dificultad para obtener buenos resultados en la etapa de OCR dificulto también la etapa de extracción de información, debido a que un carácter mal identificado dentro de una palabra provoca que ésta no sea etiquetada lo que se traduce en pérdida de información. Análogamente, caracteres mal identificados en los nombres producen información errónea dentro de la base de datos, por lo que se deben implementar funcionalidades que realicen búsquedas por aproximación y no por palabras completas.

Debido a los puntos de fallas detectados se torna necesaria la interacción humana de forma de poder supervisar y corregir eventuales problemas para obtener de esta forma resultados más confiables.

En lo que respecta a la librería Freeling, se logró crear un programa propio utilizando los módulos provistos por dicha librería, lo cual facilito en gran medida el procesamiento y búsqueda de información en los expedientes.

Si bien el estudio realizado fue hecho sobre expedientes provenientes de la Justicia Militar, los resultados obtenidos pueden ser fácilmente aplicables y extensibles a otras ramas de similares características, las cuales también manejen información en la manera que lo hace AJPROJUMI (expedientes, documentos, etc.).

6.3 Trabajo a futuro

Durante la realización de este proyecto se han logrado identificar líneas de trabajo a seguir en futuras investigaciones relacionadas al reconocimiento óptico de caracteres. En esta sección se describen posibles caminos a seguir para mejorar los resultados obtenidos.

Las pruebas de OCR realizadas sobre herramientas pagas fueron utilizando versiones de evaluación donde no se disponía de funcionalidades importantes para el marco de este proyecto, por ejemplo la ejecución en lote. Sería recomendable realizar las pruebas de algunas de estas herramientas en su versión completa para confirmar las características que se mencionan en sus sitios web.

Por otra parte la herramienta de OCR seleccionada fue Tesseract, la misma permite la posibilidad de entrenarla para un juego de caracteres específico lo cual mejora los

resultados para las imágenes que utilizan esa fuente. Si bien las imágenes de entrada no ayudan demasiado ya que en ocasiones la separación entre caracteres no es la adecuada o la intensidad entre dos letras varía, valdría la pena realizar entrenamiento para las fuentes identificadas en los expedientes.

Durante el estudio de los distintos software de reconocimiento óptico de caracteres se examinaron varias herramientas que permitían incorporar a su proceso de reconocimiento diccionarios relacionados a una temática específica, como por ejemplo la jurídica, de forma de poder especializar el reconocimiento, por lo que sería recomendable realizar pruebas sobre este tema para poder verificar si realmente el reconocimiento mejora.

Otro aspecto que no hay que descuidar en futuros trabajos es el análisis de nuevas herramientas de OCR así como la evolución de las aplicaciones analizadas en este proyecto.

Por otra parte en lo que refiere al pre procesamiento es deseable escanear las imágenes en un formato sin compresión (p.ej. TIFF) ya que al modificar imágenes en formatos comprimidos (p.ej. JPG) se pierde calidad en las mismas.

Debido a la heterogeneidad en el estado de las imágenes puede resultar positivo realizar clasificación de imágenes sobre los archivos de los expedientes de forma de agruparlas siguiendo algunos criterios basados en los formatos y características. De esa forma se puede aplicar una configuración de pre procesamiento particular a cada grupo.

En cuanto a la etapa de post procesamiento una línea a seguir es ampliar la información relacionada a los nombres que se obtuvo en la etapa de extracción de información.

Si la imagen de entrada a los procesos OCR contiene manchas, diferencias de iluminación, apuntes, etc. es muy probable que Freeling etiquete cadenas de texto sin sentido, la cual genera entradas en la aplicación web que no aportan información. En este sentido para el futuro se pueden mejorar los filtros en la búsqueda de manera de encontrar menos falsos positivos.

Otra línea pendiente en este trabajo es utilizar métodos de validación más complejos que la observación en un pequeño conjunto de expedientes. Es deseable poder obtener índices de precisión y recall así como realizar pruebas con usuarios reales para obtener información sobre satisfacción de los mismos.

La aplicación web desarrollada permite realizar búsquedas de información en un conjunto de expedientes cargados previamente; dicha aplicación actúa como una unidad independiente y no está directamente relacionada a lo ya realizado en el proyecto AJPROJUMI. Como trabajo a futuro se podría tener en cuenta la manera de integrar

ambas partes y por ejemplo poblar la base de datos ya existente en AJPROJUMI con los datos obtenidos en la aplicación desarrollada.

Un aspecto importante relativo a toda la implementación son los tiempos de procesamiento. Realizar trabajos sobre imágenes es un proceso costoso computacionalmente por lo que se debería investigar mecanismos e infraestructura para hacer estos procesos más veloces. Las etapas de OCR y análisis de texto también son costosas si se quieren utilizar en sistemas de tiempo real, por más que se logró mejorar el tiempo de ejecución de estas tareas utilizando múltiples hilos de ejecución, es un buen punto para evaluar alternativas.

Referencias

AJPROJUMI121031:
http://www.poderjudicial.gub.uy/historico-de-noticias/440-3000-expedientes-contiene-archivo-digital-del-periodo-de-facto.html (noviembre 2012).

Adaptive Threshold:
Hipermedia Image Processing Reference
Robert Fisher, Simon Perkins, Ashley Walker and Erik Wolfart
http://homepages.inf.ed.ac.uk/rbf/HIPR2/adpthrsh.htm (agosto 2012)

Chow and Kaneko:
"Boundary Detection of Radiographic Images by a Threshold Method".
C. K. Chow, T. Kaneko
IFIP Congress: 1530-1535, (1971)

Ventajas y desventajas OCR:
Sistemas de Reconocimiento Óptico de Caracteres
Joaquin Arlandis Navarro
Revista del instituto tecnológico de Informática, 2010

Herramientas OCR:
Artículo "Reconocimiento óptico de caracteres"http://es.wikipedia.org/wiki/Reconocimiento_%C3%B3ptico_de_caracteres (julio 2012)
Artículo "Los 10 Mejores Programas de Software para Escanear OCR"
http://www.tecnikeando.com/software/110411-los-10-mejores-programas-de-software-para-escanear-ocr.html (agosto 2012)

TesseractEngine:
An Overview of the Tesseract OCR Engine
Ray Smith
Google Inc, 2007

Tesseract:
Sitio web oficial de tesseract-ocr
http://code.google.com/p/tesseract-ocr/ (agosto 2012)

Django:
Sitio web oficial del framework de desarrollo web Django
www.djangoproject.com (agosto 2012)

IMPACT:

IMProving ACcess to Text
http://www.impact-project.eu (febrero 2012)

Aletheia:

Aletheia - An Advanced Document Layout and Text Ground-Truthing System for Production Environments
C. Clausner, S. Pletschacher and A. Antonacopoulos
2011 International Conference on Document Analysis and Recognition

GoogleBooks:

Proyecto Google Books
http://books.google.com/googlebooks/library.html (Agosto 2012)

ScanTailor:

Sitio web oficial de Scan Tailor
http://scantailor.sourceforge.net (Mayo 2012)

ImageMagick:

Sitio web oficial de Image Magick
http://www.imagemagick.org (Agosto 2012)
http://www.fmwconcepts.com/imagemagick/index.php (Mayo 2012)

Localtresh:

http://www.fmwconcepts.com/imagemagick/localthresh/index.php (Agosto 2012)
Efficient Implementation of Local Adaptive Thresholding Techniques Using Integral Images
Faisal Shafait, Daniel Keysers, Thomas M. Breuel
Department of Computer Science, Technical University of Kaiserslautern
Survey over image thresholding techniques and quantitative performance evaluation
Mehmet Sezgin, Bülent Sankur
Tübitak Marmara Research Center Information Technologies Research Institute Gebze, Kocaeli Turkey, Bogaziçi University Electric-Electronic Engineering Department Bebek, Istanbul Turkey

Omnipage:

Sitio web oficial de OmniPage
http://nuance.com (Marzo 2012)

Nuance:

Sitio web oficial de Nuance
http://nuance.com/for-individuals/by-product/omnipage/index.htm (Marzo 2012)

ABBYY:

Sitio web oficial de Abby FineReader
http://latam.abbyy.com (Marzo 2012)

NewSoft:

Sitio web oficial de New Soft
www.newsoftinc.com (Marzo 2012)

Readiris:

Sitio web oficial de Readiris
http://www.irislink.com/c3-2115-58/Readiris-14--OCR-Software--Scan--Convert---Manage-your-Documents-.aspx (Marzo 2012)

Iris:

Sitio web oficial de I.R.I.S.
http://www.irislink.com/c3-646-58/I-R-I-S----OCR-software-and-Document-Management-solutions.aspx

Maestro:

Sitio web oficial de Maestro Recognition Server
CIVISION Smarter Document Capture: www.cvisiontech.com

PRImA:

http://www.primaresearch.org (Agosto 2012)

Freeling:

http://nlp.lsi.upc.edu/freeling/

Ocrad:

http://www.gnu.org/software/ocrad/

Gutemberg:

Proyecto Gutemberg
http://www.gutenberg.org/

Recaptcha:

http://www.google.com/recaptcha

Tao:

Algorithms for Postprocessing OCR Results with Visual Inter- Word Constraints
Tao Hong and Jonathan J. Hull*
Center of Excellence for Document Analysis and Recognition (CEDAR) - State University of New York at Buffalo

Athento:

http://www.athento.com/ocr/

Índice de Figuras

Índice de Tablas

Anexo 1 - Tesseract: imágenes originales vs pre procesadas

El objetivo de este anexo es mostrar cómo el pre procesamiento de imágenes influye de manera positiva en la posterior etapa de procesamiento OCR, obteniendo en la mayoría de los casos una mayor tasa de acierto de los caracteres identificados. Dicho estudio se realizó sobre una selección representativa de imágenes pertenecientes a expedientes del proyecto AJPROJUMI.

Pasos realizados

Para mostrar el objetivo planteado se realizaron los siguientes pasos:

- Ejecutar cada una de las herramientas OCR sobre las imágenes de muestra elegidas.
- Ejecutar cada una de las herramientas OCR sobre las imágenes de muestra, aplicando previamente a estas ScanTailor.
- Comparar la cantidad de caracteres reconocidos en ambos casos.

Nota: Si bien la herramienta OCR escogida fue Tesseract se realizarán las pruebas con todas las herramientas OCR para mostrar que el pre procesamiento influye de manera positiva y por igual para todas las herramientas presentadas.

Ejemplo de imagen utilizada junto con su respectiva salida pre procesada con ScanTailor:

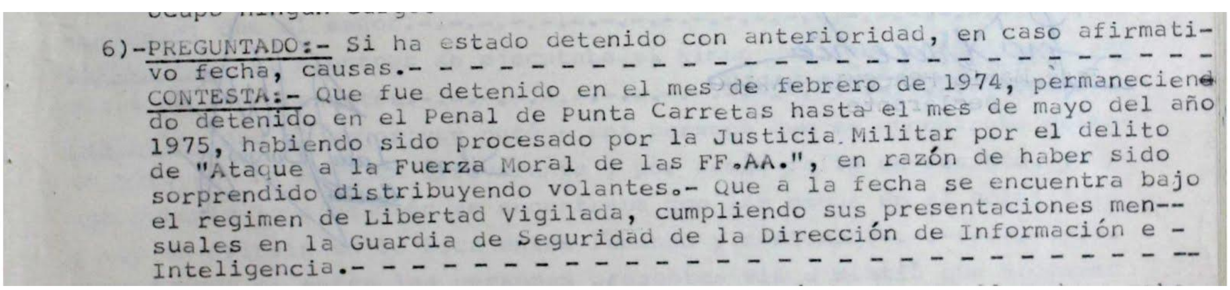
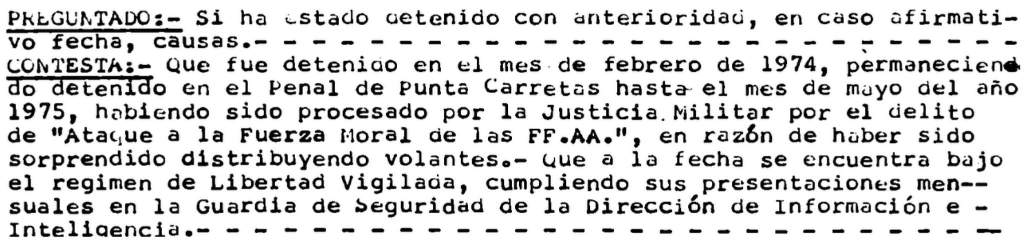

6)-PREGUNTADO:- Si ha estado detenido con anterioridad, en caso afirmati-
vo fecha, causas.- - - - - - - - - - -
CONTESTA:- Que fue detenido en el mes de febrero de 1974, permaneciendo detenido en el Penal de Punta Carretas hasta el mes de mayo del año 1975, habiendo sido procesado por la Justicia Militar por el delito de "Ataque a la Fuerza Moral de las FF.AA.", en razón de haber sido sorprendido distribuyendo volantes.- Que a la fecha se encuentra bajo el regimen de Libertad Vigilada, cumpliendo sus presentaciones men-- suales en la Guardia de Seguridad de la Dirección de Información e - Inteligencia.- - - - - - - - - - -

Figura 27 - Ejemplo de recorte de expediente sin pre procesamiento



PREGUNTADO:- Si ha estado detenido con anterioridad, en caso afirmati-
vo fecha, causas.- -
CONTESTA:- Que fue detenido en el mes de febrero de 1974, permaneciendo
do detenido en el Penal de Punta Carretas hasta el mes de mayo del año
1975, habiendo sido procesado por la Justicia Militar por el delito
de "Ataque a la Fuerza Moral de las FF.AA.", en razón de haber sido
sorprendido distribuyendo volantes.- Que a la fecha se encuentra bajo
el regimen de Libertad Vigilada, cumpliendo sus presentaciones men--
suales en la Guardia de Seguridad de la Dirección de Información e -
Inteligencia.- -

Figura 28 - Ejemplo de recorte de expediente con pre procesamiento aplicado

Resultados

Luego de correr cada una de las herramientas OCR para las imágenes de prueba se prosiguió a hacer un recuento de la cantidad de caracteres efectivamente reconocidos, los porcentajes obtenidos fueron:

Imagen	OmniPage	Presto! OCR	Abby FineReader	Readiris Pro	Tesseract	Maestro Recognition
Exp1.jpg	78.5%	44.0%	62.3%	00.0%	44.5%	90.0%
Exp2.jpg	96.6%	92.5%	91.7%	90.5%	83.8%	99.4%
Exp3.jpg	85.9%	67.4%	92.8%	19.1%	00.0%	00.0%
Exp4.jpg	97.9%	90.6%	90.6%	3.0%	11.6%	98.8%
Exp5.jpg	72.8%	76.3%	53.8%	45.7%	30.9%	75.4%
Exp6.jpg	77.0%	00.0%	77.6%	17.2%	00.0%	33.3%
Exp7.jpg	00.0%	54.5%	45.5%	0.00%	90.9%	00.0%
Exp8.jpg	97.4%	95.3%	97.9%	96.1%	95.7%	95.7%
Exp9.jpg	90.8%	56.8%	96.2%	18.2%	00.0%	84.9%
Exp10.jpg	96.7%	97.5%	96.2%	92.3%	92.0%	98.9%
% Promedio	**79.4%**	**67.5%**	**80.5%**	**38.2%**	**44.9%**	**67.6%**

Tabla 18 - Porcentaje reconocimiento sin pre procesamiento

El mismo procedimiento se realizó con las imágenes con pre procesamiento, los resultados obtenidos fueron:

Imagen	OmniPage	Presto! OCR	Abby FineReader	Readiris Pro	Tesseract	Maestro Recognition
exp1_1L.tif	72.3%	50.6%	43.4%	15.7%	33.7%	37.3%
exp1_2R.tif	73.9%	73.9%	78.3%	30.4%	60.9%	78.0%
exp2_2R.tif	98.6%	86.5%	98.6%	90.9%	97.6%	84.2%
exp4_1L.tif	95.1%	94.2 %	80.5%	3.6%	92.1%	82.7%
exp5_1L.tif	83.0%	66.9%	66.2%	50.8%	61.8%	83.2%
exp6_2R.tif	90.4%	69.2%	75.0%	51.9%	100%	92.3%
exp7_1L.tif	00.0%	36.4%	36.4%	00.0%	72.7 %	00.0%
exp8_2R.tif	95.9 %	94.1%	92.9%	88.8%	94.7%	97.0%
exp9_1L.tif	96.4%	96.4%	96.4%	60.1%	53.6%	97.1%
exp9_2R.tif	97.5%	91.6%	97.5%	81.5%	92.4%	94.9%
exp10_2R.tif	98.5%	97.0%	98.2%	90.4%	97.0%	95.6%
% Promedio	**81.96%**	**77.89%**	**78.49%**	**51.28%**	**77.86%**	**73.2%**

Tabla 19 - Porcentaje reconocimiento con pre procesamiento

Nota: Alguna de las imágenes se encuentran divididas en dos (p.ej. exp1_1L.tif y exp1_2R.tif) debido a que la herramienta Scan Tailor realiza en ocasiones divisiones en sub-imágenes con el contenido de la imagen original.

Comparando los porcentajes promedio de cada uno de los sets de pruebas se puede ver con claridad que el pre procesamiento de las imágenes aumenta la tasa de acierto en la posterior etapa de reconocimiento por OCR. De las seis herramientas estudiadas, cinco aumentaron el porcentaje de caracteres reconocidos y en particular la herramienta elegida (Tesseract) lo hizo en gran medida. Dado esto se decide aplicar pre procesamiento de imágenes a todos los expedientes analizados logrando aumentar en buena medida la efectividad del proceso de extracción de información.

Anexo 2 - Elección de cotas sobre parámetros de Localtresh

En este anexo se presenta un conjunto de pruebas llevadas a cabo a través de la herramienta de pre procesamiento Localtresh descrita en el punto **4.1.1** con el fin de poder hallar en aquellos parámetros que corresponda una cota superior e inferior. El objetivo es poder limitar el rango de valores que puedan tomar dichos parámetros, de forma de lograr resultados aceptables en la digitalización de las imágenes pertenecientes a expedientes del proyecto AJPROJUMI.

Este análisis se debió realizar ya que la herramienta Localtresh elegida como posible software de pre procesamiento a utilizar contenía varios parámetros con un amplio dominio de valores posibles a utilizar.

Localtresh (ImageMagick)

Localtresh [Ref.: **Localtresh**] es un script perteneciente a la suite de ImageMagick con el que se pueden realizar diversas operaciones para la mejora de una imagen. A modo de resumen se puede decir que este script binariza una imagen utilizando un enfoque de ventana móvil con umbral adaptativo donde para cada ventana la ubicación del pixel central se compara con alguna medida de promedio o una combinación del promedio y la desviación estándar o el promedio de la desviación estándar absoluta dentro de la ventana. Si el píxel central es más grande que esta medida por un valor de sesgo, entonces el píxel central se hace blanco, de lo contrario se hace negro.

Por otro lado este script recibe de forma opcional un conjunto de parámetros que se describirán a continuación:

- **Método (-m):** Especifica qué medida estadística se va a utilizar para el cálculo del umbral para cada ubicación de las ventanas.
  El método 1 compara el píxel central con la media de la ventana, si es mayor que el sesgo, el pixel central se hace blanco, de lo contrario negro.
  El método 2 compara el píxel central con la media de la ventana más los tiempos de polarización de la desviación estándar de la ventana, y si los pixeles del centro son más grandes, se hace blanco; de lo contrario negro.
  El método 3 compara el píxel central con la media de la ventana más los tiempos de polarización de la raíz cuadrada de la desviación absoluta promedio de la ventana, y si el pixel central es más grande, se hace blanco; de lo contrario negro.
  El método por defecto es el 1.

- **Radio (-r):** Especifica el radio de la ventana. El valor puede ser un real, pero debe ser mayor o igual a 3. El valor por defecto es 15. Para obtener resultados aceptables, el radio en general, debe ser mayor que la dimensión característica que será detectada por el umbral.

- **Sesgo (-b):** Es el parámetro de sesgo utilizado en cada uno de los dos métodos para determinar el umbral. Los valores más altos de sesgo tendrán el efecto de eliminar más ruido del resultado, pero un valor demasiado grande puede extraer parte de los objetos en primer plano que se detectan. Los valores de sesgo se expresan como (porcentaje) reales donde el sesgo >= 0. El valor por defecto es 20.

- **Infile:** Archivo de entrada al cual se le va a aplicar la transformación.

- **Outfile:** Archivo de salida resultado de la transformación.

Este punto ilustra de qué forma se obtuvieron configuraciones de parámetros que se ajustaban mejor al tipo de imágenes a ser procesadas. Para esto se tomó un subconjunto representativo de las pruebas realizadas.

La idea central consistió en analizar cada parámetro de entrada del script para obtener de esta forma una cota superior e inferior que contuviera al valor óptimo para ese parámetro. Para llevar a cabo lo descrito se definió como criterio fijar los parámetros que no estaban siendo estudiados con sus valores por defecto y variar el que estaba bajo análisis.

Las imágenes correspondientes al subconjunto de pruebas seleccionado fueron las siguientes:

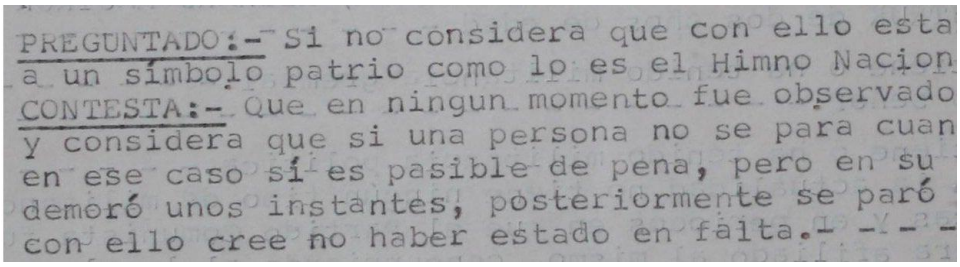

PREGUNTADO:- Si no considera que con ello esta
a un símbolo patrio como lo es el Himno Nacion
CONTESTA:- Que en ningun momento fue observado
y considera que si una persona no se para cuan
en ese caso sí es pasible de pena, pero en su
demoró unos instantes, posteriormente se paró
con ello cree no haber estado en falta.- - - -

Figura 29 - Fragmento correspondiente a la imagen 009_LC del expediente A86-T1-N333

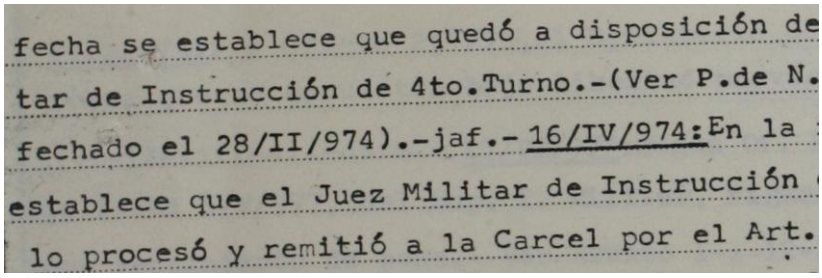

fecha se establece que quedó a disposición de
tar de Instrucción de 4to.Turno.-(Ver P.de N.
fechado el 28/II/974).-jaf.- 16/IV/974: En la
establece que el Juez Militar de Instrucción
lo procesó y remitió a la Carcel por el Art.

Figura 30 - Fragmento correspondiente a la imagen 016_RC del expediente A86-T1-N333

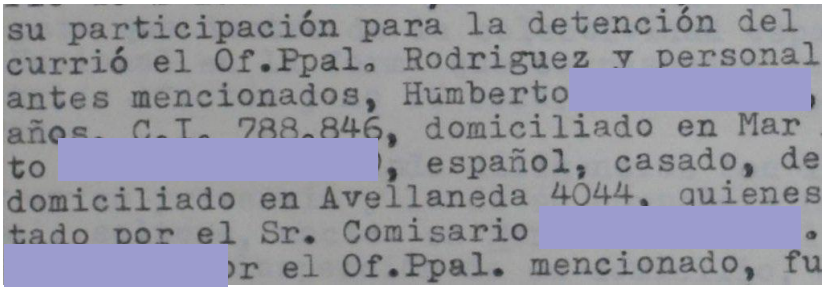

su participación para la detención del
currió el Of.Ppal. Rodriguez y personal
antes mencionados, Humberto
años, C.I. 788.846, domiciliado en Mar
to , español, casado, de
domiciliado en Avellaneda 4044, quienes
tado por el Sr. Comisario
r el Of.Ppal. mencionado, fu

Figura 31 - Fragmento correspondiente a la imagen 022_RC del expediente A86-T1-N333

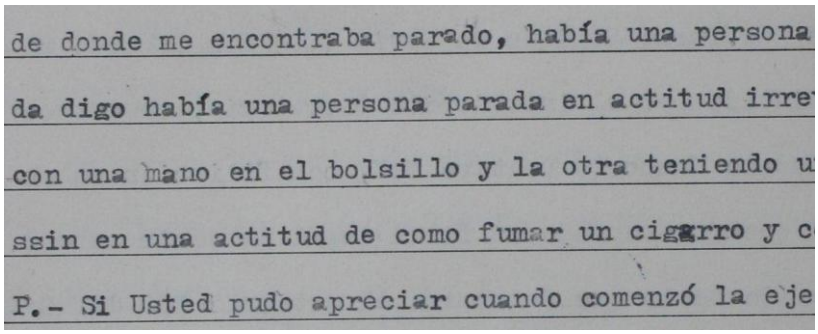

de donde me encontraba parado, había una persona
da digo había una persona parada en actitud irre
con una mano en el bolsillo y la otra teniendo u
ssin en una actitud de como fumar un cigarro y c
P.- Si Usted pudo apreciar cuando comenzó la eje

Figura 32 - Fragmento correspondiente a la imagen 039_L del expediente A86-T1-N333

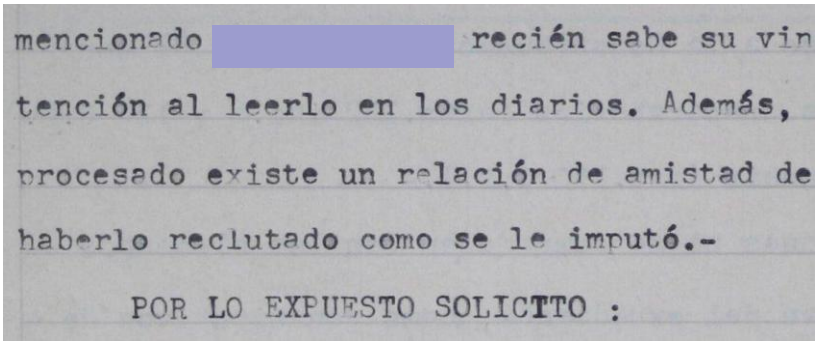

mencionado [redacted] recién sabe su vin
tención al leerlo en los diarios. Además,
procesado existe un relación de amistad de
haberlo reclutado como se le imputó.-
POR LO EXPUESTO SOLICTTO :

Figura 33 - Fragmento correspondiente a la imagen 215_L del expediente A85-T5-N1243-S-X865_86

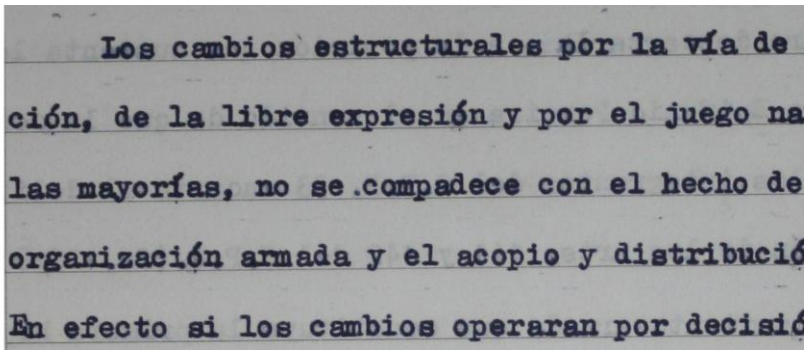

Los cambios estructurales por la vía de
ción, de la libre expresión y por el juego na
las mayorías, no se compadece con el hecho de
organización armada y el acopio y distribució
En efecto si los cambios operaran por decisió

Figura 34 - Fragmento correspondiente a la imagen 338_R del expediente A86-T3-N227-S397-X_87

Análisis del parámetro método

En este punto se muestra cómo varía la imagen original según el método (**-m**) del script que se haya seleccionado. Para un mejor análisis de los resultados, a cada imagen producto de la transformación realizada se le aplicó OCR de forma de poder comparar sobre qué imagen se realizaba una mejor extracción de datos.

A continuación se muestran las pruebas realizadas para el análisis de este parámetro.

Prueba 1

Esta prueba consiste en tomar el conjunto de imágenes seleccionadas y aplicarles la herramienta Localtresh fijando el método (-m) en uno. El resto de los parámetros se cargan con sus valores por defecto.

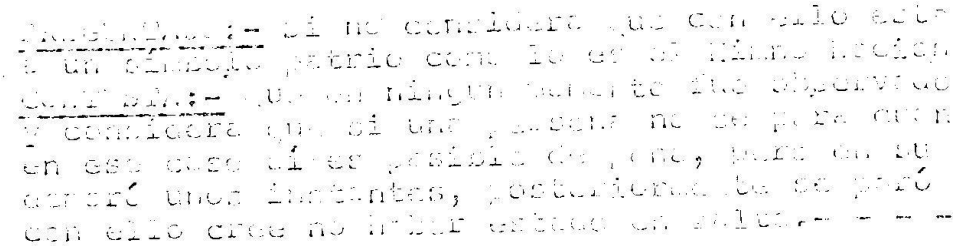


Figura 35 - Resultado de aplicar el método 1 a la fig. 29

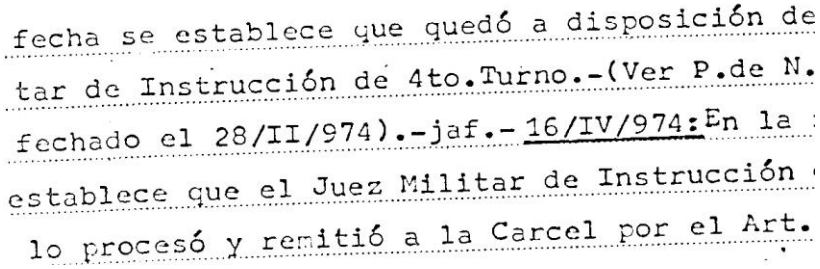


Figura 36 - Resultado de aplicar el método 1 a la fig. 30

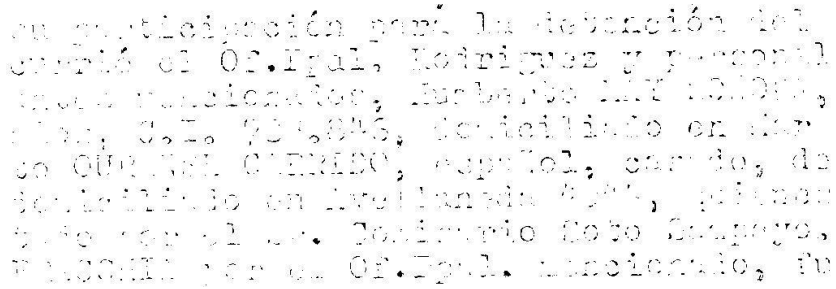


Figura 37 - Resultado de aplicar el método 1 a la fig. 31

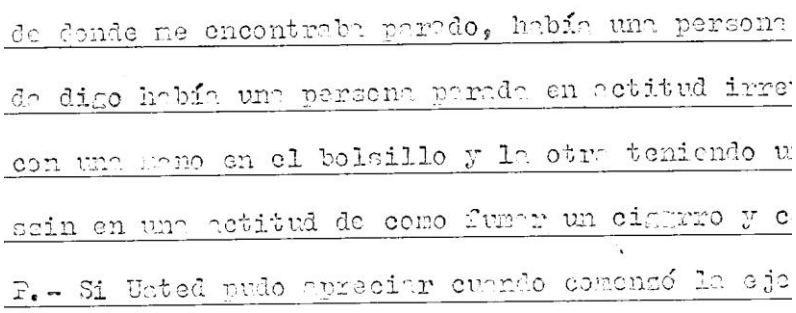


Figura 38 - Resultado de aplicar el método 1 a la fig. 32

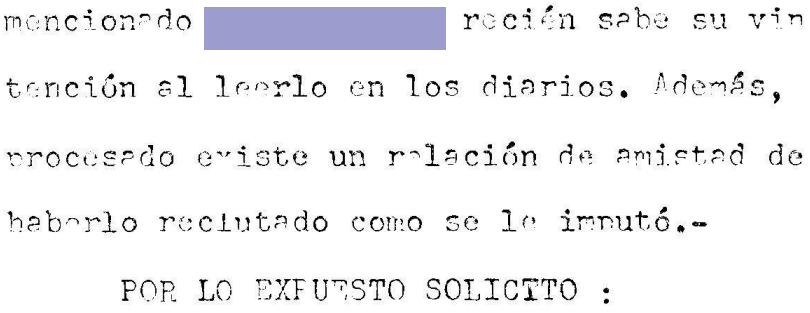


Figura 39 - Resultado de aplicar el método 1 a la fig. 33

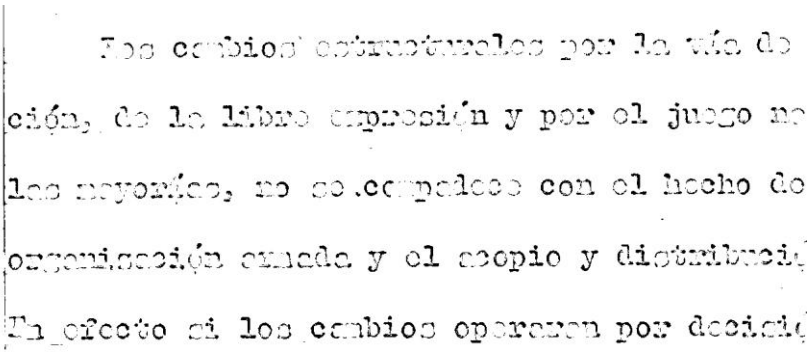


Figura 54 - Resultado de aplicar el método 1 a la fig. 34

Prueba 2

Análogamente a lo realizado en la Prueba 1, se ejecuta la herramienta Localtresh sobre las imágenes seleccionadas fijando el método (-m) en dos. El resto de los parámetros se cargan con sus valores por defecto.

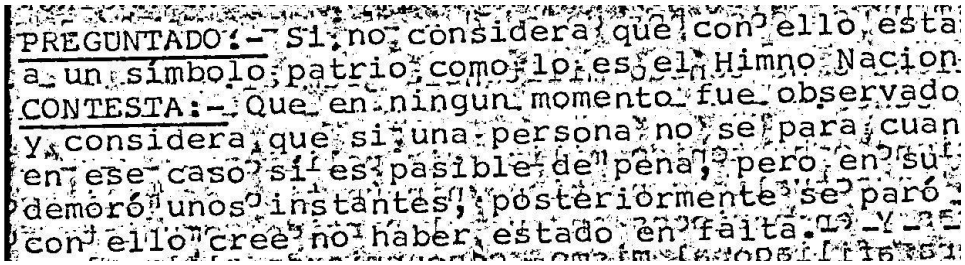

PREGUNTADO:- Si no considera que con ello esta
a un símbolo patrio como lo es el Himno Nacion
CONTESTA:- Que en ningun momento fue observado
y considera que si una persona no se para cuan
en ese caso si es pasible de pena, pero en su
demoró unos instantes, posteriormente se paró
con ello cree no haber estado en falta.- Y -

Figura 40 - Resultado de aplicar el método 2 a la fig. 29

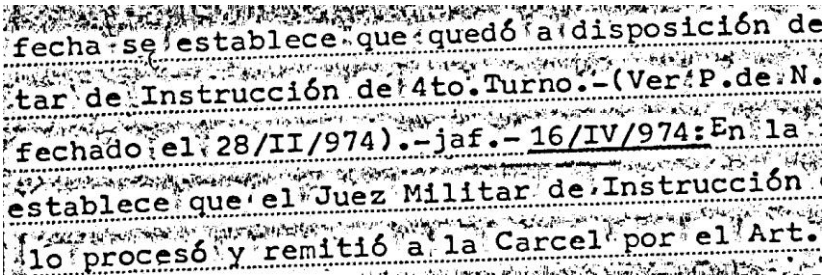

fecha se establece que quedó a disposición de
tar de Instrucción de 4to.Turno.-(Ver P.de N.
fechado el 28/II/974).-jaf.- 16/IV/974: En la
establece que el Juez Militar de Instrucción
lo procesó y remitió a la Carcel por el Art.

Figura 41 - Resultado de aplicar el método 2 a la fig. 30

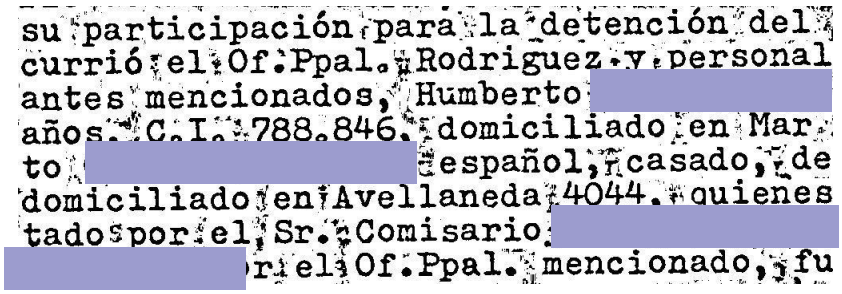

su participación para la detención del
currió el Of.Ppal. Rodriguez y personal
antes mencionados, Humberto
años. C.I. 788.846. domiciliado en Mar
to español, casado, de
domiciliado en Avellaneda 4044, quienes
tado por el Sr. Comisario
r el Of.Ppal. mencionado, fu

Figura 42 - Resultado de aplicar el método 2 a la fig. 31

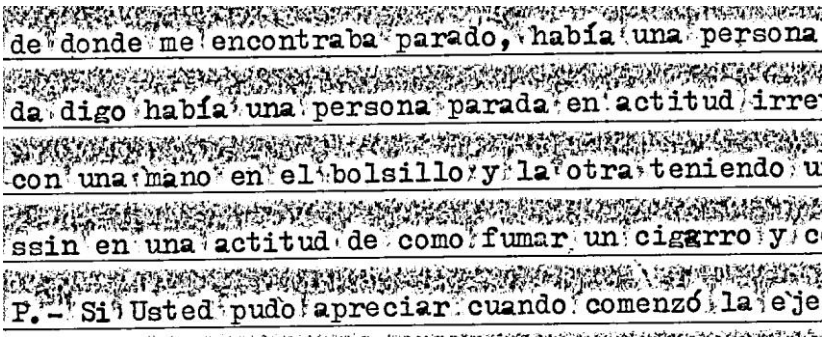

de donde me encontraba parado, había una persona
da digo había una persona parada en actitud irre
con una mano en el bolsillo y la otra teniendo u
ssin en una actitud de como fumar un cigarro y c
P.- Si Usted pudo apreciar cuando comenzó la eje

Figura 43 - Resultado de aplicar el método 2 a la fig. 32

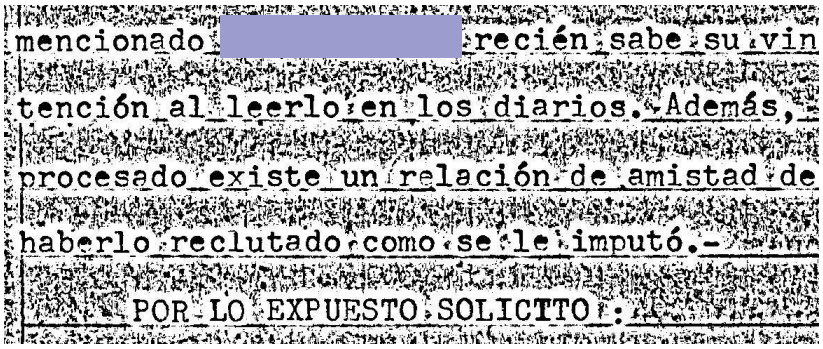

mencionado [redacted] recién sabe su vin
tención al leerlo en los diarios.-Además,
procesado existe un relación de amistad de
haberlo reclutado como se le imputó.-
POR LO EXPUESTO SOLICITO:

Figura 44 - Resultado de aplicar el método 2 a la fig. 33

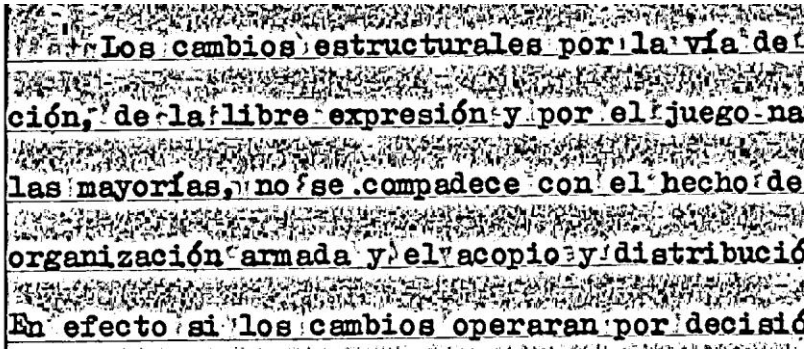

Los cambios estructurales por la vía de
ción, de la libre expresión y por el juego na
las mayorías, no se compadece con el hecho de
organización armada y el acopio y distribució
En efecto si los cambios operaran por decisió

Figura 45 - Resultado de aplicar el método 2 a la fig. 34

Prueba 3

En este punto se muestran los resultados obtenidos al aplicar Localtresh sobre las imágenes de muestra cuando el método (-m) fijado es el número tres. El resto de los parámetros se cargan con sus valores por defecto.

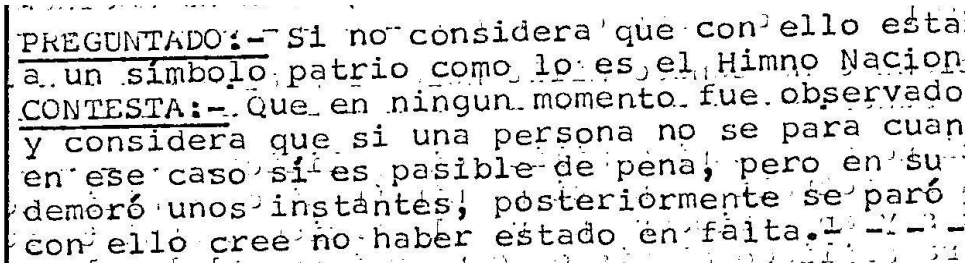

PREGUNTADO:- Si no considera que con ello esta
a un símbolo patrio como lo es el Himno Nacion
CONTESTA:- Que en ningun momento fue observado
y considera que si una persona no se para cuan
en ese caso sí es pasible de pena, pero en su
demoró unos instantes, posteriormente se paró
con ello cree no haber estado en falta.- - - -

Figura 46 - Resultado de aplicar el método 3 a la fig. 29

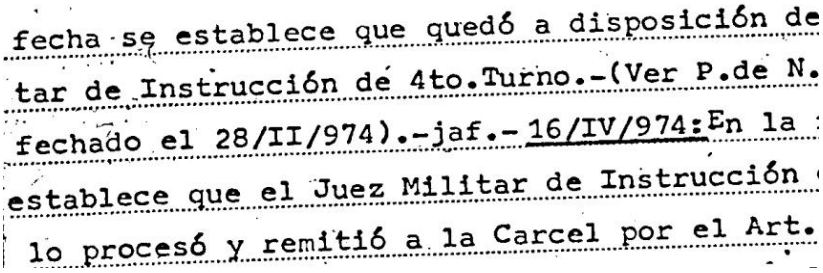

fecha se establece que quedó a disposición de
tar de Instrucción de 4to.Turno.-(Ver P.de N.
fechado el 28/II/974).-jaf.- 16/IV/974:En la
establece que el Juez Militar de Instrucción
lo procesó y remitió a la Carcel por el Art.

Figura 47 - Resultado de aplicar el método 3 a la fig. 30

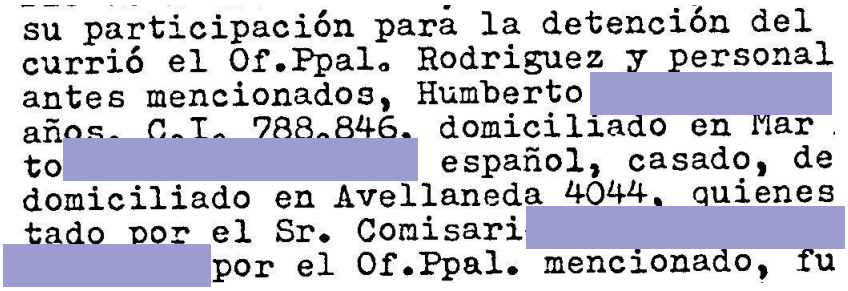

su participación para la detención del
currió el Of.Ppal. Rodriguez y personal
antes mencionados, Humberto
años. C.I. 788.846. domiciliado en Mar
to español, casado, de
domiciliado en Avellaneda 4044, quienes
tado por el Sr. Comisari
por el Of.Ppal. mencionado, fu

Figura 48 - Resultado de aplicar el método 3 a la fig. 31

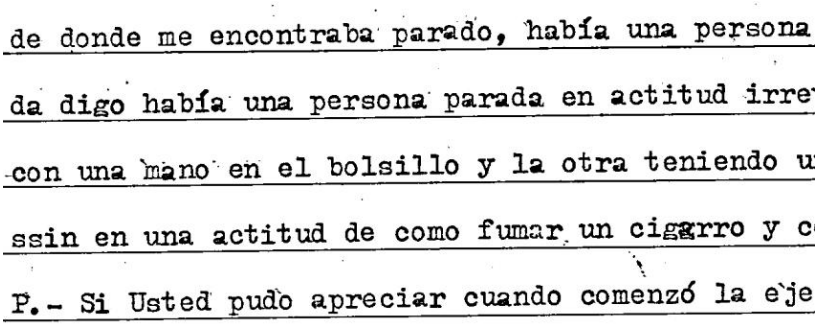

de donde me encontraba parado, había una persona
da digo había una persona parada en actitud irre
con una mano en el bolsillo y la otra teniendo u
ssin en una actitud de como fumar un cigarro y c
P.- Si Usted pudo apreciar cuando comenzó la eje

Figura 49 - Resultado de aplicar el método 3 a la fig. 32

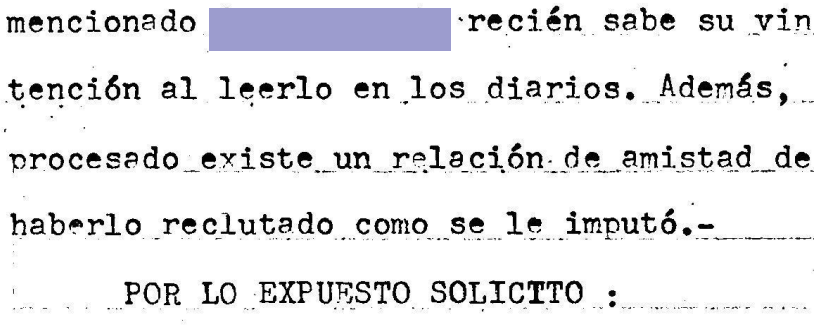

mencionado recién sabe su vin
tención al leerlo en los diarios. Además,
procesado existe un relación de amistad de
haberlo reclutado como se le imputó.-
POR LO EXPUESTO SOLICTTO :

Figura 50 - Resultado de aplicar el método 3 a la fig. 33

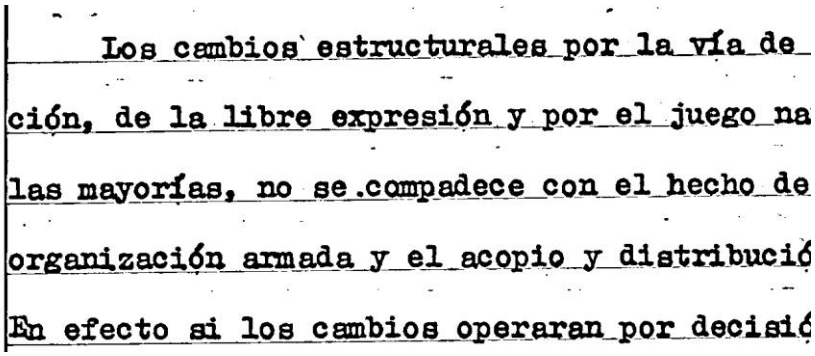

Los cambios estructurales por la vía de
ción, de la libre expresión y por el juego na
las mayorías, no se compadece con el hecho de
organización armada y el acopio y distribució
En efecto si los cambios operaran por decisió

Figura 51 - Resultado de aplicar el método 3 a la fig. 34

En las figuras antes presentadas se puede apreciar como mejora la imagen original si el método seleccionado es el número tres, no obstante, para constatar dicho resultado se

aplicó OCR [2] a cada una de las imágenes producto de las transformaciones antes mencionadas.

A continuación se muestran los resultados obtenidos de la ejecución de las pruebas:

Imagen	Método 1	Método 2	Método 3
009_LC	0	0	1
016_RC	1	0	0
022_RC	0	0	1
039_L	0	0	1
215_L	0	0	1
338_R	0	0	1
Total	1	0	**5**

Tabla 20 - Comparativa de los distintos métodos

La tabla anterior muestra el total de veces que un método superó al resto. Si el valor que se encuentra en una determinada celda es uno significa que ese método superó al resto para esa imagen que se está examinando, en caso contrario se coloca el valor cero. Como se observa en la tabla, el método que mejores resultados obtiene sobre las distintas imágenes de muestra es el número tres.

Análisis del parámetro radio

Análogamente al análisis mostrado anteriormente se realiza el estudio del parámetro radio **(-r)** para obtener el rango de valores que produzca mejores resultados sobre las imágenes originales. Este parámetro representa el radio de la ventana en la que se está trabajando, puede ser un valor real mayor o igual a 3. El valor por defecto es 15.

Para obtener resultados aceptables, el radio en general, debe ser mayor que la dimensión de los "objetos" que van a ser detectados, en este caso caracteres, por el umbral. En consecuencia, este método se aplica mejor a las imágenes de texto, objetos pequeños o bordes. Por tal motivo los valores seleccionados para las pruebas fueron los siguientes: 3, 18, 33 y 48.
Para un análisis más certero a cada imagen producto de la transformación realizada se le aplicó OCR de forma de poder comparar sobre qué imagen se realizaba una mejor extracción de datos.

A continuación se muestran las pruebas realizadas para el análisis de este parámetro.

[2] **Tesseract** fue la herramienta utilizada para estas pruebas debido a que fue seleccionada como el motor de OCR a utilizar (sección 4.2.1).

Prueba 1

La primer prueba muestra los resultados obtenidos de aplicar Localtresh a las imágenes elegidas donde el radio (-r) se fijó con el valor 3. El resto de los parámetros se cargaron con sus valores por defecto.

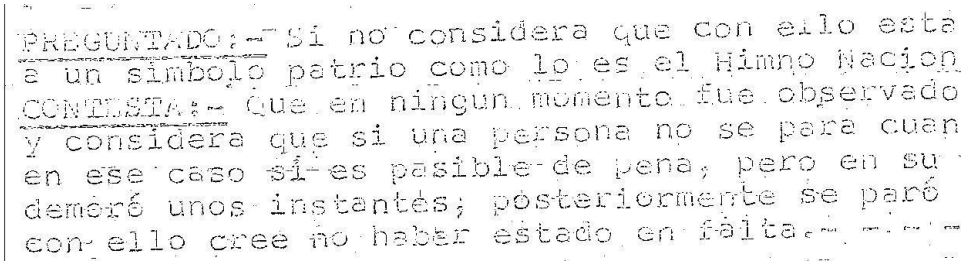

PREGUNTADO:- Si no considera que con ello esta
a un simbolo patrio como lo es el Himno Nacion
CONTESTA:- Que en ningun momento fue observado
y considera que si una persona no se para cuan
en ese caso si es pasible de pena, pero en su
demeró unos instantes; posteriormente se paró
con ello cree no haber estado en falta.-

Figura 52 - Resultado de fijar el radio en 3 aplicado a la fig. 29

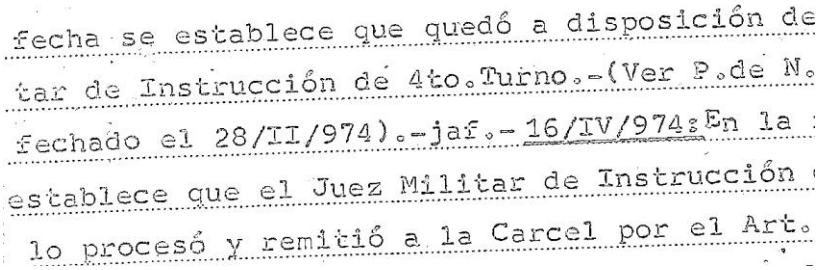

fecha se establece que quedó a disposición de
tar de Instrucción de 4to.Turno.-(Ver P.de N.
fechado el 28/II/974).-jaf.- 16/IV/974:En la
establece que el Juez Militar de Instrucción
lo procesó y remitió a la Carcel por el Art.

Figura 53 - Resultado de fijar el radio en 3 aplicado a la fig. 30

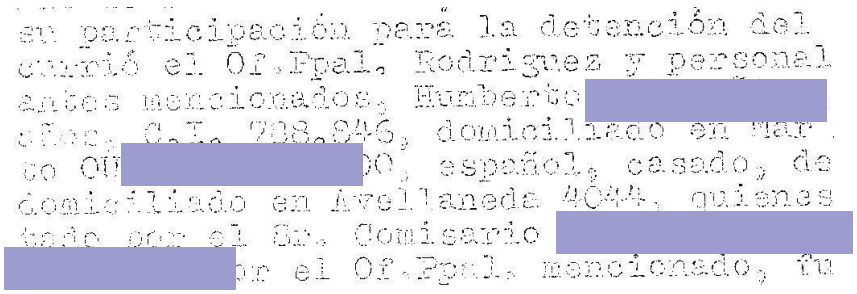

su participación para la detención del
cumplió el Of.Ppal. Rodriguez y personal
antes mencionados, Humberto
años, C.I. 738.846, domiciliado en Mar
to Of
00, español, casado, de
domiciliado en Avellaneda 4044, quienes
tado por el Sr. Comisario
por el Of.Ppal. mencionado, fu

Figura 54 - Resultado de fijar el radio en 3 aplicado a la fig. 31

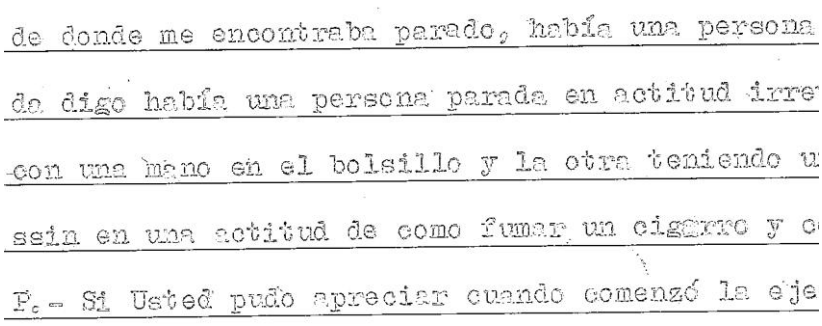

de donde me encontraba parado, había una persona
do digo había una persona parada en actitud irre
con una mano en el bolsillo y la otra teniendo u
sein en una actitud de como fumar un cigarro y c
P.- Si Usted pudo apreciar cuando comenzó la eje

Figura 55 - Resultado de fijar el radio en 3 aplicado a la fig. 32

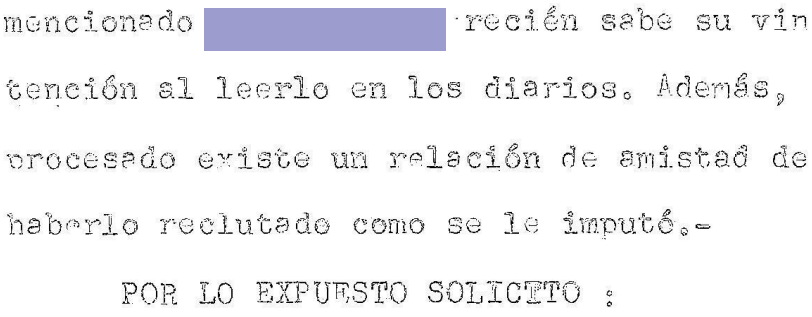

mencionado recién sabe su vin
tención al leerlo en los diarios. Además,
procesado existe un relación de amistad de
haberlo reclutado como se le imputé.-
POR LO EXPUESTO SOLICITO :

Figura 56 - Resultado de fijar el radio en 3 aplicado a la fig. 33

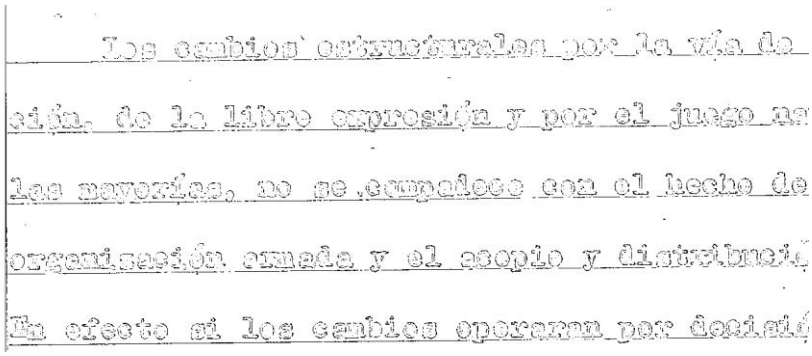

Los cambios estructurales por la vía de
sión, de la libre expresión y por el juego de
las mayorías, no se compadece con el hecho de
organización armada y el acopio y distribuci
En efecto si los cambios operarán por decisió

Figura 57 - Resultado de fijar el radio en 3 aplicado a la fig. 34

Prueba 2

Análogamente a lo realizado en la Prueba 1, se aplicó Localtresh sobre las imágenes seleccionadas fijando el radio (-r) en dieciocho. El resto de los parámetros se cargaron con sus valores por defecto.

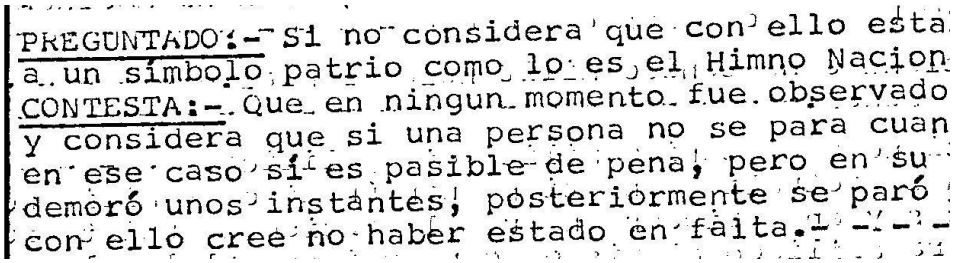

PREGUNTADO:- Si no considera que con ello esta
a un símbolo patrio como lo es el Himno Nacion
CONTESTA:- Que en ningun momento fue observado
y considera que si una persona no se para cuan
en ese caso sí es pasible de pena, pero en su
demoró unos instantes, posteriormente se paró
con ello cree no haber estado en falta.- -

Figura 58 - Resultado de fijar el radio en 18 aplicado a la fig. 29

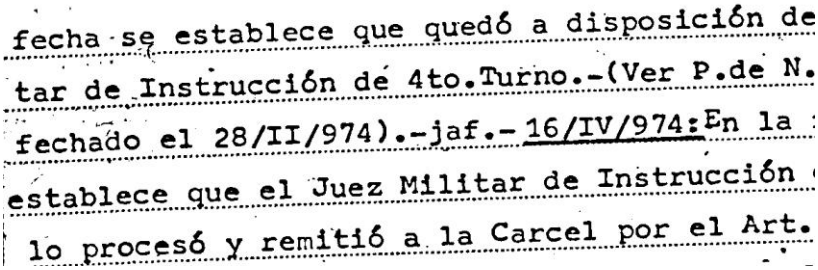

fecha se establece que quedó a disposición de
tar de Instrucción de 4to.Turno.-(Ver P.de N.
fechado el 28/II/974).-jaf.- 16/IV/974: En la
establece que el Juez Militar de Instrucción
lo procesó y remitió a la Carcel por el Art.

Figura 59 - Resultado de fijar el radio en 18 aplicado a la fig. 30

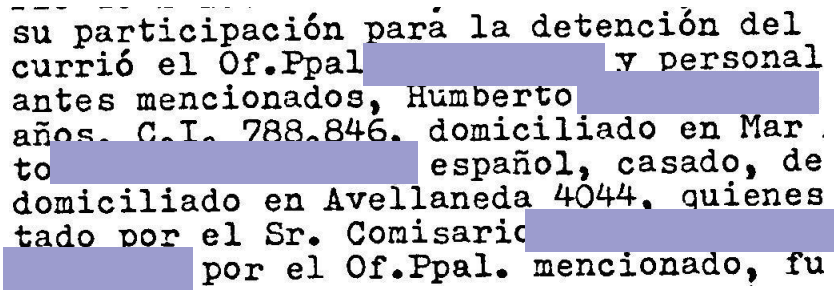

su participación para la detención del
currió el Of.Ppal [redacted] y personal
antes mencionados, Humberto [redacted]
años. C.I. 788.846. domiciliado en Mar
to [redacted] español, casado, de
domiciliado en Avellaneda 4044, quienes
tado por el Sr. Comisario [redacted]
[redacted] por el Of.Ppal. mencionado, fu

Figura 60 - Resultado de fijar el radio en 18 aplicado a la fig. 31

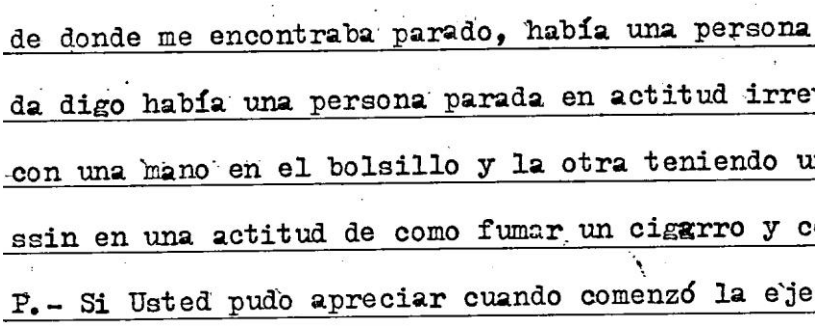

de donde me encontraba parado, había una persona
da digo había una persona parada en actitud irre
con una mano en el bolsillo y la otra teniendo u
ssin en una actitud de como fumar un cigarro y c
P.- Si Usted pudo apreciar cuando comenzó la eje

Figura 61 - Resultado de fijar el radio en 18 aplicado a la fig. 32

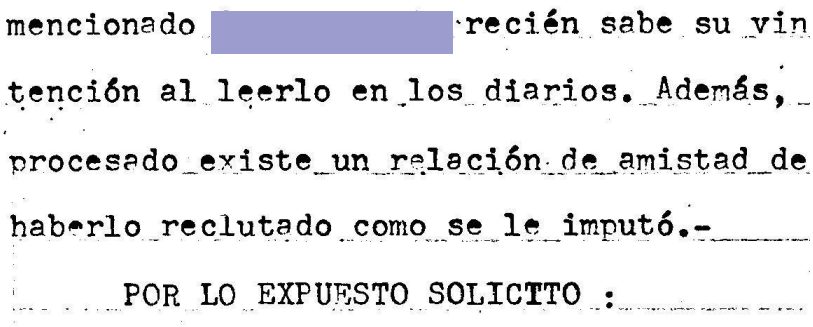

mencionado recién sabe su vin
tención al leerlo en los diarios. Además,
procesado existe un relación de amistad de
haberlo reclutado como se le imputó.-
POR LO EXPUESTO SOLICTTO :

Figura 62 - Resultado de fijar el radio en 18 aplicado a la fig. 33

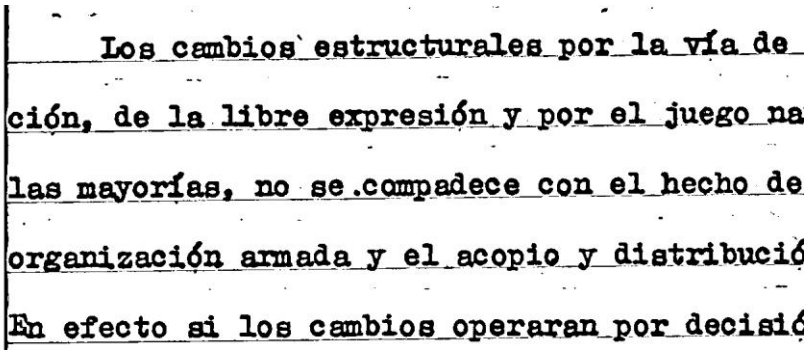

Los cambios estructurales por la vía de
ción, de la libre expresión y por el juego na
las mayorías, no se compadece con el hecho de
organización armada y el acopio y distribuci
En efecto si los cambios operaran por decisi

Figura 63 - Resultado de fijar el radio en 18 aplicado a la fig. 34

Prueba 3

En este punto se mostrarán los resultados obtenidos a partir de la ejecución de Localtresh fijando el radio (-r) en treinta y tres sobre las imágenes de muestra. El resto de los parámetros se cargaron con sus valores por defecto.

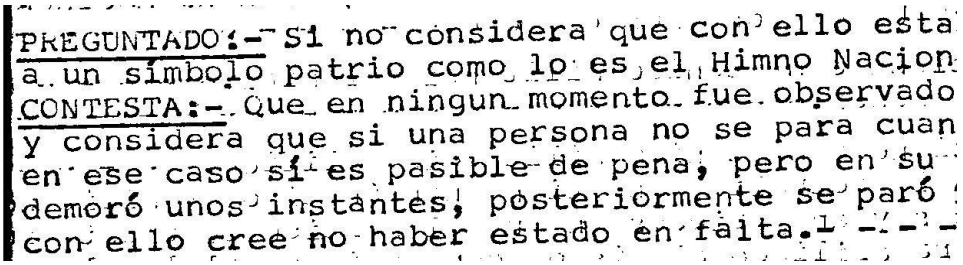

PREGUNTADO:- Si no considera que con ello esta
a un símbolo patrio como lo es el Himno Nacion
CONTESTA:- Que en ningun momento fue observado
y considera que si una persona no se para cuan
en ese caso sí es pasible de pena, pero en su
demoró unos instantes, posteriormente se paró
con ello cree no haber estado en falta.- -

Figura 64 - Resultado de fijar el radio en 33 aplicado a la fig. 29

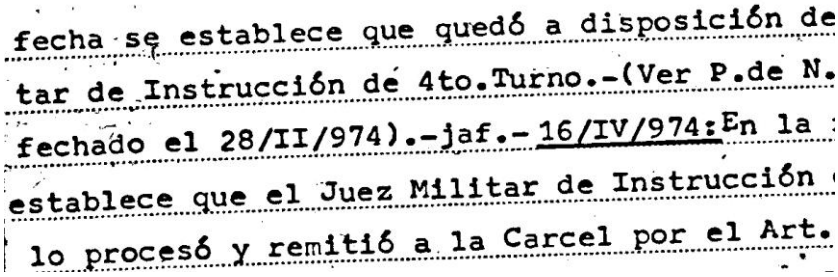

fecha se establece que quedó a disposición de
tar de Instrucción de 4to.Turno.-(Ver P.de N.
fechado el 28/II/974).-jaf.- 16/IV/974:En la
establece que el Juez Militar de Instrucción
lo procesó y remitió a la Carcel por el Art.

Figura 65 - Resultado de fijar el radio en 33 aplicado a la fig. 30

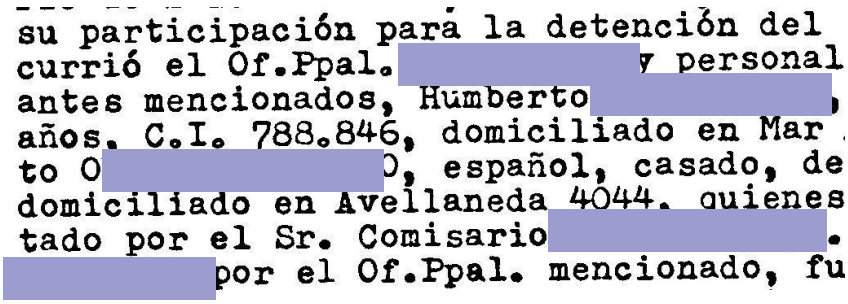

su participación para la detención del
currió el Of.Ppal. y personal
antes mencionados, Humberto
años. C.I. 788.846, domiciliado en Mar
to O , español, casado, de
domiciliado en Avellaneda 4044, quienes
tado por el Sr. Comisario .
por el Of.Ppal. mencionado, fu

Figura 66 - Resultado de fijar el radio en 33 aplicado a la fig. 31

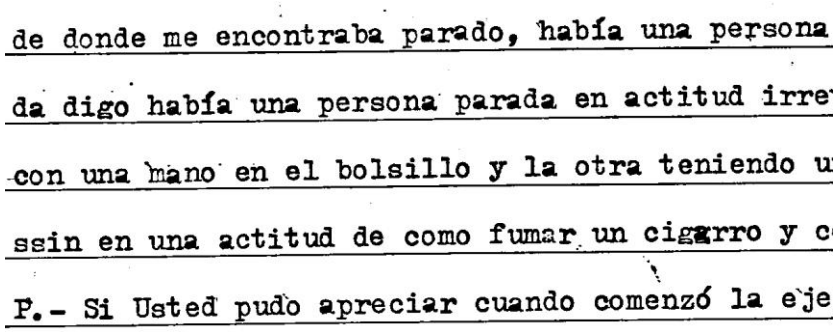

de donde me encontraba parado, había una persona
da digo había una persona parada en actitud irre
con una mano en el bolsillo y la otra teniendo u
ssin en una actitud de como fumar un cigarro y c
P.- Si Usted pudo apreciar cuando comenzó la eje

Figura 67 - Resultado de fijar el radio en 33 aplicado a la fig. 32

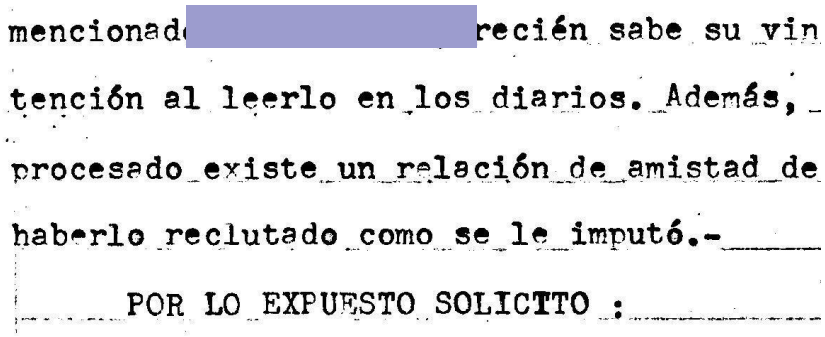

mencionad recién sabe su vin
tención al leerlo en los diarios. Además,
procesado existe un relación de amistad de
haberlo reclutado como se le imputó.-
POR LO EXPUESTO SOLICITO :

Figura 68 - Resultado de fijar el radio en 33 aplicado a la fig. 33

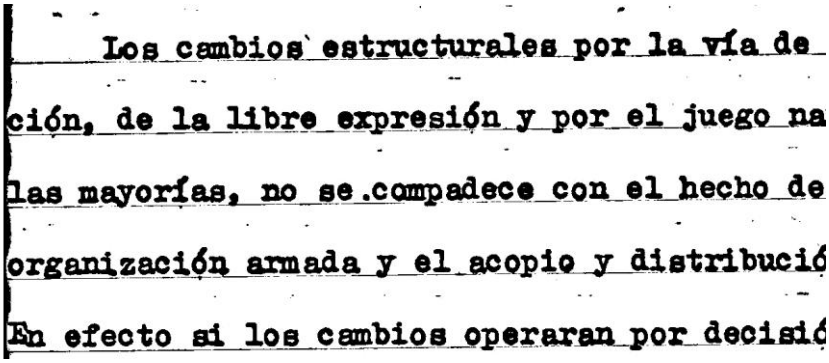

Los cambios estructurales por la vía de
ción, de la libre expresión y por el juego na
las mayorías, no se compadece con el hecho de
organización armada y el acopio y distribuci
En efecto si los cambios operaran por decisi

Figura 69 - Resultado de fijar el radio en 33 aplicado a la fig. 34

Prueba 4

Por último, en esta prueba se expondrán los resultados obtenidos de ejecutar Localtresh sobre las imágenes seleccionadas fijando el parámetro radio (-r) en 48. El resto de los parámetros se cargaron con sus valores por defecto.

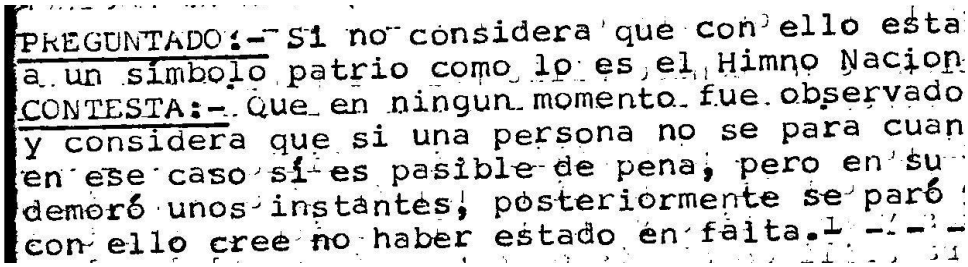

PREGUNTADO:- Si no considera que con ello esta
a un símbolo patrio como lo es el Himno Nacion
CONTESTA:- Que en ningun momento fue observado
y considera que si una persona no se para cuan
en ese caso sí es pasible de pena, pero en su
demoró unos instantes, posteriormente se paró
con ello cree no haber estado en falta.- - -

Figura 70 - Resultado de fijar el radio en 48 aplicado a la fig. 29

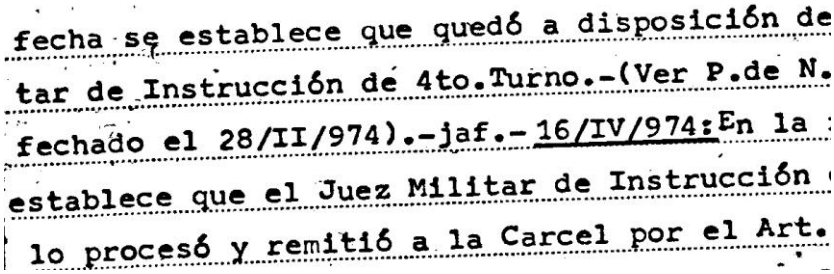

fecha se establece que quedó a disposición de
tar de Instrucción de 4to.Turno.-(Ver P.de N.
fechado el 28/II/974).-jaf.- 16/IV/974:En la
establece que el Juez Militar de Instrucción
lo procesó y remitió a la Carcel por el Art.

Figura 71 - Resultado de fijar el radio en 48 aplicado a la fig. 30

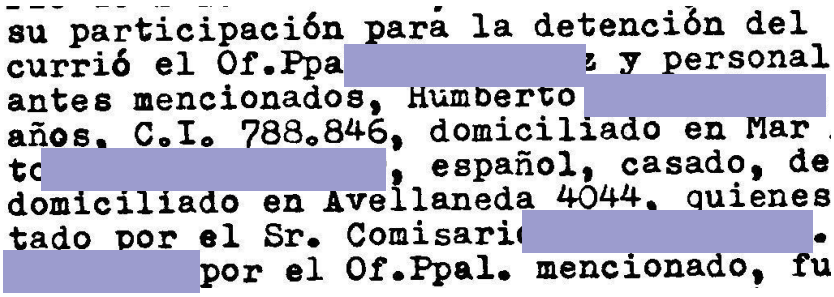

su participación para la detención del
currió el Of.Ppa z y personal
antes mencionados, Humberto
años. C.I. 788.846, domiciliado en Mar
to , español, casado, de
domiciliado en Avellaneda 4044, quienes
tado por el Sr. Comisari .
por el Of.Ppal. mencionado, fu

Figura 72 - Resultado de fijar el radio en 48 aplicado a la fig. 31

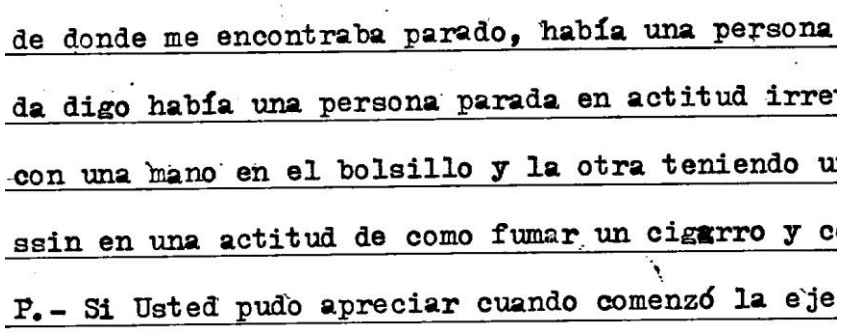

de donde me encontraba parado, había una persona
da digo había una persona parada en actitud irre
con una mano en el bolsillo y la otra teniendo u
ssin en una actitud de como fumar un cigarro y c
P.- Si Usted pudo apreciar cuando comenzó la eje

Figura 73 - Resultado de fijar el radio en 48 aplicado a la fig. 32

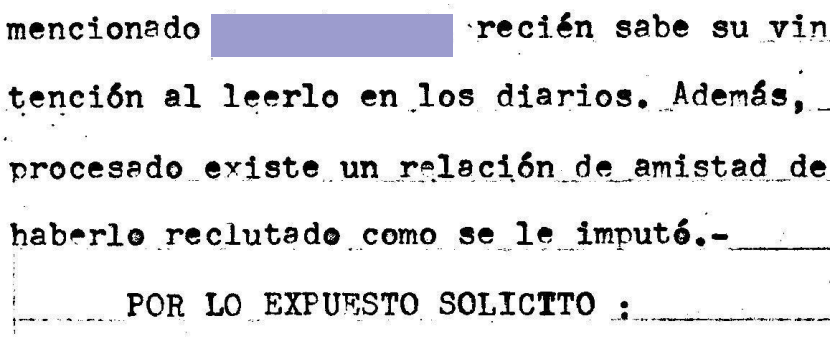

mencionado [redacted] recién sabe su vin
tención al leerlo en los diarios. Además,
procesado existe un relación de amistad de
haberlo reclutado como se le imputó.-
POR LO EXPUESTO SOLICTTO :

Figura 74 - Resultado de fijar el radio en 48 aplicado a la fig. 33

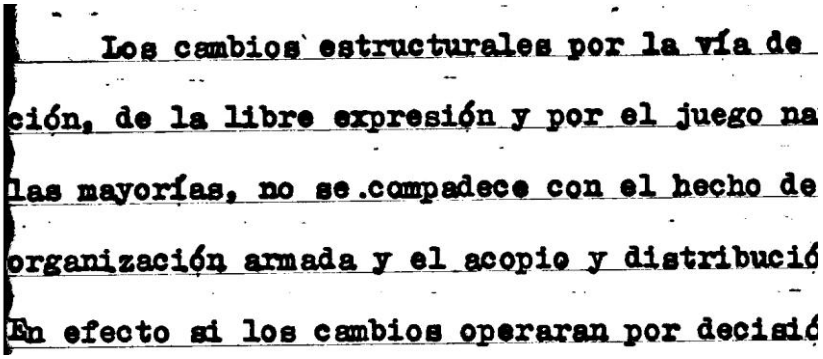

Los cambios estructurales por la vía de
ción, de la libre expresión y por el juego na
las mayorías, no se compadece con el hecho de
organización armada y el acopio y distribució
En efecto si los cambios operaran por decisió

Figura 75 - Resultado de fijar el radio en 48 aplicado a la fig. 34

A diferencia del análisis anterior aquí no se observa de forma gráfica cual es el valor del parámetro radio (-r) que proporciona mejores resultados. Para hallar una cota inferior y superior del mismo se empleó el método explicado en el punto anterior el cual compara todos los resultados obtenidos producto de la aplicación de la herramienta OCR elegida a

cada una de las imágenes alteradas por las transformaciones antes descritas. A continuación se muestran los resultados obtenidos a partir de la ejecución de las pruebas antes presentadas:

Imagen	Radio 3	Radio 18	Radio 33	Radio 48
009_LC	0	0	1	0
016_RC	0	0	0	1
022_RC	0	1	1	1
039_L	0	1	0	1
215_L	0	0	0	1
338_R	0	1	0	0
Total	0	**3**	2	**4**

Tabla 21 - Comparativa de los distintos radios

La tabla 21 muestra el total de veces que cada valor de radio probado obtuvo resultados mejores o iguales que el resto. Si el valor que se encuentra en una determinada celda es uno significa que ese radio obtuvo resultados mejores o iguales que el resto para esa imagen que se está examinando, en caso contrario el valor es cero.
Dado que los valores que mejor desempeño tuvieron fueron 18 y 48, se definen éstos como cota inferior y superior para el radio el cual se analiza más en detalle en el Anexo 3.

Análisis del parámetro sesgo

En este punto se analizará dentro de que rangos se deberá seleccionar el valor del parámetro sesgo (-b) de forma tal de obtener los mejores resultados posibles. Para llevar a cabo esto se eligió una tupla de valores (5, 10, 20, 30, 40, 50) de forma de cubrir de manera espaciada valores de porcentajes menores o iguales a 50 ya que al elegir porcentajes altos se corre un gran riesgo de eliminar parte de los objetos a reconocer.
Los valores de sesgo se expresan como porcentaje, y en caso de que no se ingrese su valor, éste por defecto es 20.

Prueba 1

En esta prueba se expondrán los resultados obtenidos de ejecutar Localtresh sobre las imágenes seleccionadas fijando el parámetro sesgo (-b) en 5. El resto de los parámetros se cargaron con sus valores por defecto.

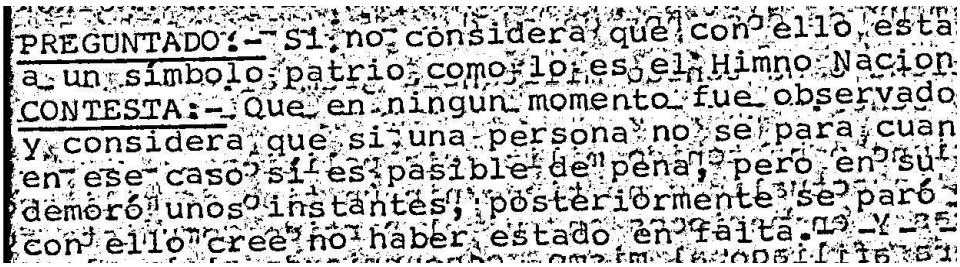

PREGUNTADO:- Si no considera que con ello esta
a un símbolo patrio como lo es el Himno Nacion
CONTESTA:- Que en ningun momento fue observado
y considera que si una persona no se para cuan
en ese caso sí es pasible de pena, pero en su
demoró unos instantes, posteriormente se paró
con ello cree no haber estado en falta.

Figura 76 - Resultado de fijar el sesgo en 5 aplicado a la fig. 29

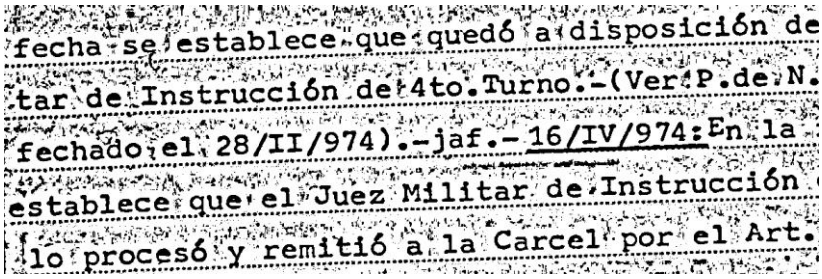

fecha se establece que quedó a disposición de
tar de Instrucción de 4to.Turno.-(Ver P.de N.
fechado el 28/II/974).-jaf.- 16/IV/974: En la
establece que el Juez Militar de Instrucción
lo procesó y remitió a la Carcel por el Art.

Figura 77 - Resultado de fijar el sesgo en 5 aplicado a la fig. 30

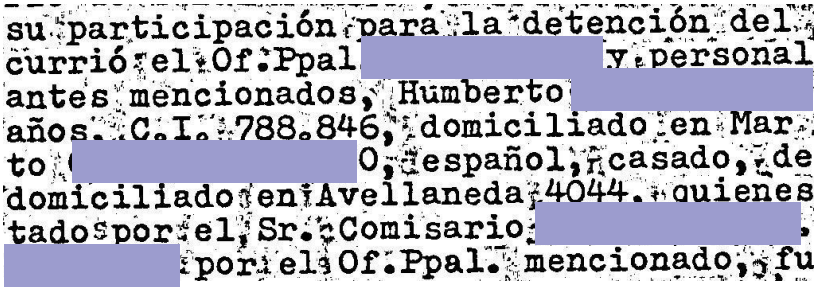

su participación para la detención del
currió el Of.Ppal [redacted] y personal
antes mencionados, Humberto [redacted]
años. C.I. 788.846, domiciliado en Mar
to [redacted] O, español, casado, de
domiciliado en Avellaneda 4044. quienes
tado por el Sr. Comisario [redacted].
[redacted] por el Of.Ppal. mencionado, fu

Figura 78 - Resultado de fijar el sesgo en 5 aplicado a la fig. 31

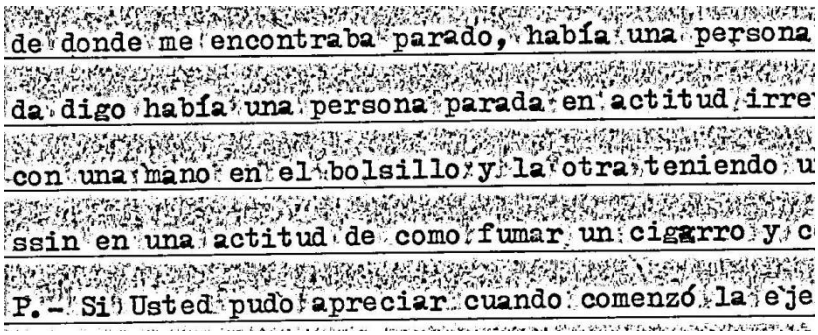

de donde me encontraba parado, había una persona
da digo había una persona parada en actitud irre
con una mano en el bolsillo y la otra teniendo u
ssin en una actitud de como fumar un cigarro y c
P.- Si Usted pudo apreciar cuando comenzó la eje

Figura 79 - Resultado de fijar el sesgo en 5 aplicado a la fig. 32

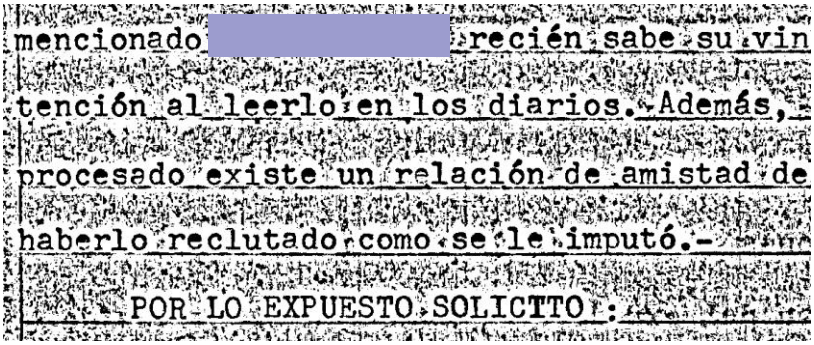

mencionado recién sabe su vin
tención al leerlo en los diarios.- Además,
procesado existe un relación de amistad de
haberlo reclutado como se le imputó.-
POR LO EXPUESTO SOLICITO:

Figura 80 - Resultado de fijar el sesgo en 5 aplicado a la fig. 33

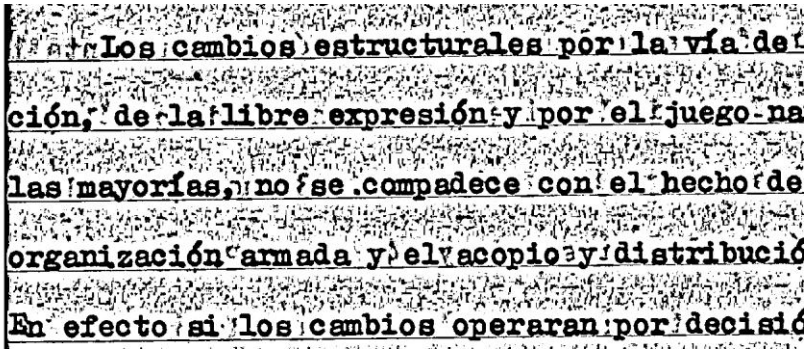

Los cambios estructurales por la vía de
ción, de la libre expresión y por el juego na
las mayorías, no se compadece con el hecho de
organización armada y el acopio y distribució
En efecto si los cambios operaran por decisió

Figura 81 - Resultado de fijar el sesgo en 5 aplicado a la fig. 34

Prueba 2

Aquí se mostrarán los resultados obtenidos al ejecutar Localtresh sobre las imágenes elegidas fijando el parámetro sesgo (-b) en 10. El resto de los parámetros se cargaron con sus valores por defecto.

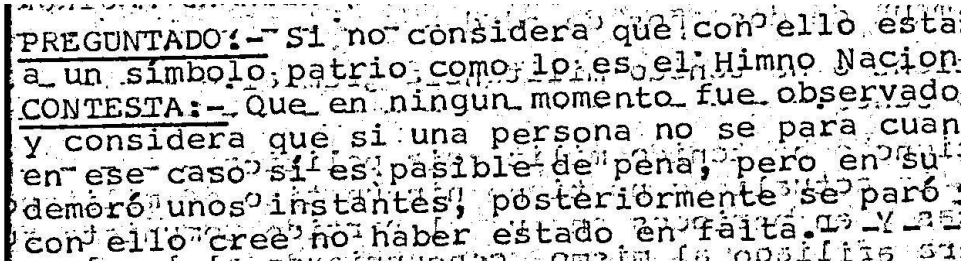

PREGUNTADO:- Si no considera que con ello esta
a un símbolo patrio como lo es el Himno Nacion
CONTESTA:- Que en ningun momento fue observado
y considera que si una persona no se para cuan
en ese caso sí es pasible de pena, pero en su
demoró unos instantes, posteriormente se paró
con ello cree no haber estado en falta.- Y -

Figura 82 - Resultado de fijar el sesgo en 10 aplicado a la fig. 29

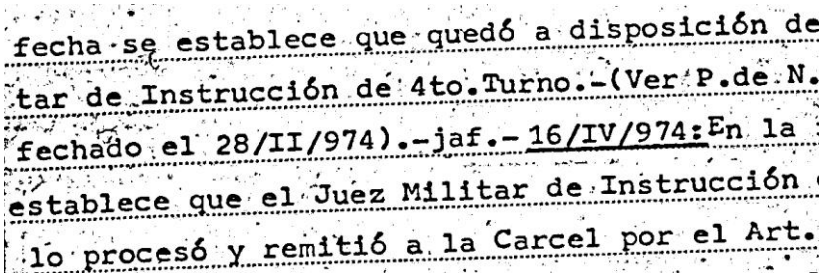

fecha se establece que quedó a disposición de
tar de Instrucción de 4to.Turno.-(Ver P.de N.
fechado el 28/II/974).-jaf.- 16/IV/974:En la
establece que el Juez Militar de Instrucción
lo procesó y remitió a la Carcel por el Art.

Figura 83 - Resultado de fijar el sesgo en 10 aplicado a la fig. 30

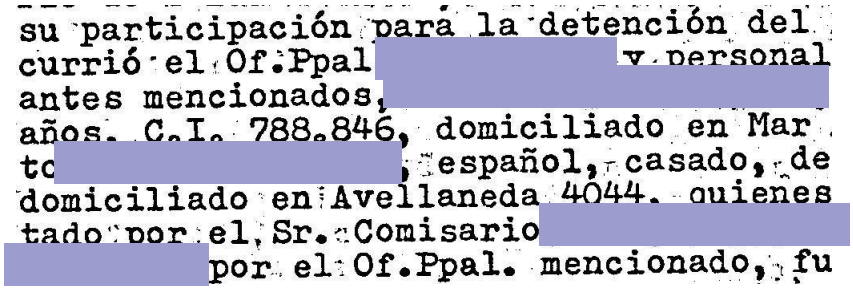

su participación para la detención del
currió el Of.Ppal y personal
antes mencionados,
años. C.I. 788.846, domiciliado en Mar
to , español, casado, de
domiciliado en Avellaneda 4044. quienes
tado por el Sr. Comisario
por el Of.Ppal. mencionado, fu

Figura 84 - Resultado de fijar el sesgo en 10 aplicado a la fig. 31

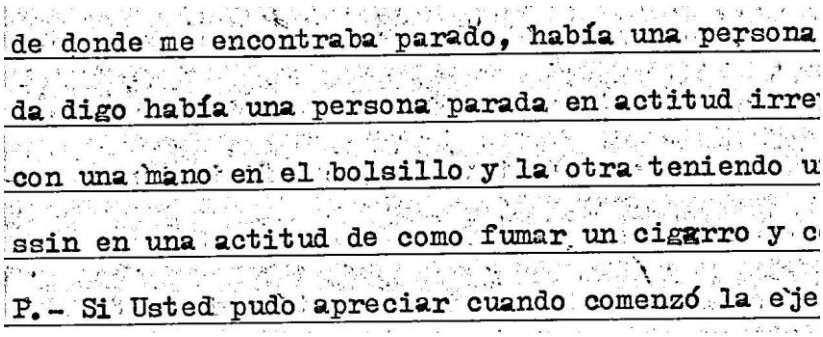

de donde me encontraba parado, había una persona
da digo había una persona parada en actitud irre
con una mano en el bolsillo y la otra teniendo u
ssin en una actitud de como fumar un cigarro y c
P.- Si Usted pudo apreciar cuando comenzó la eje

Figura 85 - Resultado de fijar el sesgo en 10 aplicado a la fig. 32

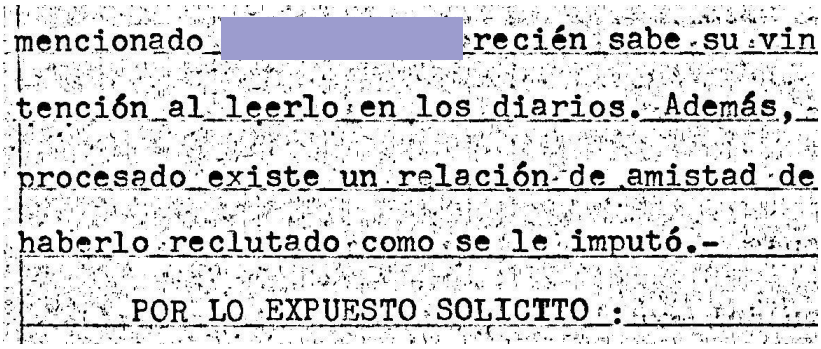

mencionado [redacted] recién sabe su vin
tención al leerlo en los diarios. Además,
procesado existe un relación de amistad de
haberlo reclutado como se le imputó.-
POR LO EXPUESTO SOLICITO :

Figura 86 - Resultado de fijar el sesgo en 10 aplicado a la fig. 33

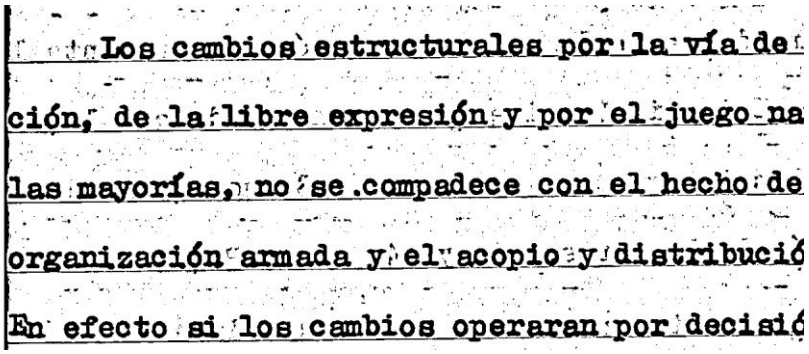

Los cambios estructurales por la vía de
ción, de la libre expresión y por el juego na
las mayorías, no se compadece con el hecho de
organización armada y el acopio y distribuci
En efecto si los cambios operaran por decisi

Figura 87 - Resultado de fijar el sesgo en 10 aplicado a la fig. 34

Prueba 3

En esta tercer prueba se presentan los resultados obtenidos al ejecutar Localtresh sobre las imágenes originales seleccionadas fijando el parámetro sesgo (-b) en 20. El resto de los parámetros se cargaron con sus valores por defecto.

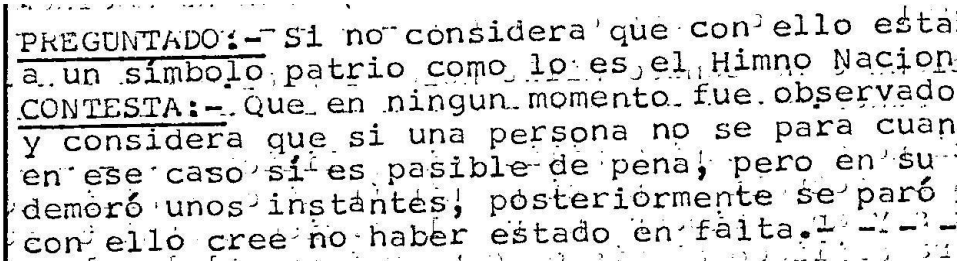

PREGUNTADO:- Si no considera que con ello esta
a un símbolo patrio como lo es el Himno Nacion
CONTESTA:- Que en ningun momento fue observado
y considera que si una persona no se para cuan
en ese caso sí es pasible de pena, pero en su
demoró unos instantes, posteriormente se paró
con ello cree no haber estado en falta.-

Figura 88 - Resultado de fijar el sesgo en 20 aplicado a la fig. 29

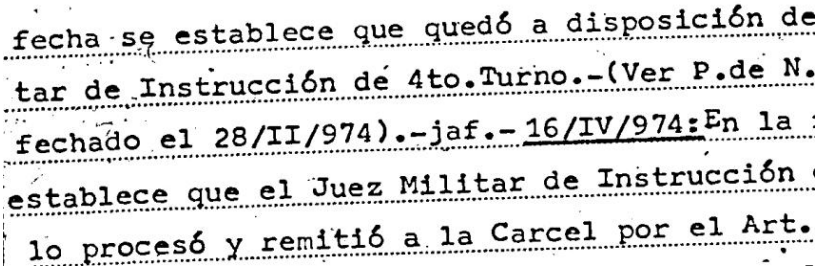

fecha se establece que quedó a disposición de
tar de Instrucción de 4to.Turno.-(Ver P.de N.
fechado el 28/II/974).-jaf.- 16/IV/974:En la
establece que el Juez Militar de Instrucción
lo procesó y remitió a la Carcel por el Art.

Figura 89 - Resultado de fijar el sesgo en 20 aplicado a la fig. 30

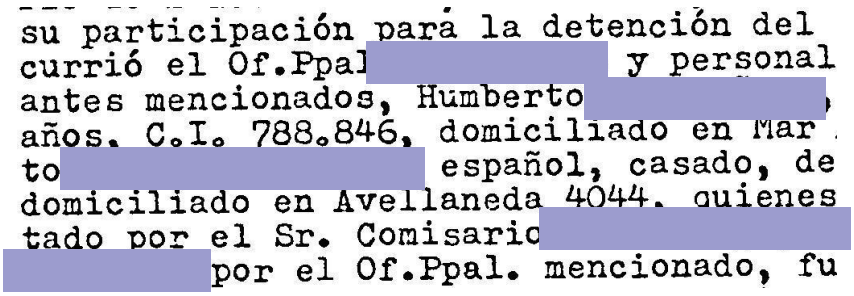

su participación para la detención del
currió el Of.Ppal [redacted] y personal
antes mencionados, Humberto [redacted]
años, C.I. 788.846, domiciliado en Mar
to [redacted] español, casado, de
domiciliado en Avellaneda 4044, quienes
tado por el Sr. Comisario [redacted]
[redacted] por el Of.Ppal. mencionado, fu

Figura 90 - Resultado de fijar el sesgo en 20 aplicado a la fig. 31

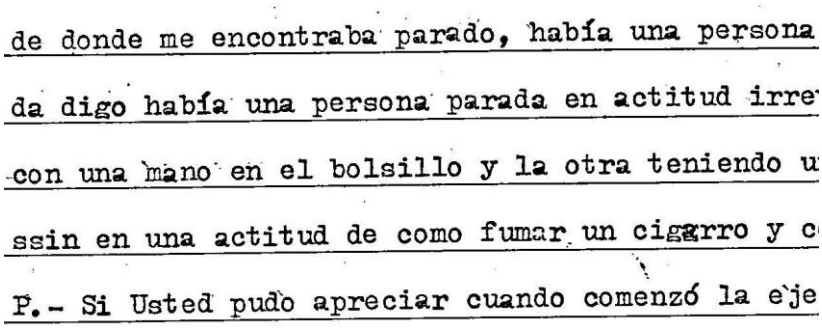

de donde me encontraba parado, había una persona
da digo había una persona parada en actitud irre
con una mano en el bolsillo y la otra teniendo u
ssin en una actitud de como fumar un cigarro y c
P.- Si Usted pudo apreciar cuando comenzó la eje

Figura 91 - Resultado de fijar el sesgo en 20 aplicado a la fig. 32

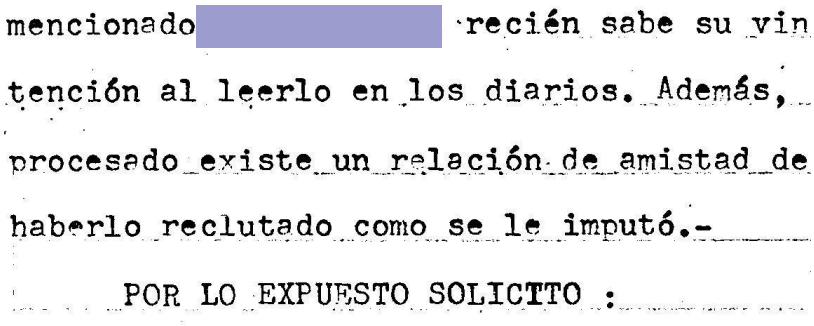

mencionado recién sabe su vin
tención al leerlo en los diarios. Además,
procesado existe un relación de amistad de
haberlo reclutado como se le imputó.-
POR LO EXPUESTO SOLICTTO :

Figura 92 - Resultado de fijar el sesgo en 20 aplicado a la fig. 33

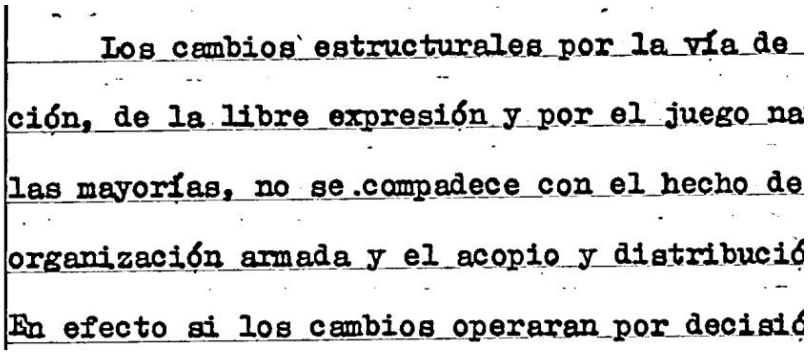

Los cambios estructurales por la vía de
ción, de la libre expresión y por el juego na
las mayorías, no se compadece con el hecho de
organización armada y el acopio y distribuci
En efecto si los cambios operaran por decisi

Figura 93 - Resultado de fijar el sesgo en 20 aplicado a la fig. 34

Prueba 4

La cuarta prueba expone los resultados que se obtuvieron al ejecutar Localtresh sobre las imágenes originales seleccionadas fijando el parámetro sesgo (-b) en 30. El resto de los parámetros se cargaron con sus valores por defecto.

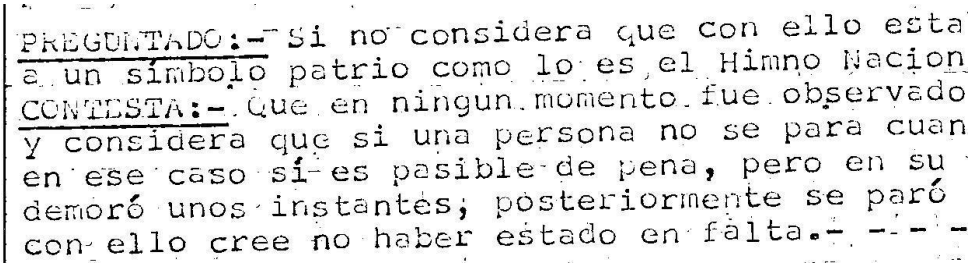

PREGUNTADO:- Si no considera que con ello esta
a un símbolo patrio como lo es el Himno Nacion
CONTESTA:- Que en ningun momento fue observado
y considera que si una persona no se para cuan
en ese caso sí es pasible de pena, pero en su
demoró unos instantes; posteriormente se paró
con ello cree no haber estado en falta.- - - -

Figura 94 - Resultado de fijar el sesgo en 30 aplicado a la fig. 29

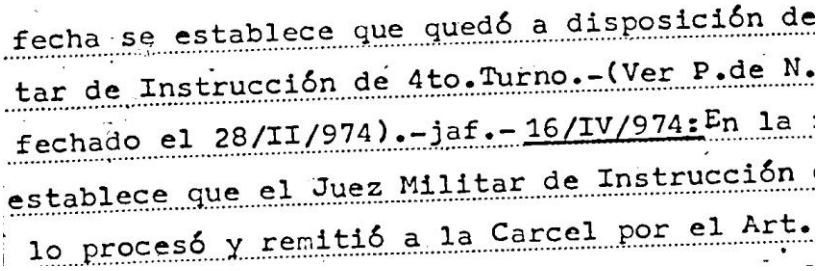

fecha se establece que quedó a disposición de
tar de Instrucción de 4to.Turno.-(Ver P.de N.
fechado el 28/II/974).-jaf.- 16/IV/974: En la
establece que el Juez Militar de Instrucción
lo procesó y remitió a la Carcel por el Art.

Figura 95 - Resultado de fijar el sesgo en 30 aplicado a la fig. 30

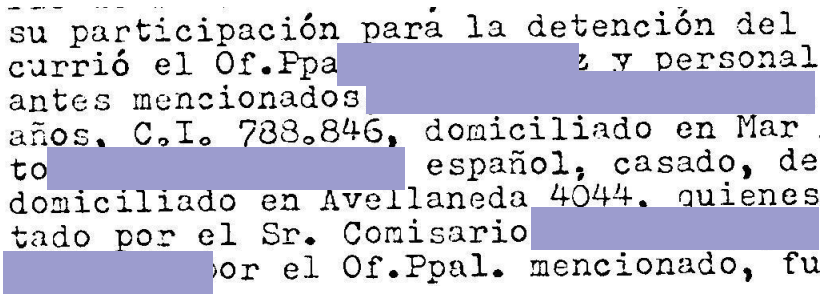

su participación para la detención del
currió el Of.Ppa [redacted] y personal
antes mencionados [redacted]
años, C.I. 788.846, domiciliado en Mar
to [redacted] español, casado, de
domiciliado en Avellaneda 4044, quienes
tado por el Sr. Comisario [redacted]
[redacted] or el Of.Ppal. mencionado, fu

Figura 96 - Resultado de fijar el sesgo en 30 aplicado a la Fig. 31

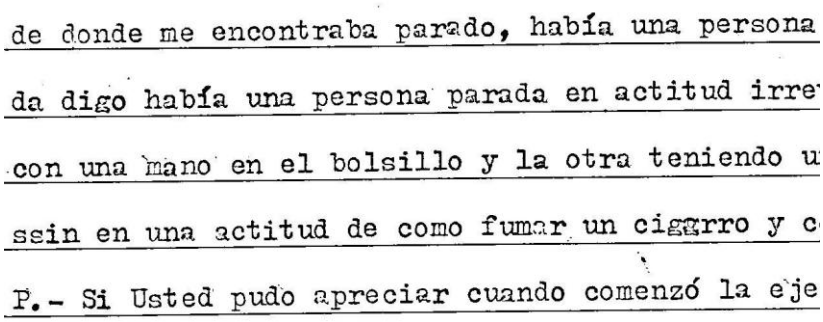

de donde me encontraba parado, había una persona
da digo había una persona parada en actitud irre
con una mano en el bolsillo y la otra teniendo u
ssin en una actitud de como fumar un cigarro y c
P.- Si Usted pudo apreciar cuando comenzó la eje

Figura 97 - Resultado de fijar el sesgo en 30 aplicado a la fig. 32

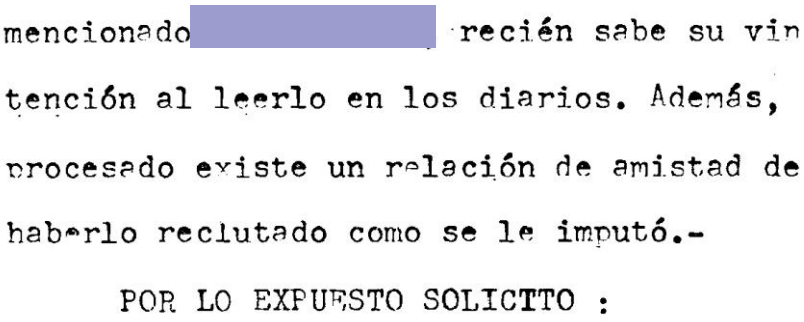

mencionado recién sabe su vin
tención al leerlo en los diarios. Además,
procesado existe un relación de amistad de
haberlo reclutado como se le imputó.-
POR LO EXPUESTO SOLICITO :

Figura 98 - Resultado de fijar el sesgo en 30 aplicado a la fig. 33

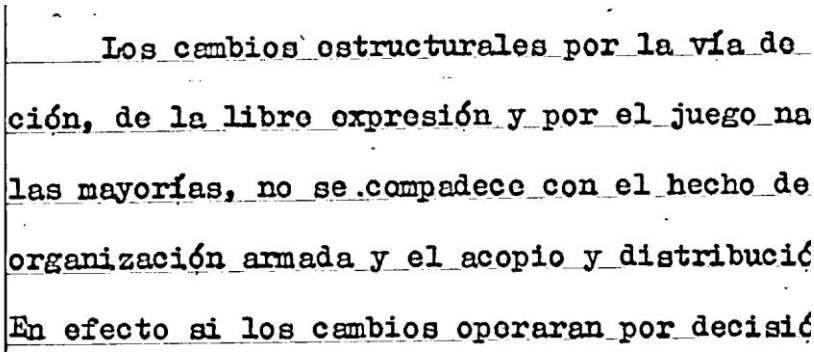

Los cambios estructurales por la vía de
ción, de la libre expresión y por el juego na
las mayorías, no se compadece con el hecho de
organización armada y el acopio y distribuci
En efecto si los cambios operaran por decisi

Figura 99 - Resultado de fijar el sesgo en 30 aplicado a la fig. 34

Prueba 5

Esta penúltima prueba muestra los resultados obtenidos al ejecutar Localtresh sobre las imágenes elegidas fijando el parámetro sesgo (-b) en 40. El resto de los parámetros se cargaron con sus valores por defecto.

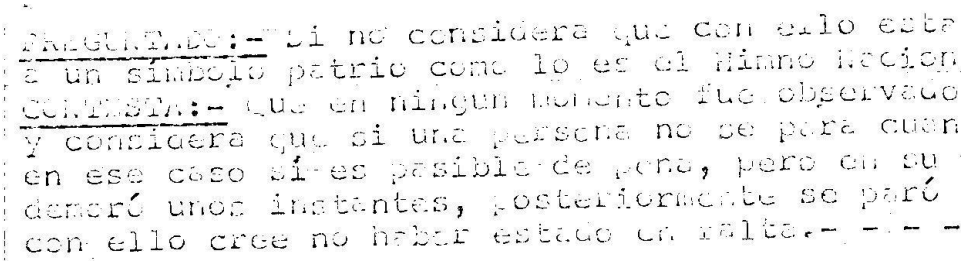


Figura 100 - Resultado de fijar el sesgo en 40 aplicado a la fig. 29

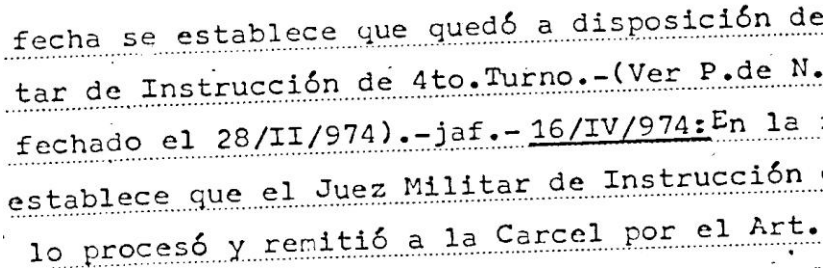


Figura 101 - Resultado de fijar el sesgo en 40 aplicado a la fig. 30

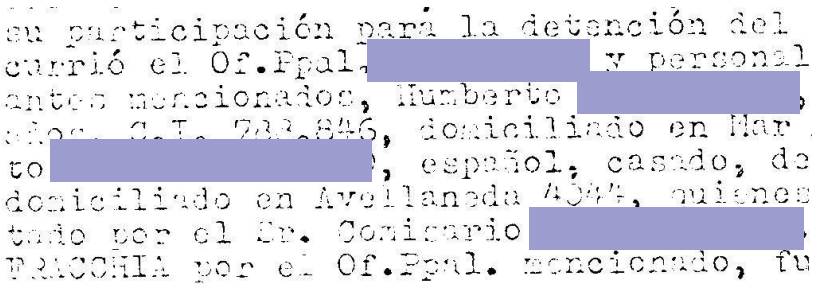


Figura 102 - Resultado de fijar el sesgo en 40 aplicado a la fig. 31

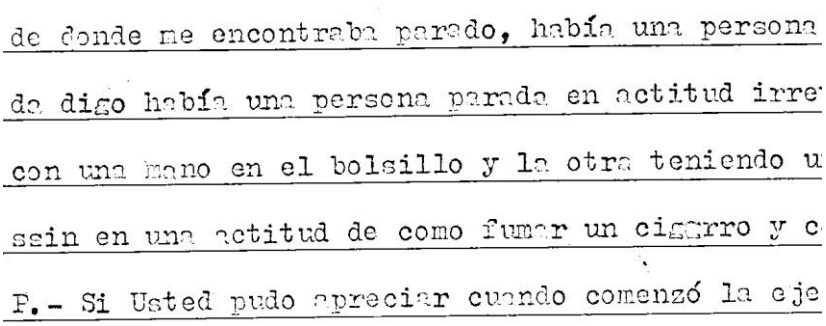

de donde me encontraba parado, había una persona
de digo había una persona parada en actitud irre
con una mano en el bolsillo y la otra teniendo u
sein en una actitud de como fumar un cigarro y c
P.- Si Usted pudo apreciar cuando comenzó la eje

Figura 103 - Resultado de fijar el sesgo en 40 aplicado a la fig. 32

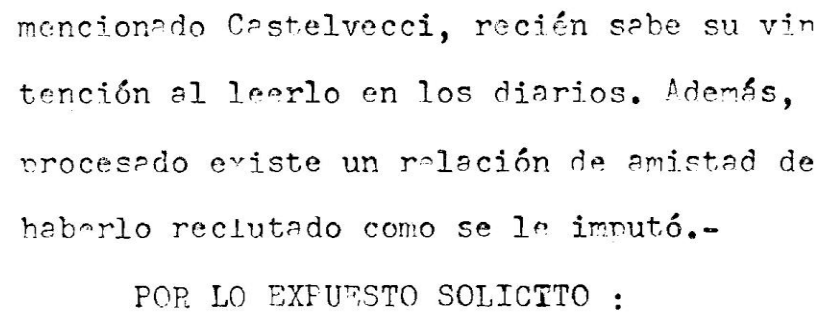

mencionado Castelvecci, recién sabe su vin
tención al leerlo en los diarios. Además,
procesado existe un relación de amistad de
haberlo reclutado como se le imputó.-
POR LO EXPUESTO SOLICITO :

Figura 104 - Resultado de fijar el sesgo en 40 aplicado a la fig. 33

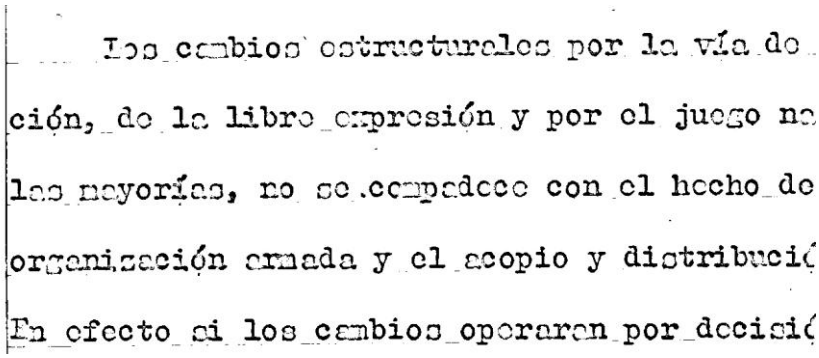

Los cambios estructurales por la vía de
ción, de la libre expresión y por el juego na
las mayorías, no se compadece con el hecho de
organización armada y el acopio y distribuci
En efecto si los cambios operaran por decisi

Figura 105 - Resultado de fijar el sesgo en 40 aplicado a la fig. 34

Prueba 6

La última prueba expone los resultados que se obtuvieron al ejecutar Localtresh sobre las imágenes originales seleccionadas fijando el parámetro sesgo (-b) en 50. El resto de los parámetros se cargaron con sus valores por defecto.

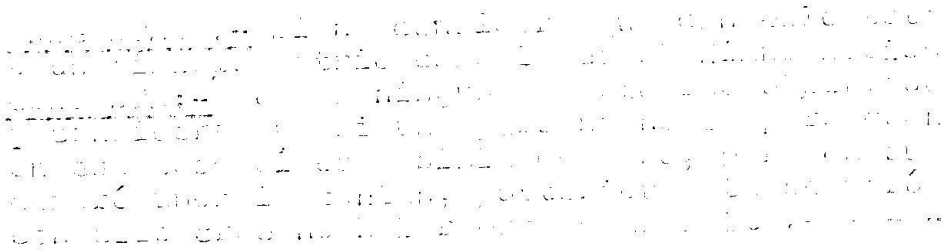


Figura 106 - Resultado de fijar el sesgo en 50 aplicado a la fig. 29

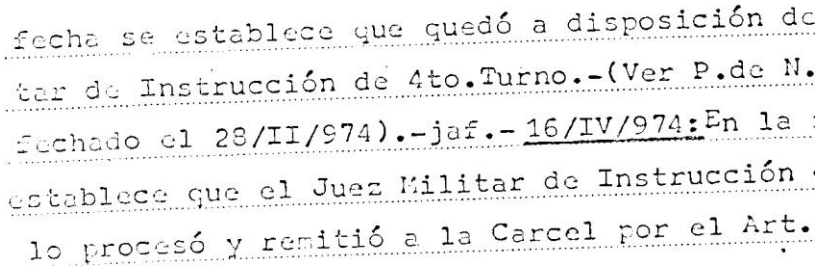


Figura 107 - Resultado de fijar el sesgo en 50 aplicado a la fig. 30

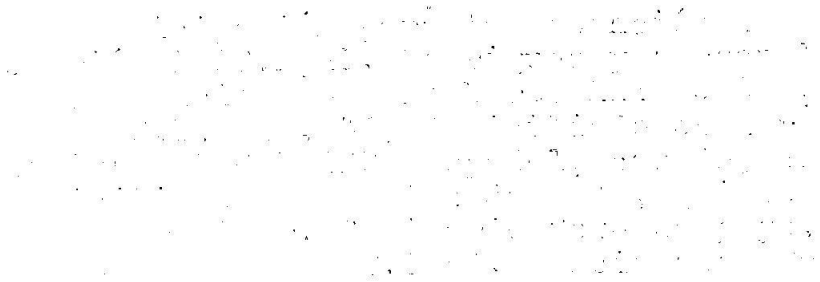


Figura 108 - Resultado de fijar el sesgo en 50 aplicado a la fig. 31

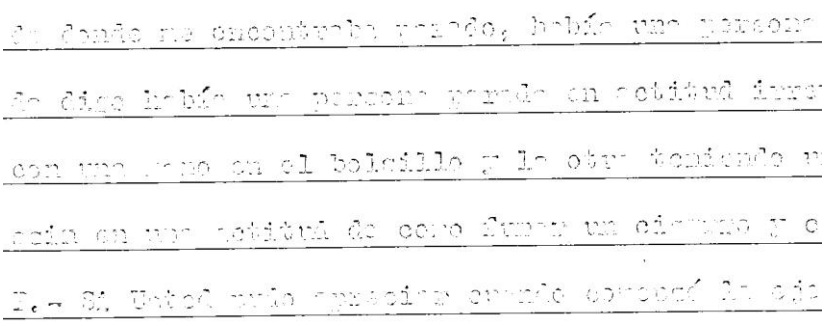


Figura 109 - Resultado de fijar el sesgo en 50 aplicado a la fig. 32

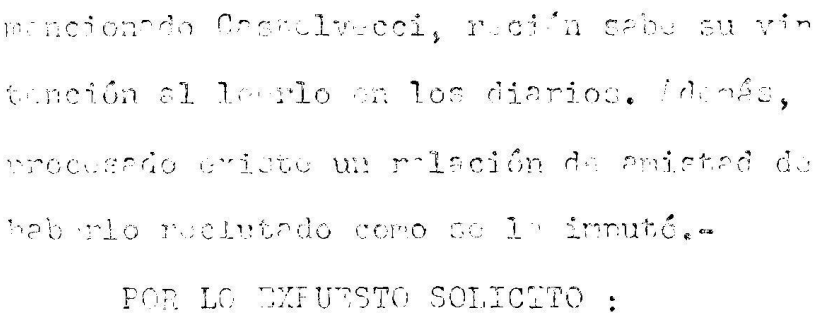


Figura 110 - Resultado de fijar el sesgo en 50 aplicado a la fig. 33

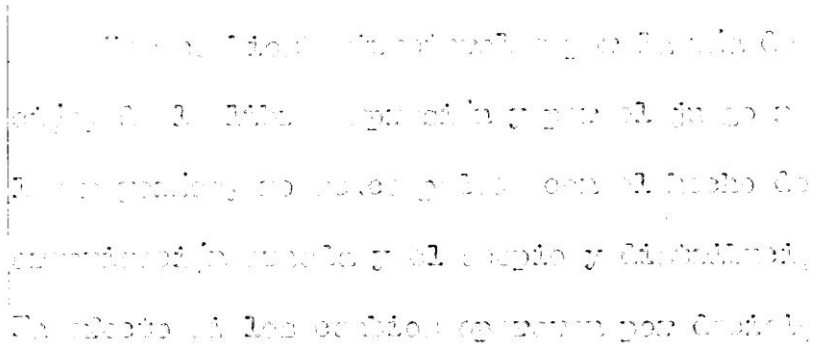


Figura 111 - Resultado de fijar el sesgo en 50 aplicado a la fig. 34

Para obtener las cotas inferior y superior asociadas al sesgo que van a delimitar al valor que mejor se adapte a las imágenes de los expedientes, se compararon los textos obtenidos producto de la aplicación de la herramienta OCR seleccionada en el punto

(sección 4.2) de forma de poder observar que archivo producía una mejor extracción de información.

A continuación se muestran los resultados obtenidos a partir de la ejecución de las pruebas antes presentadas:

Imagen	Sesgo 5	Sesgo 10	Sesgo 20	Sesgo 30	Sesgo 40	Sesgo 50
009_LC	0	0	0	1	0	0
016_RC	0	0	0	0	1	0
022_RC	0	0	1	1	0	0
039_L	0	0	0	0	1	0
215_L	0	0	0	1	0	0
338_R	0	0	0	0	1	0
Total	0	0	**1**	**3**	**3**	0

Tabla 22 - Comparativa de los distintos valores de sesgo

La tabla anterior muestra el total de veces que cada valor de sesgo probado obtuvo resultados mejores o iguales que el resto. Si el valor que se encuentra en una determinada celda es uno significa que ese valor de sesgo obtuvo resultados mejores o iguales que el resto para esa imagen que se está examinando, en caso contrario el valor es cero.
Dado que los valores que mejor desempeño tuvieron fueron 20, 30 y 40, se define como cota inferior y superior para el sesgo los valores 20 y 40 respectivamente.

Análisis del parámetro de negación

En el análisis de este último parámetro se muestra cómo varía la imagen original según si se niega o no la imagen sobre la que se está trabajando antes y después del tratamiento. Para poder realizar un mejor análisis sobre los resultados obtenidos, a cada imagen producto de la transformación realizada se le aplicó OCR de forma de poder observar sobre cuál de ellas se realizaba una mejor extracción de datos.

A continuación se muestran las pruebas realizadas para el análisis de este parámetro.

Prueba 1

Esta prueba consiste en tomar el conjunto de imágenes seleccionadas y aplicarles la herramienta Localtresh fijando el parámetro negar (-n) en sí. El resto de los parámetros se cargan con sus valores por defecto.

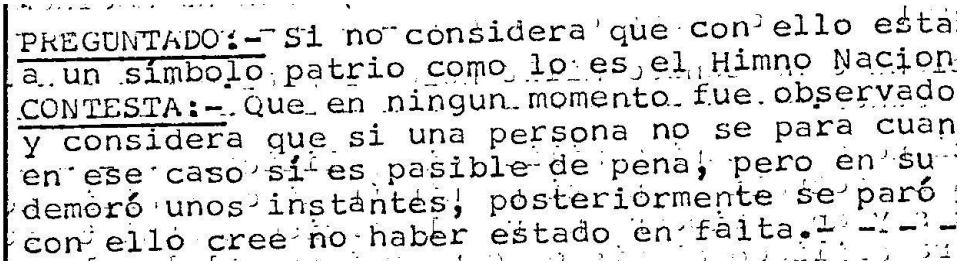

PREGUNTADO:- Si no considera que con ello esta
a un símbolo patrio como lo es el Himno Nacion
CONTESTA:- Que en ningun momento fue observado
y considera que si una persona no se para cuan
en ese caso sí es pasible de pena, pero en su
demoró unos instantes, posteriormente se paró
con ello cree no haber estado en falta.-

Figura 112 - Resultado de fijar negar en sí sobre la fig. 29

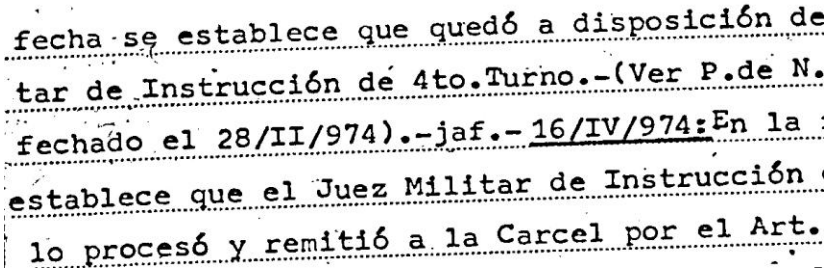

fecha se establece que quedó a disposición de
tar de Instrucción de 4to.Turno.-(Ver P.de N.
fechado el 28/II/974).-jaf.- 16/IV/974:En la
establece que el Juez Militar de Instrucción
lo procesó y remitió a la Carcel por el Art.

Figura 113 - Resultado de fijar negar en sí sobre la fig. 30

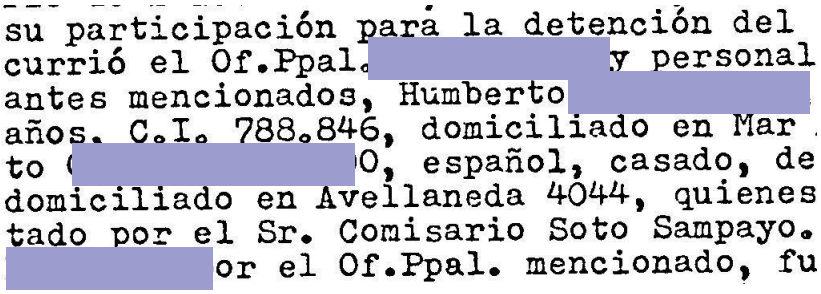

su participación para la detención del
currió el Of.Ppal. y personal
antes mencionados, Humberto
años. C.I. 788.846, domiciliado en Mar
to 0, español, casado, de
domiciliado en Avellaneda 4044, quienes
tado por el Sr. Comisario Soto Sampayo.
or el Of.Ppal. mencionado, fu

Figura 114 - Resultado de fijar negar en sí sobre la fig. 31

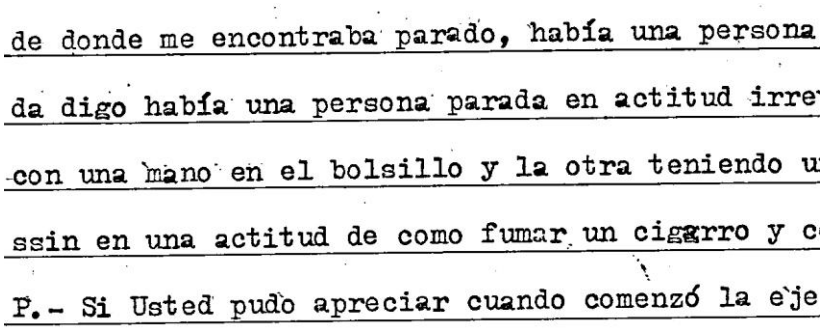

de donde me encontraba parado, había una persona
da digo había una persona parada en actitud irre
con una mano en el bolsillo y la otra teniendo u
ssin en una actitud de como fumar un cigarro y c
P.- Si Usted pudo apreciar cuando comenzó la eje

Figura 115 - Resultado de fijar negar en sí sobre la fig. 32

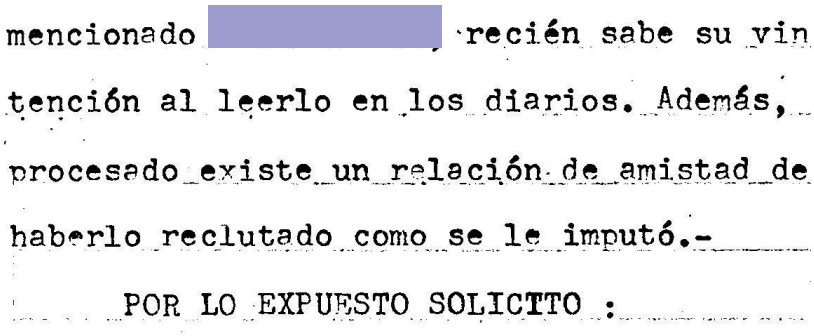

mencionado recién sabe su vin
tención al leerlo en los diarios. Además,
procesado existe un relación de amistad de
haberlo reclutado como se le imputó.-
POR LO EXPUESTO SOLICITO :

Figura 116 - Resultado de fijar negar en sí sobre la fig. 33

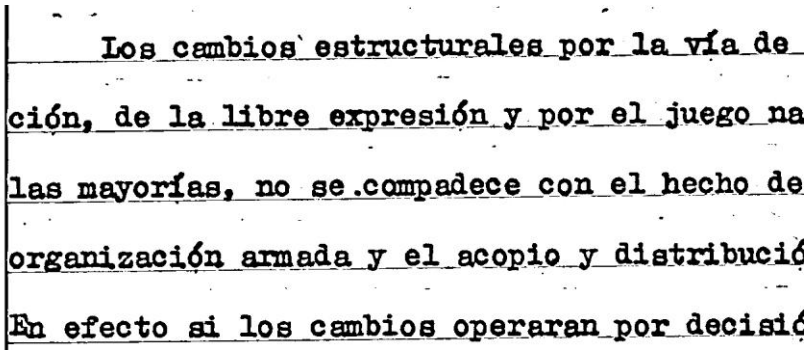

Los cambios estructurales por la vía de
ción, de la libre expresión y por el juego na
las mayorías, no se compadece con el hecho de
organización armada y el acopio y distribuci
En efecto si los cambios operaran por decisi

Figura 117 - Resultado de fijar negar en sí sobre la fig. 34

Prueba 2

Esta última prueba es inversa a la realizada en el punto anterior, es decir, se toma el conjunto de imágenes seleccionadas y se les aplica Localtresh fijando el parámetro negar

(-n) en no. El resto de los parámetros que recibe el script se cargan con sus valores por defecto.

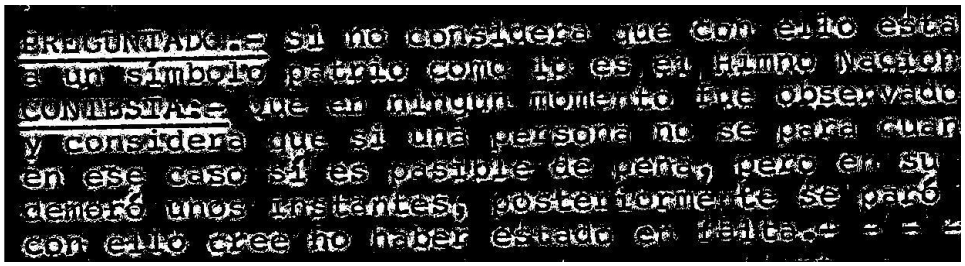


Figura 118 - Resultado de fijar negar en no sobre la fig. 29

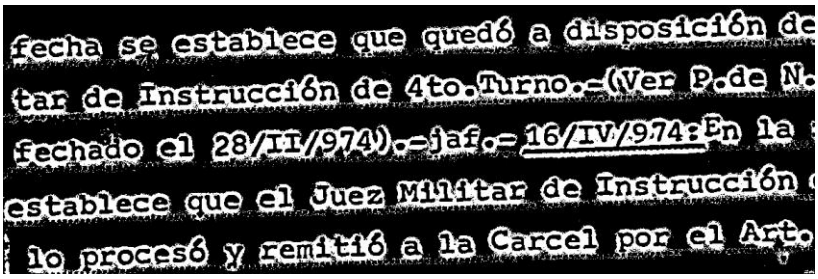


Figura 119 - Resultado de fijar negar en no sobre la fig. 30

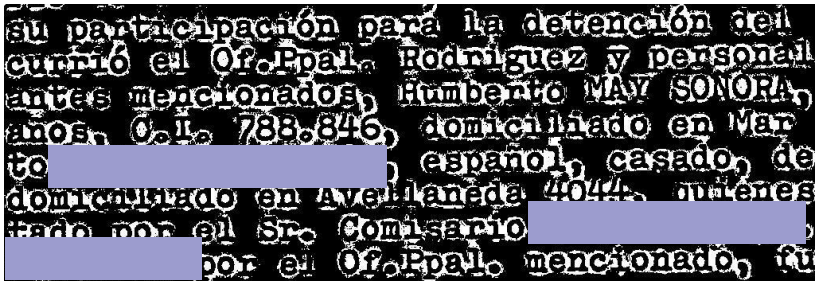


Figura 120 - Resultado de fijar negar en no sobre la fig. 31

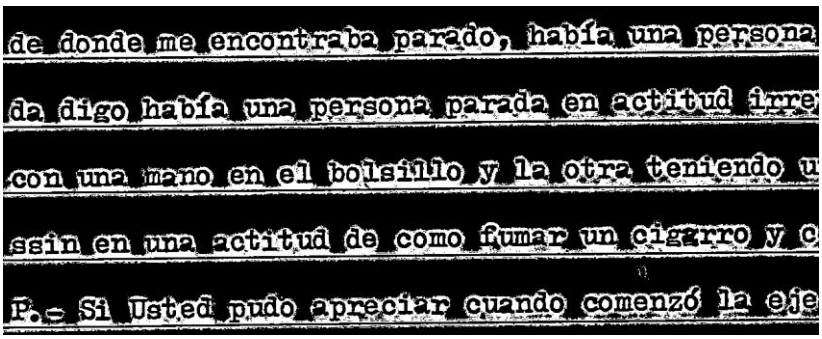


Figura 121 - Resultado de fijar negar en no sobre la fig. 32

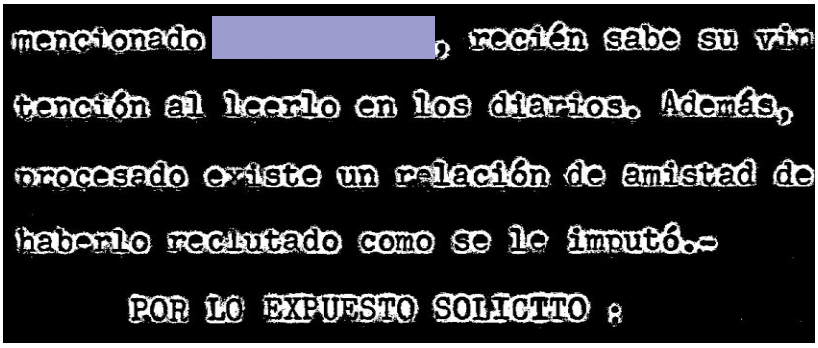


Figura 122 - Resultado de fijar negar en no sobre la fig. 33

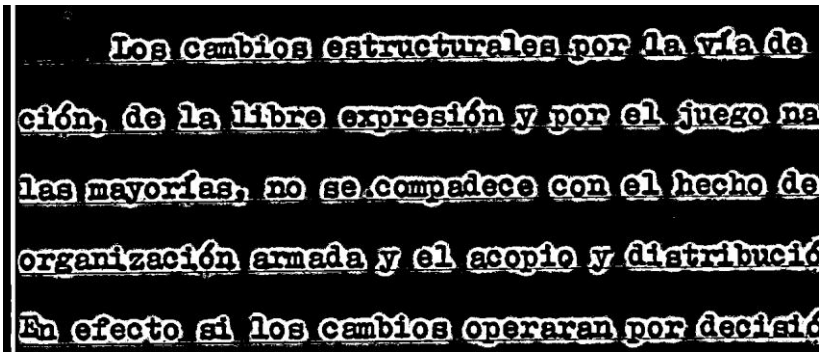


Figura 123 - Resultado de fijar negar en no sobre la fig. 34

A partir de las imágenes antes presentadas se puede concluir que el valor que mejores resultados genera sobre las imágenes de muestra es “sí”, no obstante, para verificar esto se examinaron los resultados obtenidos de forma de poder comparar sobre que archivos se realizaba una mejor extracción de datos.

A continuación se muestran los resultados obtenidos luego de ejecutadas las pruebas:

Imagen	Negar sí	Negar No
009_LC	1	0
016_RC	1	0
022_RC	1	0
039_L	1	0
215_L	1	0
338_R	1	0
Total	**6**	0

Tabla 23 - Comparativa de los valores de negación

La tabla 23 muestra el total de veces que un valor seleccionado para el parámetro que se está examinando supera al resto. Si el valor que se encuentra en una determinada celda es uno significa que ese valor elegido para el parámetro superó al resto en la imagen que se está examinando, en caso contrario se coloca el valor cero. Como se observa en la tabla, el valor que mejores resultados obtiene sobre las distintas imágenes de muestra es "sí".

Conclusiones

Debido al análisis antes expuesto se puede concluir que los parámetros método y negación no requieren de un estudio posterior en el que se halle el valor óptimo para ellos ya que éste se obtuvo en este Anexo. En cambio para los parámetros radio y sesgo se obtuvieron cotas inferiores y superiores que delimitaran la búsqueda de los valores óptimos de forma de reducir el abanico de valores posibles que pueden tomar estos parámetros. La búsqueda de valores óptimos se presenta en el Anexo 3.

Anexo 3 - Obtención de configuraciones óptimas para herramientas de pre procesamiento

En este Anexo se presentan un conjunto de pruebas realizadas sobre una selección representativa de imágenes pertenecientes a expedientes del proyecto AJPROJUMI con el objetivo de encontrar para cada herramienta de pre procesamiento la configuración de parámetros que mejor se adapte a las imágenes de muestra.

Análisis Scan Tailor

Al igual que en el Anexo 2 en esta sección se presenta una breve mención acerca de las capacidades de la herramienta así como la descripción de sus parámetros y las pruebas realizadas sobre esta.

Scan Tailor[3] [Ref.: **Scan Tailor**] es un programa de software libre diseñado para poder editar y procesar a posteriori cualquier documento escaneado. Algunas de las funcionalidades que provee esta herramienta son la eliminación de ruido, la remoción de bordes, la división en páginas del documento, el enderezamiento de la imagen, etc. Estas acciones se pueden ejecutar de forma interactiva, a través de una interfaz gráfica, o de forma automática a través de un script. Si la ejecución de la herramienta se realiza de forma automática se pueden configurar un conjunto de parámetros los cuales se describen a continuación:

- **Content-Detection** (Detección del contenido): Determina cual es la región rectangular con contenidos "útiles" o utilizables. Según el valor definido, esta detección dejará o no más porciones del archivo escaneado fuera del contenido detectado como utilizable. Los posibles valores que puede tomar este parámetro son: cautious, normal y aggressive. Su valor por defecto es normal.

- **Color-Mode** (Color del modo): Indica el modo a ser aplicado a los archivos escaneados. Los archivos resultados pueden ser imágenes en blanco y negro, imágenes en tonos de grises o imágenes de medios tonos (escala de grises o color). Este parámetro puede tomar los valores: black_and_white, color_grayscale y mixed. Su valor por defecto es black_and_white.

- **despeckle** (Reducción de ruido): El valor de este parámetro indica el grado de eliminación de ruido que se va a aplicar o no sobre el archivo escaneado. Los valores que puede tomar este parámetro son: off, cautious, normal y aggressive. Su valor por defecto es normal.

[3] Versión: 0.9.10

Además de los parámetros descritos anteriormente existe otro grupo en los que se optó por que éstos tomaran sus valores por defecto. Los más destacados dentro de este grupo son:

- **Layout-Direction** (Definición de orientación): Define qué orientación va a poseer el archivo. Los posibles valores para este parámetro son: lr (izquierda-derecha) o rl (derecha-izquierda). El valor por defecto es lr.

- **Deskew** (Enderezar): El valor de este parámetro indica si la corrección de orientación o alineamiento de la imagen se va a realizar de forma manual o automática. Los valores que puede tomar este parámetro son: auto y manual. Su valor por defecto es auto.

- **Output-Dpi** (Resolución del archivo resultado): Indica que valor de resolución va a tener el archivo resultado. Su valor por defecto es 600 dpi.

Método de evaluación

Para hallar la configuración que mejor se adaptaba a las imágenes de los expedientes del proyecto AJPROJUMI se llevó a cabo el procedimiento explicado a continuación:

- Definir un conjunto de pruebas con el fin de obtener cuál de las configuraciones analizadas se adapta mejor a las imágenes de los expedientes. La construcción de éstas se llevó a cabo combinando los posibles valores que podían tomar los parámetros antes descritos, con la salvedad de dejar fuera aquellos parámetros que fueron fijados con sus valores por defecto. A continuación se muestra la tabla que contiene la definición de las pruebas realizadas:

Nro. de Prueba	Content-detection	Color-mode	Despeckle
1	cautios	black_and_white	off
2	cautios	black_and_white	cautios
3	cautios	black_and_white	normal
4	**cautios**	**black_and_white**	**aggressive**
5	cautios	color_grayscale	off
6	cautios	color_grayscale	cautious
7	cautios	color_grayscale	normal
8	cautios	color_grayscale	aggressive
9	cautios	Mixed	off
10	cautios	Mixed	cautious
11	cautios	Mixed	normal
12	cautios	Mixed	aggressive
13	normal	black_and_white	off
14	normal	black_and_white	cautious
15	normal	black_and_white	normal
16	normal	black_and_white	aggressive
17	normal	color_grayscale	off
18	normal	color_grayscale	cautious
19	normal	color_grayscale	normal
20	normal	color_grayscale	aggressive
21	normal	mixed	off
22	normal	mixed	cautious
23	normal	mixed	normal
24	normal	mixed	aggressive
25	aggressive	black_and_white	off
26	aggressive	black_and_white	cautious
27	aggressive	black_and_white	normal
28	aggressive	black_and_white	aggresive
29	aggressive	color_grayscale	off
30	aggressive	color_grayscale	cautious
31	aggressive	color_grayscale	normal
32	aggressive	color_grayscale	aggressive
33	aggressive	mixed	off
34	aggressive	mixed	cautious
35	aggressive	mixed	normal
36	aggressive	mixed	normal

Tabla 24 - Pruebas Scan Tailor

- Ejecutar cada una de las pruebas antes definidas sobre cada una de las imágenes de la muestra.
- Ejecutar Tesseract sobre cada archivo obtenido en el punto anterior.
- Aplicar Tesseract sobre las imágenes originales.

- Comparar el archivo digitalizado correspondiente a cada imagen original versus el archivo digitalizado producto de la ejecución de cada prueba. Luego de realizado este análisis se descartan todas aquellas configuraciones que no presentan mejoras.

- Comparar las configuraciones obtenidas en el punto anterior, salvo las descartadas, de forma de seleccionar aquella que se adapta mejor al tipo de imágenes de los expedientes.

Resultados

En esta sección se muestran los resultados obtenidos a partir de la ejecución del procedimiento descrito en la sección anterior así como la conclusión alcanzada.

La siguiente tabla muestra la comparación que se realizó considerando únicamente aquellas configuraciones en las que se obtuvieron mejores resultados que los obtenidos a partir de la imagen original.

Imagen	Características de la Imagen	Mejor Config. (Nro. de Prueba)
009_LC	➢ Sin renglones ➢ Con inclinación	4, 9, 16, 21, 28 y 33
016_RC	➢ Con renglones ➢ Con gran inclinación	4, 9, 16, 21, 28 y 33
022_RC	➢ Sin renglones ➢ Con inclinación	4, 9, 16, 21, 28 y 33
039_L	➢ Con renglones ➢ Con pequeña inclinación	5, 8, 17, 20, 29 y 32
215_L	➢ Sin renglones ➢ Sin inclinación	4, 9, 16, 21, 28 y 33
338_R	➢ Sin renglones ➢ Con pequeña inclinación	4, 9, 16, 21, 28 y 33

Tabla 25 - Comparativa de configuraciones para Scan Tailor

Como se observa en la tabla anterior las configuraciones que obtuvieron mejores resultados fueron la 4, 9, 16, 21, 28 y 33. Dado que estas arrojaron resultados idénticos se optó por seleccionar aquella que transformará el archivo de entrada en una imagen binaria, blanco y negro, debido a que la herramienta OCR seleccionada para la digitalización (sección 4.2) es más efectiva sobre ese tipo de imágenes. Otras de las razones que condujeron a la elección de esa configuración fueron la selección de contenido y la eliminación de ruido. Debido a que las imágenes de este proyecto presentan un gran deterioro se eligió aplicar una fuerte eliminación de manchas y una cautelosa detección de contenido.

Análisis LocalTresh

En esta sección se presentarán las pruebas realizadas con esta herramienta, la metodología empleada para el análisis de las mismas y las conclusiones alcanzadas. Dado que este software fue explicado en detalle en el Anexo 2 aquí simplemente se expondrán las pruebas realizadas con el mismo.

Método de evaluación

Para la creación del esquema de pruebas se llevaron a cabo ciertos pasos con el fin de obtener la configuración que mejor se adaptaba a las imágenes de los expedientes del proyecto y de forma de que el tiempo y los casos de prueba no se extendieran de forma indefinida.
Los pasos llevados a cabo para el hallazgo de la configuración que mejor se adapta a las imágenes de los expedientes fueron los siguientes:

- Como primer punto se definió un conjunto de valores para los parámetros sesgo y radio de forma tal de abarcar el rango definido por las cotas superiores e inferiores obtenidas en el Anexo 2, eligiendo valores lo suficientemente distantes como para tener un número de combinaciones manejable al momento del análisis.

  El conjunto de valores definidos para los parámetros antes mencionados fueron los siguientes:

Parámetro	Valores
Radio (-r)	18, 24, 30, 36, 42, 48
Sesgo (-b)	20, 25, 30, 35, 40

Tabla 26 - Valores para sesgo y radio

- Definición del set de pruebas utilizadas para el hallazgo de la tupla de valores que mejor se adapta a las imágenes del proyecto. Estas pruebas se construyeron combinando los posibles valores de cada uno de los parámetros de la herramienta.

  A continuación se presenta un esquema conteniendo las pruebas realizadas:

Nro. de Prueba	Método	Radio	Sesgo	Negar
1	3	18	20	sí
2	3	18	25	sí
3	3	18	30	sí
4	3	18	35	sí
5	3	18	40	sí
6	3	24	20	sí
7	3	24	25	sí
8	3	24	30	sí
9	3	24	35	sí
10	3	24	40	sí
11	3	30	20	sí
12	3	30	25	sí
13	3	30	30	sí
14	3	30	35	sí
15	3	30	40	sí
16	3	36	20	sí
17	3	36	25	sí
18	3	36	30	sí
19	3	36	35	sí
20	3	36	40	sí
21	3	42	20	sí
22	3	42	25	sí
23	3	42	30	sí
24	**3**	**42**	**35**	**sí**
25	3	42	40	sí
26	3	48	20	sí
27	3	48	25	sí
28	3	48	30	sí
29	3	48	35	sí
30	3	48	40	sí

Tabla 27 - Pruebas Localtresh.

- Ejecución de cada una de las pruebas definidas en el punto anterior sobre cada una de las imágenes de la muestra.

- Ejecución del motor de OCR Tesseract sobre cada archivo obtenido en el punto anterior.

- Digitalización de las imágenes originales a través del motor de OCR Tesseract.

- Comparación del archivo digitalizado correspondiente a cada imagen original versus el archivo digitalizado producto de la ejecución de cada prueba. Luego de realizado este análisis se descartan todas aquellas configuraciones que no presentan mejoras.

- Se comparan las configuraciones obtenidas en el punto anterior, salvo las descartadas, de forma de seleccionar aquella que se adapta mejor al tipo de imágenes de los expedientes.

Resultados

En esta sección se muestran los resultados obtenidos de la ejecución del procedimiento descrito en la sección anterior así como la conclusión alcanzada.

La siguiente tabla muestra la comparación realizada tomando únicamente aquellas configuraciones en las que se obtuvieron mejores resultados que los obtenidos a partir de la imagen original.

Imagen	Características de la Imagen	Mejor Config. (Nro. de Prueba)
009_LC	➢ Sin renglones ➢ Con inclinación	24
016_RC	➢ Con renglones ➢ Con gran inclinación	---
022_RC	➢ Sin renglones ➢ Con inclinación	2
039_L	➢ Con renglones ➢ Con pequeña inclinación	24
215_L	➢ Sin renglones ➢ Sin inclinación	28
338_R	➢ Sin renglones ➢ Con pequeña inclinación	24

Tabla 28 - Comparativa de configuraciones para Localtresh

Como se observa en la tabla la configuración que obtuvo mejores resultados fue la 24 la cual está compuesta por los siguientes valores:

Parámetro	Valores
Método	3
Radio	42
Sesgo	35
Negado	yes

Tabla 29 - Valores configuración optima

Para ver la comparación entre ambas herramientas ir al capítulo **4.1.4**

Anexo 4 - Módulos de Freeling

FreeLing [Ref:Freeling] es una librería de código abierto para el procesamiento multilingüe automático, que proporciona una amplia gama de servicios de análisis lingüístico para diversos idiomas.
Ofrece a los desarrolladores de aplicaciones de Procesamiento de Lenguaje Natural funciones de análisis y anotación lingüística de textos, con la consiguiente reducción del coste de construcción de dichas aplicaciones.

La versión 3.0 provee varios módulos de análisis, cada uno de los cuales brinda distintas funcionalidades de procesamiento de texto, algunos ejemplos de estos módulos son:

- Módulo Identificador de lenguaje.
- Módulo de Tokenización.
- Módulo de detección de sentencias (Splitter).
- Módulo de detección de números.
- Módulo de detección de puntuación.
- Módulo de reconocimiento de multipalabras
- Módulo de reconocimiento de entidades con nombre (NER).
- Módulo de clasificación de entidades con nombre (NEC).

Para la implementación del programa desarrollado (“AjprojumiAnalzer”) se utilizaron algunos de estos módulos, a continuación se describirán en detalle cada uno de ellos:

Módulo de Tokenización

Este módulo convierte texto plano en un vector de palabras de acuerdo a un conjunto de reglas de tokenización definidas. Estas reglas son expresiones regulares que son emparejadas (“matcheadas”) con el comienzo de la línea de texto que se procesa. La primera regla que coincida es usada para extraer el token, el mismo es borrado de la línea y luego se repite el proceso hasta que ésta sea vacía.

API del módulo:

```
Class Tokenizer{
Public:
      ///Constructor, recibe el nombre del archivo que contiene las
reglas de tokenizacion
      Tokenizer (const std::string &);
///Tokeniza el string con las opciones por defecto
      Std::list <word> tokenize (const std::String &)
      ///Tokeniza el string con las opciones por defecto, acumula el
byte-offset de palabras
      Std::list<word> tokenize (const std::String &, unsigned long &)
};
```

Este módulo es usado para tokenizar el texto y obtener la lista de palabras contenidas en el. Esta lista será la entrada del módulo de detección de sentencias para realizar la separación del texto en oraciones.
El archivo de configuración pasado como parámetro al constructor de este módulo fue modificado tal como se indica en el Anexo 5 - sección Freeling.

Módulo de detección de oraciones (Splitter)

El módulo de detección de oraciones recibe una lista de palabras (p. ej. salida del módulo tokenizer) y las almacena hasta encontrar un indicador de límite de oración. Al finalizar el proceso se retorna una lista de sentencias.

API del módulo:

```
class splitter {
public:
      /// Constructor. Recibe un archivo con las opciones deseadas
      Splitter(conststd::string&);
      /// Añade una lista de palabras al buffer, y retorna sentencias
      completas que pueden ser construidas
      El estado booleano indica si la salida del buffer debe ser forzada
      (true) o si algunas palabras pueden permanecer en el buffer (false)
      si el splitter quiere esperar a ver lo que viene a continuación
      std::list<sentence>split(const std::list<word>&, bool);
};
```

El archivo de configuración pasado como parámetro al constructor de este módulo fue modificado tal como se indica en el Anexo 4 - sección Freeling.

Módulo analizador morfológico

El módulo analizador morfológico es un meta-módulo que no realiza ningún procesamiento por sí mismo. Es un módulo utilizado para simplificar la instanciación y el llamado a los sub-módulos (módulo de detección de fechas, módulo de reconocimiento de entidades con nombre, etc.).
Cuando este módulo es instanciado, recibe un objeto de tipo “maco_options” conteniendo información sobre que sub-módulos deben ser creados y sobre que archivos de configuración deben ser usados para crearlos.

API del módulo:

```
class maco {
public:
      /// Constructor. Recibe un conjunto de opciones
      maco(const maco_options &);
      /// Analiza la sentencia pasada como parámetro
      voidanalyze(sentence &);
      /// Analiza una lista de sentencias
      void analyze(std::list<sentence>&);
      /// Retorna una copia del análisis de la sentencia dada
      sentenceanalyze(const sentence &);
      /// Retorna una copia del análisis de la lista de sentencias dada
      std::list<sentence>analyze(const std::list<sentence>&);
};
```

En “ajprojumiAnalyzer” se utilizó un objeto maco_options en el cual se estableció la utilización de los módulos:

- Reconocimiento de multipalabras
- Detección de fechas
- Reconocimiento de entidades con nombre
- Clasificación de entidades con nombre

Módulo de detección de fechas

Este módulo es una colección de autómatas de estado finito y es lenguaje-dependiente. Al momento de instanciar éste el único parámetro necesario es el lenguaje a utilizar.

API del módulo:

```
class dates {
public:
      /// Constructor: Recibe el código del lenguaje (en nuestro caso
      “es”)
      dates(const std::string &);
      /// Analiza la sentencia especificada como parámetro
      void analyze(sentence &);
      /// Analiza una lista de sentencias
      void analyze(std::list<sentence>&);
      /// Retorna una copia del análisis de la sentencia dada
      sentence analyze(const sentence &);
      /// Retorna una copia del análisis de la lista de sentencias dada
      std::list<sentence>analyze(const std::list<sentence>&);
      };
```

Módulo de reconocimiento de multipalabras

La función principal de este módulo es agrupar tokens y representarlos como un único objeto palabra.

API del módulo:

```
class locutions: public automat {
public:
      /// Constructor, recibe el nombre del archivo que contiene las
      multipalabras a reconocer
      locutions(const std::string &);
      /// Detecta multipalabras a partir de una posición de la sentencia
dada
      bool matching(sentence &, sentence::iterator &);
      /// Analiza la sentencia especificada como parámetro
      void analyze(sentence &);
      /// Analiza las sentencias especificadas como parámetro
      void analyze(std::list<sentence>&);
      /// Retorna una copia del análisis de la sentencia dada
      sentence analyze(const sentence & );
      /// Retorna una copia del análisis de la lista de sentencias dada
      std::list<sentence>analyze(const std::list<sentence>&);
};
```

El archivo pasado como parámetro al constructor de este módulo contiene la lista de multipalabras a ser reconocidas por el analizador. El formato es de una multipalabra por línea con el siguiente formato:

```
form lema1 pos1 lema2 pos2 … [ A | I ]
```

Cualquier número de pares lema-etiqueta pueden ser asignados a la multipalabra, el módulo tagger (explicado en este mismo anexo - sección “Módulo Tagger”) seleccionará el más apropiado, al igual que con cualquier otra palabra.
El último campo especifica si la multipalabra es ambigua o no (puede ser una multipalabra o no dependiendo del contexto)

El archivo con las multipalabras utilizado en este módulo se encuentra especificado en el Anexo 5 - Sección Freeling.

Módulo de reconocimiento de entidades con nombre (NER)

Básicamente la tarea de este módulo es detectar secuencias de palabras capitalizadas (el comienzo de la misma es en mayúsculas y el resto en minúsculas) teniendo en cuenta determinado conjunto de palabras funcionales y capitalización de palabras en comienzos de oración.

API del módulo:

```
class np: public ner_module, public automat {
public:
      /// Constructor, recibe el archivo de configuración.
      np(conststd::string&);
      /// Detecta multipalabras a partir de la posición de oración dada
      bool matching(sentence &, sentence::iterator &);
      /// Analiza la sentencia especificada como parámetro
      void analyze(sentence &);
      /// Analiza las sentencias especificadas como parámetro
      void analyze(std::list<sentence>&);
      /// Retorna una copia analizada de la oración dada
      sentenceanalyze(constsentence&);
      /// Retorna una copia analizada de las oraciones dadas
      std::list<sentence>analyze(conststd::list<sentence>&);
};
```

El archivo de configuración especificado como parámetro del constructor consiste en varias secciones dentro de las que se encuentran:

- "Function Words": palabras que pueden ser encontradas embebidas dentro del nombre propio (p. ej.: Banco **de** España, Río **de la** Plata, etc.).
  Ejemplo:
  <FunctionWords>
  de
  la
  </FunctionWords>

- "SpecialPunct": es la lista de POS tags (ver sección "Módulo Tagger" de este anexo) después de los cuales una palabra capitalizada puede ser simplemente una sentencia o cláusula y no una entidad con nombre. Los casos más comunes son coma, punto, apertura de paréntesis, etc.

Ejemplo:
<SpecialPunt>
Fpa
Fp
Fpa
Fd
Fg
</SpecialPunct>

- "Ignore": contiene una lista de palabras que no son consideradas como entidades con nombre aunque aparezcan capitalizadas en la mitad de una sentencia
- "Names": contiene una lista de lemas que pueden ser nombres aunque también pueden tener otro significado.
  Ejemplo:
  <Names>
  pelé
  miren
  </Names>
- "Affixes": contiene una lista de palabras que pueden ser parte de una entidad con nombre, tanto sufijo como prefijo.
  Ejemplo:
  <Affixes>
  don PRE
  doña PRE
  jr. SUF
  </Affixes>

Nota: Para este módulo se usó el archivo de configuración por defecto de Freeling.

Módulo de clasificación de entidades con nombre (NEC)

El objetivo de este módulo es el de asignar una clase a las entidades con nombre reconocidas previamente por el módulo NER.
Cuando se clasifican las palabras, el tag es cambiado a la etiqueta de la clase correspondiente.

Las clases definidas en Freeling son:

- Personas (NP00SP00)
- Localizaciones geográficas (NP00G00)
- Organizaciones (NP00O00)
- Otros (NP00V00)

API del módulo:

```
class nec{
public:
	/// Constructor
	nec(conststd::string &);
	/// Analiza la sentencia especificada como parámetro
	void analyze(sentence &);
	/// Analiza las sentencias especificadas como parámetro
	void analyze(std::list<sentence>&);
	/// Retorna una copia analizada de la oración dada
	sentence analyze(const sentence &);
	/// Retorna una copia analizada de las oraciones dadas
	std::list<sentence>analyze(conststd::list<sentence>&);
};
```

El archivo de configuración utilizado en el constructor fue el por defecto.
La principal funcionalidad utilizada de este módulo fue la de clasificación de personas para la identificación de nombres en los expedientes analizados.

Módulo tagger (Part-of-Speech Tagger)

El módulo de tagger es el encargado de asignar los POS-tags (Part-Of-Speech tags) a cada palabra del texto analizado.
Existen dos módulos disponibles para realizar el POS-tagging. El primero es el hmm_tagger, el cual utiliza la técnica de trigramas Markovianos[4]. El segundo es el llamado relax-tagger, el cual es un híbrido ya que aparte de información estática puede ser modificado por el usuario. El primero es en general de más rapidez pero la ventaja del segundo de ellos es que permite agregar restricciones al modelo.
Debido a que no es necesaria ninguna restricción en particular optamos por utilizar el primero de los módulos:

[4] Un modelo de N-gramas intenta predecir la próxima palabra de una oración a partir de las N-1 anteriores.

API del módulo hmm_tagger:

```
class hmm_tagger: public POS_tagger {
public:
      /// Constructor
      hmm_tagger(const std::string &, const std::string &, bool, unsigned
      int);
      /// Analiza la sentencia especificada como parámetro
      void analyze(sentence &);
      /// Analiza las sentencias especificadas como parámetro
      void analyze(std::list<sentence>&);
      /// Retorna una copia analizada de la oración dada
      sentence analyze(const sentence &);
      /// Retorna una copia analizada de las oraciones dadas
      std::list<sentence>analyze(const std::list<sentence>&);
};
```

El constructor de este módulo posee los siguientes parámetros:

- Código del lenguaje (“es” para Español).
- El archivo HMM (Hidden Markov Model), el cual contiene los parámetros del modelo. Este archivo contiene los datos estáticos para el modelo Markoviano, además de probabilidades iniciales, probabilidades de transición, probabilidades léxicas, etc.

  Contiene siete secciones: <Tag>, <Bigram>, <Trigram>, <Initial>, <Word>, <Smoothing> y <Forbidden> dentro de las cuales se destacan las primeras tres en donde se definen las probabilidades de los unigramas, de las transiciones entre los bigramas y de las transiciones entre trigramas.
- Booleano que indica si las palabras que tienen información de retokenización (ej: seteadas por el diccionario) deben ser retokenizadas o no.
- Entero que indica si el tagger debe seleccionar solo un tipo de análisis en caso de ambigüedad. Valores posibles:
  - 0 - FORCE_NONE
  - 1 - FORCE_TAGGER
  - 2 - FORCE_RETOK.

Anexo 5 - Instalación de la solución

En este anexo se detallan los pasos para configurar las herramientas utilizadas para la solución brindada. Se utilizó la herramienta Scan Tailor para el pre procesamiento de imágenes, Tesseract como motor OCR y Freeling para el post procesamiento.
También se detalla la instalación y configuración de Django y la aplicación web "Ajprojumi Web".

Plataforma utilizada

Para la realización de este proyecto se utilizó la distribución de Linux: Ubuntu 11.10 (Oneiric Ocelot).
Si bien todas las herramientas son de Software Libre, el nombre de los paquetes utilizados o algún paso de configuración puede cambiar levemente.

Scan Tailor

Scan Tailor es la aplicación que se utiliza para realizar el pre procesamiento de las imágenes. Su instalación es simple ya que se encuentra en los repositorios de la distribución utilizada, solo se necesita conexión a internet y ejecutar el siguiente comando desde consola:

```
$ sudo apt-get install scantailor
```

Esto instalará la aplicación scantailor y su implementación para línea de comandos "scantailor-cli". La versión utilizada es la **0.9.10.**

Tesseract

La aplicación elegida para la etapa de procesamiento OCR es Tesseract. La versión utilizada para este proyecto es la **3.01**.
Para poder instalar Tesseract hay que instalar algunas dependencias, las mismas se obtienen de los repositorios de Ubuntu:

```
$ sudo apt-get install libpng12-dev libjpeg62-dev libtiff4-dev zlibg-dev
```

Luego se necesita descargar e instalar el paquete Tesseract. Para eso ejecutar los siguientes pasos:

```
$ wget http://tesseract-ocr.googlecode.com/files/tesseract-3.01.tar.gz
$ tar xvzf tesseract-3.01.tar.gz
$ cd tesseract-3.01
$ ./configure
$ make
$ sudo make install
$ sudo ldconfig
$ export TESSDATA_PREFIX=/usr/local/share/tessdata
```

Por último se descarta el paquete del lenguaje español. Para instalarlo se debe ejecutar:

```
$ wget http://tesseract-ocr.googlecode.com/files/spa.traineddata.gz
$ gzip -d spa.traineddata.gz
$ mv spa.traineddata $TESSDATA_PREFIX
```

Freeling

La herramienta utilizada para el procesamiento de texto es Freeling, en su versión **3.0**. Previo a la instalación de Freeling se deben instalar los siguientes paquetes:

```
$ sudo apt-get install libboost-regex-dev libicu-dev
$ sudo apt-get install libboost-filesystem-dev libboost-program-options-
dev
```

Luego se debe descargar el paquete e instalarlo siguiendo los siguientes pasos:

```
$ wget http://devel.cpl.upc.edu/freeling/downloads/16
$ tar xzvf FreeLing-3.0.tar.gz
$ cd freeling-3.0
$ ./configure
$ make
$ sudo make install
```

Para lograr los objetivos descritos en la sección 5.3 del presente informe, fue necesario modificar ciertas configuraciones de Freeling. Dichas modificaciones se detallan a continuación:

Diccionario

Para realizar búsquedas de información en los expedientes, fue necesario contar con determinadas palabras claves etiquetadas de forma única. Esto es necesario ya que con la configuración por defecto y de la manera que Freeling etiqueta las palabras, puede suceder que 2 palabras distintas posean la misma etiqueta.

Para ello fue necesario modificar el diccionario provisto por Freeling ubicado en:

```
ruta_instalacion_freeling/es/dicc.src
```

Las palabras modificadas en dicho archivo con sus correspondientes tags son las siguientes:

Palabra	Tag
procesado/procesada	PROCESADO
procesados/procesadas	PROCESADOS
encausado/encausada	ENCAUSADO
encausados/encausadas	ENCAUSADOS
ciudadano/ciudadana	CIUDADANO
ciudadanos/ciudadanas	CIUDADANOS
testigo	TESTIGO
testigos	TESTIGOS
fiscal	FISCAL
acta	ACTA
sumario	SUMARIO
caratulado	CARATULADO
persona	PERSONA
recluso	RECLUSO
detención	DETENCION
liberado	LIBERADO
libertad	LIBERTAD

Tabla 30 - Palabras agregadas a diccionario

Archivo con multipalabras

En este caso también se necesitaron tags especiales, pero ahora para multipalabras, no palabras simples. Para ello se crea un nuevo archivo (multiword.txt) que es usado por el módulo de reconocimiento de multipalabras.
Dicho archivo debe ser ubicado en el mismo directorio en donde se encuentra el ejecutable "ajprojumiAnalyzer".
Las multipalabras agregadas en este archivo fueron:

Multipalabra	Tag
juez militar de instrucción	JUEZMILITAR
sumario instruido	SUMARIOINSTRUIDO
fue preso	FUEPRESO
autos caratulados	AUTOSCARATULADOS
acta de interrogatorio	ACTA
fecha de detención	FECHADETENCION
fecha de el hecho	FECHAHECHO
fecha de procesamiento	FECHAPROCESAMIENTO
se resuelve	SERESUELVE
datos filiatorios	DATOSFILIATORIOS

Tabla 31 - Palabras agregadas a archivo de conf. de Modulo multipalabras

Archivo de configuración - módulo de reconocimiento de sentencias

El archivo de configuración del módulo de reconocimiento de sentencias también fue modificado, en particular la sección llamada "Sentence End" en la cual se establecen que caracteres son considerados como fin de línea.
El carácter agregado a esta sección fue el guión('-'), ya que al analizar los expedientes se identificó que varias oraciones culminaban con el par de caracteres '.-'. Esto mejoró considerablemente la detección correcta de sentencias, tarea fundamental para reconocer luego los datos buscados en el analizador.

Sección modificada:

```
<SentenceEnd>
. 0
? 0
! 0
- 0
</SentenceEnd>
```

Luego de cada carácter un valor binario es especificado, dicho valor específica si el carácter denota un fin de sentencia sin importar el carácter que se encuentre a continuación. En los expedientes el guión no define un fin de oración en todos los casos por eso se agregó dicho carácter con valor 0.

Archivo de configuración - módulo de tokenización

El archivo de configuración del módulo de tokenización fue modificado agregando nuevas abreviaciones. Esto es necesario ya que de otro modo se pueden tomar los puntos de las abreviaciones como fines de oración obteniendo así sentencias no válidas y por consecuencia dificultando el procesamiento del texto.
Las abreviaciones agregadas fueron las siguientes:

Abreviaciones		
t.	1er.	milit.
tech.	2do.	1o.
tel.	3er.	1a.
teléf.	4to.	2a.
telf.	5to.	3a.
ten.	1o.	fo.
tfono.	2o.	inc.
tít.	3o.	esc.
tlf.	4o.	lo.
tte.	5o.	ldo.
ud.	6o.	va.
uds.	7o.	1ra.
vda.	8o.	ciuds.
vdo.	9o.	c.i.
vid.	art.	vi.
vol.	arts.	ant.
vols.	fs.	v.
vra.	nro.	ss.
vro.	nros.	inst.
vta.	juzg.	exp.

Tabla 32 - Abreviaciones agregadas

Ajprojumi Web

A través de la aplicación web “Ajprojumi Web” se podrá ver el resultado del trabajo desarrollado en este proyecto. Esta aplicación se desarrolló utilizando el framework web Python-Django.

El primer paso para la instalación de Ajprojumi Web es la instalación de Django. La misma se realiza través del paquete python-django disponible en los repositorios de Ubuntu:

```
$ sudo apt-get install python-django
```

La aplicación se encuentra en el archivo ajprojumi_web_1.0.tar.gz, que debe descomprimir en el directorio de instalación. En este caso se eligió el directorio /usr/local/

```
$ cp ajprojumi_web_1.0.tar.gz
$ cd /usr/local/
$ tar xvzf ajprojumi_web_1.0.tar.gz
```

Dentro del directorio /usr/local/ajprojumi_web/ se encontrarán todos los archivos necesarios para la ejecución de la aplicación web. Algunos de esos archivos hay que modificarlos para que el sitio web funcione correctamente:
Dentro del directorio ajprojumi_web editar el archivo settings.py modificando el valor del campo MEDIA_ROOT:

```
MEDIA_ROOT = '/ruta/a/expedientes/'
```

Debido a conflictos en la codificación de caracteres que se generan en los archivos de salida de Tesseract, hay que cambiar la codificación por defecto del lenguaje Python. Para esto hay que editar el archivo /usr/lib/python2.7/site.py y modificar la función **setencoding():**

```
encoding = "iso-8859-1"
```

El último paso es ejecutar la aplicación, la cual correrá en localhost:

```
$ cd ajprojumi_web
$ python manage.py syncdb
$ python manage.py runserver
```

Para acceder a la aplicación abrir un navegador e ingresar a la dirección **http://localhost:8000.**

Cada vez que se quiera correr la aplicación hay que ejecutar el comando

```
$ python manage.py runserver
```

Anexo 6 - La interfaz de administración

Una potente funcionalidad que provee el framework de programación web Django es la interfaz de administración. Para cada clase implementada en una aplicación Django se puede indicar si la misma será accesible desde el panel de administración. A su vez desde este sitio también se pueden gestionar los usuarios que pueden ingresar al sitio, también se pueden dar permisos para acceder a ciertas páginas de la aplicación dependiendo del perfil de usuario. Esta última característica es la que se aprovecha para algunas páginas de ajprojumi web.

Figura 124 - Sitio de administración Django

Ingreso de nuevo usuario

Un usuario administrador puede dar de alta usuarios de consulta en el sitio, para ello debe simplemente cargarlo mediante el formulario de nuevo usuario dentro del sitio de administración.

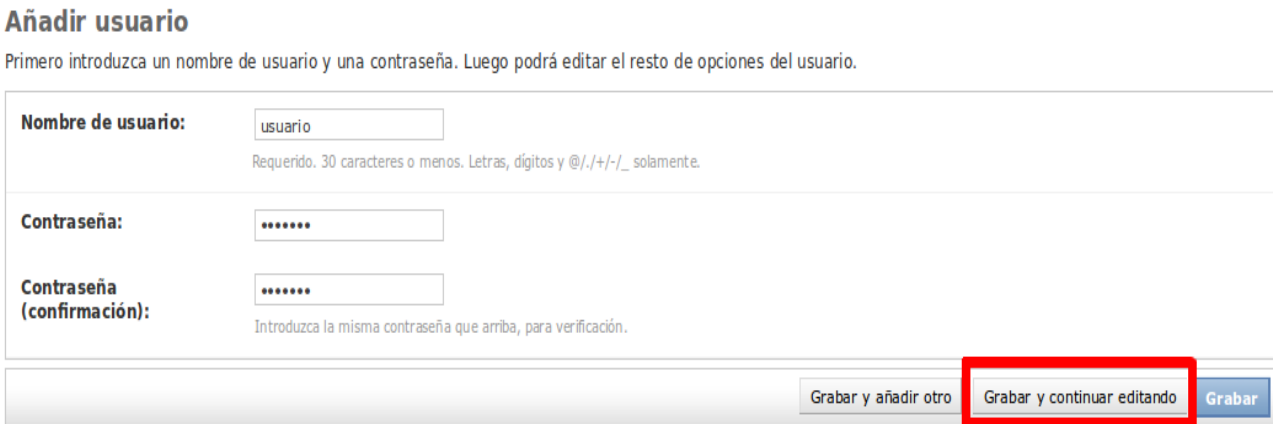


Figura 125 - Formulario nuevo usuario

Como se aprecia en la figura 125 en el formulario de nuevo usuario hay una opción para grabar y otra para grabar y seguir editando. Si se utiliza esta última se pueden editar otros datos del usuario como por ejemplo el nombre.
Si se elige la opción “Grabar y continuar editando” también se puede asignar permisos para la carga de datos en la base, esto es en la sección “Permisos”, debe estar marcada la opción **Activo** y en la sección “Permisos de usuario” elegir el campo “buscador | expediente | Can add expediente” como se muestra en la figura 126.

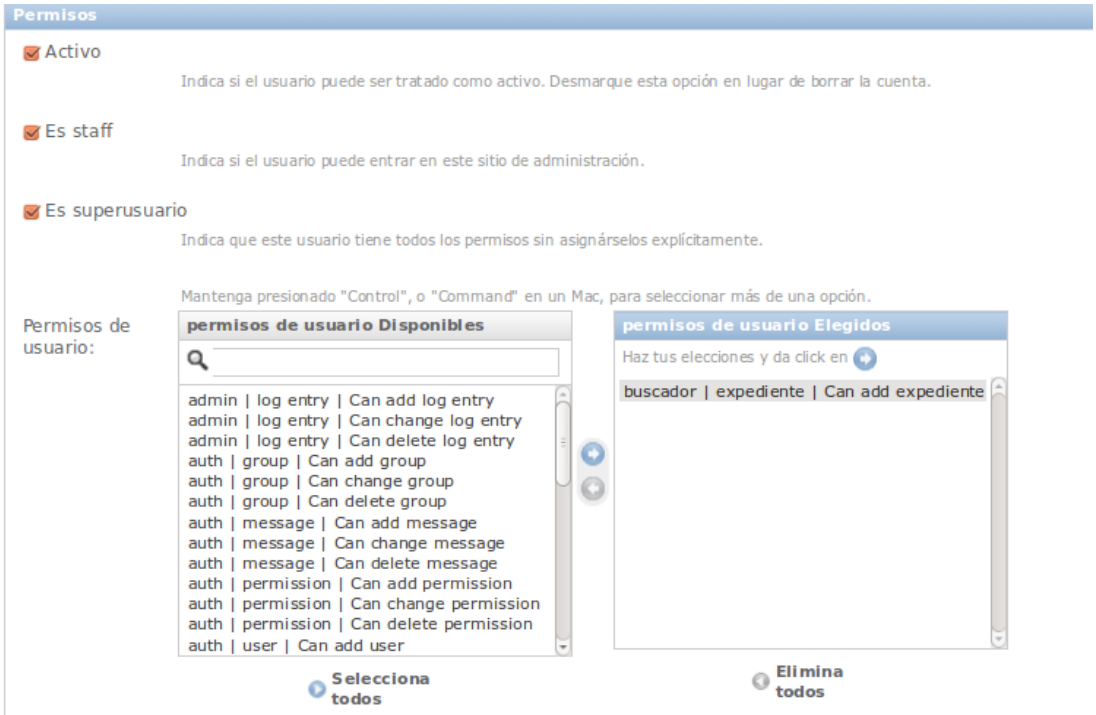


Figura 126 - Permisos de usuario

Anexo 7 - Estructura de expedientes

El objetivo de este anexo es especificar y analizar la estructura general de un expediente modelo, así como también identificar las secciones en las que se encuentran cada una de las palabras claves, necesarias para realizar las búsquedas en los expedientes. Para realizar este análisis se tuvieron en cuenta varios expedientes proporcionados por personal del Poder Judicial para tal motivo.

Si bien cada expediente varía en cuanto a su estructura, se pueden identificar las distintas secciones que los mismos poseen en la mayoría de los casos, estas son:

- Carátula
- Acta de detención
- Acta de declaración (ante juez sumariante)
- Acta de declaración (ante juez de instrucción)
- Decreto de procesamiento o de puesta en libertad
- Designación de defensor
- Acusación fiscal
- Vista a la defensa
- Sentencia definitiva de primera instancia
- Apelación
- Sentencia de segunda instancia del supremo tribunal militar
- Solicitud de libertad
- Decreto de libertad / Resolución de amparo a la ley 15.737 de amnistía

A continuación se detalla cada una de estas secciones, mostrando cómo se lograron identificar dentro de la estructura del expediente, y proporcionando también la lista de palabras clave que se identificaron dentro de la misma.

Carátula

Identificación de la sección: En general primera hoja del expediente.

Características: Contiene datos como ser número de expediente, año, fecha de iniciación, nombre del/los indagados, etc.

Palabras clave identificadas: Ninguna.

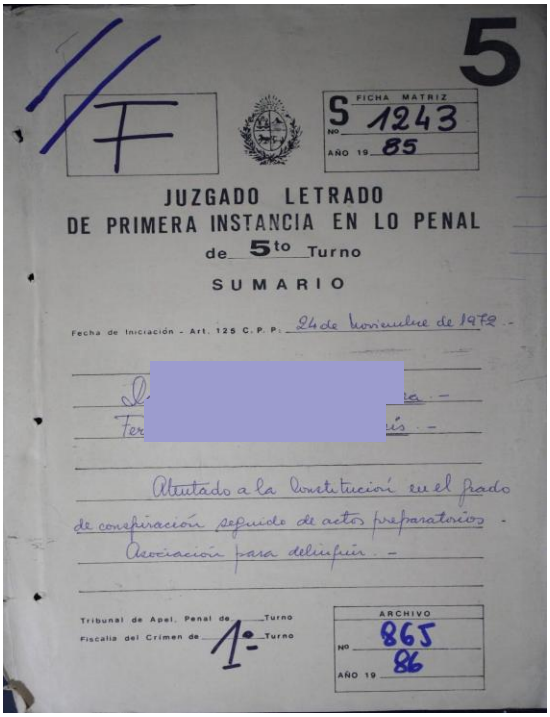


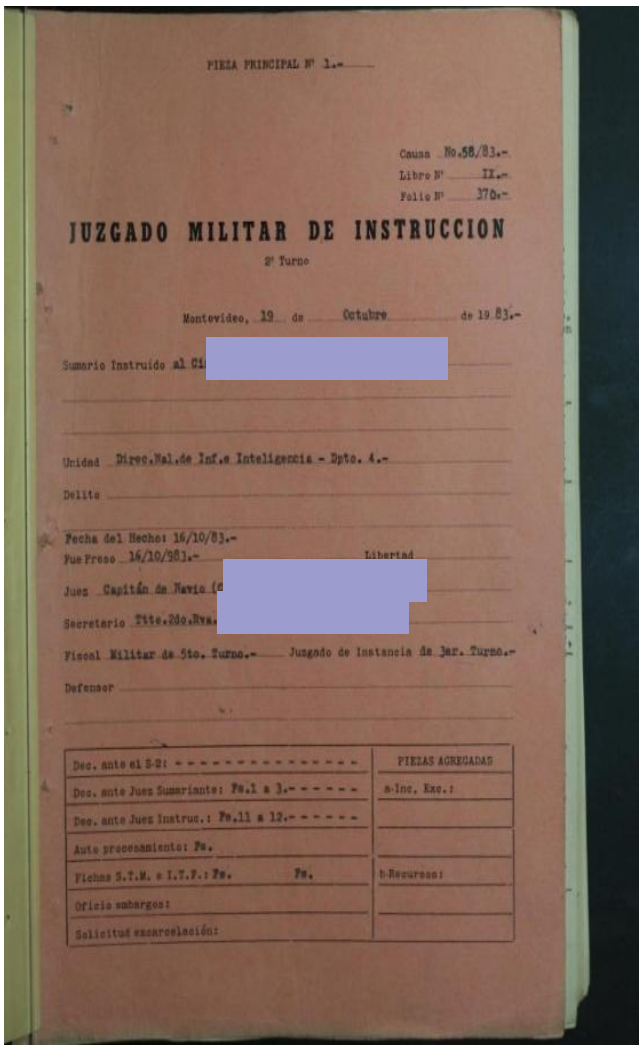
Figura 127 - Ejemplo carátula de expediente

·Acta de detención

Identificación de la sección: Ubicada a continuación de la carátula del expediente.

Características: Acta en donde se encuentran los principales datos relativos a la detención del/los indagados. Contiene datos como ser: Sumario Instruido, Unidad, Delito, Fue Preso, Fecha del Hecho, Libertad, Juez, Secretario, Fiscal y Defensor.

Palabras clave identificadas: Sumario Instruido, Sumario, Fue Preso, Libertad, Liberado, Fiscal, Fecha del Hecho.



PIEZA PRINCIPAL Nº 1.-

Causa No.58/83.-
Libro Nº IX.-
Folio Nº 370.-

JUZGADO MILITAR DE INSTRUCCION
2º Turno

Montevideo, 19 de Octubre de 19 83.-

Sumario Instruido al Ci

Unidad Direc.Nal.de Inf.e Inteligencia - Dpto. 4.-

Delito

Fecha del Hecho: 16/10/83.-
Fue Preso 16/10/983.- Libertad

Juez Capitán de Navío (

Secretario Tte.2do.Rva.

Fiscal Militar de 5to. Turno.- Juzgado de Instancia de 3er. Turno.-

Defensor

Dec. ante el S-2: - - - - - - - - - - - - - - -	PIEZAS AGREGADAS
Dec. ante Juez Sumariante: Fs.1 a 3.- - - - - -	a-Inc. Exc.:
Dec. ante Juez Instruc.: Fs.11 a 12.- - - - - -	
Auto procesamiento: Fs.	
Fichas S.T.M. e I.T.F.: Fs. Fs.	b-Recursos:
Oficio embargos:	
Solicitud excarcelación:	

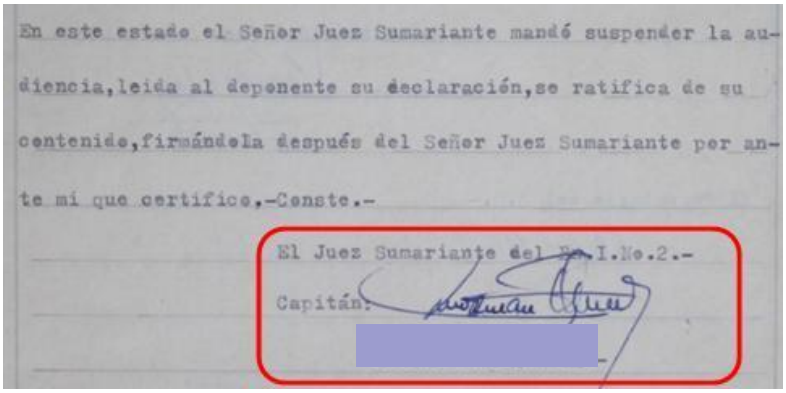
Figura 128 - Ejemplo Acta de Detención de expediente

Acta de declaración (ante juez sumariante)

Identificación de la sección: Esta sección se logra identificar por una secuencia de Preguntados/Contestados entre el juez sumariante y el indagado. Luego de esto dependiendo de lo decidido por el juez se finaliza o suspende la audiencia y pasa a firmar el juez sumariante, testigos, secretario, etc.

Características: Acta realizada por el juez sumariante en el cual se realiza un pre-sumario al indagado en cuestión.

Palabras clave identificadas: Detención, testigo, testigos, procesado, procesados, encausado, encausados, ciudadano, ciudadanos, autos caratulados, caratulado, persona, recluso, datos filiatorios, datos patronímicos, datos dactiloscópicos.


En este estado el Señor Juez Sumariante mandó suspender la audiencia,leida al deponente su declaración,se ratifica de su contenido,firmándola después del Señor Juez Sumariante por ante mi que certifico.-Conste.-

El Juez Sumariante del [redacted] I.Mo.2.-

Capitán: [redacted]

Figura 129 - Ejemplo de sección identificatoria - Firma de Juez

Acta de declaración (ante juez de instrucción)

Identificación de la sección: El acta de declaración ante el juez de instrucción siempre se encuentra a continuación de la sección de acta de declaración ante juez sumariante. En esta sección se encuentra entre otros el pedido de antecedentes del juez de instrucción al presidente del supremo tribunal militar. Esta sección es identificada claramente ya que en todas las sub-secciones se identifica el nombre/firma del juez de instrucción.

Características: Acta realizada por el juez de instrucción en el cual se le toma declaración al indagado en cuestión.

Palabras clave identificadas: Juez Militar de Instrucción, procesado, procesados, encausado, encausados, ciudadano, ciudadanos, autos caratulados, caratulado, persona, recluso, datos filiatorios, datos patronímicos, datos dactiloscópicos.

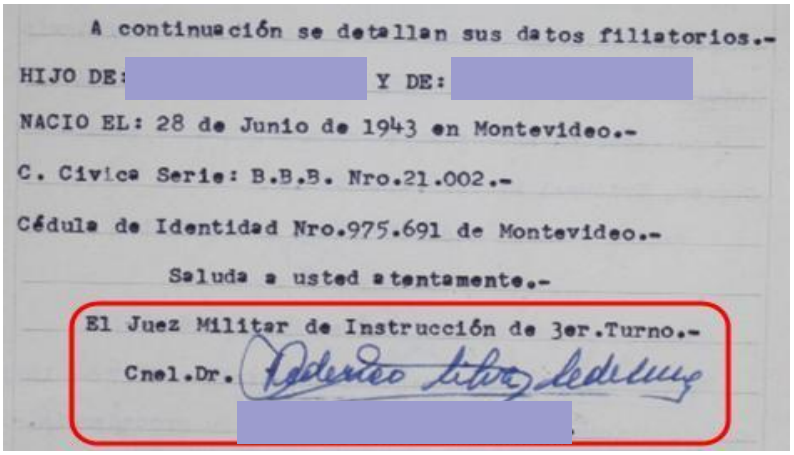

A continuación se detallan sus datos filiatorios.-
HIJO DE: [redacted] Y DE: [redacted]
NACIO EL: 28 de Junio de 1943 en Montevideo.-
C. Cívica Serie: B.B.B. Nro.21.002.-
Cédula de Identidad Nro.975.691 de Montevideo.-
Saluda a usted atentamente.-
El Juez Militar de Instrucción de 3er.Turno.-
Cnel.Dr. [redacted]

Figura 130 - Ejemplo de sección identificatoria - Firma de Juez

Decreto de procesamiento o de puesta en libertad

Identificación de la sección: Luego de la declaración ante el juez de instrucción se identifica un decreto en el cual luego de varios considerandos**,** se dicta el fallo indicando el procesamiento o la puesta en libertad del indagado.

Características: Sección en la cual se dicta el fallo para el indagado.

Palabras clave identificadas: Fallo, Se resuelve, procesado, procesados, encausado, encausados, ciudadano, ciudadanos, autos caratulados, caratulado, persona, recluso.

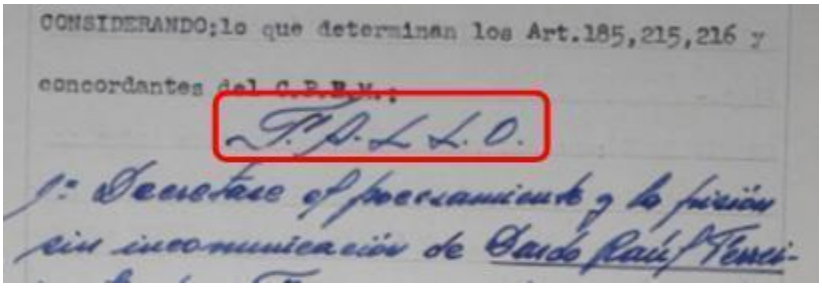

CONSIDERANDO:lo que determinan los Art.185,215,216 y
concordantes del C.P.P.M.;
F.P.L.L.O.
1º Decretase el procesamiento y la prisión

Figura 131 - Ejemplo de sección identificatoria (Fallo)

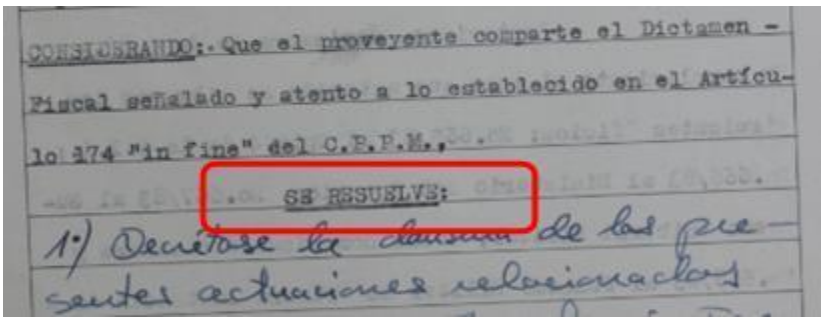

CONSIDERANDO:- Que el proveyente comparte el Dictamen -
Fiscal señalado y atento a lo establecido en el Artícu-
lo 274 "in fine" del C.P.P.M.,
SE RESUELVE:
1°) Decrétase la clausura de las pre-
sentes actuaciones relacionadas.

Figura 132 - Ejemplo de sección identificatoria (Se resuelve)

Designación de defensor

Identificación de la sección: Esta sección se puede identificar claramente luego del decreto de procesamiento/puesta en libertad. En ella se observa la designación del defensor que va a tomar la causa. En algunos expedientes también se logra ver la designación de un co-defensor.

Características: Designación del defensor que procede a tomar la causa.

Palabras clave identificadas: Procesado, procesados, encausado, encausados, ciudadano, ciudadanos, autos caratulados, caratulado, persona, recluso.

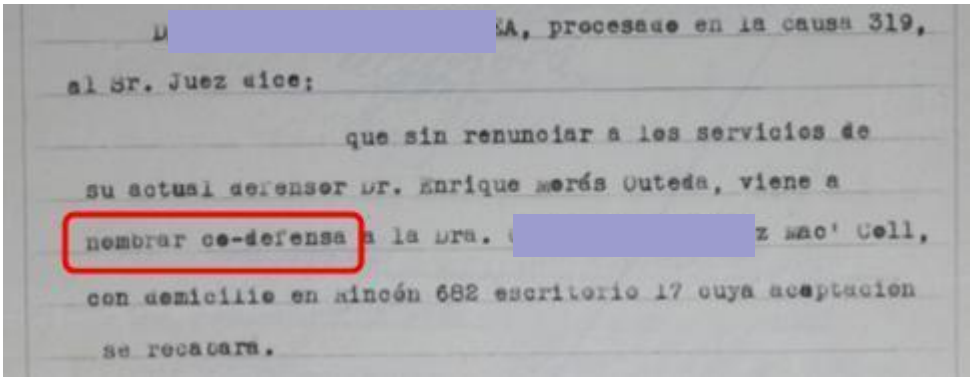

[redacted], procesado en la causa 319,
al Sr. Juez dice:
que sin renunciar a los servicios de
su actual defensor Dr. Enrique Morás Outeda, viene a
nombrar co-defensa a la Dra. [redacted] Mac' Coll,
con domicilio en Rincón 682 escritorio 17 cuya aceptacion
se recaba.

Figura 133 - Ejemplo de sección identificatoria - Designación de co-defensor

Acusación fiscal

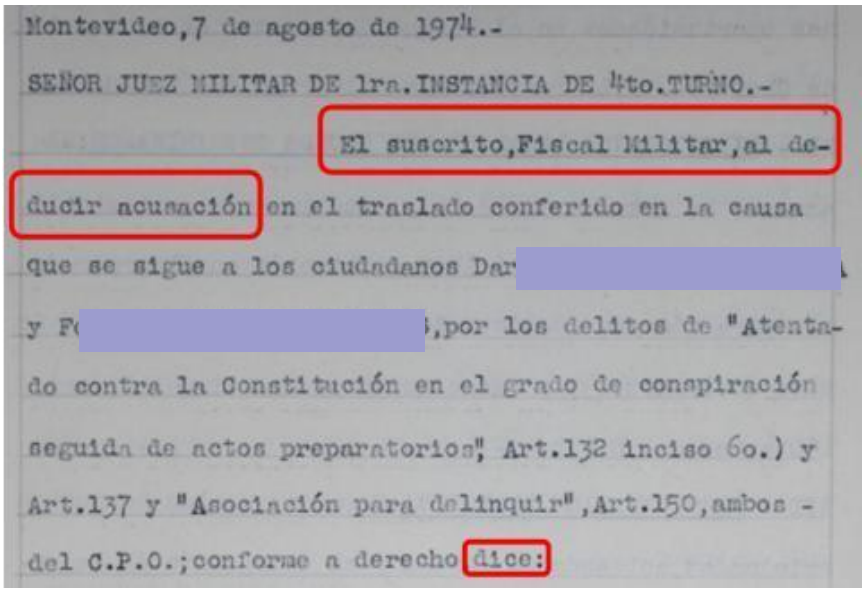
Identificación de la sección: Más adelante en el expediente se logra identificar claramente la sección de acusación fiscal, comenzando en general con las siguientes

líneas: “El suscrito, Fiscal militar, ….” prosiguiendo con una serie de pronunciamientos que se mencionan a continuación.

Características: Sección en la cual el fiscal militar declara:

- Hechos
- Tipificación legal de los hechos incriminados
- Participación delictiva
- Circunstancias atenuantes y agravantes
- Pena aconsejada
- Fundamentos de las penas

Palabras clave identificadas: Procesado, procesados, encausado, encausados, ciudadano, ciudadanos, autos caratulados, caratulado, persona, recluso.



Montevideo,7 de agosto de 1974.-

SEÑOR JUEZ MILITAR DE 1ra.INSTANCIA DE 4to.TURNO.-

El suscrito,Fiscal Militar,al de-

ducir acusación en el traslado conferido en la causa

que se sigue a los ciudadanos Dar

y F ,por los delitos de "Atenta-

do contra la Constitución en el grado de conspiración

seguida de actos preparatorios", Art.132 inciso 6o.) y

Art.137 y "Asociación para delinquir",Art.150,ambos -

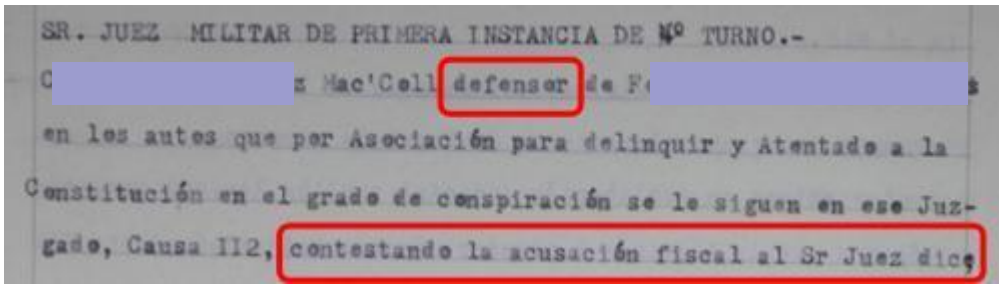
del C.P.O.;conforme a derecho dice:

Figura 134 - Ejemplo de sección identificatoria - Acusación fiscal

Vista a la defensa

Identificación de la sección: A continuación de la acusación fiscal el defensor del indagado responde a dicha acusación. Se identifica claramente porque está a continuación de la sección anterior y porque se nombra explícitamente a la defensora que está respondiendo en defensa del acusado (en general expresado de la siguiente forma: “contestando la acusación fiscal al Sr Juez ...”)

Características: Respuesta del defensor a la acusación del fiscal. **Palabras clave identificadas:** Procesado, procesados, encausado, encausados, ciudadano, ciudadanos, autos caratulados, caratulado, persona, recluso.


SR. JUEZ MILITAR DE PRIMERA INSTANCIA DE Nº TURNO.-

C [redacted] z Mac'Cell defensor de F [redacted] s

en los autes que por Asociación para delinquir y Atentado a la

Constitución en el grado de conspiración se le siguen en ese Juz-

gado, Causa 112, contestando la acusación fiscal al Sr Juez dice

Figura 135 - Ejemplo de sección identificatoria - Vista a la defensa

Sentencia definitiva de primera instancia

Identificación de la sección: En el borde izquierdo de la página se puede identificar la palabra "SENTENCIA". Se observan secciones características como ser: VISTO, RESULTANDO, CONSIDERANDO, ETC. Al final de esta sección se encuentra la sección FALLO en donde se dicta la sentencia definitiva.

Características: Aquí se encuentra la sentencia de primera instancia para el procesado.

Palabras clave identificadas: Fallo**,** Se resuelve, Detención, Fecha de detención, Procesado, procesados, encausado, encausados, ciudadano, ciudadanos, autos caratulados, caratulado, persona, recluso.


D.t. Nº 422283

AÑO DE LA INTEGRALIDAD

Estado de la Defensa

fojas 47.

SENTENCIA

No. 103/975.

Montevideo, 9 de A B R I L de 1975.-

VISTOS: Para sentencia definitiva de

Primera Instancia esta causa seguida

a los procesados DAIG [redacted] A, oriental,

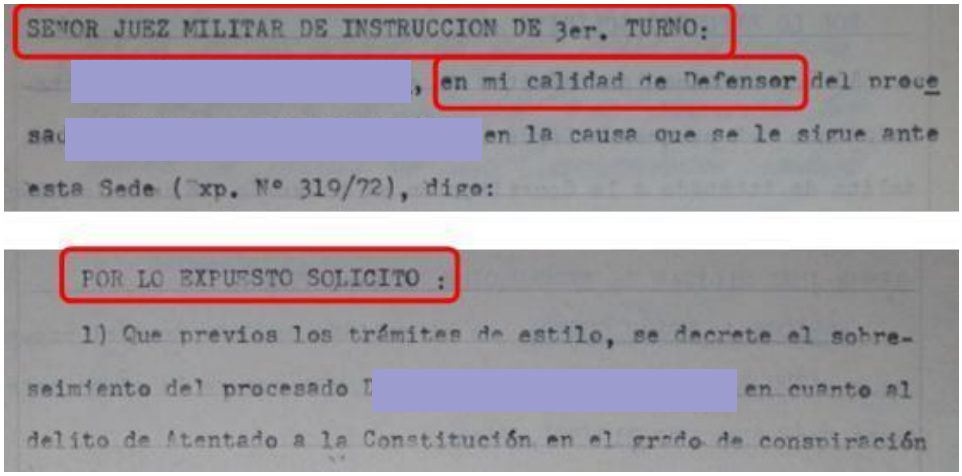
Figura 136 - Ejemplo de sección identificatoria - Sentencia de primera instancia

Apelación

Identificación de la sección: Se detecta a continuación de la sentencia definitiva de primera instancia. El defensor apela dicha sentencia respondiendo al juez en general con el formato: “Señor Juez Militar de Instrucción de X Turno, en mi calidad de defensor del procesado X, …. . Por lo expuesto solicito ...”

Características: En esta sección la defensa apela a la sentencia dictada por el juez.

Palabras clave identificadas: Juez Militar de Instrucción, Procesado, procesados, encausado, encausados, ciudadano, ciudadanos, autos caratulados, caratulado, persona, recluso.


SEÑOR JUEZ MILITAR DE INSTRUCCION DE 3er. TURNO:

, en mi calidad de Defensor del proce

sa en la causa que se le sigue ante

esta Sede (Exp. Nº 319/72), digo:

POR LO EXPUESTO SOLICITO :

1) Que previos los trámites de estilo, se decrete el sobre-

seimiento del procesado en cuanto al

delito de Atentado a la Constitución en el grado de conspiración

Figura 137 - Ejemplo de sección identificatoria - Apelación

Sentencia de segunda instancia del supremo tribunal militar

Identificación de la sección: Ídem a sentencia definitiva de primera instancia.

Características: Ídem a sentencia definitiva de primera instancia.

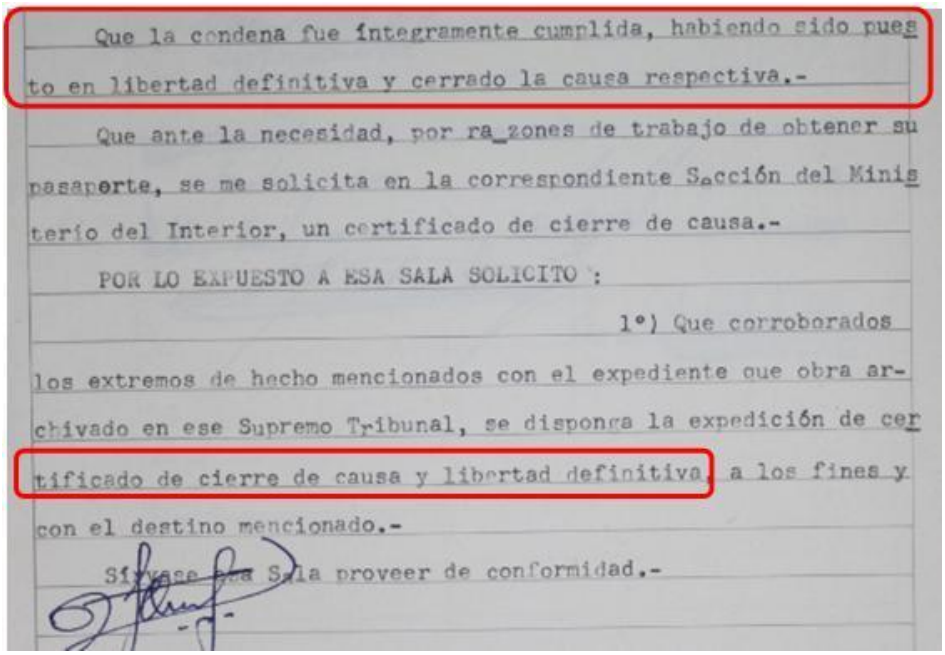
Palabras clave identificadas: Fallo**,** Se resuelve, Detención, Fecha de detención, Procesado, procesados, encausado, encausados, ciudadano, ciudadanos, autos caratulados, caratulado, persona, recluso.

Solicitud de libertad

Identificación de la sección: En esta sección se expresa por parte del defensor la solicitud de libertad del procesado. Se identifica porque contiene frases tales como “condena íntegramente cumplida”, “solicitud de libertad” y al final firma el defensor por dicho pedido.

Características: Solicitud de libertad pedida por el defensor para el procesado.

Palabras clave identificadas: Procesado, procesados, encausado, encausados, ciudadano, ciudadanos, autos caratulados, caratulado, persona, recluso.



Que la condena fue íntegramente cumplida, habiendo sido puesto en libertad definitiva y cerrado la causa respectiva.-

Que ante la necesidad, por razones de trabajo de obtener su pasaporte, se me solicita en la correspondiente Sección del Ministerio del Interior, un certificado de cierre de causa.-

POR LO EXPUESTO A ESA SALA SOLICITO :

1°) Que corroborados los extremos de hecho mencionados con el expediente que obra archivado en ese Supremo Tribunal, se disponga la expedición de certificado de cierre de causa y libertad definitiva, a los fines y con el destino mencionado.-

Sírvase esa Sala proveer de conformidad.-

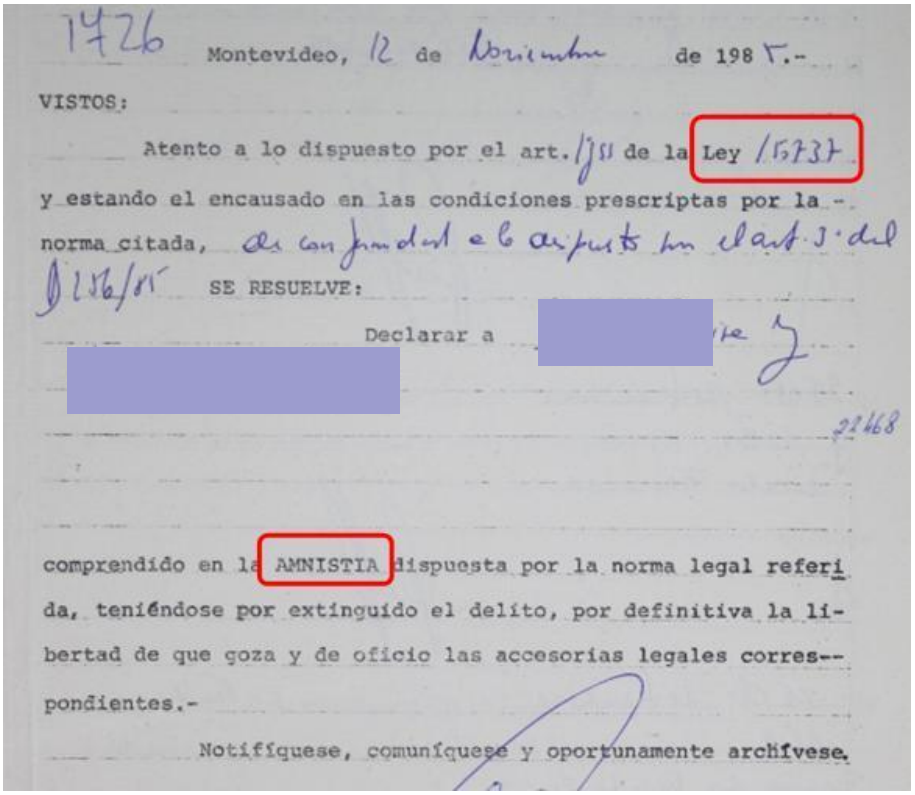
Figura 138 - Ejemplo de sección identificatoria - Solicitud de libertad

Decreto de libertad / Resolución de amparo a la ley 15.737 de amnistía

Identificación de la sección: Situada a continuación de la “solicitud de libertad”, en donde el Juez dicta a él/los condenados como libres de sus penas si correspondiera. Se puede identificar claramente, porque se nombra directamente la ley con su correspondiente número y se declara a los encausados como libres de ningún delito.

Características: Se decreta la libertad de el/los procesados.

Palabras clave identificadas: Se resuelve, procesado, procesados, encausado, encausados, ciudadano, ciudadanos, autos caratulados, caratulado, persona, recluso.



1476 Montevideo, 12 de Noviembre de 198 7.-

VISTOS:

Atento a lo dispuesto por el art. 1/51 de la Ley 15737 y estando el encausado en las condiciones prescriptas por la norma citada, de conformidad a lo dispuesto en el art. 3 del D 256/85 SE RESUELVE:

Declarar a [redacted] 22468

comprendido en la AMNISTIA dispuesta por la norma legal referida, teniéndose por extinguido el delito, por definitiva la libertad de que goza y de oficio las accesorias legales correspondientes.-

Notifíquese, comuníquese y oportunamente archívese.

Figura 139 - Ejemplo de sección identificatoria - Decreto de libertad