PRISCUS: Reconocedor Óptico de Caracteres Manuscritos y Antiguos

Eduardo Cuevas Farfán, Pilar Gómez-Gil
Coordinación de Ciencias Computacionales.
Instituto Nacional de Astrofísica, Óptica y Electrónica
Tonantzintla, Puebla, México
eduardo.cuevas@ieee.org, pgomez@acm.org

RESUMEN

PRISCUS es un proyecto para la construcción de un sistema de reconocimiento óptico de caracteres basado en redes neuronales artificiales (RNA) y otros componentes de inteligencia computacional y entendimiento de lenguaje, especializado en el reconocimiento de textos manuscritos y textos antiguos. EL objetivo de PRISCUS es desarrollar un software inteligente integral, que supere la problemática intrínseca que se presenta en los reconocedores convencionales al trabajar con letra cursiva y textos antiguos. La construcción de PRISCUS implica la mejor implementación posible a problemas que aún son abiertos, tales como: segmentación de escritura continua, entendimiento de texto completo en base a clasificación de caracteres y palabras, e identificación de palabras en imágenes digitales que contienen diferentes tipos información, tal como renglones, fotos, ruido, etc. En este artículo, se explica de manera breve los componentes del proyecto y se presentan resultados obtenidos a la fecha, así como las perspectivas del trabajo faltante.

I. INTRODUCCIÓN

Mucha de la información que se necesita actualmente para desarrollar nuestras actividades diarias se encuentra documentada de manera escrita, como apuntes de alguna junta o de una clase; inclusive mucha de esta información fue documentada hace muchos años y es preservada en bibliotecas por su valor histórico. Dado el uso de tecnologías electrónicas como Internet, es necesario que esta información esté disponible en un formato digital. Aun así, no basta con tener la información digitalizada como una imagen; es necesario que esta información se encuentre en un formato editable para su mejor aprovechamiento. Bajo la primicia anterior surgen diferentes sistemas para el reconocimiento óptico de caracteres (OCR por sus siglas en ingles), sin embargo a la fecha no existe alguno capaz de lidiar efectivamente con la problemática que representa el reconocimiento de texto manuscrito o de documentos antiguos.

El reconocimiento de textos manuscritos y en especial de los antiguos tiene implícito una serie de características que resulta un reto para las tecnologías de reconocimiento de patrones, como se expone en [1]. Entre estas se encuentra que los textos en letra cursiva tienen mayor cantidad de ornamentos, resultan de difícil segmentación, un mismo carácter puede tener diferentes aspectos según su lugar en la palabra, y distintas letras puede verse muy similares (ver figura 1). Adicionalmente, los textos manuscritos varían mucho dependiendo de factores como el tipo de pluma, la edad del autor ó inclusive el estado de ánimo. Además, los problemas aumentan cuando estos textos fueron escritos hace varias décadas dado que el paso del tiempo puede dañar los textos deteriorando el papel, la textura y la tinta. Reconocer este tipo de textos puede ser complicado hasta para un humano.

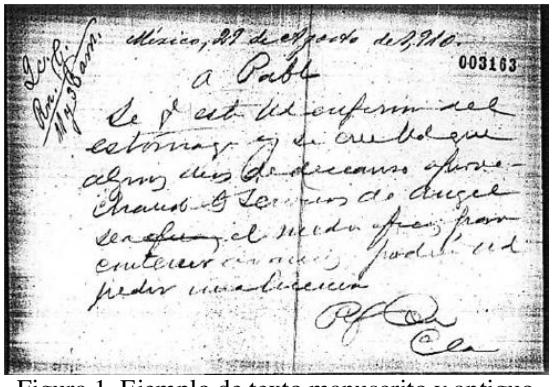

Figura 1. Ejemplo de texto manuscrito y antiguo [1]

Visto desde un enfoque formal de reconocimiento de patrones, el reconocimiento de este tipo de

texto tiene entre otras implicaciones [1,3]: que no se tengan prototipos evidentes para definir cada una de las clases del problema, que pueda haber variaciones significativas entre miembros de la misma clase lo que lleva a que no se puedan ocupar directamente similitudes métricas como la distancia Euclidiana, que el sistema sea poco tolerante al ruido, a la inclinación, a muestras incompletas y de diferentes tamaños, que el reconocimiento sea de forma "*off-line*" e independiente del escritor, y que la segmentación sea mucho más complicada que la típica manera de picos y valles.

Debido a que las RNA son modelos matemáticos, inspirados en sistemas biológicos, son capaces de "aprender" el comportamiento de un sistema a partir de un entrenamiento, poseen características de generalización, abstracción y aprendizaje que pueden lidiar con las implicaciones descritas anteriormente [2]. Las RNA pueden encontrar la solución de problemas que no tienen una solución evidente.

La investigación que se realiza en el proyecto PRISCUS está encaminada a ocupar diferentes arquitecturas y teorías de las RNA, y combinarlas otras teorías de reconocimiento de patrones y entendimiento de lenguaje para resolver la problemática que ya se ha mencionado.

Aquí se presenta un resumen de los resultados de de la investigación realizada a la fecha en este proyecto, así como los retos pendientes a resolver.

La organización de este documento es la siguiente: en la segunda sección se explica el modelo PRISCUS. El reconocimiento de caracteres y la formación de palabras estan explicados en la sección tres y cuatro respectivamente. La quinta sección habla de la interfaz y el sistema administrativo del software. Las conclusiones y el trabajo futuro están en la sexta sección.

II. MODELO PRISCUS

El modelo PRISCUS se compone de las siguientes partes (Ver figura 2) [1]:

- Digitalización. Creación de una imagen digital del documento a color o escala de grises.
- Pre-Procesamiento. Clarificar la imagen, eliminar ruido, convertir la imagen en blanco y negro.
- Segmentación de palabras. A partir del mapa de bits de todo el documento, obtener las palabras presentes.
- Entrenamiento de segmentación. Sistema adaptivo para identificar los puntos de segmentación de las palabras.
- Segmentación de caracteres. Basado en el aprendizaje obtenido en el entrenamiento de segmentación, obtiene los segmentos donde es muy probable que haya un caracter.
- Entrenamiento del reconocimiento. Sistema adaptivo que identifica el carácter contenido en un segmento de palabra.
- Reconocimiento de caracteres aislados. Recibe los segmentos con caracteres y devuelve cual es el caracter mas parecido.
- Identificación de palabras. Ocupa el reconocimiento de los caracteres y un diccionario, identifica cual es la palabra más parecida.
- Corrección del estilo. Crea oraciones bien formadas ocupando las palabras identificadas y las reglas de gramática. De aquí se obtiene una transcripción del documento.

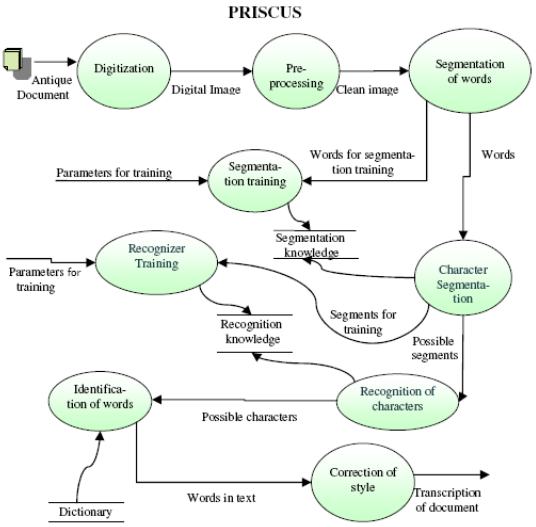

Figura 2. Modelo PRISCUS [1]

Al día de hoy, en este proyecto se ha investigado y trabajado en la segmentación de palabras, el reconocimiento de caracteres aislados, el reconocimiento de palabras basado en un diccionario y los requerimientos del sistema como un software integral. En las siguientes secciones se resumen los avances.

III. RECONOCIMIENTO DE CARACTERES AISLADOS

Para el reconocimiento de caracteres aislados se ha implementado un modelo basado en redes SOM (self-organized maps). Las redes SOM o mapas auto-organizados son un tipo de red neuronal que trata de agrupar el espacio de entradas según las características comunes entre las diferentes entradas [2]. Las redes SOM tienen un tipo de entrenamiento no supervisado en el que las neuronas de salida compiten para ser la neurona ganadora. Al final del entrenamiento, el resultado de una entrada es una neurona activada que nos dice a qué grupo o clase pertenece la entrada.

La razón para elegir una red SOM es que se puede tener diferentes prototipos de salida diferentes para cada clase, muy necesario para el problema del texto manuscrito en el que los elementos de una clase pueden ser muy diferentes.

En esta parte de la investigación, se probaron diferentes topologías de redes SOM para encontrar la más óptima para el problema dando como resultado una red con salida de 2x30 [4]. Hasta este punto la red fue entrenada ocupando como corpus una colección de telegramas antiguos de la biblioteca de la Universidad de las Américas Puebla, de manera que solo reconocía 26 clases distintas de un solo autor. Posteriormente la red SOM fue entrenada ocupando como corpus la base de datos NIST19 para ampliar su capacidad de reconocimiento a 52 clases e independiente del autor. El resultado fue una red de con salida de 4X75 capaz de reconocer hasta el 56% de los caracteres presentados [3].

Aunque el reconocimiento de caracteres no es tan alto como se esperaría, el resultado del entrenamiento otorga un archivo de frecuencias, en el que se menciona otras opciones de letras que podrían ser los reconocidos y que se ocupa para el reconocimiento de las palabras.

IV. RECONOCIMIENTO DE PALABRAS

La reconstrucción de una palabra podría haberse realizado simplemente concatenando los caracteres que arroja el reconocimiento de caracteres. Sin embargo para subsanar los errores que podría haber en este reconocimiento, un sistema autómata de estados finitos determinístico (AFD) se utiliza como solución al reconocimiento de palabras. Un autómata de estados finitos es un modelo del sistema complejo donde existen una cantidad determinada de nodos y condiciones predefinidas para el salto de nodos. La razón por la que un AFD se utiliza como solución al reconocimiento de palabras es que permite "jugar" con el reconocimiento de los caracteres para encontrar una palabra correcta [3].

Dado que para la construcción de un AFD se requieren conocer todas las posibles soluciones, (en este caso palabras las que se pueden reconocer) el AFD que se construyó tiene inicialmente un vocabulario restringido pero no fijo, en el que se pueden agregar palabras y crear los autómatas de dicha palabra en tiempo de ejecución. Este aspecto se ocupó adicionalmente para crear un perfil de usuario, en el que cada usuario puede tener su propio diccionario para el reconocimiento de las palabras.

Para que el AFD no sea equivalente una comparación de matrices, se ocupa la tabla de frecuencias generada en el reconocimiento de caracteres. De esta manera, si los caracteres reconocidos en primera instancia no logran modelar una palabra, se evalúan otros caracteres que pueden ser candidatos fuertes para formar una palabra. Y si en un momento dado un mismo reconocimiento puede generar dos palabras, se ocupa la misma tabla de frecuencias para decidir cuál de las dos palabras es la que más que se parece a la imagen.

El resultado del reconocimiento de palabras basado en un AFD y ocupando el reconocimiento de caracteres de la red SOM arrojó, en el mejor de los casos experimentales, un porcentaje de 66% de palabras reconocidas con un diccionario de 40 palabras. Tratando de mejorar este porcentaje se corrigió la inclinación de las imágenes y se alcanzó un reconocimiento del 86.51% sobre el mismo diccionario [3]. De esta manera el porcentaje de reconocimiento de la red SOM fue mejorado y evidenciando la necesidad de un módulo para el pre-procesamiento de las imágenes.

V. SISTEMA INTEGRAL

Los avances revisados hasta este punto son funcionales en sí, pero no representan la idea inicial de un sistema integral orientado al usuario. Por esta razón, se ha iniciado con el desarrollo de este sistema, comenzado con la interfaz y la incorporación de los módulos de software existentes y programados a la fecha. Para esto, se ha seguido un modelo de desarrollo de software espiral, basado en una metodología integral e iterativa orientada a objetos. En la figura 3 observamos el prototipo obtenido a la fecha de esta interfaz.

La idea fundamental es conseguir un software con una interfaz amigable al usuario que permita de

manera intuitiva su uso. El reconocimiento debe poder guardarse en documentos de texto comerciales (.doc) y que el software proporcione la posibilidad de dar formato básico al texto. También debe ser capaz de manejar más de una imagen al mismo tiempo. El software también debe poder administrar todos los archivos relacionados con el reconocimiento de un documento a manera de proyecto.

Para la realización de esta parte del proyecto se hizo una investigación sobre las características de reconocedores de texto manuscrito comerciales. De estos, para aquellos con versiones de prueba disponibles se realizó un análisis de funcionalidad y usabilidad, y de los que no tenían versiones de prueba disponibles a potenciales consumidores, se averiguaron sus características más importantes según el fabricante. Este análisis llevó a enriquecer las especificaciones de funcionalidad del software Priscus.

Inspirados en la investigación anterior se desarrollo una primera versión de interfaz de PRISCUS (Ver figura 3.) y se comenzó con la programación de sus funciones. Al día de hoy se tienen programadas todas las funciones de los menús Proyecto, Imagen y Documento.

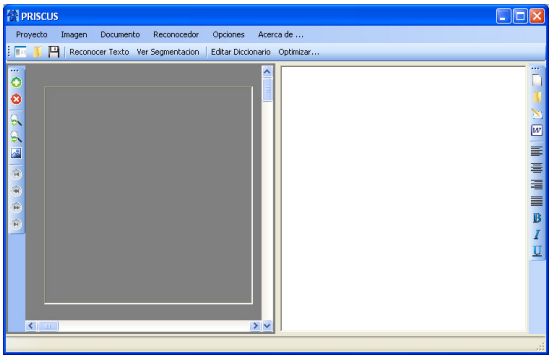

Figura 3. Ventana principal del prototipo PRISCUS

VI. CONCLUSIONES Y TRABAJO FUTURO

El reconocimiento óptico de caracteres, en especial de caracteres manuscritos y antiguos, significa un reto para las ciencias computacionales. Es evidente que una la combinación de conocimientos es necesaria para lograr resultados plenamente exitosos. En este caso las redes neuronales y los autómatas de estados finitos han sido mezclados para mejorar rendimientos, pero aún se tiene mucho trabajo por hacer. Es clara la necesidad de un módulo para el pre-procesamiento de las imágenes que reduzca los niveles de ruido, la inclinación y otros factores que dificulten el reconocimiento. Basándonos en el modelo de PRISCUS, un módulo para la corrección de la gramática contribuiría sustancialmente con la solución de este problema.

De igual manera, la integración de todos los módulos en un software orientado al usuario lleva este trabajo de investigación a un plano en el que cualquier persona, sea cual sea su actividad, se sirva del trabajo realizado para facilitar su forma de vida, que al final de cuentas es una razón de ser para la ciencia y la tecnología.

VII. AGRADECIMIENTOS

Eduardo Cuevas F. agradece al INAOE y a la Coordinación de Computación el apoyo otorgado para la realización de la estancia profesional de licenciatura que apoya el presente proyecto.

VIII. REFERENCIAS

[1] Gómez Gil, Pilar et al. "The Role of Neural Networks in the Interpretation of Antique Handwritten Documents." Hybrid Intelligent Systems Analysis and Design Series: Studies in Fuzziness and Soft Computing, Castillo, O.; Melin, P.; Kacprzyk, J.; Pedrycz, W. (Eds.). Vol. 208. pp. 269-281.

[2] Haykin, S. Neural Networks, a comprenhensive foundation. Second Edition, Upper Saddle River: NJ, Prentice Hall. 1999.

[3] Murillo Gil, Marbella. Reconocimiento de Textos Manuscritos Breves Basado en un Procesamiento de Palabras Completas. Tesis de Maestría. Departamento de Computación, Electrónica, Física e Innovación. Universidad de las Américas Puebla. 2007

[4] De los Santos Torres, Guillermo. Reconocedor de Caracteres Manuscritos. Tesis de Maestría. Departamento de Ingeniería en Sistemas Computacionales. Universidad de las Américas Puebla. 2003