

Autores: Gabriel Constenla (grconstenla@gmail.com)

Ing. Sergio Omar Aguilera (sergio.aguilera@ub.edu.ar)

Unidad de Investigación: Grupo de Software Libre – Facultad de Tecnología Informática – Universidad de Belgrano.

Título: Aplicación de Redes Neuronales en el reconocimiento óptico de caracteres.

Resumen

En una primer etapa este trabajo se enfocará en adaptar una red neuronal convencional y de uso general para su utilización en la fase de reconocimiento del Sistema OCR a desarrollar.

Se pondrá especial énfasis en posibilitar el crecimiento incremental de la solución para facilitar futuras investigaciones sobre el tema, apuntando con esta modalidad a la creación, a mediano o largo plazo, de una solución de interés académico, científico y comercial.

Específicamente se desarrollarán dos aplicaciones. La primera, **miniOCR**, será una aplicación capaz de reconocer dígitos aislados dibujados a mano en la misma a través del mouse. Luego, se desarrollará **Calculadora**, una segunda aplicación basada en **miniOCR**, que permita aplicar a un caso de ejemplo las conclusiones del primer desarrollo. En esta, el usuario podrá obtener respuestas de la computadora a operaciones aritméticas simples dibujadas en su interfaz de usuario.

Contexto

Todo esto se encuentra inserto dentro del área de Investigación de Agentes y Sistemas Inteligentes. Perteneciente al Área de Investigación en Software Libre, que reside en la Facultad de Tecnología Informática. La base de la I+D ha sido desarrollada como Tesina de Grado del Alumno Gabriel Constenla, de la carrera de Ing. en Informática.

Introducción

Los seres humanos aprenden a leer durante sus primeros años de educación. En general, no presentan dificultad en comprender diversos estilos de escritura, caracteres artísticos, con ornamentos o partes faltantes, o palabras con errores de ortografía, tachadas o con marcas indeseadas. Las palabras o caracteres pueden presentar distorsiones importantes, no obstante el ser humano es capaz de utilizar su experiencia y su capacidad para analizar el contexto para comprenderlos sin dificultad.

A pesar de ser un problema que tuvo más de cinco décadas de investigación intensiva, la capacidad de lectura de la computadora está muy por detrás de la de su contra parte humana. La computadora presenta severos fallos al intentar leer caracteres con distorsiones, los cuales pueden ser provocadas, por ejemplo, por degradación de los documentos fuente o por los estilos de escritura, si se trata de caracteres manuscritos en vez de documentos impresos.

Podría decirse que los Sistemas OCR son uno de los campos de investigación más importantes en el área de reconocimiento de patrones. Los Sistemas OCR lograron fomentar tal interés sobre los investigadores debido a sus grandes posibilidades de aplicación. Algunas de las aplicaciones incluyen: digitalización de textos impresos, procesamiento de cheques, distribución de correo, lectura automática de formularios, dispositivos de ayuda para lectura para ciegos, automatización de entrada de datos, interacción con el usuario.

En la actualidad, el problema de los OCR para textos impresos puede considerarse resuelto, a pesar de no poseer un reconocimiento del cien por ciento. Sin embargo, el reconocimiento de texto manuscrito es un área que aún hoy se encuentra vigente y es objeto de investigación activa. Las soluciones populares producto de dichas investigaciones apuntan a utilizar métodos estadísticos o redes neuronales que son especialmente aptas para resolver este tipo de problemas, como el algoritmo K-NN o los mapas auto-organizados de Kohonen, respectivamente.

Sin embargo, este trabajo se enfocará en adaptar una red neuronal convencional y de uso general para su utilización en la fase de reconocimiento del Sistema OCR a desarrollar.

Se pondrá especial énfasis en posibilitar el crecimiento incremental de la solución para facilitar futuras investigaciones sobre el tema, apuntando con esta modalidad a la creación, a mediano o largo plazo, de una solución de interés comercial y científico.

Específicamente se desarrollarán dos aplicaciones. La primera, **miniOCR**, será una aplicación capaz de reconocer dígitos aislados dibujados a mano en la misma a través del mouse. Luego, se desarrollará **Calculadora**, una segunda aplicación basada en **miniOCR**, que permita aplicar a un caso de ejemplo las conclusiones del primer desarrollo. En esta, el usuario podrá obtener respuestas de la computadora a operaciones aritméticas simples dibujadas en su interfaz de usuario.

Los objetivos primarios de este trabajo de I+D son:

- ❖ Desarrollar **miniOCR**, una aplicación capaz de reconocer dígitos aislados dibujados a mano en pantalla por el usuario a través del mouse. Utilizará una red neuronal convencional adaptada para este fin. Debe comprender también la recolección de muestras, el entrenamiento y medición de rendimiento. La tasa de reconocimiento debe ser de un 90% de aciertos o superior de acuerdo al **Medidor de Rendimiento**. Su diseño y arquitectura deben posibilitar el crecimiento incremental para facilitar futuras investigaciones.
- ❖ Desarrollar **Calculadora**, una aplicación que permita aplicar a un caso de ejemplo a **miniOCR**, donde el usuario pueda obtener respuestas de la computadora a operaciones aritméticas simples (suma, resta, multiplicación y división) dibujadas. Tendrá en cuenta la precedencia de operadores y su modificación a través de paréntesis.

La Hipótesis es que en función de las características de aprendizaje de las Redes Neuronales feedforward de tres capas, convencional y de uso general, se puede desarrollar un programa de reconocimiento de texto manuscrito. Implica esto un conjunto de operaciones de adaptación y soporte a fin de capacitarla para operar en el contexto descripto.

A continuación se exponen los límites del presente trabajo:

- ❖ El trabajo se enfocará sobre los Sistemas OCR de procesamiento offline, de acuerdo a su definición en el mismo.
- ❖ Se busca tanto para **miniOCR** como para **Calculadora** una tasa de reconocimiento aceptable a los fines del trabajo, definida en un 90% o superior de aciertos según el **Medidor de Rendimiento**. Es decir, habrá casos en los que la aplicación no pueda reconocer correctamente el carácter ingresado.
- ❖ Tanto **miniOCR** como **Calculadora** intentarán reconocer toda figura ingresada como un carácter reconocible.
- ❖ Para **miniOCR** y **Calculadora**, se utilizará una cantidad de muestras para el entrenamiento que puede no ser suficiente para presentar un rendimiento homogéneo frente a usuarios con estilos de escritura que sean demasiado diferentes a los presentados durante el entrenamiento.
- ❖ **Calculadora** se desarrollará para ejemplificar las posibilidades de la tecnología, no se busca una aplicación plenamente funcional.
- ❖ El algoritmo de segmentación de símbolos de **Calculadora** es simplista, y fue desarrollado de tal manera a modo de cumplir con los objetivos de la aplicación. Los caracteres deben estar separados horizontalmente de modo que una línea vertical recta pueda pasar entre ellos, y solamente se puede escribir una línea por vez.

Líneas de Investigación y desarrollo

Desarrollo de aplicación que permita el reconocimiento de caracteres manuscritos utilizando Redes Neuronales feedforward.

La idea es sentar las bases de una I+D que permita desarrollar productos de transferencia para reconocimiento de caracteres manuscritos, utilizando herramientas de software libre.

Resultados Obtenidos/Esperados

miniOCR es la primera aplicación desarrollada en el presente trabajo. Es capaz de reconocer dígitos dibujados manualmente utilizando el mouse. Abarca todo el proceso necesario para llegar a dicho fin.

Entorno de Desarrollo

- ❖ IDE: Eclipse 3.5.1 (<http://www.eclipse.org/>)
- ❖ Lenguaje: Java 1.6.17 (<http://java.sun.com/>)
- ❖ Librerías: Encog (<http://www.heatonresearch.com/>)
- ❖ Sistema Operativo: Windows XP Professional Service Pack 3
- ❖ Hardware: Pentium 4 3.2, 3 GB RAM

Módulos

miniOCR está compuesto por cuatro módulos independientes, listados a continuación:

1. **El Generador de Muestras** se utiliza para facilitar la toma de muestras de distintos individuos. El mismo fue diseñado como un módulo independiente para que pueda ser distribuido entre varias personas y de ese modo obtener sus muestras. Tener en cuenta que las muestras no sufrirán ningún tipo de procesamiento en esta etapa.
2. **El Entrenador** permite entrenar la red neuronal a ser utilizada por el **Clasificador**. Se precisa para su ejecución un conjunto de muestras generadas por el **Generador de Muestras**. El entrenamiento termina cuando se alcanza el error deseado, solicitado al usuario al inicio de la ejecución.
3. **El Clasificador** es el módulo que lleva a cabo el objetivo de la aplicación, el reconocimiento de dígitos. El archivo de red neuronal generado con el **Entrenador** permite reconstruir la red para su utilización. Al igual que con los módulos anteriores, el **Clasificador** reconocerá los caracteres por defecto del cero al nueve, o se podrá modificar este comportamiento utilizando un archivo de texto plano.
4. **El Medidor de Rendimiento** es el módulo que permite evaluar el rendimiento de una red neuronal entrenada por el **Entrenador**. El mismo presenta un conjunto de muestras generadas con el **Generador de Muestras** (preferiblemente no usadas durante el entrenamiento) a la red neuronal entrenada y compara sus respuestas con la esperada. Luego, emite un informe de porcentajes y cantidades de aciertos para cada caracter y en total.

Medición de Rendimiento

Se utilizó el módulo **Medidor de Rendimiento** para evaluar el rendimiento de cada uno de los casos y seleccionar el mejor.

Para esto se generaron 700 nuevas muestras (70 para cada caracter), que no fueron utilizadas durante el entrenamiento.

Se pudo verificar en los resultados que los casos **5**, **6**, **8** y **9** superaron la tasa de rendimiento del 90% planteada en el trabajo como aceptable, siendo el caso **9** el de mejor rendimiento con un 98% de aciertos. Además, el caso **9** superó la tasa de rendimiento en las mediciones individuales de rendimiento para cada caracter, lo cual no sucedió en ninguno de los otros casos.

Es importante destacar que esta medida de rendimiento está sujeta a las muestras utilizadas para la medición, al usar otras muestras se obtendrán valores que podrían diferir considerablemente de los dados.

La **Calculadora** es una aplicación basada en miniOCR que permite dibujar operaciones aritméticas simples (suma, resta, multiplicación y división) y obtener su resultado. Se tiene en consideración la precedencia de los operadores y es posible modificarla utilizando paréntesis.

El objetivo de esta aplicación es el de aplicar la tecnología desarrollada en **miniOCR** a un caso práctico, para demostrar sus posibilidades con especial énfasis en la modificación del paradigma para la interacción del usuario con la computadora.

Para implementar esta aplicación fue necesario investigar y desarrollar sobre los siguientes puntos:

- ❖ Nuevas Clases
- ❖ Generación de Muestras para las Nuevas Clases
- ❖ Nuevo Entrenamiento
- ❖ Segmentador de Imágenes
- ❖ Evaluador de Expresiones Aritméticas

Nuevas Clases

Se sumaron las siguientes clases a los dígitos:

- ❖ '+': suma
- ❖ '-': resta
- ❖ 'x': multiplicación
- ❖ '%': división
- ❖ '(': paréntesis abierto
- ❖ ')': paréntesis cerrado

La selección del símbolo '%' para la división no es arbitraria. Si se hubiera utilizado el símbolo '/', el mismo hubiera sido modificado por el corrector de inclinación y se confundiría con un '1'.

Se incorporó a los módulos **Generador de Muestras** y **Entrenador de miniOCR** un archivo de texto plano con todos los caracteres que se quieren reconocer, para que **miniOCR** modifique su comportamiento para funcionar con los nuevos caracteres. Esto permite utilizar los módulos ya desarrollados sin necesidad de modificaciones en el código.

Generación de Muestras para las Nuevas Clases

Se generaron 360 muestras de los caracteres nuevos a reconocer. Es decir, 60 muestras por cada caracter nuevo.

Nuevo Entrenamiento

Se realizaron diversos entrenamientos para **Calculadora**, definiendo distintas configuraciones de error deseado y cantidad de muestras, teniendo en consideración la experiencia con el entrenamiento de **miniOCR**.

Todos los casos de entrenamiento para **Calculadora** se realizaron en una red neuronal configurada en 48 neuronas en la capa de entrada, 24 neuronas en la capa oculta y 16 neuronas en la capa de salida, con el fin de reconocer 16 clases.

Se realizaron cuatro casos de entrenamiento:

- **Caso 1:** 1680 muestras, 0.15 error deseado
- **Caso 2:** 1680 muestras, 0.09 error deseado
- **Caso 3:** 1860 muestras, 0.15 error deseado
- **Caso 4:** 1860 muestras, 0.09 error deseado

Se utilizó el mismo conjunto de 1500 muestras con el que se entrenó **miniOCR**, y se le agregaron 180 muestras nuevas (30 por cada caracter nuevo) para el caso de 1680 muestras totales y 360 muestras nuevas (60 por cada caracter nuevo) para el caso de 1860 muestras totales.

Medición de Rendimiento

Se utilizó el módulo **Medidor de Rendimiento** para evaluar el rendimiento de cada uno de los casos y seleccionar el mejor.

Para esto se utilizaron las 700 muestras con las que se midió el rendimiento de **miniOCR**, y se les agregaron 420 muestras (70 nuevas muestras por cada caracter nuevo), llegando a un total de 1120 muestras.

Se pudo verificar en los resultados que los casos **2** y **4** superaron la tasa de rendimiento del 90% planteada en el trabajo como aceptable, siendo el caso **4** el de mejor rendimiento con un 96.6% de aciertos. Además, el caso 4 superó la tasa de rendimiento en las mediciones individuales de rendimiento para cada caracter, lo cual no sucedió en ninguno de los otros casos.

Es importante destacar que esta medida de rendimiento está sujeta a las muestras utilizadas para la medición, al usar otras muestras se obtendrán valores que podrían diferir considerablemente de los dados.

Futuras Investigaciones

A continuación se enumeran algunas de las más plausibles para ser desarrolladas en base al presente trabajo:

Ampliar el set de caracteres reconocibles

La primera opción que surge es la de entrenar a **miniOCR** para que sea capaz de reconocer un set de caracteres mayor. Por ejemplo se podría intentar incorporar el alfabeto completo.

En conclusión, es probable que sea necesario optimizar el pre procesamiento de **miniOCR** e implementar un extractor de características más minucioso. Así mismo, para reconocer casos especiales que son difíciles de diferenciar podría ser útil incorporar otras redes neuronales que se encarguen de dicha tarea.

Reconocimiento de líneas de texto

El principal desafío al momento de intentar realizar el reconocimiento sobre líneas de texto es el de ubicar cada caracter individual sobre la imagen.

Esta tarea implicaría una modificación en el pre procesamiento para lograr dicha división. Así mismo, se pueden agregar otras optimizaciones como la corrección de la inclinación de la línea.

Una vez aislado el caracter, el procesamiento seguiría su curso actual.

Continuación de la investigación en el área de los OCR.

Reconocimiento de caracteres en imágenes escaneadas o fotos digitales

Reconocimiento de líneas de texto en imágenes escaneadas o fotos digitales

Formación de Recursos Humanos

Equipo de Software Libre: Gabriel Constenla (UBLug), Diego Ardizzone (Becario), Ivan Puente Mendoza (Becario), Ing. Sergio Aguilera (Director).

Antecedentes:

Tesina: Distorsión de sonidos con Redes Neuronales. Ing. Leandro Jorge Perez. FTI – UB. 2008.

Tesina: Reconocimiento óptico de dígitos con Redes Neuronales. Gabriel Constenla. FTI – UB. 2010.

Bibliografía.

- ❖ "A generic 2D approach of handwriting recognition" Centre d'Expertise Parisien, Francia
Sylvain Chevalier, Edouard Geoffrois, Françoise Preteux, Mélanie Lemaitre
- ❖ "Artificial Intelligence Illuminated" Jones and Bartlett Publishers, 2004. *Ben Coppin*
- ❖ "Artificial Intelligence: A Systems Approach" Infinity Science Press, 2008 *M. Tim Jones*
- ❖ "Character Recognition Systems: A Guide For Students And Practitioners" Wiley, 2007
Mohamed Cheriet, Nawwaf Kharma, Cheng-Lin Liu, Ching Y. Suen
- ❖ "Digital Recognition using Neural Network" University Sains Malaysia, Malasia
Saleh Ali K. Al-Omari, Putra Sumari, Sadik A. Al-Taweel, Anas J.A. Husain
- ❖ "Handwritten Digit Recognition Using Image Processing and Neural Networks" Proceedings of the World Congress on Engineering 2007 Vol I, Inglaterra. *Faisal Tehseen Shah, Kamran Yousaf*
- ❖ "Introduction to Encog 2.2 for Java" Heaton Research, Inc., 2009. *Jeff Heaton*
- ❖ "Introduction to Neural Networks for Java". Heaton Research, Second Edition, 2008. *Jeff Heaton*
- ❖ "Reliable Recognition of Handwritten Digits Using A Cascade Ensemble Classifier System and Hybrid Features" Concordia University, Canadá, 2006. *Ping Zhang*
- ❖ "Utilizing Information Technology Systems Across Disciplines". IGI Global, 2009. *Evon M. O.*