

Máster de Visión Artificial

Asignatura: Aplicaciones Industriales

Lectura OCR de crotales - Práctica obligatoria

Orión García Gallardo

INTRODUCCIÓN

Desde 1998, en el sector vacuno se ha desarrollado un sistema de trazabilidad que permite un mayor control a lo largo de toda la cadena alimentaria en aras de una mayor transparencia, lo que ha permitido restablecer la confianza del consumidor tras la crisis de las encefalopatías espongiformes transmisibles y garantizar la seguridad alimentaria de los productos de carne de vacuno.

En la página web del Ministerios de Agricultura, Alimentación y Medio Ambiente [1] se describe el sistema de trazabilidad del ganado bovino, cómo debe actuar el ganadero en diferentes supuestos y se ofrece información sobre la normativa española y comunitaria sobre identificación y registro del ganado bovino.

Con el objetivo de facilitar la tarea de los ganaderos y trabajadores encargados de la manipulación del ganado está practica tiene como objetivo el de desarrollar un sistema en el cual el empleado se centraría principalmente en extraer los crotales de los animales y depositarlos en una cinta. A partir de este momento una cámara tomaría una instantánea de este crotal y mediante una aplicación se devolvería el número que contienen.

Dicho aplicación se va a dividir en dos prácticas. La primera se centraría en la preparación de la imagen, y la segunda, y que es objeto de este documento, en el reconocimiento del texto de la imagen resultante anterior.

Como resultado de esta práctica, partiendo de las imágenes de crotales contenidas en una carpeta, se devolverá por consola el número reconocido en cada imagen.

Este documento se divide en varias secciones. En la primera de ellas se describe el método desarrollado. A continuación se explica las pruebas realizadas sobre la muestra y las mejoras que se han introducido al ver los resultaos. Y por último se comenta las conclusiones obtenidas y los flancos que quedan abiertos después de la finalización de esta práctica.

DESCRIPCIÓN DEL MÉTODO DESARROLLADO

Para la implementación de esta aplicación se hace uso de las funciones que aporta la librería ImageJ [2]. ImageJ es un programa de dominio público de procesamiento de imágenes inspirado en NIHImage [3] para los Macintosh. Se ejecuta en cualquier ordenador que tenga instalada máquina virtual de Java 1.4 o posterior y permite aplicar todo tipo de operaciones tales como mostrar, editar, analizar, procesar, guardar e imprimir sobre imágenes de 8-bits, 16-bits y 32-bits.

Otra de las librerías usadas es Tess4j [4] que implementa teresact [5]. Tesseract está considerado actualmente como uno de los motores OCR libres con mayor precisión disponible

[6][7]. La librería Tess4j permite llamar de forma fácil y sencilla a una serie de métodos que aportan la funcionalidad descrita en [4].

En el diseño de esta práctica se ha supuesto que el sistema recibe como entrada imágenes binarias donde sólo aparecen los números de los crotales (fig 1).

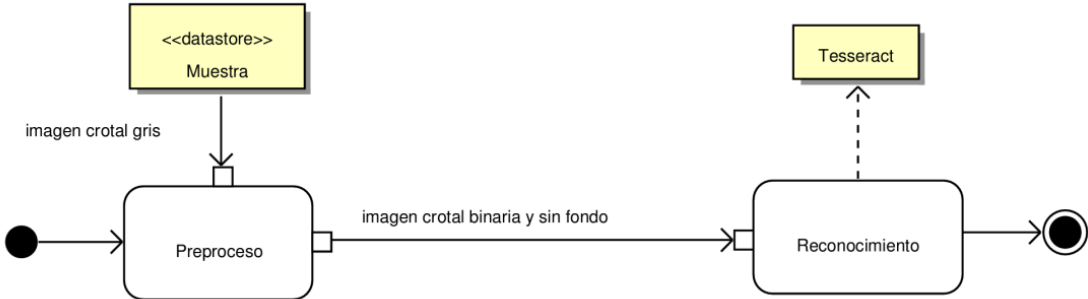


Fig 1. Diseño general del sistema completo.

El sistema presentado en este documento se compone de una serie de pasos detallados a continuación. El primero de ellos consiste en leer las imágenes supuestamente preprocesadas de un directorio de entrada. Seguidamente se dilatan las imágenes para engrosar los dígitos que se encuentran impresos en la imagen y se guarda la imagen resultante en una carpeta de salida. A continuación se pasa al reconocimiento de caracteres propiamente dicho, donde la imagen es procesada por la librería Tess4j. Teniendo en cuenta el contexto, es decir, que los crotales sólo pueden proporcionar números, se hace la suposición que el resultado de este procesado ha de ser un número entero. En caso de que lo devuelto no sean un número, un algoritmo intenta corregir la cadena de caracteres devuelta. Para ello se ha tenido en cuenta los típicos errores de reconocimiento óptico de caracteres numéricos que se especifican en la siguiente tabla (tabla 1). De tal manera que los caracteres son reemplazados por sus correspondientes números.

Digito erróneo	Digito Correcto
O	0
D	0
C	0
I	1
L	1
Z	2

Digito erróneo	Digito Correcto
S	5
\$	5
T	7
B	8
P	8
&	8
g	9

Tabla 1. Típicos errores de reconocimiento óptico de caracteres numéricos

Finalmente se presenta el número resultante por pantalla junto con el nombre de la correspondiente imagen.

PRUEBAS REALIZADAS SOBRE LA MUESTRA

Dentro de las pruebas realizadas se observa como el porcentaje de números correctamente detectados baja notablemente en el caso de que no se corrijan los caracteres numéricos devueltos (Tabla 2).

Algoritmo	Eficacia
Algoritmo sin corrección de caracteres	30%
Algoritmo con corrección de caracteres	60%

Tabla 2. Eficacia de los algoritmos

Los resultados concretos de ambos algoritmos pueden verse en la siguiente tabla (tabla 3).

Imagen	Número reconocidos algoritmo sin corrección de caracteres	Número reconocidos algoritmo sin corrección de caracteres
Crotal1.TIF	1225	1225
Crotal10.TIF		
Crotal2.TIF	3839	3839
Crotal3.TIF	4640	4640
Crotal4.TIF	0189	0189
Crotal5.TIF	7918 1	7918 1
Crotal6.TIF	7 if	7 if
Crotal7.TIF	03422 25	03422 25
Crotal8.TIF	Z646	2646
Crotal9.TIF		

Tabla 3. Resultados obtenidos con los dos algoritmos

CONCLUSIONES

- Resumen de resultados obtenidos.
Como se puede observar en la tabla anterior (Tabla 3) la corrección numérica supone un incremento en la eficiencia a la hora de detectar números de cortales correctamente. Esto demuestra que a la hora de reconocer textos es importantísimo el contexto en el que nos movamos. Esto queda demostrado por ejemplo a la hora de reconocer el crotal de la imagen crotal4.tiff donde ni siquiera para un humano sería posible distinguir entre la letra l y el número 1.
- Mejoras que podrían realizarse.
El algoritmo de tesseract funciona mejor cuanto más largo es el texto a reconocer. Para comprobar si el sistema mejora en eficiencia a la hora de devolver los números reconocidos podría ser una buena opción probar duplicar o triplicar las imágenes.
- Problemas que no se sabe cómo solucionar.
El punto más destacable a mejorar debería ser la eficiencia. Actualmente en el caso mejor un 60% de los casos se detectan correctamente y se debería buscar mejorar este porcentaje. Es posible que para poder mejorar esta eficiencia sea más conveniente mejorar la eficiencia del proceso de preprocesado que suministra las imágenes a este reconocedor OCR para que estas tengan una mejor calidad.

REFERENCIAS

- [1] <http://www.magrama.gob.es/es/ganaderia/temas/trazabilidad-animal/identificacion-animal/bovino/>
- [2] <http://rsb.info.nih.gov/ij/>
- [3] <http://rsb.info.nih.gov/nih-image/>
- [4] <http://tess4j.sourceforge.net/>
- [5] An Overview of the Tesseract OCR Engine, Ray Smith, ICDAR, 2007
- [6] Canonical Ltd. (February 2011). "OCR". Retrieved 2011-02-11.
- [7] Willis, Nathan (September 2006). "Google's Tesseract OCR engine is a quantum leap forward". Retrieved 2008-07-18.