|  |  |  |
| --- | --- | --- |
| **T3** | **# 11 Areli Cervantes Flores**  **# 23 José Javier Muñoz Cabrera** | **8 A** |

**Introducción**

El uso de aplicaciones OCR permite extraer el contenido textual de documentos es una tecnología que trata de emular el ojo humano para reconocer objetos. OCR es un software que permite el reconocimiento óptico de los caracteres de una imagen (documento, fotografía escaneada), de forma que estos se vuelven comprensibles o reconocibles para un ordenador, obteniendo como resultado final un archivo en formato de texto editable. El formato del archivo de salida puede ser un (txt, pdf, etc.) dependiendo de las características del software.

**Como funciona OCR**

Para reconocer los caracteres, el software inspecciona la imagen pixel a pixel, buscando formas que coincidan con los rasgos de los caracteres. En función del nivel de complejidad o grado de desarrollo del software, éste buscará coincidencias con los caracteres y fuentes disponibles en el programa, o tratará de identificar los caracteres a través del análisis de sus características, de forma que el reconocimiento de los mismos no se limite exclusivamente a un determinado número de fuentes.

El OCR puede analizar los elementos del documento (bloques de texto, imágenes, tablas), examinando los espacios en blanco y descomponiendo el texto en líneas, palabras y caracteres.

**Requerimientos**

* Equipo de cómputo
* Spyder(Python 3.5)
* Imágenes segmentadas a partir de MatLab

**Resumen**

El programa funciona de la siguiente manera mediante un conjunto de imágenes previamente segmentadas se generará un dataset el cual contendrá 14 características de cada una de las imágenes. Primero se tomara una imagen prueba de cada una de las carpetas y se eliminara esa imagen de la carpeta. Se guardarán las imágenes de prueba en una carpeta llamada *testset* para que después se puedan clasificar. Después se recorrerá cada una de las imágenes dentro de las carpetas, se calcularan sus características y estas se almacenaran en una matriz bidimensional. Esta matriz se escribirá en un archivo de datos con extensión .csv al cual se le dará un formato para acomodar cada una de las características en filas y columnas y así poder asignarles una clase. Para la clasificación se pedirá la ruta de la imagen que se deseara clasificar se obtendrán características de esta imagen y se comparara utilizando el método de clasificación KNN. Al finalizar este proceso la computadora será capaz de clasificar la imagen y determinar a qué clase pertenece según sus características. Para este ejemplo solo se utilizaran los números del 0 al 9.

**Pasos:**

Para hacer el programa se tomaron como base una serie de imágenes que ya se encontraban segmentadas y divididas en carpetas. Para poder caracterizar estas imágenes y obtener los datos de entrenamiento se realizaron los siguientes pasos:

**Primero**

Para que se puedan realizar las pruebas de clasificación correspondientes se tomó una imagen de prueba de cada una de las carpetas y se guardaron en una carpeta llamada *testset*. Se eliminaron estas imágenes de las carpetas para después poder clasificarlas.

**Segundo**

Lo primero que se tuvo que hacer después de tomar las imágenes de prueba fue recorrer imagen por imagen dentro de cada una de las carpetas.

**Tercero**

Ya que una imagen es una matriz de píxeles lo que se hizo después fue obtener el tamaño de la imagen. Posteriormente se tuvo que recorrer cada una de las posiciones de la matriz y validar el número de unos en toda la imagen. Esto se dividió entre el tamaño de la imagen y el resultado se tomó como la primera característica para el dataset.

La segunda característica se obtuvo calculando el ancho de la imagen sobre el alto de la misma.

La tercera característica se calculó dividiendo el ancho de la imagen entre dos para así obtener el número de unos que hay a la mitad de la imagen de forma vertical. Después se dividió el total de unos encontrados entre el tamaño de la imagen.

La cuarta característica se calculó dividiendo el ancho de la imagen entre 4 para así obtener el número de unos que hay a 1/4 de la imagen de forma vertical. Después se dividió el total de unos encontrados entre el tamaño de la imagen.

La quinta característica se calculó dividiendo el ancho de la imagen entre 4 y multiplicándolo por dos para así obtener el número de unos que hay a 3/4 de la imagen de forma vertical. Después se dividió el total de unos encontrados entre el tamaño de la imagen.

La sexta característica se calculó dividiendo el alto de la imagen entre dos para así obtener el número de unos que hay a la mitad de la imagen de forma Horizontal. Después se dividió el total de unos encontrados entre el tamaño de la imagen.

La séptima característica se calculó dividiendo el alto de la imagen entre 4 para así obtener el número de unos que hay a 1/4 de la imagen de forma Horizontal. Después se dividió el total de unos encontrados entre el tamaño de la imagen.

La octava característica se calculó dividiendo el alto de la imagen entre 4 y multiplicándolo por dos para así obtener el número de unos que hay a 3/4 de la imagen de forma horizontal. Después se dividió el total de unos encontrados entre el tamaño de la imagen.

La novena característica es la relación de cortes que se encuentran a la mitad de la imagen de forma vertical, es decir los cambios que hace entre unos y ceros con respecto al tamaño total de la imagen.

La décima característica es la relación de cortes que se encuentran a 1/4 de la imagen de forma vertical.

La décima primera característica es la relación de cortes que se encuentran a 3/4 de la imagen de forma vertical.

La décima segunda característica es la relación de cortes que se encuentran a la mitad de la imagen de forma horizontal.

La décima tercera característica es la relación de cortes que se encuentran a 1/4 de la imagen de forma horizontal.

La décima cuarta característica es la relación de cortes que se encuentran a 3/4 de la imagen de forma horizontal.

Esto se realizó para cada una de las imágenes. Al mismo tiempo que se obtienen las características estas se van almacenando en una matriz bidimensional.

**Cuarto**

Una vez obtenidas todas las características y ordenadas en la matriz, se generara un archivo de datos con extensión .csv en donde se pasaran los datos de la matriz ordenando estos por filas y columnas. Esto para poder etiquetarlos según la clase a la que pertenezcan. Al final se generó el archivo con todas las clases y sus características en la carpeta en donde se encuentra el programa.

**Quinto**

Después de haber generado el *dataset* se le pedirá al usuario la ruta de la imagen prueba para clasificarla. Una vez obtenida la imagen se calcularan sus características y se preguntara al usuario cuantos vecinos gustaría tomar para aplicar el método de clasificación *knn* en base al *dataset* ya generado. *Knn* es un método de clasificación que sirve para etiquetar una nueva instancia según las características que comparte con sus vecinos más cercanos es decir se etiquetan según las características más comunes a la clase a la que más se aproxime según su distancia euclidiana con respecto a las clases.

**Sexto**

Por último el programa nos imprimirá en pantalla como resultado la distancia euclidiana de la nueva instancia y la clase a la que pertenece según sus características.