

PLATAFORMA DE EVALUACIÓN DE ALGORITMOS DE RECONOCIMIENTO DE CARACTERES NUMÉRICOS EN IMÁGENES DIGITALES

Ing. Janeth Arias, Ing. Oscar Guaypatin Pico, Alberto Arellano
Universidad Técnica de Cotopaxi, Escuela Superior Politécnica de Chimborazo

janeth.arias@utc.edu.ec, oscar.guaypatin@utc.edu.ec, aarellano@esPOCH.edu.ec

Resumen

En este trabajo se presenta un estudio comparativo entre tres algoritmos de reconocimiento de caracteres numéricos en imágenes digitales: KNN, TESSERACT y Redes Neuronales Artificiales, con el fin de determinar el de mayor grado de precisión, utilizando el método inductivo y experimental se midieron los siguientes indicadores: nivel de precisión, tiempo empleado en el reconocimiento, consumo de memoria ram y consumo de cpu. Se utilizó OpenCV, Python y C++ para el desarrollo de una aplicación que evaluó cada una de las técnicas indicadas en nuestra muestra ponderada de 30 fotografías tomadas a medidores de energía eléctrica. Como resultado de la investigación se determinó que KNN es la técnica con mayor porcentaje de precisión alcanzando el 49,33%, el tiempo promedio empleado para el reconocimiento fue de 1,2 segundos, el consumo de memoria ram fue en promedio 15,7 megabytes, y el uso de cpu fue de 11,64%.

Abstract

This paper presents a comparative study between three numerical character recognition algorithms in digital images: KNN, TESSERACT and Artificial Neural Networks, in order to determine the highest degree of accuracy, using the inductive and experimental methods were measured Indicated: accuracy level, time spent on recognition, ram consumption and cpu consumption. OpenCV, Python and C++ were used for the development of an application that evaluated each of the techniques indicated in our weighted sample of 30 photographs taken at electric power meters. As a result of the investigation it was determined that KNN is the technique with the highest percentage of accuracy reaching 49.33%, the average time used for the recognition was 1.2 seconds, ram memory consumption was on average 15.7 megabytes, And the use of cpu was 11.64%.

Keywords: Knn, Tesseract, Opencv, Artificial Neuronal Networks, Python.

I. INTRODUCCIÓN

Como se indica en [1], [2] y [3], la visión artificial o visión por computador tiene un amplio campo de aplicación en áreas tan diversas como la medicina, la agricultura, redes eléctricas inteligentes, entre otras; utilizándose en tareas como el reconocimiento de patrones, la detección de movimiento, el reconocimiento de caracteres, etc.

El Reconocimiento de Caracteres, OCR (Optical Character Recognition), es una tecnología que engloba un conjunto de técnicas basadas en estadísticas, en las formas de los caracteres,

transformadas y en comparaciones, que complementándose entre sí, se emplean para distinguir de forma automática entre los diferentes caracteres alfanuméricos existentes en una imagen digital [4].

Para las aplicaciones de reconocimiento de dígitos existen una variedad de algoritmos, el presente trabajo de investigación se centra en el desarrollo de una plataforma basada en un computador de bajo recursos que permita estudiar y evaluar los algoritmos de aprendizaje supervisado: KNN y ANN, por su simplicidad y grado de precisión en el reconocimiento de caracteres; y contrastarlos frente a Tesseract [5],

una aplicación opensource de reconocimiento óptico de caracteres, inicialmente desarrollado por HP y que actualmente google continua con su desarrollo.

Como es de suponerse, los recursos computacionales en los computadores de placa reducida(SBC) como la Raspberry Pi, son muy escasos, por esta razón es de fundamental importancia determinar el algoritmo de reconocimiento de dígitos más idóneo que mantenga un alto grado de precisión y un bajo consumo de los recursos computacionales de los que disponen esta tipo de computadores.

II. MARCO TEÓRICO

La visión por computador lleva asociada una enorme cantidad de conceptos relacionados con hardware, software y también con desarrollos teóricos. En esta sección se verán los pasos fundamentales para llevar a cabo una tarea de visión artificial.

Para cumplir satisfactoriamente su tarea, todo sistema de reconocimiento de caracteres debe realizar cuatro fases, como se muestra en la Figura 1.



Figura 1. Fases del Sistema de Reconocimiento de Caracteres.

La fase de pre procesamiento tiene como objetivo principal mejorar la imagen de forma que el objetivo final de la tarea tenga mayores posibilidades de éxito.

La fase de segmentación tiene como meta principal la detección de los contornos de los caracteres o símbolos de la imagen tomando en cuenta la información de intensidad de los

mismos. Además permite la descomposición del texto en diferentes entidades lógicas suficientemente invariantes para no depender del estilo de la escritura y que puedan ser suficientemente significativas para reconocerlas. Los principales métodos utilizados para esta fase son: método de agrupamiento, método basado en histogramas, método de crecimiento de regiones, método de particionamiento gráfico entre otros.

Una vez realizada la segmentación, se tiene una imagen normalizada en la que se encuentra la información susceptible de ser “reconocida”. La información así representada, una matriz bidimensional de valores binarios, niveles de gris o color RGB, no codifica de forma óptima las características más discriminativas del objeto al que representa. La extracción de las características es una de las fases más difíciles en los sistemas de reconocimiento de caracteres, puesto que es muy difícil escoger un conjunto de características óptimo. Para esta fase se utilizan entre otros los siguientes métodos: análisis de componentes principales (PCA), análisis discriminante lineal (LDA), análisis independiente de componentes (ICA) y análisis discriminante no lineal (NDA) [4].

Una vez que se obtienen las características más importantes de cada uno de los caracteres incluidos en la imagen digital, el siguiente paso es realizar el reconocimiento del mismo en base a un patrón predefinido, básicamente existen dos métodos de reconocimiento: offline y online. Los métodos offline más utilizados son: Clustering, Feature Extraction, Pattern Matching y Redes Neuronales, mientras que el clasificador k-NN y el algoritmo basado en dirección son los métodos online más comunes [6].

OpenCV (Open source Computer Vision) es una librería abierta que contiene más de 500 algoritmos optimizados para el análisis de imágenes y video. Desde su apareamiento en 1999, esta librería ha sido adoptada como herramienta de desarrollo por la comunidad de investigadores en el área de visión por computador. Originalmente fue desarrollada por Intel bajo la supervisión de Gary Bradski jefe de grupo de investigación de visión artificial avanzada [7].

III. METODOLOGIA DE LA INVESTIGACION

La presente investigación es del tipo cuasi experimental ya que se escoge la técnica de reconociendo de dígitos más adecuada en imágenes para lo cual se evaluarán: KNN, Tesseract y Redes Neuronales Artificiales en fotografías tomadas a medidores de energía eléctrica con el objetivo de obtener la lectura de consumo de manera automática, además los datos de las pruebas realizadas son generados por el autor de esta investigación.

De acuerdo a [8], los diseños cuasi experimentales manipulan deliberadamente al menos una variable independiente para observar su efecto y relación con una o más variables dependientes. En el caso específico de la presente investigación se determinará el porcentaje de precisión en el reconocimiento de los dígitos así como el tiempo, el consumo de memoria ram y la cantidad de procesamiento utilizados para esta tarea.

Para la verificación del adecuado nivel de precisión en el reconocimiento de dígitos en imágenes se seleccionó como población y muestra una serie de fotografías a medidores de energía de marca Hiking de la Empresa Eléctrica Riobamba, se realizaron 2 tomas diarias una en la mañana y otra en la tarde durante 25 días. La figura 2 muestra la imagen original del medidor.



Figura 2. Medidor de Energía Eléctrica

Una vez que las herramientas software fueron seleccionadas, se procedió a desarrollar la aplicación de visión artificial que permita verificar el adecuado funcionamiento de cada una de las técnicas de reconocimiento de dígitos seleccionadas, la figura 3 muestra las fases involucradas en esta aplicación.

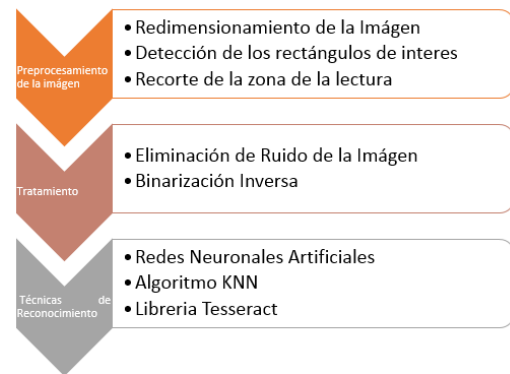


Figura 3. Fases del Reconocimiento de Dígitos.

Para el redimensionamiento de la imagen, las fotografías de muestra de los medidores de energía eléctrica fueron tomas en un rango de 15 cm a 30 cm con lo que el tamaño de la imagen varia notablemente, mediante varias pruebas se determinó que el tamaño adecuado para que la imagen seleccionada tenga un nivel adecuado de reconocimiento debe ser de 453 x 487 pixeles.

Para segmentar la fotografía redimensionada del medidor de energía eléctrica, se utilizó el método de localización de objetos basado en el algoritmo para detectar blobs, los objetos dentro de la imagen se delimitaron en rectángulos que podrían contener áreas de interés como por ejemplo la zona que muestra el consumo de energía en la imagen. En esta fase del pre procesamiento se utilizan las funciones de dilatación y erosión de OpenCV. La figura 4 muestra el resultado de esta operación.



Figura 4. Detección de Blobs en la imagen

Para finalizar esta primera etapa se recorta el blobs de forma rectangular en el cual se encuentra la medición del consumo de energía, desde la posición inicial 144x170 pixeles hasta la posición final 315x201 pixeles, la figura 5 muestra el blob seleccionado, para esto se utiliza la función getRectSubPix.



Figura 5. Lectura obtenida de la imagen

Uno de los problemas comunes que se presentan al momento de analizar imágenes obtenidas desde dispositivos de captura como las cámaras fotográficas es el ruido de compresión, cuantización y de sensibilidad del sensor de la cámara. Para resolver este problema existen varias técnicas, una de ellas es el suavizado de la imagen mediante la aplicación de filtros, para completar esta fase se utilizó el filtro gaussiano. La figura 6 muestra el resultado de esta operación:



Figura 6. Filtro Gaussiano aplicado a la imagen

La binarización inversa es un proceso previo a la utilización de las funciones para el reconocimiento de patrones, que se basan en la detección de puntos blancos sobre fondos negros. Debido a que los caracteres de la imagen de la figura 6 son blancos en fondo negro, es necesario binarizar la imagen y posteriormente hallar su complemento. La figura 7 muestra este resultado.



Figura 7. Binarización inversa de la imagen

IV. RESULTADOS

Para determinar la técnica más adecuada en el reconocimiento de dígitos en imágenes se establecieron tres escenarios de pruebas con 30 muestras ponderadas de imágenes tomadas a

medidores de energía eléctrica, los escenarios planteados se detallan a continuación:

- Escenario 1: Reconocimiento mediante KNN
- Escenario 2: Reconocimiento mediante Tesseract
- Escenario 3: Reconocimiento mediante Redes Neuronales Artificiales

En los escenarios planteados, se mide los efectos que la variable independiente produce en la variable dependiente, para esto los parámetros que se establecen como indicadores son los siguientes:

- **I1. Precisión:** Relación entre el número de dígitos reconocidos sobre el número total de dígitos en la imagen original.
- **I2. Tiempo Procesamiento:** Tiempo en segundos que se tarda cada una de las técnicas seleccionadas en realizar el proceso de reconocimiento.
- **I3. Consumo RAM:** Cantidad de memoria RAM del computador SBC utilizada para el proceso de reconocimiento de cada una de las algoritmos.
- **I4. Consumo CPU:** Porcentaje de consumo de CPU del computador SBC utilizado para el proceso de reconocimiento de cada una de los algoritmos.

Las figuras 8, 9,10 y 11 muestran los resultados de las pruebas realizadas con cada uno de los indicadores mencionados anteriormente.

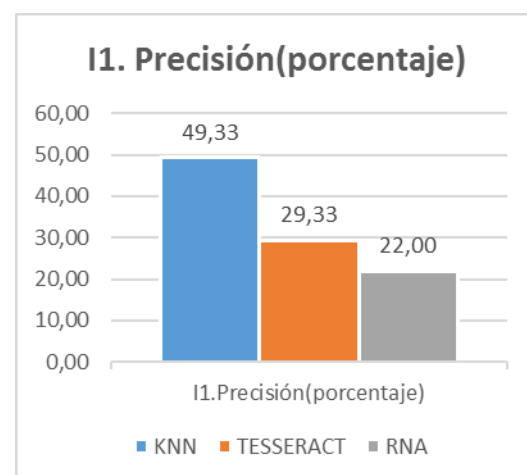


Figura 8. Porcentaje de Precisión

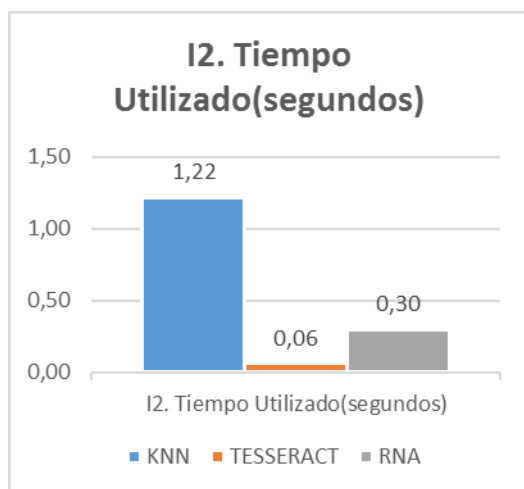


Figura 9. Tiempo utilizado por cada Algoritmo

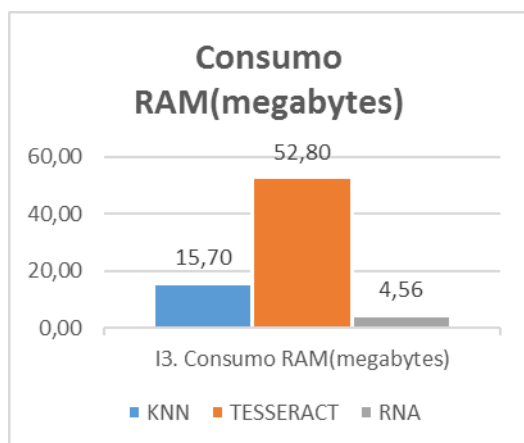


Figura 10. Consumo de memoria RAM de cada Algoritmo.

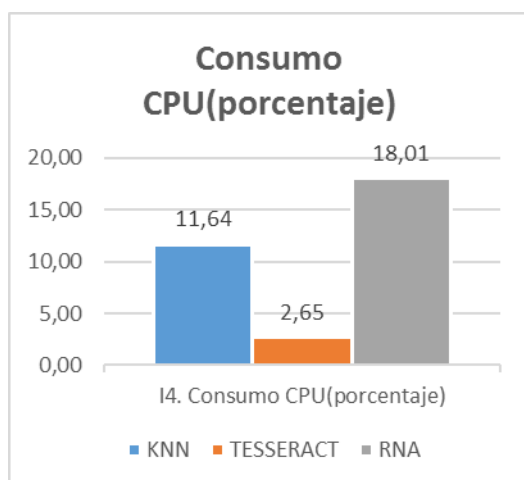


Figura 11. Consumo de CPU de cada Algoritmo.

V. CONCLUSIONES

- Se desarrolló una plataforma opensource mediante el sistema operativo GNU/LINUX, la librería de visión artificial OpenCV y los lenguajes de programación Python y C++, para la evaluación del nivel de precisión en el reconocimiento de la lectura del consumo de las imágenes tomadas a medidores de energía eléctrica
- En las pruebas realizadas para determinar la técnica de reconocimiento de caracteres numéricos con mayor grado de precisión se utilizó una muestra ponderada de treinta fotografías tomadas a medidores de energía eléctrica en las cuales el consumo se indica en un número de cinco dígitos; obteniéndose los siguientes resultados: 49,33% de precisión con la técnica KNN, 29,33% con TESSERACT y 22% con Redes Neuronales Artificiales, utilizando un tiempo promedio de 1.22 , 0.06 y 0.3 segundos respectivamente en cada una de las técnicas señaladas.
- En lo relacionado con el consumo de recursos utilizados para el reconocimiento de los dígitos los resultados fueron los siguientes: KNN consumió en promedio 15.7 megabytes de memoria RAM y 11.64% de consumo de CPU; TESSERACT consumió 52.8 megabytes de memoria RAM y 2.65% de CPU; finalmente las Redes Neuronales Artificiales consumieron 4.56 megabytes de memoria RAM y 18.01% de CPU.
- De los resultados obtenidos de la prueba no paramétrica de Kruskal-Wallis se pudo determinar que la técnica que mejor rango promedio presenta es KNN esto se corroboró con el porcentaje de precisión que se obtuvo al momento de realizar el reconocimiento de los dígitos por cada una de los algoritmos seleccionados.

REFERENCIAS

- [1] T. Moeslund y E. Granum, «A Survey of Computer Vision-Based Human Motion Capture,» *Computer Vision and Image Understanding*, 2001.
- [2] G. Pajares, VISION POR COMPUTADOR: IMAGENES DIGITALES Y APLICACIONES, Mexico: RA-MA, 2001.
- [3] R. López Beltrán y E. Sotter Solano, «Aplicación del sistema Robot Vision PRO para operaciones automáticas de control de calidad,» *Revista Científica Ingeniería y Desarrollo*, n° 9, 2001.
- [4] C. J. Sánchez Fernández y V. Sandonís Consuegra, «Reconocimiento Óptico de Caracteres (OCR),» *Universidad Carlos III*, 2014.
- [5] M. Nayak y N. Ajit Kumar, «Odia Characters Recognition by Training Tesseract OCR,» *International Journal of Computer Applications* , 2013.
- [6] D. Suruchi y C. Anjali , «Survey of Methods for Character Recognition,» *International Journal of Engineering and Innovative Technology (IJEIT)*, vol. 1, 2012.
- [7] R. Laganière, OpenCV Computer Vision Application, Packt Publishing, 2014.
- [8] R. Hernández Sampieri, Metodología de la Investigación, Mexico: McGRAW-HILL, 2010.