

## Tarea N° 3 Procesamiento de Señales e Imágenes

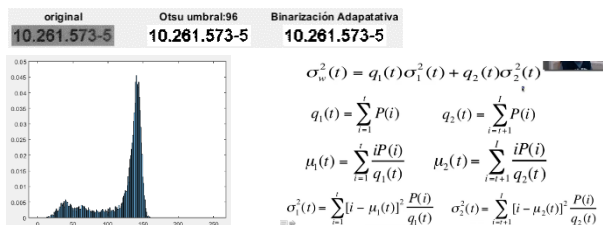
Nicolas Farfán Cheneaux  
Universidad de Santiago de Chile

**Motivación** - El reconocimiento de imágenes digitales es un área importante que hoy en día se ve reforzado por la inteligencia artificial y el aprendizaje automático, tiene diversas utilidades tanto en la comunidad científica y cotidiana, en el presente trabajo se hace un acercamiento al reconocimiento de dígitos mediante el procesamiento de imágenes y su posterior calificación y análisis.

### 1. Contexto del problema, alcances

La tarea por realizar es reconocer los dígitos de un Rut dada una imagen, las máquinas por sí solas no pueden entender el lenguaje humano, pero con una base de conocimiento planteada o modelos “ideales” es posible aplicar técnicas de reconocimiento. Cabe recalcar que el presente Proyecto se rige por los símbolos entregados y estos no pueden ser ideales o parecidos a los que se va a leer por lo que puede haber imprecisiones en la lectura e identificación, además el programa ha sido probado solo con los ejemplos dados, más no con otros Ruts por lo que está sujeto a errores.

2.1 Preparación de la imagen: Lo primero es binarizar la imagen es decir separar los elementos importantes del fondo que no es relevante, y los dígitos, para esto se implementó el algoritmo Otsu el cual generó el siguiente histograma de umbral para el siguiente Rut como ejemplo.



El siguiente paso es identificar los componentes conexos, y dado que está prohibido utilizar la función **regiónprops**, se utiliza una estrategia transformando los índices usados por el vector fila resultante para obtener las bounding boxes de cada componente y generar el arreglo pedido, de la siguiente forma

Componentes Conexos  
10.261.573-5

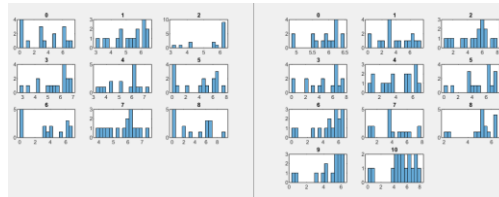
Una vez obtenidos las bounding boxes y coloreadas de cada componente conexo se procede a extraer los bordes para poder identificar cada dígito, se ve de la siguiente forma:

Bordes Componentes  
10.261.573-5

2.2 Estrategia de reconocimiento de la imagen: Lo primero es dividir cada componente en 16 partes iguales / cuadrantes, es decir se redimensiona la imagen a 128x64 pixeles, lo cual resulta en cada componente de tamaño 32x16 pixeles, esto se realiza para cada modelo y componente.

**Para cada modelo:** Se guarda el porcentaje de 1's o partes de color blanco de cada cuadrante, es decir un arreglo bidimensional de 16 filas y 11 columnas, al ser 16 el número de cuadrantes de cada imagen y 11 el número de modelos, la parte blanca simboliza el borde de la imagen, finalmente se divide esto por el número de muestras y se obtiene un promedio “ideal” que cada dígito a evaluar más adelante debería cumplir.

**Para cada componente o dígito identificado:** Se utiliza exactamente la misma estrategia de dividir los cuadrantes en 16 y obtener el porcentaje de borde de cada cuadrante y se guarda en un arreglo de 16x9, 9 dígitos a evaluar y 16 cuadrantes por c/u, se puede observar los histogramas de 16 bandas y su concatenación.



**Identificación del símbolo:** Finalmente después de tener guardado ambos arreglos, para cada símbolo se hace una diferencia con cada modelo para cada cuadrante y se obtiene su valor absoluto de esta resta es decir en *cuanto difiere* (sea más / menos para finalmente calcular el mínimo de estas diferencias, el cual será el dígito que más se aproxima al que se está evaluando.

Cabe resaltar que existe diferencias entre los modelos e imágenes de Rut entregados por ejemplo el caso del 3, 5, 6 como se puede observar tiene varias diferencias, por lo que la solución propuesta no identificará correctamente estos dígitos.



Cómo se puede apreciar existe varias distinciones con los siguientes dígitos por lo que van a existir fallas en la detección del símbolo. Sin embargo, los demás símbolos que tienen características similares los reconoce sin problemas. Se evalúan todos los resultados de todas las imágenes de prueba:

Rut	Otsu	Matches	Adaptativo	Matches
15.407.110-5	186871188	3	101071100	4
10.261.573-5	102018780	5	102010700	5
24.017.026-4	210170009	5	210170204	6
24.017.026-4	280170206	6	210170204	7
10.261.573-5	102010700	5	102010700	5
10.261.573-5	182818788	4	102010700	5
24.017.026-4	288178286	4	210170201	6
10.457.232-4	104072004	6	101870014	4
15.068.718-7	100087187	7	100007107	5

**Ventajas y desventajas, solución propuesta:**

Se puede observar de la tabla que existe un mejor desempeño de matches usando el algoritmo adaptativo 47 vs 45, sin embargo estos resultados son similares y no difieren ni tienen un gran impacto en el resultado, el método usado es el que determina el desempeño, esta solución es simple dado que aproxima muy de cerca la mayoría de los números a evaluar, por lo que se puede implementar fácilmente las desventajas son que se aproxima correctamente a algunos símbolos parecidos porque los modelos no están adecuados al input que se va a tener, por lo que tiene alta tasa de error.

### 2. Conclusión

El proyecto ha sido terminado exitosamente y se ha logrado aprender y aplicar nuevas técnicas de procesamiento y reconocimiento de imágenes, si bien los resultados no son precisos al 100% se podrían mejorar ajustando los modelos y detectando singularidades en algunos dígitos.

**Referencia Alg. Otsu:**

<http://www.labbbookpages.co.uk/software/imgProc/otsuThreshold.html>