

# Procesamiento Digital: Matching de imágenes Template

Matías Villalón

**Abstract**—En este informe se presenta el método, resultado y conclusión obtenidos durante la aplicación de la técnica Template Matching, con el fin de identificar patrones dentro de un conjunto de 14328 imágenes, buscando una precisión aceptable y bajo uso de recursos computacionales como ram y CPU. Los resultados obtenidos son producto del preprocesamiento combinado con el uso de la transformada de wavelet, logrando un análisis y reconocimiento de bajos recursos a costa de una reducida precisión.

**Index Terms**—Wavelet, Template, Matching, Preprocesamiento, Redimensionar, Reconocimiento de características.

## I. INTRODUCCIÓN

EL reconocimiento de imágenes es un área del procesamiento digital, siendo utilizada en distintos ámbitos, tales como seguridad, automatización y análisis de imágenes. Una de las técnicas más utilizadas en este ámbito es el Template Matching, el cual permite identificar los patrones dentro de una imagen mediante el uso de las plantillas del tipo Template. Esta técnica, se basa en la correlación, resultando efectiva para identificar imágenes con características similares, siendo principalmente usada cuando se busca un bajo requerimiento de parte del equipo utilizado. El presente informe busca explorar estrategias usando Template Matching que mejoren la eficiencia de la técnica, demostrando su uso con una gran cantidad de imágenes en el menor tiempo posible.

## II. MARCO TEÓRICO

El Template Matching es una técnica usada para reconocer objetos o características dentro de imágenes, basándose en la comparación entre una plantilla Template y distintas zonas de la imagen. La comparación se centra principalmente en calcular el grado de similitud entre el Template y la imagen, lo que permite identificar donde se encuentra el patrón buscado. Aunque es una técnica simple de realizar, su utilidad se encuentra en lo fácil de adaptarla a diferentes contextos dado su baja necesidad de recursos en casos como la detección de formas y análisis de características en las imágenes. Para mejorar su eficiencia, se suele manejar las imágenes en escala de grises o aplicar métodos de preprocesamiento como el redimensionamiento, disminuyendo la cantidad de información procesada y reduciendo el uso de ram y procesador..

This paper was produced by the IEEE Publication Technology Group. They are in Piscataway, NJ.

Manuscript received April 19, 2021; revised August 16, 2021.

## III. MÉTODO EXPERIMENTAL

Durante el trabajo, se utilizaron distintos procedimientos en las imágenes, estos fueron realizados con el fin de facilitar el análisis de la imagen y poder clasificar si correspondían a uno de los tres modelos de los templates, siendo estos del tipo circ, box o cross. A continuación se detallan los múltiples pasos aplicados a las imágenes y a los templates, abarcando desde las transformaciones hasta el proceso de matching y clasificación.

- **Conversión a 128x128 pixeles** Los datos de las imágenes del archivo ZIP utilizado para el trabajo fueron redimensionados a 128x128 pixeles, una gran diferencia en cuanto a su versión original de 512x512 pixeles.
- **Aplicación de Wavelet** Se aplica la transformación de la Wavelet a las imágenes, con el fin de reducir el ruido y acelerar el procesamiento al trabajar con sus coeficientes.
- **Normalización de los datos** Normalizamos los datos obtenidos de los coeficientes para evitar problemas con el posterior matching, ya que dejamos todos los datos entre 0 y 255.
- **Matching** Se realiza una iteración de las imágenes y plantillas procesadas con wavelet, utilizando cv2.matchTemplate para comparar la correlación de las imágenes y la plantilla en diferentes posiciones, e interpretar los resultados, se aprovecha el uso de un umbral para intentar reducir errores.
- **Clasificación** finalmente los datos obtenidos son comparados para comprobar el porcentaje de error del proceso realizado, esto se realiza mediante la identificación anterior de si era un box, circ o cross y comparando dicho resultado con el nombre que le correspondía originalmente, en caso de que el tipo de objeto asignado encaje con su nombre se considera un acierto, caso contrario es un error.

## IV. RESULTADOS

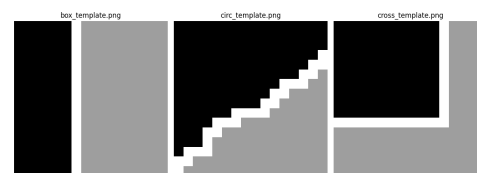


Fig. 1. Templates en escala de grises.

La figura 1 muestra las templates utilizadas como plantillas para el matching en escala de grises, en base a estas formas es que se identificara a que tipo pertenece cada una de las imágenes del archivo zip.

La estrategia re dimensionar las imágenes y utilizar las transformaciones de la wavelet resulto ser de gran utilidad para acelerar el proceso, se lograron reducir los tiempos de análisis exitosamente, dando como resultados entre 6 – 7 segundos en el matching dada la gran cantidad de elementos innecesarios que elimina esta combinación. Lamentablemente el costo de aumentar la capacidad y mejorar el rendimiento fue la capacidad de determinar correctamente a que tipo pertenece cada imagen, dando como resultado una tasa de error cercana al 50%.

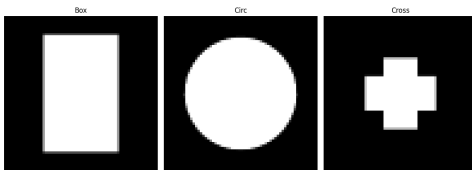


Fig. 2. Batch con wavelets.

La imagen 2, muestra las imágenes extraídas del archivo zip posterior a haber aplicado re dimensionamiento, transformada wavelet y normalización, resultan fácilmente identificables y no han perdido ninguna de sus características de forma que dificulte su reconocimiento.

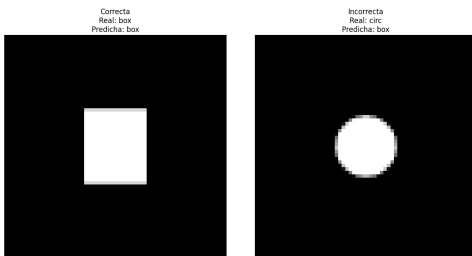


Fig. 3. Comparación de imagen correcta e incorrecta.

Finalmente, la figura 3 muestra una imagen correctamente identificada en el lado izquierdo y una imagen incorrecta en el lado izquierdo, a la vez que se indica que la predicción de la imagen incorrecta correspondía a un tipo box mientras que la imagen real correspondía a una imagen del tipo circ.

## V. CONCLUSIÓN

En conclusión, el pre procesamiento de imágenes resulta útil para reducir los tiempos necesarios para identificar a que tipo de imágenes nos enfrentamos, en este caso se logro parcialmente cumplir con el objetivo, ya que se logro un tiempo sumamente bajo para el procesamiento de las 14328 imágenes, lamentablemente este resultado fue a costa de la precisión al momento de identificar el tipo de imagen correctamente. Para futuras referencias seria recomendable el sacrificio de una parte de la velocidad de procesamiento a cambio de una mayor precisión, algunas técnicas que pudieron haber sido aplicadas para mejorar este proceso son la binarización dado que las imagenes se veian en solo dos colores hubiera sido una alternativa fiable.

## REFERENCES

- [1] *Template Matching..* Template Matching [https://docs.adaptive-vision.com/4.7/studio/machine\\_vision\\_guide/TemplateMatching.html](https://docs.adaptive-vision.com/4.7/studio/machine_vision_guide/TemplateMatching.html). [web].
- [2] *OpenCV Matching.* [https://docs.opencv.org/4.x/d4/dc6/tutorial\\_py\\_template\\_matching.html](https://docs.opencv.org/4.x/d4/dc6/tutorial_py_template_matching.html) [web].
- [3] *Cross correlation..* <https://www.scientific.net/AMR.989-994.3827> [web].
- [4] *¿Que es image matching?.* <https://blog.roboflow.com/image-matching/> [web].