



UNIVERSIDAD PONTIFICIA CATÓLICA DE VALPARAÍSO

## Proyecto II

PROCESAMIENTO DIGITAL MULTIMEDIA – EIE401

Profesor: Jorge Cárdenas

### 1. The Fastest Correlation Verification

En diferentes procesos en ingeniería y astronomía, se está usando de forma más recurrente la de identificación de patrones en imágenes. Por ejemplo, en la inspección de piezas mecánicas para identificación de fallos, circuitos electrónicos para identificar componentes, o en astronomía para identificación de objetos estelares[1].

Para realizar estas tareas de identificación de patrones, se utilizan varias técnicas, entre ellas el *template matching*. Esto puede tener multiples variación dependiendo del tratamiento que se haga a la imagen de forma preliminar, por ejemplo cambios de color, canales, transformaciones etc, [2]. Conceptualmente debes usar unas imágenes de referencia o plantillas, para que, al realizar una operación sobre la imagen de tu interés, puedas calcular el nivel de similitud entre la imagen y tu patrón de referencia [4].

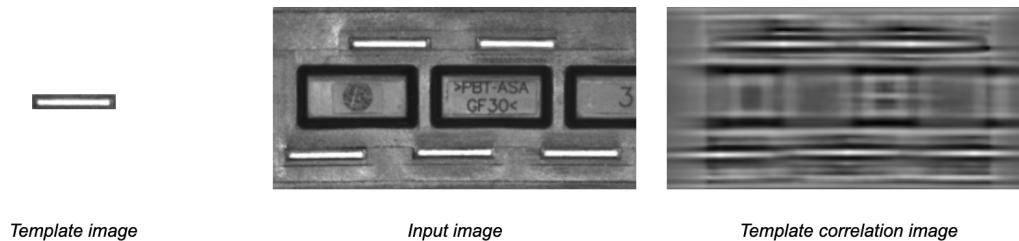


Figura 1: Template matching para reconocimiento de patrones.

Como es habitual, estos procesos no se realizan sobre una imagen sino sobre conjuntos de datos grandes. En este caso debes tomar el set de datos proporcionado con imágenes de 3 clases: cross, box, y circle. Las imágenes tienen unas medidas de 512x512, y tu objetivo es clasificarlas a partir de 3 imágenes *template* proporcionadas con medidas 32x32.

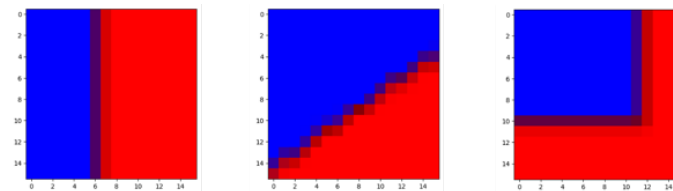


Figura 2: Patrones de referencia

Desarrolla el código para procesar las imágenes y hacer la operación de matching con los templates de tal forma que puedas identificar a que clase pertenece cada imagen[3]. El *matching* se puede calcular usando una medida de correlación, a partir de la cual, deberás inferir que tanto se aproxima la imagen a tus patrones de referencia. Debes

cuantificar el error en la clasificación, es decir saber cuantas clasificaste bien o mal, y debes presentar dicho error. Así mismo, debido a que el proceso involucra tantas imágenes, debes buscar la forma más rápida de hacer tu proceso. Para esto puedes recurrir a técnicas de multi-procesamiento en python o a operación en batches con pytorch.

**NOTA** En este caso vamos a evaluar a partir del mejor resultado de clasificación y la menor velocidad de procesamiento. El resultado de clasificación debe calcularse como un número entre 0 y 1, a partir del número de éxitos de clasificación y el número total de imágenes procesadas. En el notebook del proyecto, encontrarás la forma de medir el tiempo de procesamiento de tu código.

**NOTA** Dichos valores debe quedar **evidentes** en tu reporte escrito, y deben coincidir con los que aparezcan en el notebook.

**NOTA** Debido a la cantidad de imágenes y el tiempo de procesamiento requerido para cada trabajo, deben asegurarse de guardar el notebook con todo el procesamiento y las imágenes generadas de muestra que evidencien tu procedimiento.

- **Desarrollo de la solución (70 %):** En el siguiente link encuentras el notebook que te puede servir como guía para el trabajo [https://github.com/Jorgecardenas1/EIE\\_401\\_MULTIMEDIA/tree/main/Proyectos/Proyecto%202](https://github.com/Jorgecardenas1/EIE_401_MULTIMEDIA/tree/main/Proyectos/Proyecto%202)
  - El trabajo debe realizarse en Jupyter Notebook, donde el orden en la presentación, estructura del código, comentarios y explicaciones el funcionamiento serán evaluados (20 %).
  - Debes incluir todos los archivos para ejecutar tu solución y así comprobar que funcione correctamente. (80 %)
  - La solución debe reposar en tu repositorio personal en GitHub.
  - La hora de entrega será como máximo las 23:59 de la fecha establecida de entrega. Cualquier retraso en la fecha y hora de entrega se sanciona con un 10 % de la nota final del proyecto.
- **Reporte (30 %):** En el mismo enlace encuentras el archivo con el formato para realizar tu informe en Overleaf.
  - Debe realizar un informe de tu trabajo en formato de paper IEEE. Se valora la calidad del texto, la explicación y comprensión del procedimiento y los resultados. (50 %)
  - El informe NO puede ser superior a 2 páginas de contenido (sin incluir referencias). (20 % calificación binaria)
  - El informe debe contener, abstract, introducción, sección de Marco Teórico, Metodología, Resultados y Conclusiones. (20 % calificación binaria)
  - Debe proporcionar por lo menos cuatro (4) referencias para el trabajo. (10 % calificación binaria)
  - El reporte debe ser nombrado Apellido\_Nombre.cédula\_Proyecto#.pdf
  - La hora de entrega será como máximo las 23:59 de la fecha establecida de entrega. Cualquier retraso en la fecha y hora de entrega se sanciona con un 10 % de la nota final del proyecto.

## Referencias

- [1] P Frick et al. *Scaling and correlation analysis of galactic images*. URL: <https://academic.oup.com/mnras/article/327/4/1145/1007769>.
- [2] Ji Jun Sun. "Astronomical image matching based on the cross-correlation algorithm". En: vol. 989-994. Trans Tech Publications Ltd, 2014, págs. 3827-3833. ISBN: 9783038351733. DOI: 10.4028/www.scientific.net/AMR.989-994.3827.
- [3] British Columbia university. *Template Matching*. 2018. URL: [https://www.cs.ubc.ca/~lsigal/teaching18\\_Term1.html](https://www.cs.ubc.ca/~lsigal/teaching18_Term1.html).
- [4] Zebra Adaptive Vision. "Template matching". En: (2017). URL: [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).