

Procesamiento Digital Multimedia: Proyecto 2

Esteban Córdova, *Ingeniería Civil en Telecomunicaciones*

Abstract—En este proyecto, se logra clasificar 14.328 imágenes en tres categorías: box, cross, y circ mediante técnicas de pre-procesamiento y comparación por template matching. El proceso consiste en extraer características desde regiones segmentadas de las imágenes y compararlas con plantillas predefinidas. La implementación propuesta alcanzó una tasa de error de clasificación del 5.72 % con un tiempo total de ejecución de 1 minuto y 54 segundos, demostrando un buen rendimiento tanto en precisión como en eficiencia.

I. INTRODUCCIÓN

El template matching es una técnica que permite identificar patrones dentro de una imagen comparándolos con plantillas de referencia. Esta herramienta es útil en tareas de clasificación donde los objetos tienen formas bien definidas. En este proyecto se utiliza la correlación cruzada normalizada el cual permite una comparación robusta, ya que normaliza la intensidad de la plantilla y la imagen, mitigando variaciones de iluminación, resultando la comparación de cada imagen con los templates box, circ y cross mas eficaz. Mediante el procesamiento por regiones, rotación de plantillas y evaluar la similitud, se busca una clasificación precisa y eficiente sobre el conjunto de 14.328 imágenes.

II. MARCO TEÓRICO

El template matching es una técnica que busca la mejor coincidencia entre una plantilla y partes de una imagen, comparándolas mediante una métrica de similitud, comúnmente la correlación cruzada normalizada. Esta métrica, que varía entre -1 y 1, permite identificar coincidencias robustas ante variaciones de iluminación, aunque es sensible a cambios de escala o rotación. Por ello, en este proyecto se complementa con segmentación por regiones y rotación de plantillas.

Esta técnica se aplica en diversas áreas, como la inspección visual automática en la industria, el reconocimiento de estructuras en astronomía y la relación entre imágenes 2D y estructuras 3D. Su implementación sencilla y bajo costo la hacen útil cuando se requieren soluciones rápidas y efectivas.

III. METODOLOGÍA

Para resolver el problema de clasificación de imágenes, se aplico un enfoque basado en template matching. Inicialmente, se exploraron otras alternativas, como la aplicación de filtros suavizados, normalización por convolución kernel, y binarización previa. Sin embargo, estos métodos no entregaban resultados satisfactorios o presentaban tiempos de ejecución excesivos, como ocurrió al intentar guardar y comparar imágenes binarizadas durante mas de una hora sin éxito. Finalmente, se opto por una estrategia mas simple y eficiente, detallada a continuación:

A. Carga y extracción de imágenes

Se descomprimió un archivo ZIP con 14.328 imágenes de entrada estas se convierten a formato RGB, manteniendo su color.

B. Conversión a escala de grises

Utilizando funciones, los templates fueron convertidos a grises. Del mismo modo, las imágenes fueron convertidas a grises y guardadas en una carpeta llamada imagenes2, con el fin de reducir la complejidad y centrarse en la estructura.

C. Segmentación por regiones

Cada imagen de 512 x 512 pixeles se divide en cuatro cuadrantes: superior izquierda, superior derecha, inferior derecha e inferior izquierda.

D. Procesamiento por region

Cada zona se suaviza mediante un filtro bilateral para preservar bordes y se binariza utilizando un umbral adaptativo (Gaussian)

E. Rotación de plantillas

Se rotan las plantillas (box, cross, circ) en 0°, 90°, 180° y 270° para coincidir con la orientación de cada cuadrante

F. Calculo de similitud

Se aplica la correlación cruzada normalizada entre cada región y su plantilla rotada correspondiente. Se obtiene un valor promedio para cada clase.

G. Clasificación

Se asigna a la imagen la clase cuyo promedio de correlación fue mayor.

H. Evaluación de desempeño

se construye una matriz de confusión y se calcula el porcentaje de error global y por clase. Además muestra el tiempo del proceso template matching.

IV. RESULTADOS

Durante el desarrollo del proyecto se exploraron multiples enfoques para el procesamiento y clasificación de imágenes mediante el template matching. A continuación, se detallan los resultados obtenidos en cada etapa para los intentos realizados y el desempeño del metodo final seleccionado.

A. Enfoques preliminares

Para resolver el problema de clasificación de imágenes, se aplicó un enfoque basado en *template matching*. Inicialmente, se exploraron otras alternativas que se mostrarán a continuación. Sin embargo, estos métodos no entregaron resultados satisfactorios o presentaron tiempos de ejecución excesivos. Finalmente, se optó por una estrategia más simple y eficiente, detallada a continuación:

- **Aplicación de kernel + binarización + normalización:** Aplique una convolución con kernel, seguida de normalización y binarización de las imágenes. Sin embargo, esto arrojó un error de clasificación del 70%, posiblemente debido a la pérdida de información útil durante el binarizado.
- **Eliminación de la binarización y almacenamiento de imágenes:** Intente comparar imágenes en escala de grises sin binarización, guardando los resultados en una carpeta. Este enfoque fue inviable, ya que la ejecución duro por más de 63 minutos sin completarse.
- **Comparación en escala de grises (sin binarización ni suavizado):** Intente una versión más simple, donde solo se convertían las imágenes a escala de grises sin aplicar filtrado ni binarización. Este método redujo el error a 54%, pero fue insuficiente para una clasificación robusta.
- **Aplicación de desenfoque gaussiano accidental:** Sin darme cuenta, aplique un desenfoque gaussiano (*Gaussian blur*) a las imágenes antes del procesamiento por plantilla. Este error provocó que las imágenes de clase *box* fueran clasificadas incorrectamente en todos los casos, generando un 100% de error en esa categoría, aunque el rendimiento en otras clases era óptimo.

B. Metodo Final

El enfoque final utilizó imágenes en escala de grises, aplicó suavizado bilateral por región, binarización adaptativa, y comparación por correlación cruzada normalizada con rotación de plantillas por cuadrante. Cabe destacar que inicialmente se probó este mismo método utilizando imágenes cargadas directamente en memoria, obteniéndose un error ligeramente mayor del 7%. Cuando convertí las imágenes previamente a escala de grises y las almacene en una carpeta, el error se redujo. Con este método, se logró:

- **Error de clasificación global:** 5,72%
- **Tiempo total de ejecución:** 1 minuto y 54 segundos
- **Precisión global:** 94,28%

TABLE I
MATRIZ DE CONFUSIÓN

Real \ Predicha	box	circ	cross
box	3700	289	0
circ	0	4416	0
cross	452	78	5393

Ejemplos correctos:

- Clase real: *box* - Clase inferida: *box*
- Clase real: *circ* - Clase inferida: *circ*
- Clase real: *cross* - Clase inferida: *cross*

Ejemplos incorrectos:

- Clase real: *box* - Clase inferida: *circ*
- Clase real: *cross* - Clase inferida: *box*

C. Visualización de ejemplos

Se incluyen imágenes correctamente clasificadas y ejemplos de clasificación errónea para ilustrar el comportamiento del sistema. Estos ejemplos reflejan tanto el éxito del método propuesto como las limitaciones ante patrones ambiguos. Por ultimo la Figura 3 se puede observar los templates después del procesamiento

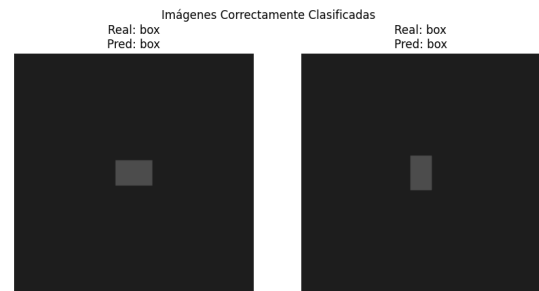


Fig. 1. Imágenes correctamente clasificadas

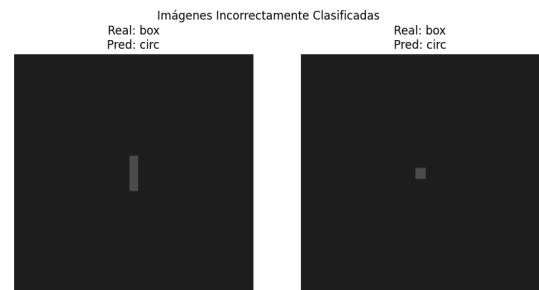


Fig. 2. Imágenes correctamente clasificadas

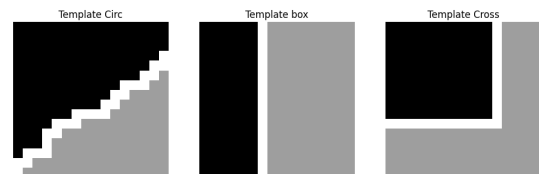


Fig. 3. Templates después del procesamiento

V. CONCLUSIÓN

El presente proyecto demostró la eficacia del método de template matching con preprocesamiento por regiones para la clasificación de imágenes en las categorías *box*, *cross* y *circ*. A pesar de haber explorado inicialmente diversos enfoques, fue la estrategia basada en escala de grises, suavizado bilateral, binarización adaptativa y comparación mediante correlación cruzada normalizada la que logró un equilibrio óptimo entre precisión y eficiencia. Con una tasa de error baja y un tiempo de ejecución rápido, el sistema alcanzó una clasificación confiable sobre 14.328 imágenes. Los resultados obtenidos validan la solidez del enfoque y su potencial aplicación en contextos reales donde se requiera clasificación rápida y precisa.

REFERENCES

- [1] British Columbia university. *Template Matching*. 2018. Available at: https://www.cs.ubc.ca/~lsigal/teaching18_Term1.html
- [2] Zebra Adaptive Vision. "Template matching". En: (2017). https://docs.adaptive-vision.com/4.7/studio/machine_vision_guide/TemplateMatching.html.
- [3] What is Template Matching? An Introduction. <https://blog.roboflow.com/template-matching/>
- [4] Template matching — OpenCV con Python. Available: [omes-va.com/template-matching-opencv-python/](https://docs.opencv.org/4.x/python_tutorials/py_tutorials/py_imgproc/py_template_matching.html)
- [5] Automated spectral classification using template matching To cite this article: Fu-Qing Duan et al 2009 Res. Astron. Astrophys. 9 341