

Reconstrucción de Imagen BMP a través de Transformaciones a Nivel de Bits

Desafío I - Informática II, Semestre 2025-1

Daniel Rúa, Tomás Mesa

Ingeniería Electrónica
Universidad de Antioquia

Resumen—Este documento presenta el desarrollo de un proyecto cuyo objetivo es reconstruir una imagen original a partir de una imagen distorsionada que fue sometida a varias transformaciones a nivel de bits (operaciones XOR, desplazamientos y rotaciones), sin conocer el orden exacto de las mismas. Se detalla el análisis del problema, la propuesta de solución implementada en C++ usando el framework Qt, experimentos realizados y consideraciones sobre la integridad de los datos y la eficiencia del método.

Index Terms—Imagen BMP, Operaciones a nivel de bits, Ingeniería inversa, C++, Qt, Transformaciones, Enmascaramiento.

I. INTRODUCCIÓN

En el contexto de la seguridad digital y el procesamiento de imágenes, es común aplicar técnicas de transformación y enmascaramiento para proteger la información visual. El presente proyecto tiene como objetivo reconstruir una imagen original a partir de una imagen distorsionada mediante el análisis y reversión de una serie de operaciones a nivel de bits aplicadas de forma desconocida. Se utiliza el lenguaje C++ y el framework Qt, haciendo énfasis en el uso de punteros, arreglos y memoria dinámica, sin recurrir al uso de estructuras ni STL.

II. ANÁLISIS DEL PROBLEMA

El desafío consiste en:

- Analizar una imagen BMP de 24 bits denominada ID, resultado de múltiples transformaciones.
- Utilizar una imagen aleatoria (IM) y una máscara (M) para replicar o revertir operaciones aplicadas a la imagen.
- Extraer información de archivos de rastreo generados durante el enmascaramiento.
- Identificar y revertir las operaciones (XOR, desplazamientos y rotaciones) para reconstruir la imagen original.

III. DISEÑO Y METODOLOGÍA DE LA SOLUCIÓN

III-A. Estructura del Algoritmo

Se propone una solución en la que:

1. Se implementan funciones específicas para realizar operaciones a nivel de bits: XOR, desplazamiento y rotación.
2. Se realizan experimentos para analizar el impacto de dichas operaciones en la integridad de los datos.

3. Se verifica el enmascaramiento mediante la comparación de la imagen transformada y la máscara con la información de archivos de rastreo.
4. Se diseña un algoritmo para deducir el orden de las operaciones y revertirlas para obtener la imagen original.

III-B. Herramientas y Restricciones

La solución se implementa en C++ utilizando Qt. Se hace uso de punteros, arreglos y memoria dinámica; no se utilizan estructuras (structs) ni la STL. Se debe mantener un control riguroso de la evolución de la solución mediante commits en un repositorio público.

IV. ALGORITMOS IMPLEMENTADOS

A continuación se presentan fragmentos de código relevantes a las operaciones a nivel de bits:

IV-A. Operación XOR

IV-B. Rotación y Desplazamiento de Bits

IV-C. Verificación del Enmascaramiento