INFORME DE ANALISIS DE LECTURA

KEWI YILIAN MOYA CATAÑO 1033487777

MARDEY SOLANYI ARIAS GAVIRIA 1039689519

La empresa informá2 nos pide la creación de un algoritmo que revierta una imagen distorsionada a la original. Nos proporcionan que la imagen se les pudo hacer echo unas transformaciones que son a nivel de bits y una máscara que se le sumará a un valor de la imagen. Primero nos centraremos en hacer pruebas con transformaciones y la máscara para entender cómo funcionan y afecta los datos.

Análisis de las transformaciones

Tenemos que las trasformaciones son:

* Un xOr con una imagen de ruido
* Desplazamientos de bits a la derecha o izquierda
* Rotaciones a la derecha o la izquierda

Análisis de los desplazamientos

En el caso del desplazamiento de bits a la derecha o a la izquierda sabemos que el máximo rango de movimiento puede ser de 8 bits. Sabemos que esta transformación se les hace a todos los bytes RGB de la imagen. Realizando algunos experimentos descubrimos que los desplazamientos tienen perdida de la información.

Les mostrare un ejemplo con dos desplazamientos una de 8 bits a la derecha y otro a la de 4 bits a la izquierda:

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| 1 | 0 | 1 | 0 | 1 | 1 | 1 | 1 |

Bit original:

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

Primer desplazamiento:

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

Segundo desplazamiento:

Como pueden ver en este ejemplo los datos que se desplazan hacia un lado se pierden, y en caso de que se vuelva a desplazar, pero en su contraria, seguirá existiendo pérdida importante de información. En resumen, este operador daña la integridad de los datos y no sería útil para codificar ya que son operaciones irreversibles

Por otro lado, si esta información se pudiera almacenar en algún lugar, deberían ser específicos en un archivo aparte para posteriormente incorporarlos a la imagen original, sin embargo, este no es el caso.

Análisis de las rotaciones

En el caso de las rotaciones de bits a la derecha o a la izquierda sabemos que la máxima cantidad de movimientos es de 8 bits. Esta transformación se les aplica a todos los bytes RGB de la imagen.

Realizando algunos experimentos descubrimos que las rotaciones no tienen pérdida de la información.

Como la rotación de bit hace que el bit de un extremo pase a un extremo a otro no hay modificación de los datos ya que se hacen solo con la misma información, haciendo posible que vuelvan a su estado original.

Un ejemplo muy específico en este caso es que roten 4 bits a la izquierda y luego que se roten 4 bits a la derecha:

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| 1 | 1 | 0 | 1 | 1 | 0 | 1 | 1 |

Bit original:

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| 1 | 0 | 1 | 1 | 1 | 1 | 0 | 1 |

Primera rotación:

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| 1 | 1 | 0 | 1 | 1 | 0 | 1 | 1 |

Segunda rotación:

Cómo podemos visualizar en este caso muy específico, observamos que con las rotaciones que fueron la misma cantidad de bit, pero a direcciones contrarias, vemos que volvieron a los datos originales esto significa que las rotaciones pueden volver a su estado original y que no habrá pérdida de información. Como conclusión las rotaciones mantienen la integridad de los datos.

Análisis de xOr

En ese caso es un poco más complejo ya que la forma de hacer esa transformación tenemos una imagen que se va a utilizar para comparar cada a byte con la operación xOr de la imagen original. Con esto nos conlleva que toda la imagen va a ser transformada a partir de una imagen aleatoria del mismo tamaño.

Forma

El contenido generado por IA puede ser incorrecto.Primero vamos a ver cómo funciona xOr.

Teniendo en cuenta el caso en el que tomemos los primeros pixeles de cada imagen la originaria y las distorsionada. Y de eso tomamos el primer byte de cada uno y realizamos la operación y con el resultado realizamos la misma operación, pero se hace contra en byte de la imagen distorsionada.

M/O: byte de la imagen original

M/D: byte de la imagen distorsionada

C/R: resultado de xOr de M/O y M/D

C/Q: resultado de xOr de C/R y M/D

v C/Q

Tabla #2

Tabla #1

|  |  |  |
| --- | --- | --- |
| **M/O** | **M/D** | **C/R** |
| 0 | 1 | 1 |
| 1 | 1 | 0 |
| 1 | 0 | 1 |
| 0 | 0 | 0 |
| 1 | 1 | 0 |
| 1 | 0 | 1 |
| 0 | 1 | 1 |
| 1 | 0 | 1 |

|  |  |  |
| --- | --- | --- |
| **C/R** | **M/D** | **C/Q** |
| 1 | 1 | 0 |
| 0 | 1 | 1 |
| 1 | 0 | 1 |
| 0 | 0 | 0 |
| 0 | 1 | 1 |
| 1 | 0 | 1 |
| 1 | 1 | 0 |
| 1 | 0 | 1 |

Como podemos ver comparando los resultados de la operación xOr hecho en las tablas nos damos cuenta a qué podemos volver a la información original,

Y esto se comprueba ya que M/O y C/Q, Sus valores de byte son los mismo, llegando a la conclusión que si guardad la integridad de los datos.

Conclusión del análisis de las trasformaciones.

Después de analizar cada tipo de transformación nos damos cuenta de que verdaderamente las únicas 2 transformaciones que se pueden hacer a la imagen sin perder información sería la rotación y la operación xOr. Dado que el desplazamiento pierde la información, siendo imposible regresar a la imagen original. Y esto nos lleva que la transformación de desplazamiento de bit es una trampa para no poder regresar a la imagen original.

* Transformaciones que si se utilizan: xOr y rotaciones.
* Transformaciones que no se utilizan: desplazamientos.

Análisis de la mascara

***En proceso.***