Trabajo en grupo Arquitectura de Computadores 2021-2022

|  |  |  |
| --- | --- | --- |
| **A** | **Fernández Méndez**  **González Sierra**  **Fuertes Camporro**  **Baños Gómez** | **Lara**  **Fernando José**  **Luis**  **Álvaro** |
| Grupo | Apellidos | Nombre |

|  |  |  |
| --- | --- | --- |
| **71.711.034-R**  **71.754.511-P**  **71.672.531-T**  **71.908.915-J** | **UO276026**  **UO277938**  **UO264691**  **UO245852** | **UO276026@uniovi.es**  **UO277938@uniovi.es**  **UO264691@uniovi.es**  **UO245852@uniovi.es** |
| DNI | Matrícula | e-mail de contacto (@uniovi.es) |

|  |  |  |
| --- | --- | --- |
| **1** | Comparativa de arquitecturas |  |
| Nº Trabajo | Trabajo en grupo de Arquitectura de Computadores | Calificación |

|  |
| --- |
| Comentarios sobre la corrección |
|  |

Asignatura

ARQUITECTURA DE COMPUTADORES

**Curso 2021-2022**

|  |  |  |
| --- | --- | --- |
|  | | **Área de Arquitectura y Tecnología de Computadores**  *Departamento de Informática de la Universidad de Oviedo* |
| **1** | | Comparativa de arquitecturas | | |
| Nº Trabajo | | Trabajo en grupo de Arquitectura de Computadores | | |

|  |  |
| --- | --- |
| Apellidos | Nombre |
| **Fernández Méndez** | **Lara** |
| Tareas realizadas | |
| Código del programa | |
| Memoria del trabajo | |
| Mediciones de tiempos | |
|  | |
|  | |
|  | |
|  | |
|  | |
|  | |

|  |  |
| --- | --- |
| Apellidos | Nombre |
| **González Sierra** | **Fernando José** |
| Tareas realizadas | |
| Código del programa | |
| Memoria del trabajo | |
| Mediciones de tiempos | |
|  | |
|  | |
|  | |
|  | |
|  | |
|  | |

|  |  |
| --- | --- |
| Apellidos | Nombre |
| **Fuertes Camporro** | **Luis** |
| Tareas realizadas | |
| Código del programa | |
| Memoria del trabajo | |
| Mediciones de tiempos | |
|  | |
|  | |
|  | |
|  | |
|  | |
|  | |

|  |  |
| --- | --- |
| Apellidos | Nombre |
| **Baños Gómez** | **Álvaro** |
| Tareas realizadas | |
| Código del programa | |
| Memoria del trabajo | |
| Mediciones de tiempos | |
|  | |
|  | |
|  | |
|  | |
|  | |

1. Resumen
2. Introducción

En este trabajo vamos a desarrollar un algoritmo para el procesamiento de imágenes.

En nuestro caso realizaremos una fusión entre dos imágenes.

Posteriormente se realizarán unos análisis de los resultados obtenidos.

1. Método de solapamiento: Fusión de capas

El objetivo del algoritmo es realizar una fusión entre dos imágenes de forma solapada, la primera será la escultura “La bailarina” y la segunda un fondo degradado de blanco y negro.

Importante que ambas imágenes tengan el mismo tamaño para aplicar el algoritmo.

La forma de funcionamiento de nuestro algoritmo se basa en recorrer las imágenes simultáneamente, obtener un píxel de cada una y descomponerlos en las tres componentes de los colores, “RGB” (Red, Green, Blue). Una vez realizada la descomposición, se aplica la fórmula para cada uno de los componentes (c) de ambas fotos:

Imagen que contiene Texto

Descripción generada automáticamente

Y à Imagen 1 X à Imagen 2 I à Imagen destino

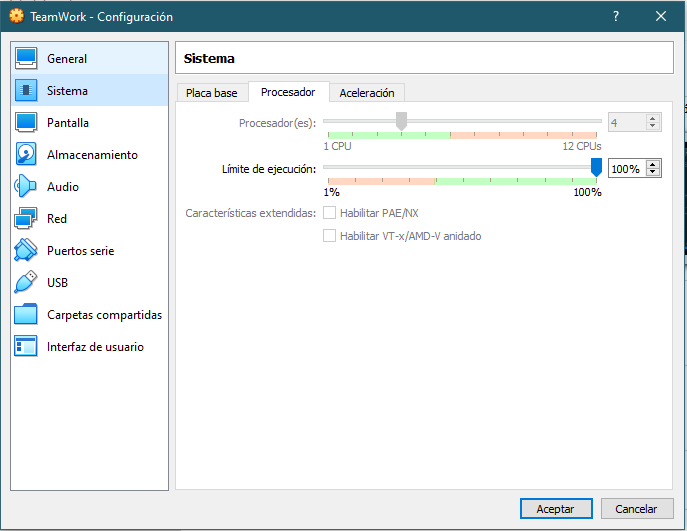
Al terminar de aplicar la formula se habrán fusionado todos los componentes de todos los pixeles de las imágenes, dando lugar a una sola imagen que sea la forma solapada de ambas.

* 1. Entorno de trabajo

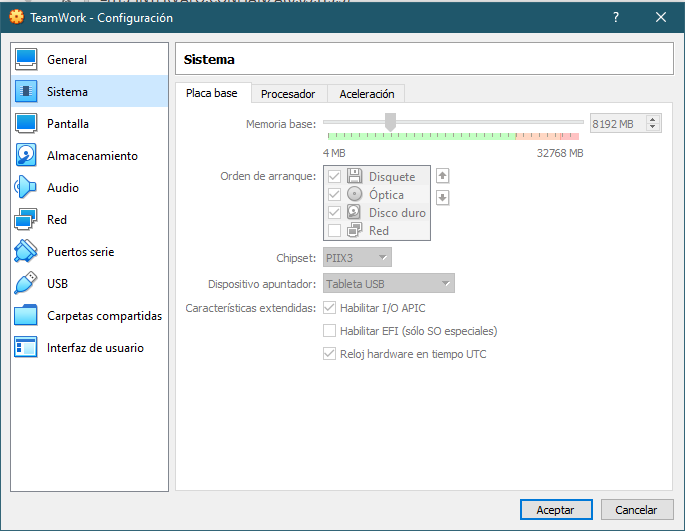
Para la realización de este trabajo hemos utilizado una máquina virtual con el sistema operativo Ubuntu, una distribución de Linux. Tanto Ubuntu como Linux son software libre, es decir, accesible para todos los usuarios de forma gratuita y, además, modificable.

En nuestro caso hemos usado el programa VirtualBox, de Oracle para poder acceder a la máquina virtual.

Las especificaciones del entorno de trabajo son:



**Figura 1.** Número de núcleos de la máquina virtual.



**Figura 2.** Memoria usada para la máquina virtual.

* 1. Programa, Compilación y Ejecución

Realizamos un programa en Visual Studio con C++ en el que hacíamos la selección de los pixeles de las fotos y su descomposición en componentes mediante punteros. Y aplicando el algoritmo ya mencionado en el apartado 2.0 para todos los pixeles y componentes obtenemos.

Para la compilación de ese programa utilizamos la ventana de comandos de Ubuntu, la carpeta Makefile nos ayuda para este proceso así que haciendo el comando: make release, haremos la compilación y la llevaremos a la carpeta release, el nombre del archivo compilado será single-thread.

Tras la compilación del programa podemos ejecutarlo con el comando: reléase/single\_thread.

La ejecución nos mostrará por ventana todas las imágenes implicadas en el proceso, las dos de fusión y, tras la aplicación del algoritmo, la resultante, esta última se guardará en la carpeta desde donde se ejecuta el comando.

* 1. Medición de tiempos

Tomamos tiempos diez veces para realizar una media con la que trabajar. *Figura 3.*

De ella a su vez podemos obtener una desviación típica, es decir lo que varía los datos a partir de la media. *Figura 4.*

Luego calculamos los intervalos de confianza con un nivel de confianza del 95%. *Figura 4.*



**Figura 3.** Tiempos medidos.

Tabla

Descripción generada automáticamente

**Figura 4.** Media, Desviación Típica e Intervalo de Confianza 95% inferior y superior.

1. Conclusiones

La realización de los filtros en imágenes es un proceso muy sencillo ya que solo programa muy sencillo de obtención de pixeles y componentes, y posteriormente la aplicación de un algoritmo. Por lo que copiando el proyecto y cambiando el algoritmo podemos hacer otro filtro totalmente distinto.

1. Referencias

Ubuntu: <https://ubuntu.com>

Linux: <https://www.linux.org>

Oracle VirtualBox: <https://www.virtualbox.org>