Proyecto Integrador de Lenguaje Ensamblador y Arquitectura

CI-0119

Documento de Diseño

Jeremy Espinoza
Daniel Pinto
Andrew Umaña

UCR - ECCI

Tareas

Etapa I

Etapa inicial donde se inicializa el proyecto y el estudiante empieza a construir las bases para el dispositivo.

Uso de github

1. Aprender a utilizar herramientas de github.
2. Utilizar github para crear una lista de temas e introducción.

Filtros

1. Investigar filtros de cámaras fotográficas y analizar su validez para efectos de este proyecto.
2. Escoger filtros interesantes, mínimo 4, pero mantener un número mayor con varias posibilidades, de lo contrario el proyecto puede demorarse si un filtro da problemas en implementación y no se tiene otras opciones preparadas.

Producto final (Resumen etapa I)

1. Analizar la comunicación entre los dispositivos y como se simulará.
2. Aprender y entender varios conceptos; bitmap (.bmp), drivers, pruebas de validación.
3. Realizar una figura expandida del diseño inicial.

Temas y conocimientos

1. Investigar los temas necesarios para el proyecto.
2. Investigar los conocimientos necesarios para el proyecto.
3. Investigar las habilidades necesarias para el proyecto.
4. Empezar a aplicar y/o memorizar lo necesario en preparación para las futuras etapas del proyecto.

Github

1. Subir el material finalizado de la etapa I al repositorio de github.

Debido al diseño del curso, se añadirán tareas en las próximas etapas y se expandirán con el progreso del proyecto, cuando tengamos una visión más completa y con más entendimiento y experiencia del tema que se está trabajando.

Etapa II

Código.

1. Aplicar las habilidades y conocimientos en simics para aplicar el diseño del modelo.
2. Subir el código en el repositorio.

Repositorio

1. Actualizar tareas.
2. Investigar el uso de wiki para proyectos.
3. Aplicar y actualizar el wiki con el resumen del proyecto, temas, detalles, etc.

Etapa III

Repositorio

1. Actualizar tareas.
2. Actualizar wiki.
3. Actualizar código con una versión más completa del diseño.

Diagramas

1. Investigar el uso adecuado de diagramas.
2. Aplicar lo aprendido y crear diagramas que expliquen el funcionamiento del modelo.

Etapa IV

Repositorio

1. Actualizar tareas.
2. Actualizar wiki.
3. Actualizar código con la versión final del diseño.

Demo

1. Investigar y analizar la mejor forma de exponer una demostración del diseño.
2. Aplicar lo aprendido y crear el demo utilizando todo el material que se considere relevante.
3. Presentar el demo.

Temas, conocimientos y habilidades

- Programación en lenguajes de alto nivel (C/C++, Python, …)
- Programación en lenguajes de bajo nivel (x86-64)
- Habilidades en programación del paradigma Programación Orientada a Objetos (P.O.O)
- SIMICS/DML
- Uso de Linux (Ubuntu)
- Uso de Git
- Conocimientos de hardware y sus interacciones a bajo nivel
- Manipulacion de imagenes
- Aritmética Binaria y Manipulación de Bits
- Diseño de Software
- Creación y uso de pruebas de validación
- Habilidades blandas de comunicación, coordinación y trabajo en equipo
- Capacidad de planificación y desarrollo de proyectos
- Capacidad de investigación
- Implementación de conceptos aprendidos del curso Lenguaje Ensamblador

Filtros

Se investiga y preparan 6 filtros diferentes para la primera fase, dependiendo de su factibilidad en este proyecto se escogerán los que sean posibles de implementar para efectos de este curso.
Para ejemplificar las diferencias cada filtro se usará la misma foto base, estos filtros son ejemplos para demostrar los filtros en esta etapa de conceptualización y es posible que difieran del producto final ya que dependen de la intensidad del filtro, herramientas utilizadas, etc.

Ilustración 1 - Fotografía base en Cerro Chirripó, escogida por la claridad de colores

Sepia

Un filtro muy popular, produce tonos con colores apagados y oscuros en una escala que otorga un toque nostálgico y de antaño.

Ilustración 2 - Fotografía bajo el filtro sepia.

Negativo

Un filtro que invierte los colores en una fotografía.

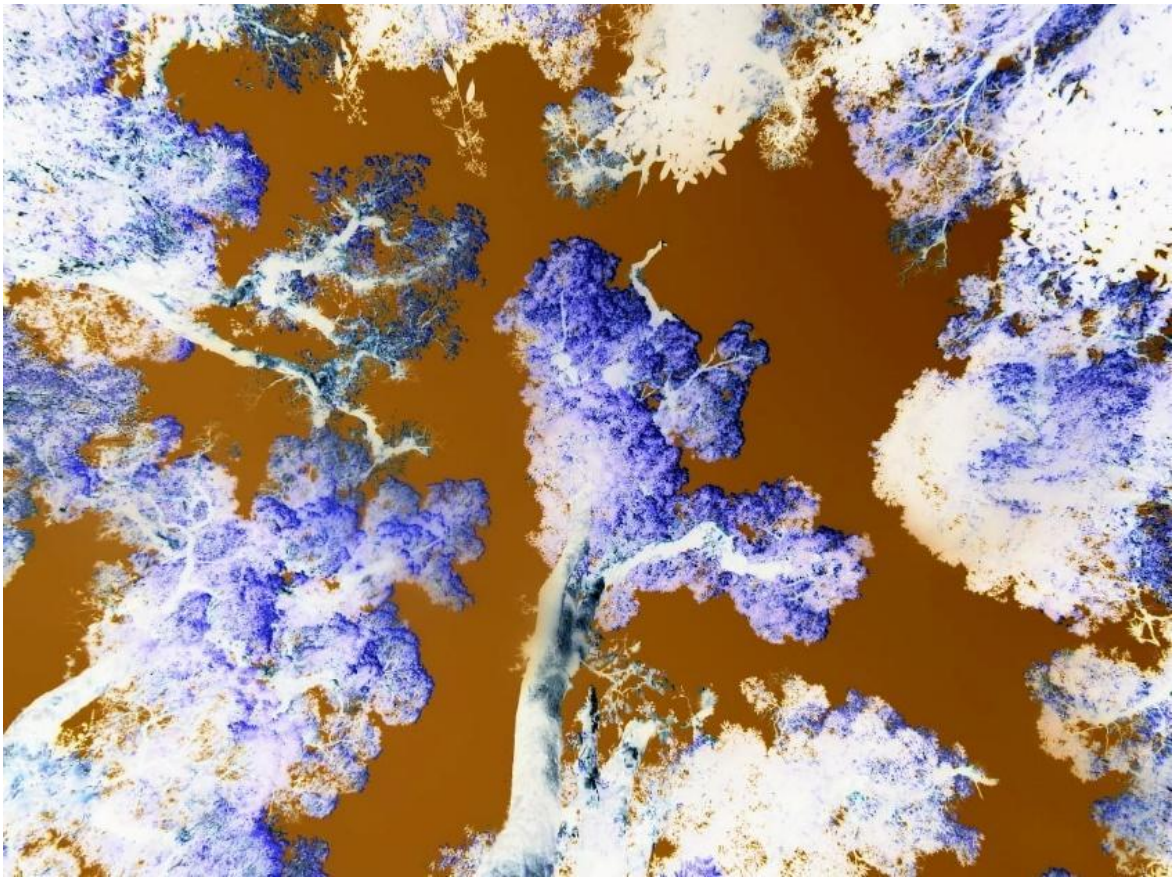

Ilustración 3 - Fotografía bajo el filtro negativo

Blanco y Negro

Un filtro que mutea todos los colores y los transforma en una escala de grises, permite apreciar y resaltar o esconder ciertos detalles.

Ilustración 4 - Fotografía bajo el filtro blanco y negro

Heat Map

Un filtro que transforma los colores en un mapa de calor, dependiendo de las opciones escogidas en el filtro este puede cambiar los colores radicalmente.

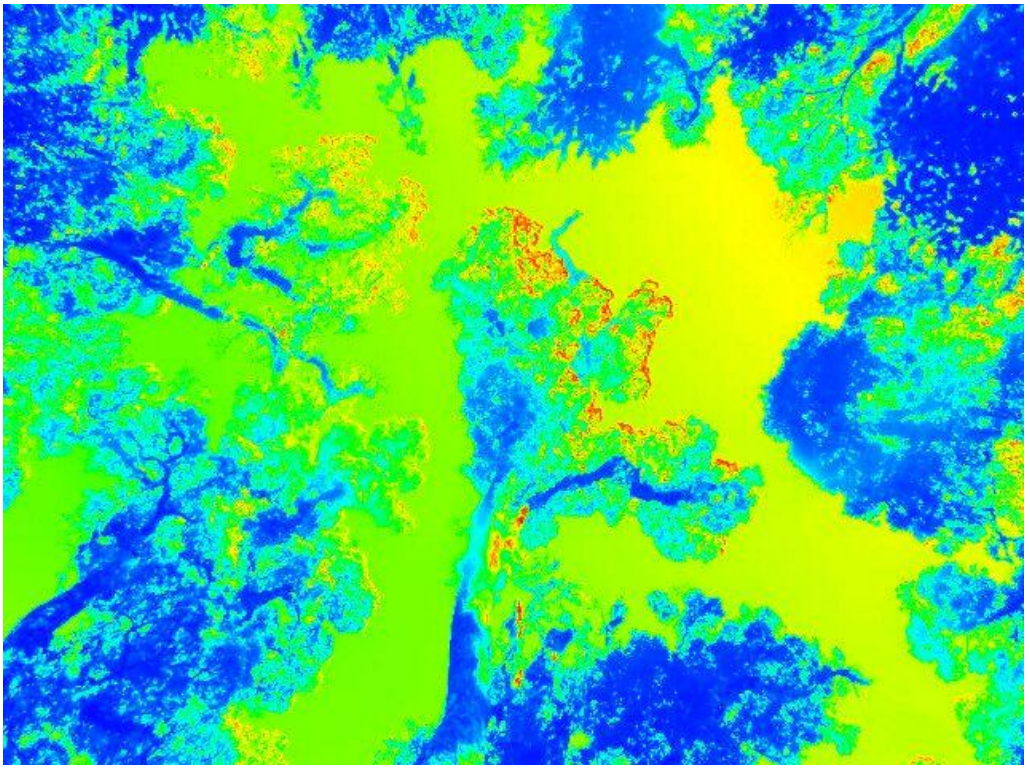

Ilustración 5 - Fotografía bajo el filtro heat map.

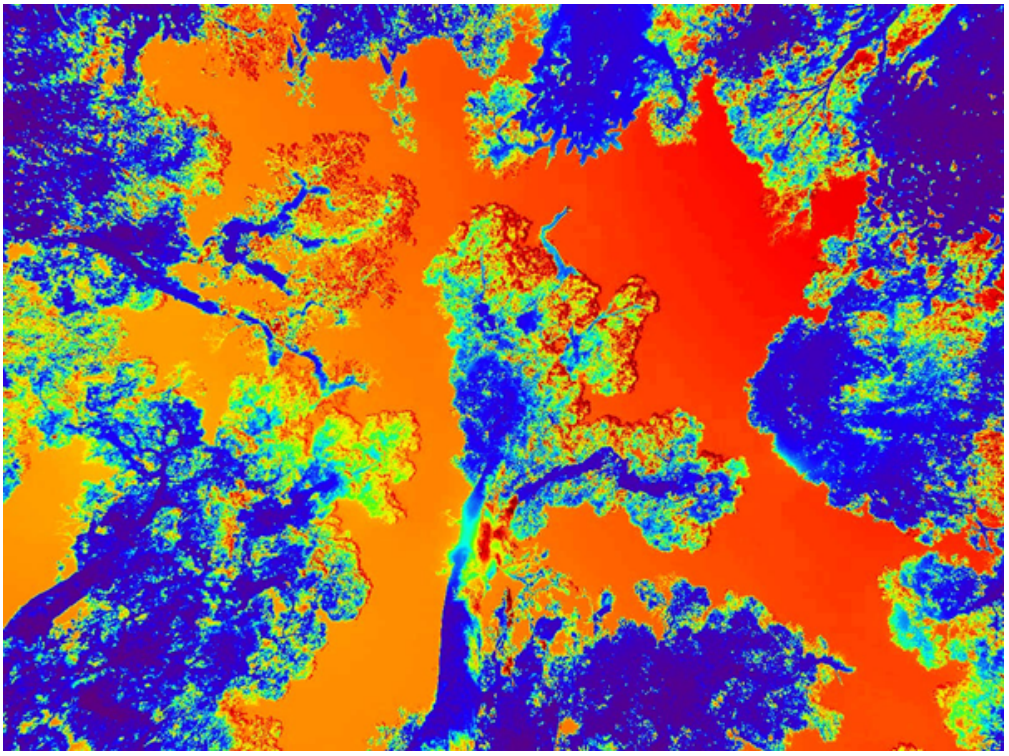

Ilustración 6 - Fotografía bajo el filtro heat map con diferentes ajustes.

Bokeh

Un filtro que transforma la foto con el zoom del lente, creando un efecto de enfoque en una localidad específica de la fotografía mientras que el resto se difumina.

Ilustración 7 - Fotografía bajo el filtro bokeh, con un enfoque al centro de la imagen.

Amaro

Un filtro más básico, añade tintes azules y amarillos e incrementa el contraste, nitidez y el brillo.

Ilustración 8 - Fotografía bajo el filtro Amaro, recalcando los colores vividos.

1977

Un filtro que cambia los colores y les proporciona un tinte rojo y marrón, creando un efecto de una foto tomada en los 70s.

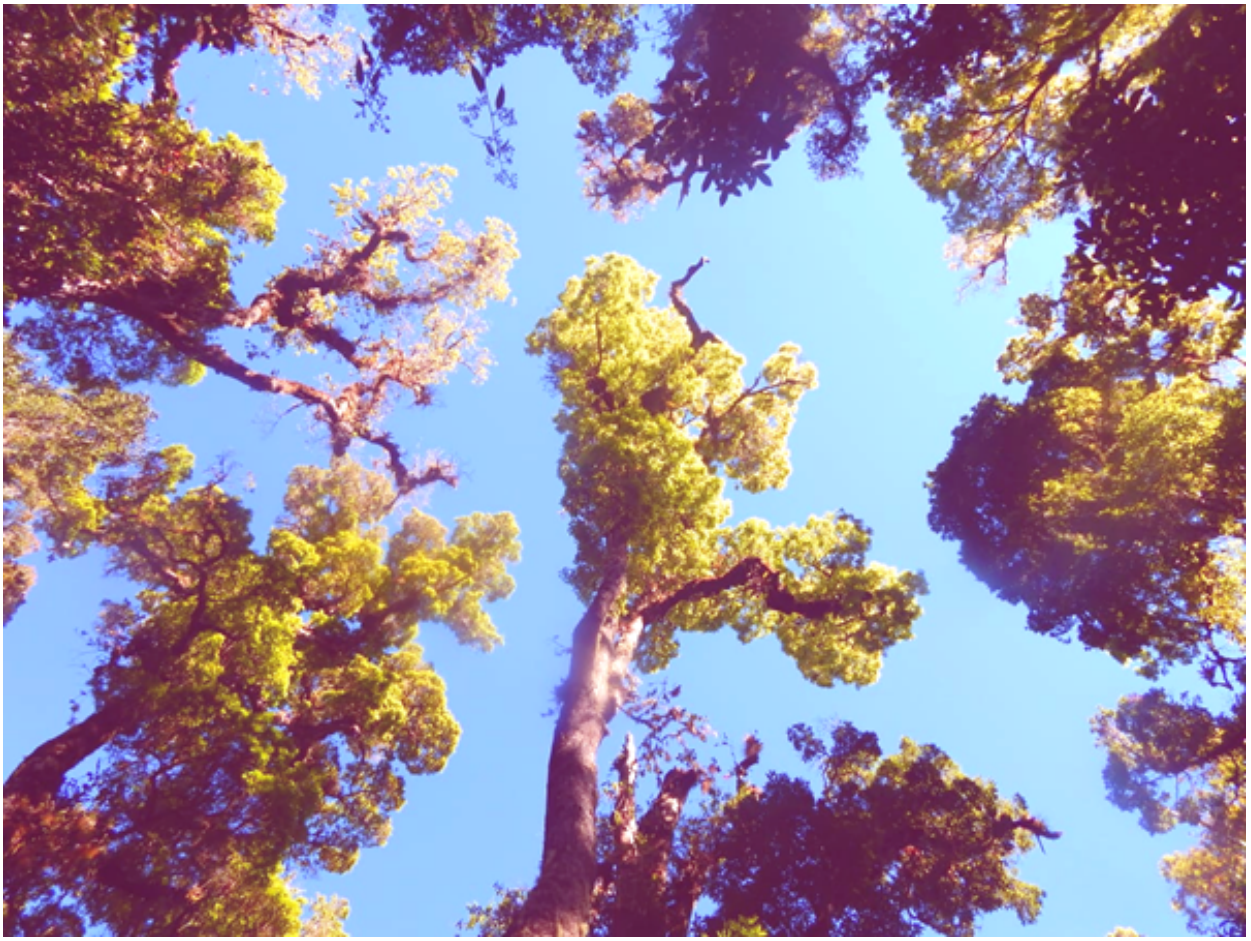

Ilustración 9 - Fotografía bajo el filtro 1977, transformando los colores a tonos más apagados y con un tinte rojizo.

Resumen del producto final

El producto final consiste en un dispositivo virtualizado (1) que captura imágenes de formato bitmap y utiliza no menos de cuatro filtros antes de almacenar el resultado con sus metadatos respectivos (fecha de captura, tamaño, etc.) en una pc virtualizada. Los filtros por desarrollar para el dispositivo son el Sepia, Negativo, Heat Map, Bokeh, Amaro y 1977.

La pc utiliza el sistema operativo Ubuntu y está conectada con el dispositivo utilizando PCI-Express (5) y un driver (2) que controla el dispositivo. La PC contará con un programa de pruebas (4) e interfaz (3) que permite verificar el funcionamiento del dispositivo.