

Informe Actividades Versiones Finales Programación Python

Estimado Profesor:

Aquí le describiré lo más relevante del Estudio que hice y algunos detalles en la programación para que pueda transmitirle todo lo recopilado al nuevo tesista que continuará con el tema del Chroma Key aplicado en el video ya que hasta ahora está funcionando perfectamente para imágenes. Cabe destacar que igual dentro de los códigos fuentes también explico con más detalles lo que hace cada una de las líneas de la programación en Python.

Antecedentes Extras Importantes que se usaron para la programación:

- Sistema Operativo: Ubuntu 12.04 LTS – núcleo kernel: 3.5.0-37-generic
- Python 2.7
- VLC
- Instalación – Compilación de v4l2loopback

Bueno, la primera parte de la programación se basó en como poder enviar el video que estaba capturando por medio de mi webcam a una tubería de video virtual, para hacer esto tuve que investigar que herramienta me permitiría dicho proceso. Al cabo de varias pruebas hechas de primera en Debian (con el módulo vloopback) y luego finalmente encontrar un estándar de trabajo en Ubuntu por medio del módulo v4l2loopback (el cual lo tuve que instalar y compilar manualmente por medio de varios paso a seguir en donde para tomar evidencia de las cosas que iba haciendo era tomar pantallazos de cada uno de

los etapas, las cuales están detalladas en el informe de Trabajo de Titulo con más detalles).

Luego de la instalación de v4l2loopback logre pasar a la programación en Python, para ello tuve que usar librerías como lo son:

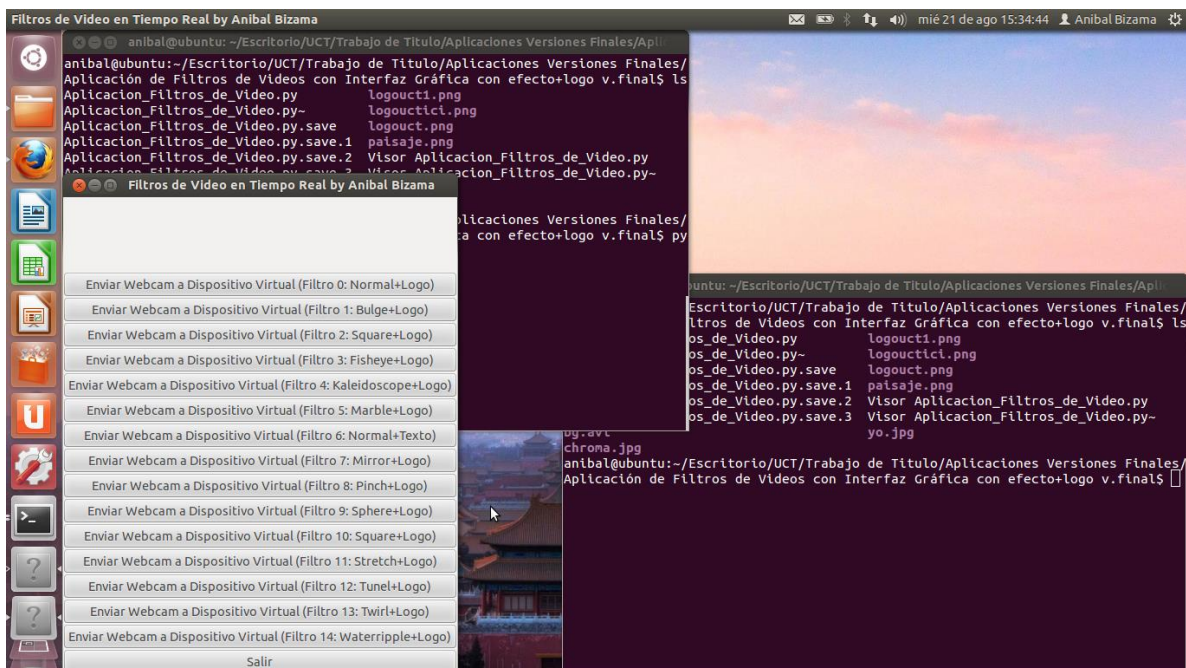
- **sys** (funcionalidad relacionada con el propio intérprete de Python).
- **os** (engloba funcionalidad relativa al sistema operativo).
- **pygtk y gtk** (para poder acceder a los controles de Gtk+ y poder crear el menú y botones de la interfaz gráfica, entre otros).
- **object** (es una parte fundamental para Gstreamer ya que se encarga de crear un ciclo o bucle en el que gst habita y hace su trabajo).
- **pygst y gst** (Se importa gst y pygst en la version 0.10 para cargarlos y poder transmitir video por Gstreamer).

1. Explicación pasos más importantes en programación código **Aplicacion_Filtros_de_Video.py**:

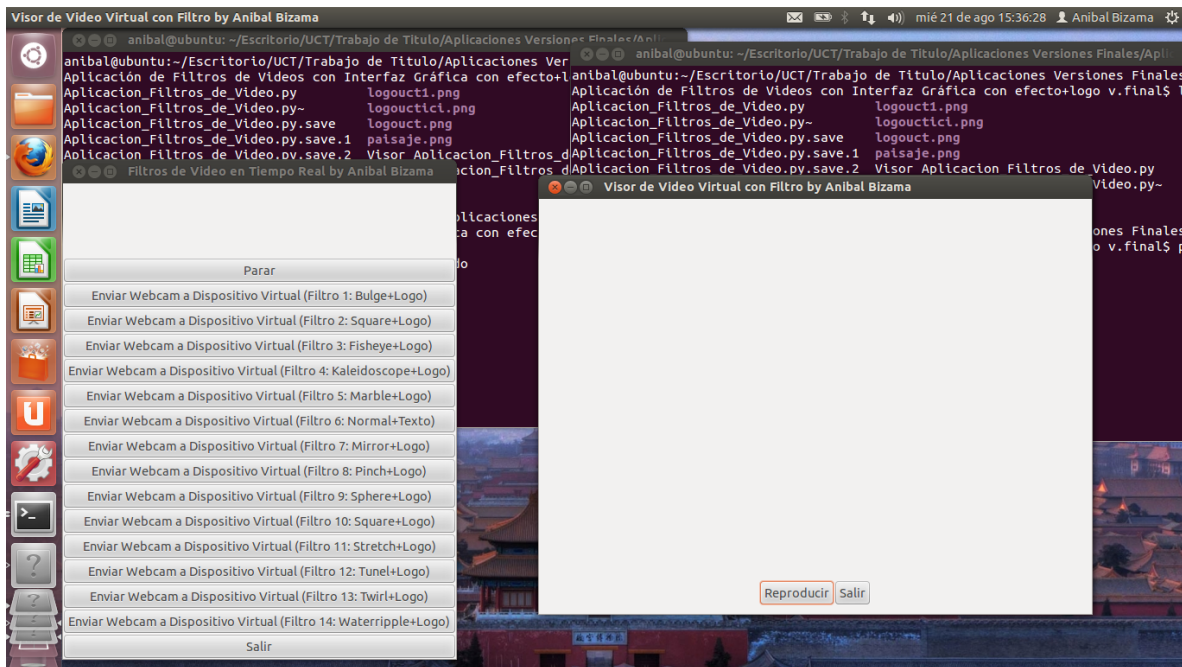
- Después de definir las librerías a usar empecé a crear el entorno gráfico donde al ejecutar nuestro código **Aplicacion_Filtros_de_Video.py** nos desplegará como primera cosa la información sobre el programa con el logo UCT y descripción del tema tratado.
- Luego paso a la creación de botones para enviar el video de la webcam para la tubería de video virtual, para esto tuve que darle la función a este botón para transmitir el video de forma correcta.

- Luego le indicamos a los botones que cuando se le hagan un clic emita la señal "clicked" que llamara a la función "start_stop".
- La función "start_stop" se ejecuta al hacer clic sobre el botón "Enviar Webcam a Dispositivo Virtual" si queremos detener el flujo de video hacemos nuevamente clic en "Parar" y se detendrá el proceso.
- El paso más importante de todos es la configuración de la tubería **gstreamer** para aplicarle los filtros de video en tiempo real(como lo son video + texto, video + logoUCT transparente, video + logo + texto , video + efectos gstreamer) para ello:

`self.player0 = gst.parse_launch("aquí va la configuración completa que le queremos dar a la tubería, nombrando nuestro input de video (webcam – video0) y nuestra salida que será el flujo de video virtual (video1(virtual) + efectos hechos por filtros de video en tiempo real)")`
- También se creó una función denominada "exit" para salir de la ventana.



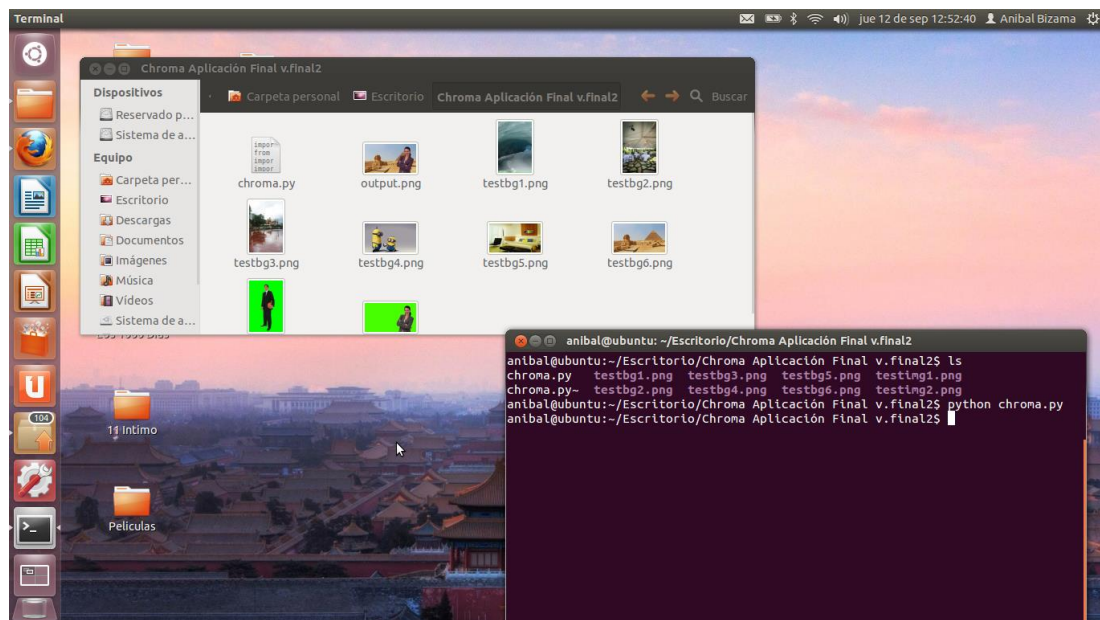
2. También se creó un código aparte denominado **“Visor Aplicacion_Filtros_de_Video.py”** incorporado con entorno gráfico, que se hizo con el propósito de poder visualizar la salida de video (video virtual) generada y no usar un reproductor de video instalado en el sistema operativo Ubuntu (como es el caso de VLC por ejemplo).

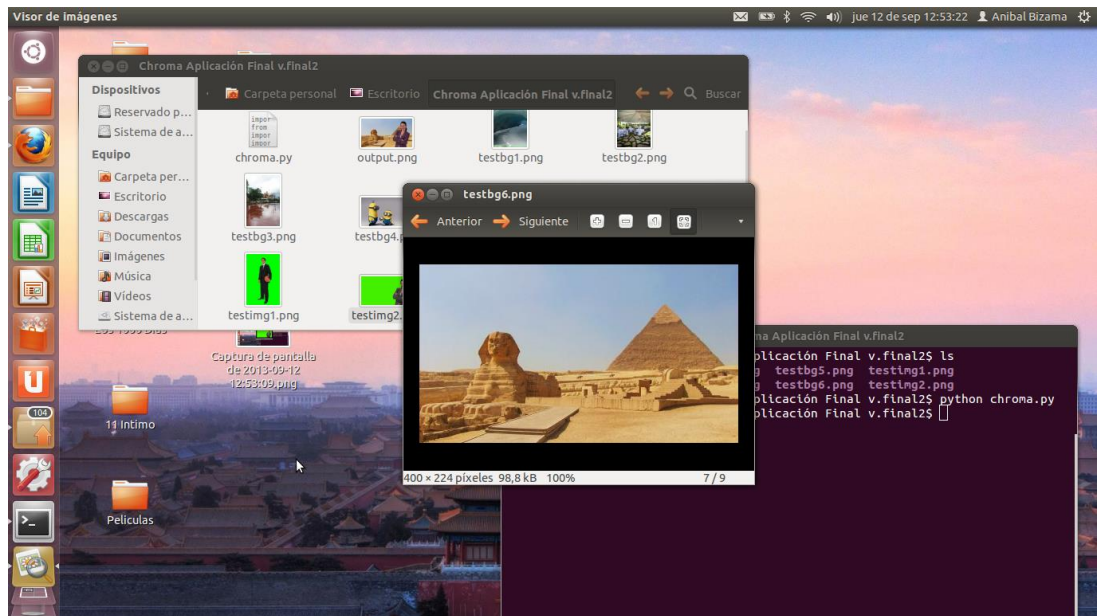
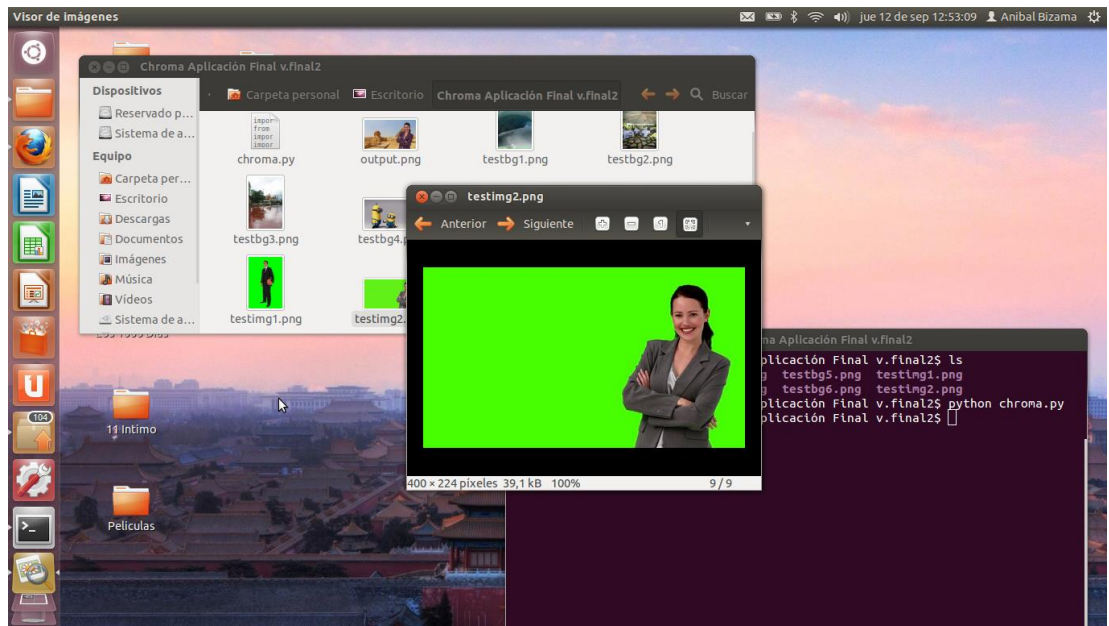


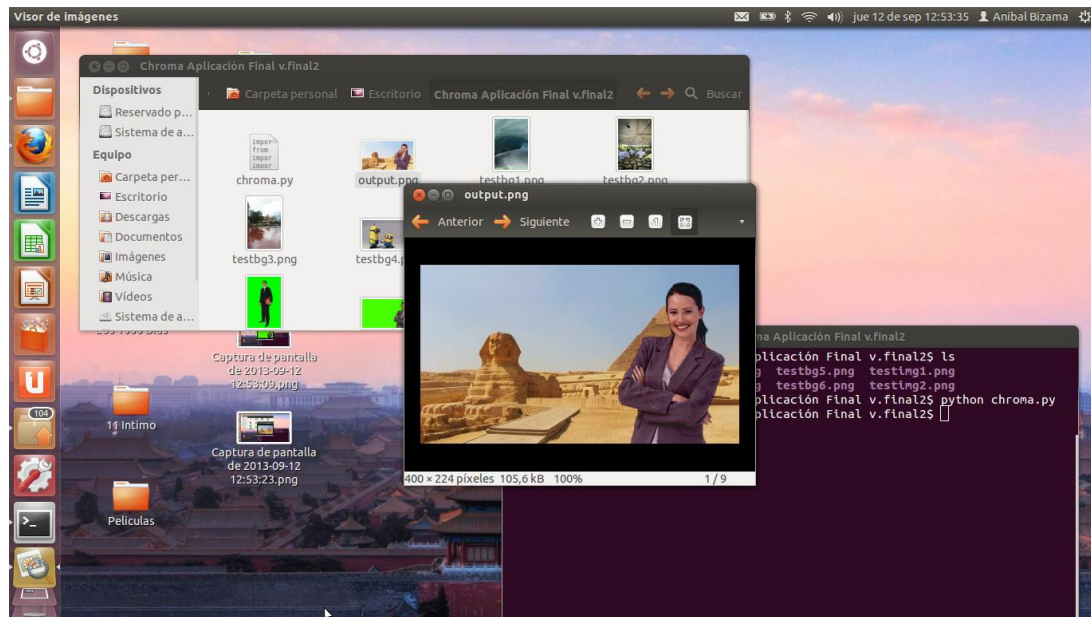
- Con lo cual se logró de un Chroma Key forma perfecta por medio de la calibración en los parámetros y variables de la fórmula matemática.

Las librerías más relevantes que se usan en este código son:

- **Image:** Para poder trabajar y manipular imágenes a tratar con el algoritmo matemático para el Chroma.
- **Numpy:** que constituye un gran aporte como una biblioteca de funciones matemáticas de alto nivel para operar con vectores o matrices en la manipulación de las imágenes.
- **Math:** librería que posee una gran parte de las funciones matemáticas básicas para el uso en el algoritmo a dentro de la programación.







- Como algo importante en la programación: se creó un array a partir de la imagen para vectorizar en una función de enmascaramiento para generar una máscara. Luego se convierte el array a imagen y se convierte la imagen de entrada a RGB para facilitar el trabajo en el código. También se sustraen los verdes de la entrada de la imagen con fondo verde. Para finalmente en la función `colorclose()` aplicar el algoritmo matemático (en el formato **`math.sqrt(acá_la_función_matemática)`**) que hará el proceso completo en la manipulación con las imágenes para generar como solución final un output con una imagen con el efecto Chroma Key.
- El objetivo principal en este caso era intentar de aplicar, en la tubería virtual de video creada con `gststreamer` en el código anterior ("**Aplicacion_Filtros_de_Video.py**"), el efecto en tiempo real del Chroma Key para ver su comportamiento en sí. Pero lo cual no pude encontrar la solución óptima por medio de mis pruebas hechas directo del código que me funcionaba con imágenes. Para ello estuve trabajando esforzadamente ya sea modificando el formato para vídeo, calibrando los ajustes para la visualización , probando nuevos métodos, etc. dentro del código en Python pero no logre tener buenos resultados con las pruebas hechas. Por lo cual le enviare el código que me ha estado

funcionando de forma correcta (al 100%) que es el Chroma tratado con imágenes. Esta descripción espero sea de gran aporte para Ud. Profesor para que le pueda comunicar al nuevo alumno tesista todo lo que se trabajó en el tema y se pueda lograr la última etapa final.