

# Guía técnica para trabajar con OpenShot en Linux Mint

Diagnóstico de pantalla negra, configuración de exportación y render  
alternativo con FFmpeg

Linux Mint 22.1 XFCE | OpenShot | FFmpeg | 2026

<b>Empresa</b>	TECPROG WORLD E.I.R.L.
<b>RUC</b>	20608743252
<b>Tipo de contribuyente</b>	Empresa Individual de Responsabilidad Limitada
<b>Estado del contribuyente</b>	Activo
<b>Condición del contribuyente</b>	Habido
<b>Domicilio fiscal</b>	Mza. C Lote 43 Urb. Los Nísperos, San Martín de Porres, Lima, Lima
<b>Actividad principal</b>	7110 – Actividades de arquitectura e ingeniería y actividades conexas de consultoría técnica
<b>Sitio web</b>	<a href="https://tecprog-world-store.github.io/">https://tecprog-world-store.github.io/</a>

Documento técnico para edición de video, documentación de software, cursos de Linux y  
producción audiovisual aplicada a ingeniería.

4 de junio de 2026

# Índice

<b>1. Objetivo de la guía</b>	<b>2</b>
<b>2. Contexto del problema</b>	<b>2</b>
<b>3. Flujo regular recomendado para trabajar con OpenShot</b>	<b>2</b>
3.1. Crear un proyecto limpio . . . . .	2
3.2. Importar el video fuente . . . . .	2
3.3. Colocar el clip desde el segundo cero . . . . .	2
<b>4. Causas frecuentes de pantalla negra al exportar</b>	<b>3</b>
4.1. Diferencia entre resolución de captura y resolución de exportación . . . . .	3
4.2. Uso de perfiles incoherentes . . . . .	3
4.3. Códecs de video y audio no recomendados . . . . .	3
4.4. Aceleración de hardware . . . . .	3
<b>5. Configuración recomendada de OpenShot</b>	<b>4</b>
5.1. Perfil del proyecto . . . . .	4
5.2. Exportación recomendada para videos 16:9 . . . . .	4
5.3. Exportación recomendada para monitor 1024x768 . . . . .	4
<b>6. Crear un perfil personalizado 1024x768</b>	<b>4</b>
<b>7. Configuración de rendimiento de OpenShot</b>	<b>5</b>
<b>8. Limpieza de caché de OpenShot</b>	<b>5</b>
<b>9. Verificación técnica del video fuente</b>	<b>5</b>
<b>10. Conversión preventiva del video fuente con FFmpeg</b>	<b>5</b>
<b>11. Render alternativo desde terminal si OpenShot falla</b>	<b>6</b>
11.1. Script de render con FFmpeg . . . . .	6
11.2. Ejecución del script . . . . .	9
<b>12. Render manual de un solo archivo sin usar el proyecto OpenShot</b>	<b>10</b>
<b>13. Recomendaciones para futuros videos</b>	<b>10</b>
13.1. Grabar en resolución compatible con la publicación . . . . .	10
13.2. Usar fondo corporativo para capturas 4:3 . . . . .	10
13.3. Guardar una copia del proyecto antes de exportar . . . . .	10
<b>14. Conclusión</b>	<b>10</b>

## 1 Objetivo de la guía

El objetivo de esta guía es documentar un flujo regular de trabajo con OpenShot Video Editor en Linux Mint y explicar cómo diagnosticar un problema frecuente donde el video exportado aparece con fondo negro o sin imagen visible. La guía parte de un caso práctico en el que la previsualización de OpenShot mostraba correctamente la grabación de pantalla, pero el archivo exportado quedaba negro o con una relación de aspecto inadecuada. El procedimiento integra configuraciones internas de OpenShot, revisión de perfiles de video, control de códecs y un método alternativo con FFmpeg desde terminal.

## 2 Contexto del problema

En proyectos de grabación de pantalla, especialmente cuando se usa Linux Mint XFCE con más de un monitor, OpenShot puede mostrar correctamente el video en la vista previa y, aun así, generar una exportación defectuosa. Esta situación puede presentarse cuando el monitor grabado tiene una resolución 4:3, por ejemplo 1024x768, mientras el proyecto o la exportación se configura como HD 720p 16:9 o incluso como 4K. En ese escenario, OpenShot intenta adaptar el material de entrada a un lienzo con otra geometría, lo que puede producir barras negras, recortes, escalado incorrecto o, en ciertos casos, un render final sin imagen.

## 3 Flujo regular recomendado para trabajar con OpenShot

### 3.1 Crear un proyecto limpio

Al iniciar un proyecto de edición, se recomienda guardar el archivo `.osp` desde el comienzo. Esto permite conservar rutas, clips, capas, cortes y configuraciones generales de la edición.

```
Archivo -> Guardar proyecto como...  
Ejemplo:  
/home/pc/Escritorio/Textmaker.osp
```

### 3.2 Importar el video fuente

El video fuente debe importarse desde la sección de archivos del proyecto. Es recomendable evitar mover o renombrar el archivo original después de importarlo, ya que OpenShot almacena referencias a la ruta del archivo.

```
Archivo de entrada:  
Videograbacion 2026-06-04 20:58:49.mp4
```

### 3.3 Colocar el clip desde el segundo cero

Antes de diagnosticar otros problemas, se debe confirmar que el clip inicia en el segundo cero de la línea de tiempo. Si el clip empieza después, la exportación contendrá un tramo negro al inicio.

```
Inicio correcto:  
00:00:00 | video  
  
Inicio incorrecto:  
00:00:00 | espacio vacio | video inicia despues
```

## 4 Causas frecuentes de pantalla negra al exportar

### 4.1 Diferencia entre resolución de captura y resolución de exportación

Una causa frecuente es grabar desde un monitor con resolución 1024x768 y exportar como 1280x720 o 1920x1080. El primer formato tiene relación 4:3, mientras que los perfiles HD y Full HD comunes tienen relación 16:9.

Caso	Relación	Observación
<b>Monitor 1024x768</b>	4:3	Captura cuadrada o clásica de escritorio.
<b>HD 1280x720</b>	16:9	Perfil común para YouTube.
<b>Full HD 1920x1080</b>	16:9	Perfil común para videos modernos.
<b>4K 3840x2160</b>	16:9	Excesivo para capturas simples de escritorio.

Si se exporta un video 4:3 dentro de un lienzo 16:9, es normal observar barras negras laterales. Sin embargo, si además existe incompatibilidad de códec o aceleración de hardware, puede aparecer una salida completamente negra.

### 4.2 Uso de perfiles incoherentes

Si el título de la ventana de OpenShot indica HD 720p 30 fps, pero en la exportación avanzada aparece 4K UHD 2160p 23.98 fps, existe una incoherencia de proyecto. Esta diferencia puede forzar escalados innecesarios y producir resultados inestables.

```
Proyecto:
HD 720p 30 fps
```

```
Exportacion erronea:
4K UHD 2160p 23.98 fps
3840 x 2160
```

### 4.3 Códecs de video y audio no recomendados

Para exportar archivos .mp4, se recomienda usar `libx264` para video y `aac` para audio. En algunos casos, usar `libmp3lame` como audio dentro de MP4 puede funcionar, pero no es la combinación más estándar para publicación en plataformas web.

```
Video recomendado: libx264
Audio recomendado: aac
Formato recomendado: mp4
Pixel format recomendado: yuv420p
```

### 4.4 Aceleración de hardware

OpenShot puede usar decodificación por hardware según la configuración del sistema. En equipos antiguos, controladores gráficos limitados o escritorios XFCE con compositores específicos, esta opción puede provocar errores de previsualización o render. Para máxima estabilidad, se recomienda exportar usando CPU.

## 5 Configuración recomendada de OpenShot

### 5.1 Perfil del proyecto

Antes de editar, seleccionar un perfil coherente con el video fuente:

```
Archivo -> Elegir perfil
```

Si el monitor grabado fue 1024x768, se puede crear un perfil personalizado 4:3. Si se desea publicar en YouTube con lienzo 16:9, se puede trabajar en 1280x720, pero se debe decidir si se aceptan barras laterales, recorte o fondo personalizado.

### 5.2 Exportación recomendada para videos 16:9

Para videos que ya fueron grabados o adaptados a 16:9:

```
Perfil: Web
Objetivo: CPU MP4 (h.264)
Perfil de video: HD 720p 30 fps (1280x720)
Calidad: Alta
```

En configuración avanzada:

```
Formato de video: mp4
Codec de video: libx264
Tasa de bits: 8 Mb/s o 12 Mb/s
Codec de audio: aac
Muestreo de audio: 48000 Hz
Canales: Estereo
Tasa de audio: 192 kb/s
```

### 5.3 Exportación recomendada para monitor 1024x768

Si se desea conservar la resolución original del monitor 2:

```
Ancho: 1024
Alto: 768
Relacion de aspecto: 4:3
FPS: 30
Entrelazado: No
Formato: mp4
Codec de video: libx264
Codec de audio: aac
```

## 6 Crear un perfil personalizado 1024x768

Si OpenShot no muestra un perfil XGA 1024x768, se puede crear uno manualmente.

```
mkdir -p ~/.openshot_qt/profiles
nano ~/.openshot_qt/profiles/xga_1024x768_30fps
```

Pegar el siguiente contenido:

```
description=Monitor 2 XGA 1024x768 30 fps
frame_rate_num=30
frame_rate_den=1
width=1024
height=768
progressive=1
```

```
sample_aspect_num=1
sample_aspect_den=1
display_aspect_num=4
display_aspect_den=3
colorspace=709
```

Luego cerrar completamente OpenShot, volver a abrirlo y seleccionar el nuevo perfil desde:

```
Archivo -> Elegir perfil
```

## 7 Configuración de rendimiento de OpenShot

En OpenShot se puede revisar:

```
Editar -> Preferencias -> Rendimiento
```

Para máxima compatibilidad:

```
Modo de decodificacion por hardware: Sin aceleracion
Tarjeta grafica de decodificacion por hardware: no usar si falla
Tarjeta grafica de codificacion por hardware: no usar si falla
Usar renderizacion Blender GPU: desactivado
```

Después de cambiar estas opciones, se recomienda cerrar y volver a abrir OpenShot.

## 8 Limpieza de caché de OpenShot

Si OpenShot conserva datos defectuosos de previsualización o render, limpiar la caché puede ayudar.

```
rm -rf ~/.cache/openshot_qt
```

Luego abrir nuevamente el proyecto.

## 9 Verificación técnica del video fuente

Antes de culpar a OpenShot, se debe inspeccionar el archivo original con ffprobe.

```
ffprobe -hide_banner "Videograbacion 2026-06-04 20:58:49.mp4"
```

También se puede mostrar información específica del primer stream de video:

```
ffprobe -v error -select_streams v:0 \
-show_entries stream=codec_name,width,height,r_frame_rate,avg_frame_rate,pix_fmt \
-of default=noprint_wrappers=1 \
"Videograbacion 2026-06-04 20:58:49.mp4"
```

## 10 Conversión preventiva del video fuente con FFmpeg

Si OpenShot muestra el video en previsualización pero exporta negro, se recomienda convertir primero el archivo fuente a un MP4 más compatible.

```
ffmpeg -i "Videograbacion 2026-06-04 20:58:49.mp4" \
-c:v libx264 \
-preset veryfast \
-crf 18 \
-pix_fmt yuv420p \
```

```
-r 30 \
-c:a aac \
-b:a 192k \
"Videograbacion_convertida_openshot.mp4"
```

Luego se debe importar el archivo convertido en OpenShot y eliminar el clip anterior del proyecto.

## 11 Render alternativo desde terminal si OpenShot falla

Si después de ajustar perfiles, códecs, caché y aceleración el archivo exportado sigue saliendo negro, se puede renderizar desde terminal usando FFmpeg. Este método evita el exportador interno de OpenShot y procesa directamente los clips del archivo `.osp`.

### 11.1 Script de render con FFmpeg

Crear el archivo:

```
nano render_texmaker_ffmpeg.sh
```

Pegar el siguiente contenido:

```
#!/usr/bin/env bash
set -euo pipefail

# Renderiza un proyecto OpenShot con FFmpeg.
# Uso:
#  bash render_texmaker_ffmpeg.sh
#  bash render_texmaker_ffmpeg.sh /ruta/proyecto.osp /ruta/salida.mp4

PROJECT="${1:-/home/pc/Escritorio/Textmaker.osp}"
OUTPUT="${2:-/home/pc/Escritorio/Textmaker_render_ffmpeg.mp4}"

WORKDIR="$(mktemp -d)"
CONCAT_LIST="$WORKDIR/concat_list.txt"
SEGMENT_DIR="$WORKDIR/segments"
mkdir -p "$SEGMENT_DIR"

cleanup() {
    rm -rf "$WORKDIR"
}
trap cleanup EXIT

echo "====="
echo " Render FFmpeg para proyecto OpenShot"
echo "====="
echo "Proyecto: $PROJECT"
echo "Salida  : $OUTPUT"
echo

if ! command -v ffmpeg >/dev/null 2>&1; then
    echo "ERROR: ffmpeg no esta instalado."
    echo "Instala con: sudo apt update && sudo apt install ffmpeg"
    exit 1
fi

if ! command -v ffprobe >/dev/null 2>&1; then
    echo "ERROR: ffprobe no esta instalado."
    echo "Instala con: sudo apt update && sudo apt install ffmpeg"
    exit 1
```

```
fi

if ! command -v python3 >/dev/null 2>&1; then
    echo "ERROR: python3 no esta instalado."
    echo "Instala con: sudo apt update && sudo apt install python3"
    exit 1
fi

if [ ! -f "$PROJECT" ]; then
    echo "ERROR: no existe el archivo de proyecto: $PROJECT"
    exit 1
fi

echo "[1/5] Analizando proyecto OpenShot..."

python3 - "$PROJECT" "$WORKDIR/clips.tsv" <<'PY'
import json
import sys
from pathlib import Path

project_path = Path(sys.argv[1])
out_path = Path(sys.argv[2])

data = json.loads(project_path.read_text(encoding="utf-8", errors="replace"))
project_dir = project_path.parent

files = {}
for f in data.get("files", []):
    fid = str(f.get("id", ""))
    path = f.get("path", "")
    if fid and path:
        p = Path(path)
        if not p.is_absolute():
            p = (project_dir / p).resolve()
        files[fid] = str(p)

rows = []
for clip in data.get("clips", []):
    file_id = str(clip.get("file_id", ""))
    if file_id not in files:
        continue

    p = Path(files[file_id])
    if p.suffix.lower() not in [".mp4", ".mkv", ".mov", ".avi", ".webm", ".m4v"]:
        continue

    try:
        position = float(clip.get("position", 0.0))
        start = float(clip.get("start", 0.0))
        end = float(clip.get("end", 0.0))
        layer = int(float(clip.get("layer", 0)))
    except Exception:
        continue

    rows.append((position, layer, start, end, str(p)))

rows.sort(key=lambda r: (r[0], r[1], r[2]))

if not rows:
```

```

candidates = []
for p in files.values():
    pp = Path(p)
    if pp.suffix.lower() in [".mp4", ".mkv", ".mov", ".avi", ".webm", ".m4v"]:
        candidates.append(str(pp))
candidates = sorted(set(candidates))
if candidates:
    rows = [(0.0, 0, 0.0, 0.0, candidates[0])]

with out_path.open("w", encoding="utf-8") as out:
    for position, layer, start, end, path in rows:
        out.write(f"{position}\t{layer}\t{start}\t{end}\t{path}\n")

print(f"Clips detectados: {len(rows)}")
for r in rows:
    print(f" pos={r[0]:.3f}s start={r[2]:.3f}s end={r[3]:.3f}s file={r[4]}")
PY

if [ ! -s "$WORKDIR/clips.tsv" ]; then
    echo "ERROR: no se encontraron clips de video dentro del proyecto."
    exit 1
fi

echo
echo "[2/5] Verificando videos fuente..."
while IFS= read -r video; do
    [ -z "$video" ] && continue
    if [ ! -f "$video" ]; then
        echo "ERROR: no se encontro video fuente: $video"
        exit 1
    fi
    echo "Fuente: $video"
    ffmpeg -v error -select_streams v:0 \
        -show_entries stream=codec_name,width,height,r_frame_rate,avg_frame_rate,pix_fmt \
        -of default=noprint_wrappers=1 "$video" || true
    echo
done <<(awk -F '\t' '{print $5}' "$WORKDIR/clips.tsv" | sort -u)

echo "[3/5] Renderizando segmentos con FFmpeg..."
: > "$CONCAT_LIST"

i=0
while IFS=$'\t' read -r position layer start end video; do
    i=$((i + 1))
    seg="$SEGMENT_DIR/segment_$(printf "%04d" "$i").mp4"

    echo "Segmento $i: $video"
    echo " inicio=$start fin=$end"

    if awk "BEGIN {exit !((($end) > ($start))}"; then
        duration="$(awk "BEGIN {printf \"%.6f\", ($end) - ($start)}")"
        ffmpeg -y \
            -ss "$start" \
            -i "$video" \
            -t "$duration" \
            -vf "scale=1024:768:flags=lanczos,setsar=1,fps=30,format=yuv420p" \
            -c:v libx264 \
            -preset veryfast \

```

```

        -crf 18 \
        -c:a aac \
        -b:a 192k \
        -ar 48000 \
        -ac 2 \
        -movflags +faststart \
        "$seg"
    else
        ffmpeg -y \
            -i "$video" \
            -vf "scale=1024:768:flags=lanczos,setsar=1,fps=30,format=yuv420p" \
            -c:v libx264 \
            -preset veryfast \
            -crf 18 \
            -c:a aac \
            -b:a 192k \
            -ar 48000 \
            -ac 2 \
            -movflags +faststart \
            "$seg"
    fi

    printf "file '%s'\n" "$seg" >> "$CONCAT_LIST"
done < "$WORKDIR/clips.tsv"

echo
echo "[4/5] Uniendo segmentos..."
if [ "$i" -eq 1 ]; then
    cp "$SEGMENT_DIR/segment_0001.mp4" "$OUTPUT"
else
    ffmpeg -y -f concat -safe 0 -i "$CONCAT_LIST" -c copy -movflags +faststart "$OUTPUT"
fi

echo
echo "[5/5] Verificando archivo final..."
ffprobe -v error -select_streams v:0 \
    -show_entries stream=codec_name,width,height,r_frame_rate,avg_frame_rate,pix_fmt \
    -of default=noprint_wrappers=1 "$OUTPUT" || true

echo
echo "====="
echo " Render finalizado"
echo "====="
echo "Archivo generado: $OUTPUT"
echo "Para reproducirlo: xdg-open \"$OUTPUT\""

```

## 11.2 Ejecución del script

Conceder permisos de ejecución:

```
chmod +x render_texmaker_ffmpeg.sh
```

Ejecutar con rutas por defecto:

```
./render_texmaker_ffmpeg.sh
```

Ejecutar indicando proyecto y salida:

```
./render_texmaker_ffmpeg.sh \
```

```
/home/pc/Escritorio/Textmaker.osp \  
/home/pc/Escritorio/Textmaker_render_ffmpeg.mp4
```

Abrir el resultado:

```
xdg-open /home/pc/Escritorio/Textmaker_render_ffmpeg.mp4
```

## 12 Render manual de un solo archivo sin usar el proyecto OpenShot

Si el proyecto `.osp` no es necesario y solo se quiere convertir el video fuente, se puede usar:

```
ffmpeg -i "Videograbacion 2026-06-04 20:58:49.mp4" \  
-vf "scale=1024:768:flags=lanczos,setsar=1,fps=30,format=yuv420p" \  
-c:v libx264 \  
-preset veryfast \  
-crf 18 \  
-c:a aac \  
-b:a 192k \  
-ar 48000 \  
-ac 2 \  
-movflags +faststart \  
"salida_final_1024x768.mp4"
```

## 13 Recomendaciones para futuros videos

### 13.1 Grabar en resolución compatible con la publicación

Si el video será publicado en YouTube, se recomienda grabar directamente en 1280x720 o 1920x1080. Si se graba desde un monitor 1024x768, se debe decidir si se conserva 4:3 o se edita sobre un lienzo 16:9 con fondo corporativo.

### 13.2 Usar fondo corporativo para capturas 4:3

Una solución estética consiste en colocar el video 4:3 al centro y agregar una imagen de fondo 16:9 con identidad corporativa. Esto evita barras negras y mantiene una presentación profesional.

```
Pista 2: video 1024x768  
Pista 1: fondo corporativo 1920x1080
```

### 13.3 Guardar una copia del proyecto antes de exportar

Antes de exportar, guardar una copia del proyecto permite retroceder si el archivo `.osp` queda configurado con perfiles erróneos.

```
Textmaker.osp  
Textmaker_respaldo_antes_exportar.osp
```

## 14 Conclusión

Cuando OpenShot muestra correctamente la previsualización, pero el archivo exportado aparece negro, el problema suele estar asociado a incompatibilidades de perfil, relación de aspecto, aceleración de hardware, códec o caché de render. La primera corrección consiste en alinear el perfil del proyecto con la resolución real del video fuente y exportar con `mp4`, `libx264`, `aac` y `yuv420p`. Si OpenShot continúa fallando, el render con FFmpeg desde terminal permite recuperar el flujo de producción y generar un archivo final estable para redes sociales o documentación técnica.

## Contacto institucional

<b>Empresa</b>	TECPROG WORLD E.I.R.L.
<b>RUC</b>	20608743252
<b>Actividad principal</b>	Actividades de arquitectura e ingeniería y actividades conexas de consultoría técnica
<b>Ubicación</b>	San Martín de Porres, Lima, Perú
<b>Sitio web</b>	<a href="https://tecprog-world-store.github.io/">https://tecprog-world-store.github.io/</a>