Usage Examples for New Features

GUI Usage Examples

Example 1: Processing a Video with Custom Suffix and No Audio

Scenario: You want to blur humans in a video, use a custom filename, and remove audio for privacy.

Steps:
1. Launch the GUI: `python gui_enhanced.py`
2. Click "Browse File" and select your video (e.g., `meeting.mp4`)
3. Choose your preferred mask type (Black Mask or Blur)
4. In the **Output Settings** section:
- Change "Filename suffix" to: `-privacy`
- **Uncheck** "🔊 Keep audio in output videos"
5. Click "🚀 Process Media"

Result:
- Original file: `meeting.mp4`
- Output file: `meeting-privacy.mp4` (without audio)

Example 2: Batch Processing with Custom Suffix

Scenario: You have a folder of vacation photos and want to blur people with a custom suffix.

Steps:
1. Launch the GUI: `python gui_enhanced.py`
2. Click "Browse Folder" and select your folder
3. Set folder media filter to "Images Only"
4. Choose your mask type (e.g., "Blur")
5. In the **Output Settings** section:
- Change "Filename suffix" to: `-vacation-blurred`
6. Click "🚀 Process Media"

Result:
- `photo1.jpg` → `photo1-vacation-blurred.jpg`
- `photo2.jpg` → `photo2-vacation-blurred.jpg`
- `photo3.png` → `photo3-vacation-blurred.png`

Example 3: Processing Video with Audio Preserved

Scenario: You want to blur humans in a presentation video but keep the audio.

Steps:
1. Launch the GUI: `python gui_enhanced.py`
2. Click "Browse File" and select `presentation.mp4`
3. Choose "Black Mask" for fastest processing

4. In the **Output Settings** section:
- Set “Filename suffix” to: `-anonymous`
- **Keep** “🔊 Keep audio in output videos” checked
5. Click “🚀 Process Media”

Result:
- Original: `presentation.mp4` (with audio)
- Output: `presentation-anonymous.mp4` (with audio preserved)

CLI Usage Examples

While the CLI doesn’t directly support these options yet, you can achieve similar results by using the Python API:

Example: Python Script with Custom Settings

```python
from pathlib import Path
from blur_humans import HumanBlurProcessor

# Create processor with custom settings
processor = HumanBlurProcessor(
    model_name='yolov8n-seg.pt',
    mask_type='blur',
    blur_intensity=151,
    blur_passes=3,
    filename_suffix='-custom',  # Custom suffix
    keep_audio=False  # Remove audio
)

# Process a video
video_path = Path('my_video.mp4')
processor.process_video(video_path, confidence=0.5)

# Result: my_video-custom.mp4 (without audio)
```

Example: Batch Processing with Python API

```python
from pathlib import Path
from blur_humans import HumanBlurProcessor

# Create processor with settings
processor = HumanBlurProcessor(
    model_name='yolov8n-seg.pt',
    mask_type='black',
    filename_suffix='-batch-processed',
    keep_audio=True  # Keep audio
)

# Process directory
directory = Path('./my_media_folder')
successful, total = processor.process_directory(
    directory,
    confidence=0.5,
    media_type='videos'  # Only process videos
)

print(f"Processed {successful}/{total} files")
```

Common Use Cases

Use Case 1: Social Media Content

Goal: Create privacy-friendly content for social media
- **Suffix:** `-social`
- **Audio:** Keep (checked)
- **Mask Type:** Blur

Use Case 2: Training Materials

Goal: Anonymize training videos
- **Suffix:** `-training-anon`
- **Audio:** Keep (checked)
- **Mask Type:** Black Mask

Use Case 3: Surveillance Footage

Goal: Blur faces in surveillance videos
- **Suffix:** `-surveillance`
- **Audio:** Remove (unchecked)
- **Mask Type:** Blur

Use Case 4: Documentation Photos

Goal: Anonymize people in documentation
- **Suffix:** `-docs`
- **Audio:** N/A (images)
- **Mask Type:** Black Mask

Use Case 5: Research Data

Goal: Anonymize research participants
- **Suffix:** `-research-anonymized`
- **Audio:** Remove (unchecked)
- **Mask Type:** Black Mask

Tips and Best Practices

Filename Suffix Tips

1. **Use descriptive suffixes** that indicate the processing type
   - Good: `-privacy`, `-blurred`, `-anonymous`
   - Avoid: `-1`, `-new`, `-temp`

2. **Keep suffixes short** to avoid overly long filenames
   - Good: `-proc`
   - Avoid: `-this-is-a-very-long-suffix-name`

3. **Use hyphens or underscores** for readability
   - Good: `-batch-1`, `_processed`
   - Works: `processed` (no separator)

4. **Be consistent** across batches for easy organization

Audio Handling Tips

1. **Check ffmpeg availability** before processing videos
   - Run: `ffmpeg -version` in terminal
   - Install if needed

2. **Consider file size**
   - Removing audio can significantly reduce file size
   - Useful when audio is not needed

3. **Privacy considerations**
   - Remove audio when it contains sensitive information
   - Keep audio when voice content is not identifying

4. **Testing**
   - Test with a small video first before batch processing
   - Verify the output meets your requirements

Troubleshooting

Issue: Audio checkbox disabled

Solution: Install ffmpeg on your system

Issue: Custom suffix not applied

Solution: Make sure the suffix field is not empty (use at least `-` or `_`)

Issue: Output file overwrites existing file

Solution: Use a unique suffix or move existing processed files

Issue: Large video files taking too long

Solution:
- Use “Black Mask” instead of “Blur” for faster processing
- Consider unchecking audio to skip audio extraction/merging

Session Persistence

During a single session:
- Your custom suffix is **remembered** between different file selections
- Audio handling preference is **saved** for the session
- All other settings are maintained

After closing and reopening the GUI:
- Settings return to defaults
- Suffix returns to `-background`
- Audio handling returns to “Keep” (checked)

Keyboard Shortcuts (GUI)

While in the GUI:
- `Alt+F` - Focus on filename suffix field
- `Alt+A` - Toggle audio checkbox (if available in your OS)
- `Ctrl+O` - Open file browser (may work depending on system)

Command Line Alternative

If you prefer command-line control, you can modify the `blur_humans.py` script to add CLI arguments for these features:

```
# Example of what could be added to CLI
python blur_humans.py video.mp4 \
    --mask-type blur \
    --suffix "-custom" \
    --no-audio
```

Note: This requires modifying the CLI argument parser in `blur_humans.py`

Integration with Workflows

Workflow 1: Automated Processing Pipeline

```
#!/bin/bash
# Process videos in batches with different settings

# Batch 1: Privacy videos (no audio)
python gui_enhanced.py --batch \
    --input ./privacy_videos/ \
    --suffix "-privacy" \
    --no-audio

# Batch 2: Training videos (with audio)
python gui_enhanced.py --batch \
    --input ./training_videos/ \
    --suffix "-training" \
    --keep-audio
```

Workflow 2: Organize by Date

```
# Use date-based suffixes
DATE=$(date +%Y%m%d)
python process_with_suffix.py --suffix "-processed-${DATE}"
```

Questions & Support

If you have questions about using these new features:
1. Check the built-in Help (❓ Help button in GUI)
2. Review this examples document
3. Contact: apps@globalemancipation.ngo

Happy Processing! 🚀