Octavia - Standard Video Translator: User Flow Documentation


Document Information

- **Title**: User Flow Documentation for Standard Video Translator

- **Version**: 1.0

- **Date**: January 2025

- **Feature**: Standard Video Translator

- **Platform**: Cross-platform Desktop (Flutter UI + Python Backend)
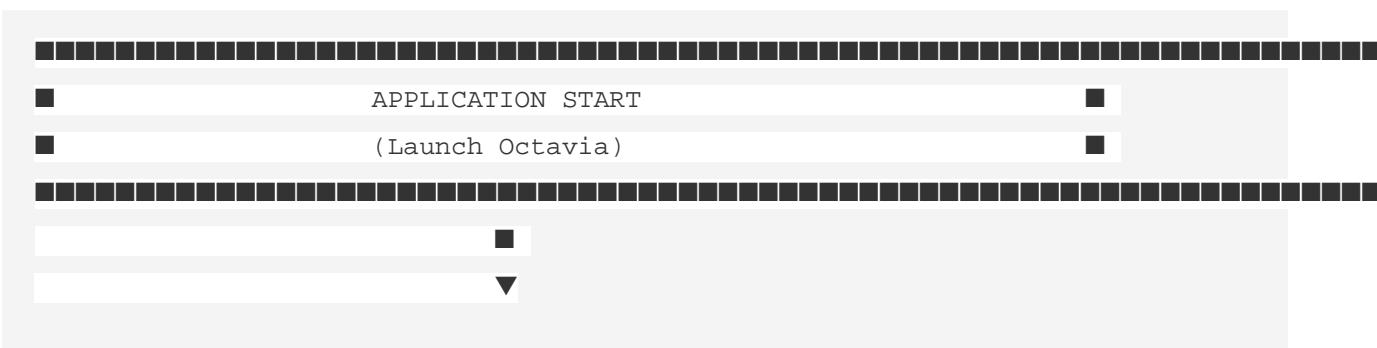
Overview

This document outlines the complete user flow for Octavia's Standard Video Translator feature, mapping the journey from initial video upload through processing to final output delivery. The flow is designed for simplicity, requiring minimal user input while providing transparency through AI-driven automation.

Main User Flow

Flow 1: Standard Video Translation (Primary Flow)

```
███████████████████████████████████████████████████████████████
█                    APPLICATION START                       █
█                    (Launch Octavia)                        █
███████████████████████████████████████████████████████████████
                              ■
                              ▼
```
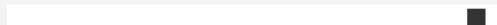
SCREEN 1: UPLOAD & LANGUAGE SELECTION

- Drag & Drop Zone (or Browse Button)
- Original Language Dropdown (Auto-detected)
- Target Language Dropdown (Manual Selection)
- AI Insight Banner
- "Start Translation" Button

[Valid Input]   [Invalid Input]

▼   ▼
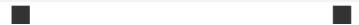
Validation   Error Toast
Passed   Display

▼

SCREEN 2: PROCESSING DASHBOARD

- Overall Progress Bar (Percentage)
- Step-by-Step Timeline:
  - Splitting (✓/■/■)
  - Transcribing (✓/■/■)
  - Translating (✓/■/■)
  - Dubbing (✓/■/■)
  - Syncing (✓/■/■)
  - Merging (✓/■/■)
- Current Status Text (e.g., "Dubbing chunk 56/82")
- ETA Display
- Action Buttons:
  - Pause (saves checkpoint)
  - Resume (from checkpoint)
  - Cancel (exports partials)

- • "Play Sample Chunk" Button (mid-process)
- • Collapsible Technical Logs

[Processing Complete]   [User Action]

[Pause]   [Cancel]
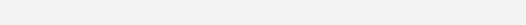
■ Save   ■ Export
■ State   ■ Partial

■ SCREEN 3: OUTPUT REVIEW & EXPORT

- • Video Player (Left):
  - – Full video preview
  - – Seekable timeline
  - – "Check Spots" (5 random 30s clips)
- • Stats Panel (Right):
  - – Sync: 98%
  - – Duration: Exact Match
  - – Size: Optimized
- • Action Buttons:
  - – "AI Refine" (1 free per run)
  - – "Download MP4"
  - – "Include original files in .zip" (checkbox)
- • Feedback Slider: "Pacing Good?" (1-5)
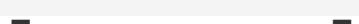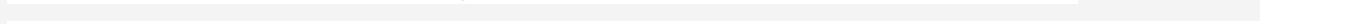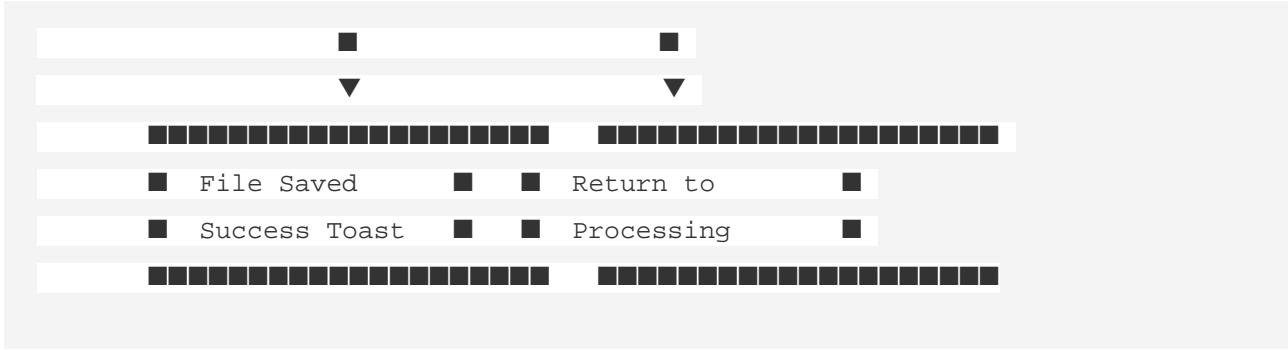
[Download]   [Refine]

```
        ■            ■
        ▼            ▼
  ■■■■■■■■■■■■■■■■  ■■■■■■■■■■■■■■■■■■
  ■  File Saved   ■  ■  Return to        ■
  ■  Success Toast ■  ■  Processing      ■
  ■■■■■■■■■■■■■■■■  ■■■■■■■■■■■■■■■■■■
```

Detailed Flow Steps

Step 1: Upload & Language Selection

Entry Point: User launches Octavia and navigates to Standard Video Translator

User Actions: 1. **Upload Video** - Option A: Drag & drop video file into upload zone - Option B: Click "Browse Files" button - Supported formats: MP4, AVI, MOV - Max size: 50GB

1. **AI Auto-Detection** (Background)

2. System extracts first 30 seconds

3. Runs faster-whisper (tiny model) for quick language detection

4. Auto-populates "Original Language" dropdown

Displays metadata (duration, size)

Language Selection

Original Language:

- Pre-filled with auto-detected result

- User can override if incorrect

- Dropdown with 100+ languages

Target Language:

- User manually selects from dropdown

- System validates (prevents same-language selection)

- Shows popular suggestions

AI Insight Display

10. Banner appears: "AI will split into ~7k chunks and ensure perfect 10-hour sync"

Provides transparency without complexity

Start Translation

13. Button enabled only after:

• Valid video file uploaded

• Both languages selected

• Languages are different

14. Clicking initiates pipeline

Error Handling: - Invalid format → Toast: "Invalid format—try MP4" - File too large → Toast: "File exceeds 50GB limit" - Same languages → Disable start button, show hint - No file → Disable start button

Exit Points: - Cancel → Return to upload screen, clear selections - Start → Proceed to Processing Dashboard

---

Step 2: Processing Dashboard

Entry Point: After "Start Translation" clicked

System Actions (Automated): 1. **Pre-Analysis** (5-30s) - Extract full metadata via FFprobe - AI orchestrator determines optimal chunk size (30-120s) - Generate chunk plan - Update progress: "Analyzing video structure..."

1. **Splitting** (10-20% of total time)

2. Split video/audio into chunks

3. Optional vocal separation if BGM detected

4. Update progress: "Splitting into X chunks..."

Step icon: Scissor (✓ when complete)

STT Extraction (20-30% of total time)

7. Transcribe each chunk in parallel

8. Generate SRT files with timestamps

9. Update progress: "Transcribing chunk X of Y..."

Step icon: Microphone (✓ when complete)

Translation (15-25% of total time)

12. Translate transcripts with AI condensation
13. Ensure ≤1.2x original length per segment
14. Update progress: "Translating chunk X: Condensed to fit 3s slot"

Step icon: Globe (✓ when complete)

TTS Generation (20-30% of total time)

17. Generate dubbed audio per segment
18. AI selects optimal voice/speed
19. Update progress: "Generating audio for chunk X..."

Step icon: Speaker (✓ when complete)

Sync & Stitching (10-20% of total time)

22. Adjust audio speeds to match durations
23. Stitch segments together
24. Update progress: "Syncing chunk X..."

Step icon: Sync (✓ when complete)

Final Merge (5-10% of total time)

27. Combine all segments
28. Global duration check
29. Update progress: "Finalizing output..."
30. Step icon: Merge (✓ when complete)

User Actions: - **Pause**: Saves checkpoint, pauses processing - **Resume**: Reloads from checkpoint, continues - **Cancel**: Exports partial results (zipped), returns to upload - **Play Sample Chunk**: Preview 10-30s dubbed segment mid-process - **View Logs**: Expand collapsible technical logs

Progress Indicators: - Overall progress bar (0-100%) - Step-by-step timeline with icons - Current status text - ETA calculation (dynamic based on hardware)

Error Handling: - Chunk failure → Retry 2x, log error, continue - OOM → Reduce batch size, notify user - GPU unavailable → Fallback to CPU, update ETA - Fatal error → Save checkpoint, show error, allow resume

Exit Points: - Processing complete → Proceed to Output Review - Cancel → Export partials, return to upload - Fatal error → Show error screen, option to retry

---

Step 3: Output Review & Export

Entry Point: Processing completes successfully

User Actions: 1. **Preview Video** - Full video player with seekable timeline - Play/pause controls - Switch between "Full Preview" and "Check Spots" (5 random 30s clips)

1. **Review Stats**

2. Sync accuracy: 98%

3. Duration match: Exact Match

File size: Optimized

AI Refine (Optional, 1 free per run)

6. Click "AI Refine" button

7. Enter prompt (e.g., "Softer voice?")

8. System reruns TTS/translation for affected segments

9. Returns to processing dashboard briefly

Returns to review with updated output

Download

12. Click "Download MP4" button

13. Optional: Check "Include original files in .zip"

14. File saves to default location

Success toast notification

Feedback (Optional)

17. Adjust "Pacing Good?" slider (1-5)

18. Feeds into local AI fine-tuning

Error Handling: - Player load failure → Show error, retry button - Download failure → Retry mechanism - Refine failure → Show error, allow retry

Exit Points: - Download complete → Success screen, option to start new translation - Refine requested → Return to processing dashboard - Close app → Save state, exit

---

Alternative Flows

Flow 2: Pause & Resume

```
Processing Dashboard
    ■
    ■■→ [Pause Clicked]
    ■    ■
    ■    ■■→ Save checkpoint (pickle state)
    ■    ■■→ Pause all processing threads
    ■    ■■→ Update UI: "Paused"
    ■    ■■→ Show "Resume" button
    ■
    ■■→ [Resume Clicked]
        ■
        ■■→ Load checkpoint
        ■■→ Resume from last chunk
        ■■→ Continue processing
```

Flow 3: Cancel & Partial Export

```
Processing Dashboard
    ■
    ■■→ [Cancel Clicked]
        ■
        ■■→ Stop all processing
        ■■→ Package completed chunks:
        ■    ■■→ Translated SRT files
        ■    ■■→ Generated audio segments
        ■    ■■→ Progress log
```

```
        ■■→ Create .zip file
        ■■→ Show download option
        ■■→ Return to Upload Screen
```

Flow 4: Error Recovery
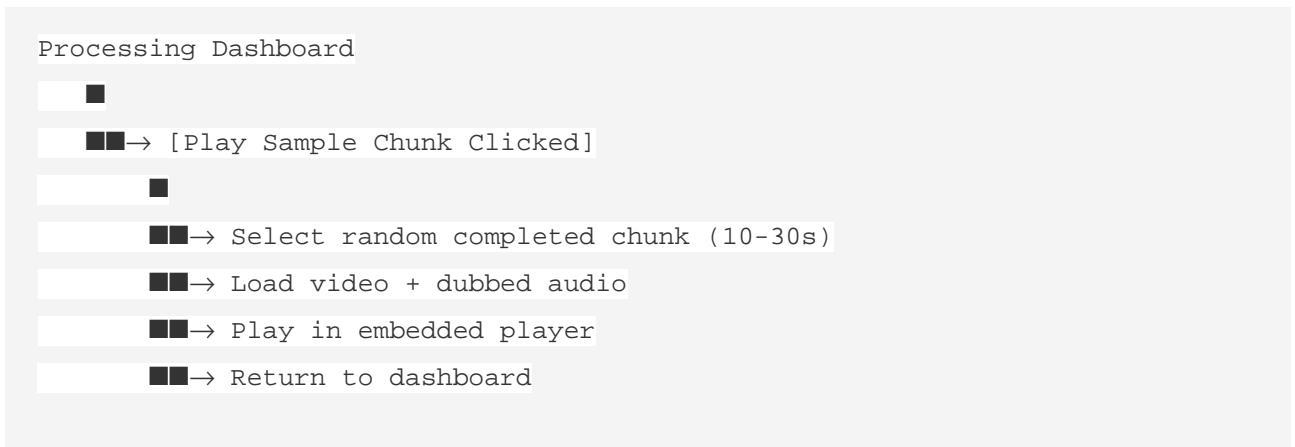
```
Any Processing Step
    ■
    ■■→ [Error Detected]
    ■      ■
    ■      ■■→ Retry (2 attempts)
    ■      ■    ■
    ■      ■    ■■→ [Success] → Continue
    ■      ■    ■■→ [Failure] → Log error, skip chunk
    ■      ■
    ■      ■■→ [Fatal Error]
    ■           ■
    ■           ■■→ Save checkpoint
    ■           ■■→ Show error screen:
    ■           ■    ■■→ Error message
    ■           ■    ■■→ "Retry" button
    ■           ■    ■■→ "Cancel & Export" button
    ■           ■
    ■           ■■→ [User Choice]
    ■                ■■→ Retry → Resume from checkpoint
    ■                ■■→ Cancel → Export partials
```

Flow 5: Long Video Handling

```
Upload Screen
    ■
    ■■→ [Video > 1 hour detected]
         ■
         ■■→ AI adjusts chunk size (larger for efficiency)
         ■■→ Show notification: "Long video detected. Estimated time: X hours"
         ■■→ Option to proceed or split manually
         ■
```

```
              ■■→ [User proceeds]
                    ■■→ Continue with optimized settings
```

Flow 6: Preview During Processing

```
Processing Dashboard
    ■
    ■■→ [Play Sample Chunk Clicked]
         ■
          ■■→ Select random completed chunk (10-30s)
          ■■→ Load video + dubbed audio
          ■■→ Play in embedded player
          ■■→ Return to dashboard
```

Edge Cases & Special Scenarios

Edge Case 1: No Audio Detected

- **Detection**: During STT, no speech found

- **Action**: Show warning, proceed with silent output

- **User Notification**: "No speech detected. Output will be silent video."

Edge Case 2: Multilingual Content

- **Detection**: STT detects language switches

- **Action**: Split segments per language, translate separately

- **User Notification**: "Multiple languages detected. Processing per segment."

Flow 7: AI Refine Request

```
Output Review Screen
    ■
    ■■→ [AI Refine Clicked]
        ■
        ■■→ Show prompt input:
    ■       ■■→ Text field: "Describe desired change"
    ■       ■■→ Examples: "Softer voice?", "Faster pace?"
    ■
        ■■→ [User submits]
        ■       ■
        ■       ■■→ Return to Processing Dashboard
        ■       ■■→ Rerun TTS/Translation for affected segments
        ■       ■■→ Update progress: "Refining based on feedback..."
        ■       ■■→ Return to Output Review with updated output
        ■
        ■■→ [Limit reached]
                ■■→ Show: "Refine limit reached. Upgrade for more."
```

Edge Case 3: Rare Language

- **Detection**: WER >15% during STT
- **Action**: Continue with warning
- **User Notification**: "Accuracy may vary for this language. Suggest re-upload."

Edge Case 4: Hardware Limitations

- **Detection**: GPU unavailable or low RAM
- **Action**: Fallback to CPU, adjust batch sizes
- **User Notification**: "GPU not detected. Processing on CPU. ETA doubled."
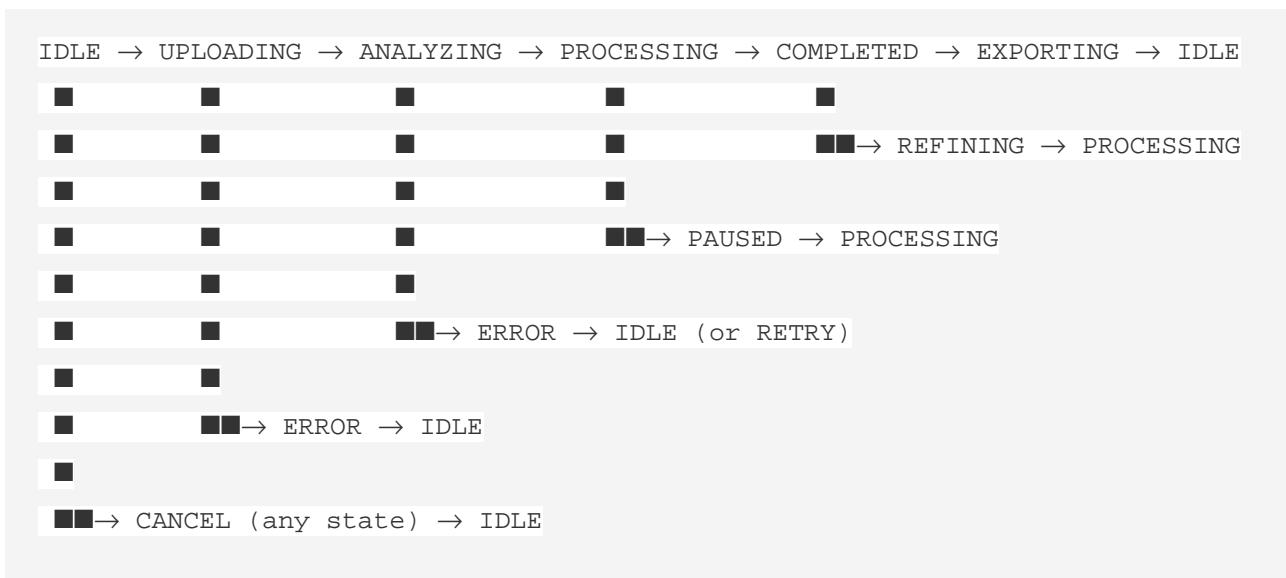
State Management

Application States

1. **IDLE**: Initial state, upload screen ready

2. **UPLOADING**: File being processed/validated

3. **ANALYZING**: Pre-analysis in progress

4. **PROCESSING**: Pipeline execution active

5. **PAUSED**: Processing paused, checkpoint saved

6. **COMPLETED**: Processing finished, ready for review

7. **ERROR**: Fatal error occurred, recovery options shown

8. **REFINING**: AI refine in progress

9. **EXPORTING**: Download in progress

State Transitions

```
IDLE → UPLOADING → ANALYZING → PROCESSING → COMPLETED → EXPORTING → IDLE
  ■         ■           ■           ■            ■
  ■         ■           ■           ■          ■■→ REFINING → PROCESSING
  ■         ■           ■           ■
  ■         ■           ■         ■■→ PAUSED → PROCESSING
  ■         ■           ■
  ■         ■         ■■→ ERROR → IDLE (or RETRY)
  ■         ■
  ■       ■■→ ERROR → IDLE
  ■
■■→ CANCEL (any state) → IDLE
```

User Interface Elements

Screen 1: Upload & Language Selection

- **Components**:

- Drag & drop zone (large, centered)

- File browser button

- Language dropdowns (2)

- AI insight banner

- Start button

- Cancel button

Screen 2: Processing Dashboard

- **Components**:

- Progress bar (horizontal, percentage)

- Step timeline (vertical, icon-based)

- Status text (dynamic)

- ETA display

- Action buttons (Pause/Resume/Cancel)

- Sample chunk player (embedded)

- Logs section (collapsible)

Screen 3: Output Review & Export

- **Components**:

- Video player (left, 70% width)

- Stats panel (right, 30% width)

- Action buttons (Refine/Download)

- Feedback slider

- Checkbox (zip option)

Data Flow

Input Data

- Video file (binary)

- Original language (string, auto-detected)

- Target language (string, user-selected)

Processing Data

- Chunk metadata (JSON: paths, durations)
- Transcripts (SRT files)
- Translated texts (strings)
- Audio segments (WAV files)
- Synced chunks (MP4 files)

Output Data

- Final video (MP4)
- Optional intermediates (ZIP: SRTs, audio)

Internal State

- Pipeline state (pickle: progress, checkpoints)
- AI logs (text: decisions, errors)

Performance Considerations

Expected Timings

- **10-minute video**: 5-10 minutes processing (mid-range hardware)
- **10-hour video**: 5-10 hours processing (mid-range hardware)
- **Upload**: <5 seconds
- **Pre-analysis**: 5-30 seconds
- **Preview**: Instant (if chunk ready)

Optimization Points

- Parallel processing for STT/TTS (multi-threaded)

- GPU acceleration when available

- Progressive cleanup (delete chunks after use)

- Checkpointing every 10% progress

Accessibility Features

- **Keyboard Navigation**: Full keyboard support for all actions

- **Screen Reader**: ARIA labels on all interactive elements

- **High Contrast**: Dark mode with sufficient contrast ratios

- **Focus Indicators**: Clear focus states for all inputs

- **Error Announcements**: Screen reader announcements for errors

---

Success Metrics

User Experience

- **Time to First Action**: <5 seconds (upload ready)

- **Error Rate**: <2% fatal errors

- **User Satisfaction**: >4/5 rating

- **Completion Rate**: >90% users complete flow

Technical Performance

- **Duration Accuracy**: 100% match (exact)

- **Sync Accuracy**: >98%

- **STT Accuracy**: >95% WER

- **Processing Speed**: 1-2x realtime on mid-range hardware

Future Enhancements (v2)

- **Advanced Mode**: User-configurable parameters

- **Multi-Role Dubbing**: Assign different voices to speakers

- **Subtitle Editing**: Inline subtitle editor

- **Batch Processing**: Multiple videos at once

- **Cloud Processing**: Optional cloud offload for long videos

- **Real-time Preview**: Live preview during processing

Conclusion

This user flow document provides a comprehensive map of the Standard Video Translator feature, ensuring a smooth, intuitive experience while maintaining transparency through AI-driven automation. The flow prioritizes simplicity for casual users while supporting power users through optional refinements and previews.

Document End