stream2clip 사용 설명서

적용 버전	v0.19
대상	Windows 설치 프로그램, 실행 파일 및 파이썬 패키지 사용자
배포처	github.com/BudongJW/stream2clip

1. 설치

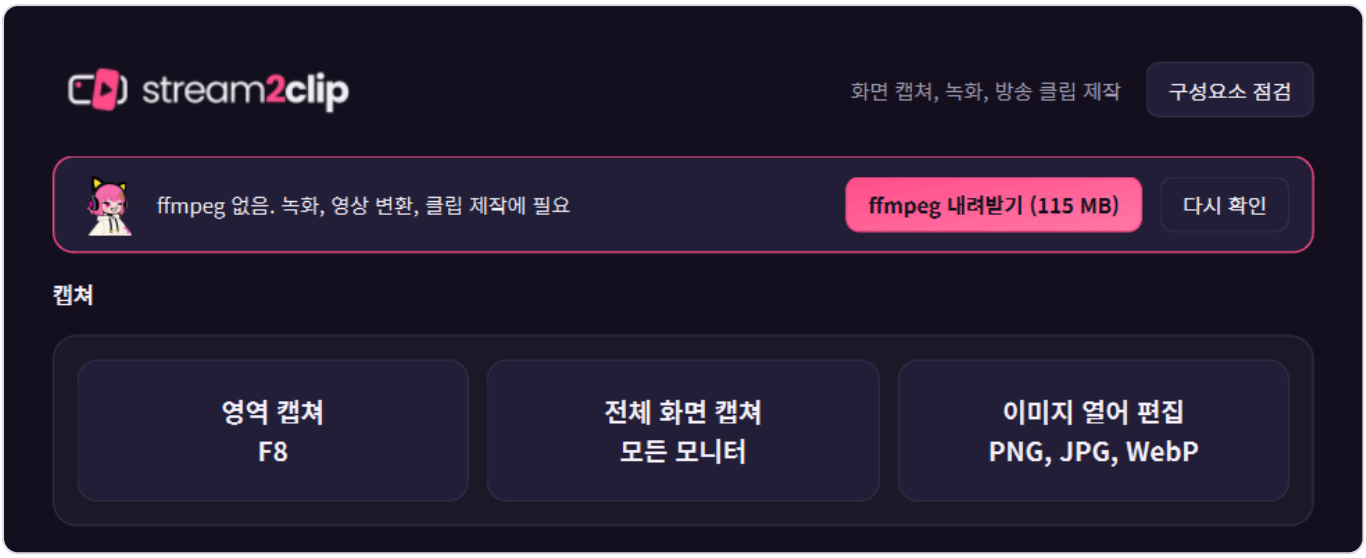
1.1 Windows 설치 프로그램

1. 릴리즈 페이지에서 Windows용 설치 프로그램(`setup.exe`) 다운로드 후 실행
2. 첫 화면에서 언어 선택: 한국어, 日本語, English. 선택한 언어가 설치 화면과 앱의 표시 언어로 적용
3. 설치 위치 확인. 관리자 권한 불필요
4. 추가 작업 화면에서 **ffmpeg 내려받기** 선택 유지 후 설치. 영상 변환에 필요한 ffmpeg(115MB)를 설치 과정에서 내려받으며, ffmpeg가 이미 있는 PC에서는 이 항목이 표시되지 않음
5. 시작 메뉴에서 stream2clip 실행

- ffmpeg 내려받기를 해제하였거나 실패한 경우: 앱 시작 화면 상단의 **ffmpeg 내려받기** 버튼으로 다시 진행
- 설치 없이 사용: zip 파일 다운로드 후 원하는 폴더에 압축 해제, `stream2clip-gui.exe` 실행. ffmpeg는 시작 화면의 **ffmpeg 내려받기** 버튼으로 준비
- 구성요소 확인: 시작 화면 우측 상단의 **구성요소 점검**에서 기능별 필요 구성요소의 유무와 해결 방법 표시. 명령줄에서는 `stream2clip setup`



구성요소 점검 창. ffmpeg와 얼굴 검출 모델은 이 창에서 내려받기 가능



ffmpeg가 없는 PC의 시작 화면. 안내는 ffmpeg가 준비되면 사라짐

- 앱의 표시 언어는 설치 후에도 런처 하단에서 변경 가능
- 제거: Windows 설정의 설치된 앱 목록에서 stream2clip 제거. 내려받은 ffmpeg도 함께 삭제

참고

- 코드 서명 미적용. Windows SmartScreen 경고가 표시될 수 있음
- Smart App Control 사용 환경에서는 실행이 차단될 수 있으며, 이 경우 파이썬 패키지 설치 사용
- 파일 무결성은 릴리즈 페이지의 SHA-256 값과 아래 명령의 결과를 대조하여 확인

```
Get-FileHash .\stream2clip-*-windows-x64.zip -Algorithm SHA256
```

1.2 파이썬 패키지

- 요구 사항: Python 3.10 이상, ffmpeg
- 지원 운영체제: Windows, macOS, Linux

```
pip install "stream2clip[gui,ingest,obs] @ git+https://github.com/BudongJW/stream2clip"
```

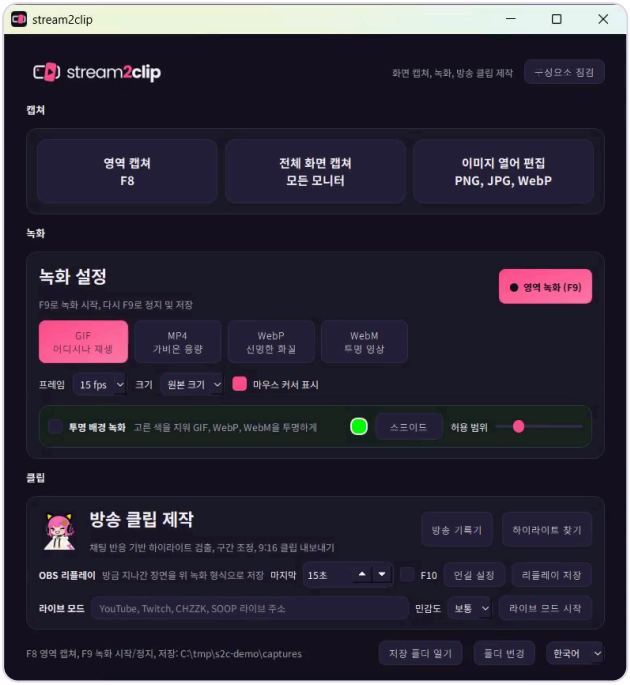
```
stream2clip gui
```

선택 항목	기능
`asr`	자막 받아쓰기(Whisper)
`detect`	아바타 위치 자동 검출

선택 항목	기능
`upload`	YouTube 업로드

선택 항목은 설치 명령의 대괄호 안에 추가. 자막 받아쓰기와 YouTube 업로드는 파이썬 패키지에서 제공.

2. 기본 사용법



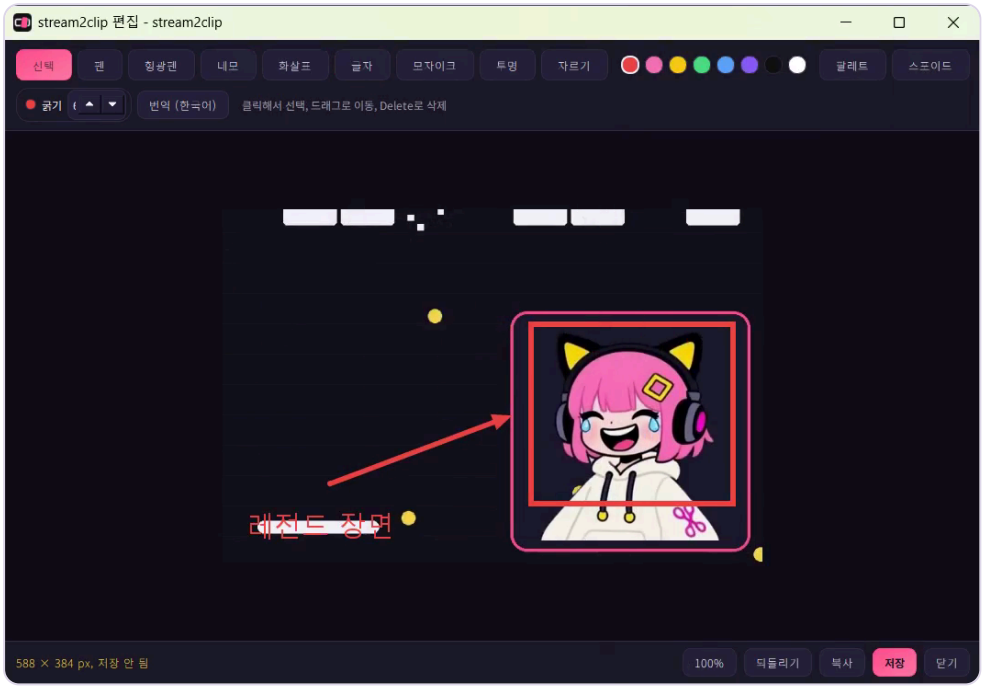
런처 화면. 상단부터 캡쳐, 화면 녹화, 클립 제작 영역

단축키	기능
`F8`	화면 영역 캡쳐 후 편집기 실행
`F9`	영역 지정 후 녹화 시작. 다시 누르면 정지 및 저장
`F10`	OBS 리플레이 버퍼 저장 및 클립 변환. 런처에서 사용 설정한 경우에 한함

- 단축키는 다른 프로그램 사용 중에도 동작
- 저장 위치 기본값은 `동영상\stream2clip`. 런처 하단의 **폴더 변경**으로 변경
- 화면 언어는 한국어, 日本語, English 지원. 변경 사항은 재시작 후 적용
- 설정은 자동 저장되어 다음 실행 시 복원. 비밀번호는 저장하지 않음

3. 화면 캡쳐 및 편집

- `F8`: 화면 정지 후 드래그한 영역을 편집기로 전달
- **전체 화면 캡쳐**: 모든 모니터 캡쳐
- **이미지 열어 편집**: PNG, JPG, WebP 파일 편집



편집기 화면

키	도구 및 동작
`V` `P` `H`	선택, 펜, 형광펜
`R` `A` `T`	네모, 화살표, 글자
`M` `E` `C`	모자이크, 투명, 자르기
`I`	스포이드(화면에서 색상 추출)
`[` `]`	굵기 감소, 증가
`Ctrl` + `Z`, `Ctrl` + `Y`	되돌리기, 다시 실행
`Ctrl` + `C`, `Ctrl` + `S`	복사, 저장
`Delete`	선택한 주석 삭제

- 투명 도구: 드래그 시 해당 영역 삭제, 클릭 시 클릭한 색상을 이미지 전체에서 투명 처리
- 자르기 도구: 원본을 유지한 상태로 남길 영역만 지정

4. 화면 녹화

1. 런처에서 출력 형식, 프레임, 크기 선택
2. `F9` 입력 후 녹화 영역 드래그. 즉시 녹화 시작
3. `F9` 재입력 시 정지 및 저장

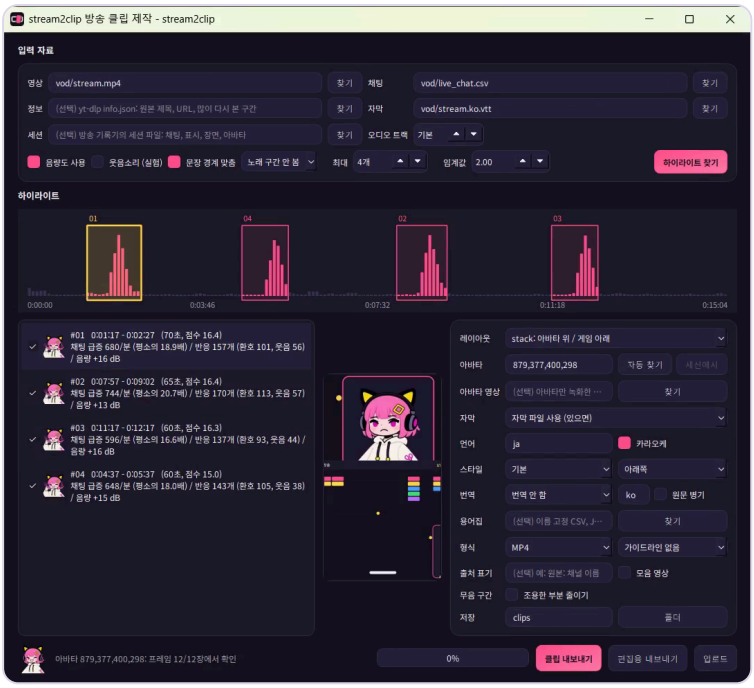
형식	특징
GIF	재생 호환성 높음. 투명도는 2단계
MP4	파일 용량 작음
WebP	화질 우수. 부드러운 투명 가장자리 지원
WebM	투명 영상 지원

4.1 투명 배경 녹화

- 지정한 색상(그린스크린 등)을 제거한 GIF, WebP, WebM 출력
- **스포이드**로 제거할 색상 지정, **허용 범위**로 유사 색상의 제거 범위 조정

5. 하이라이트 클립 제작

런처의 **하이라이트 찾기**에서 검출, 구간 조정, 내보내기를 수행.



하이라이트 창. 분홍 막대는 검출 구간, 노란 테두리는 선택한 구간

5.1 입력 자료

항목	내용
영상	다시보기 파일 또는 방송 주소
채팅	채팅 다시보기 파일. 방송 주소 입력 시 자동 수집
정보	선택 항목. 원본 제목과 주소를 설명란 작성에 사용
자막	선택 항목. SRT, WebVTT 파일

5.2 방송 주소로 시작

- 영상 칸에 방송 주소 입력 시 채팅 다시보기와 방송 정보를 자동 수집
- 내보내기 단계에서 선택한 구간의 영상만 다운로드
- 수집 자료는 저장 폴더의 `sources` 폴더에 방송별로 보관하며 재실행 시 재사용
- 미리보기는 해당 구간의 다운로드 완료 후 표시
- 음량, 웃음소리, 노래 구간 분석은 영상 파일 입력 시 사용 가능
- 지원 대상: YouTube, CHZZK, SOOP 다시보기

5.3 작업 절차

1. **하이라이트 찾기** 실행. 채팅 반응이 평소보다 높은 구간이 타임라인과 목록에 표시
2. 타임라인에서 구간 조정. 가운데 드래그는 이동, 가장자리 드래그는 시작과 끝 조정, 휠 또는 더블클릭은 확대

3. 내보낼 구간 선택 후 레이아웃과 형식 지정

4. 클립 내보내기 실행. 저장 폴더에 클립, 자막 파일(`.ass`), 설명란(`.txt`) 생성

영상, 채팅, 정보, 자막 파일은 창에 끌어다 놓아 입력 가능.

5.4 모음 영상과 편집 프로그램 연계

항목	내용
출처 표기	입력한 문구를 클립 왼쪽 위에 작게 표시. 예: 원본: 채널 이름
배경 음악, 효과음	배경 음악에 직접 내려받은 음원 파일 또는 음원 목록의 이름 입력. 클립 소리 아래에 깔리고 말하는 동안 더 줄어듦. 효과음은 내장 5종 또는 자동(채팅 반응에 맞춤)을 최고 순간, 시작, 끝에 배치. 음원의 출처와 라이선스는 `stream2clip sounds add` 로 기록하면 설명란 크레디트와 점검에 사용. MP4, WebM에서 사용
음원 목록	배경 음악 줄의 목록에서 내려받은 파일을 라이선스, 제목, 작가, 주소, 크레디트 문구와 함께 추가. 입력 중에 설명란 크레디트와 점검 결과 표시. 고른 곡은 배경 음악으로 사용
음량	-14 LUFS로 맞추기 선택 시 클립 소리를 YouTube 기준 음량에 맞춤. 한 번 측정한 후 일정하게 조절(MP4, WebM)
조용한 부분 줄이기	클립 안에서 0.7초 넘게 조용한 부분을 짧은 쉼으로 단축. 자막도 함께 이동. 음량 기준이므로 게시 전 결과 확인 필요
모음 영상	선택 시 내보낸 클립을 방송 순서대로 이어 붙인 `compilation.mp4` 와 챕터 목록 `compilation.txt` 를 함께 저장
편집용 내보내기	체크한 구간을 컷 목록 `highlights.edl` , 타임라인 마커 `markers.edl` , 타임스탬프 목록 `timestamps.txt` 로 저장. DaVinci Resolve, Premiere Pro 등에서 가져와 이어서 편집

5.5 영상 형태와 시작 부분

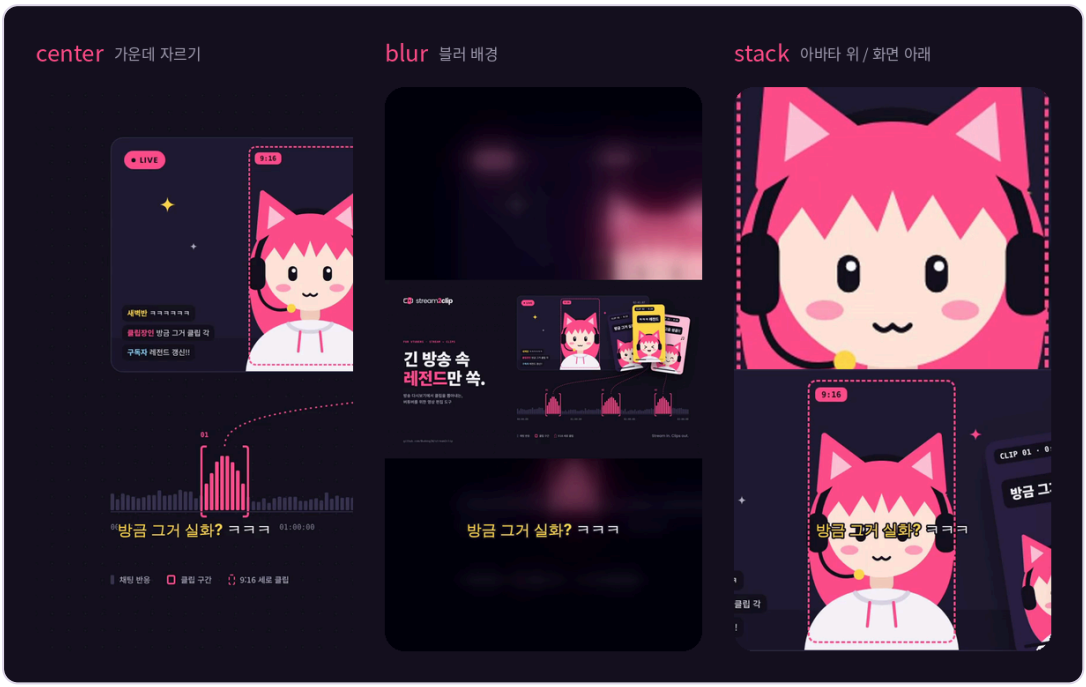
항목	내용
형태	Shorts(세로, 15~90초), 가로 클립(1~3분), 가로 모음(클립 6개를 이은 10분 전후 영상과 챕터) 중 선택. 레이아웃, 모음 영상, 상황 설명, 자막 방식(가로는 문장 자막, Shorts는 카라오케), 구간 수가 함께 바뀌며 하이라이트 찾기 전에 고르면 구간 길이도 적용. 실제 클립 채널은 대부분 가로 영상

항목	내용
상황 설명	클립 첫 3초 동안 위쪽에 방송 주제 한 줄 표시. 방송 제목의 괄호 속 게임 이름 등을 사용
최고 순간 미리 보기	클립 맨 앞에 최고 순간 2초를 먼저 보여 준 후 본편 시작(MP4, WebM)
썸네일	최고 순간 근처의 가장 선명한 장면에 제목 앞부분을 크게 넣은 1280x720 이미지 (`clip_01.jpg`). 아바타 위치를 지정한 경우 아바타 중심으로 확대
제목, 해시태그	클립마다 최고 순간의 대사, 탤런트 이름, 기획사 이름으로 제목 제안. 업로드 창의 제목 기본값. 기획사가 없는 한국어 제목은 `[이름] 대사` . 해시태그는 설명란 끝에 추가(한국어는 치지직, SOOP 플랫폼 태그와 키리누키를 더해 최대 5개). 과장된 표현은 점검에서 경고
합방 상대, 채널 링크	합방 상대(쉼표로 구분)는 제목 끝에 `(w.이름)` . 채널 링크는 설명란의 원본 채널 줄 아래에 추가. 연락 문구 선택 시 가이드라인이 따로 없는 스트리머용 문구 추가. 설명란은 제목과 같은 언어로 쓰고 방송 날짜 줄 포함
업로드 분류	업로드 창의 분류. 자동은 원본 방송이 게임 방송이면 게임, 그 밖은 인물 및 블로그. 직접 고르기도 가능
효과	최고 순간 한 곳에만 짧은 확대와 자막 단어 강조(절제), 채팅 반응 꼬리표 추가(예능). 소리가 커지는 순간이나 VTube Studio 핫키를 누른 순간이 뚜렷하지 않으면 넣지 않음. 움직임 줄이기 선택 시 확대 대신 단어를 더 크게 강조
재사용 콘텐츠 점검	자막, 상황 설명, 미리 보기 없이 잘라내기만 한 클립은 경고. YouTube 재사용 콘텐츠 정책상 게시 전에 직접 해설이나 자막 추가 필요

5.6 레이아웃

이름	구성
blur	방송 화면을 중앙에 배치하고 상하 여백을 흐림 처리한 배경으로 채움
center	화면 중앙부를 세로 화면에 맞춰 확대. 좌우 영역 잘림
fit	화면 전체가 보이도록 축소
stack	상단 아바타, 하단 게임 화면. **자동 찾기**로 아바타 위치 검출
pip	게임 화면을 전체에 채우고 구석에 아바타 삽입. **자동 찾기**로 아바타 위치 검출

이름	구성
아바타 영상	stack, pip에서 선택 사용. 아바타만 녹화한 영상을 지정하면 초록 배경을 제거하여 합성. 투명 배경 영상은 그대로 사용



레이아웃 예시. 브랜드 이미지를 방송 화면으로 사용하여 작성

5.7 검출 옵션

- 임계값: 낮을수록 검출 구간 증가, 높을수록 반응이 큰 구간만 검출
- 최대 개수: 검출할 구간 수의 상한
- 노래 구간: 노래와 겹치는 구간 표시 또는 제외. 노래 방송은 저작권 문제로 클립 제작이 제한되는 경우가 많음

5.8 방송 자료 일괄 다운로드

영상 전체가 필요한 경우 아래 명령으로 영상, 채팅, 방송 정보를 `vod` 폴더에 저장.

```
stream2clip.exe fetch "방송 주소" -o vod --video
```

6. 자막 및 번역

항목	내용
자막 파일	SRT, WebVTT 지원. YouTube 자동 자막 사용 가능
카라오케	발화 단어에 맞춰 색상 변경. 단어별 시각 정보가 있는 자막에 적용

항목	내용
스타일	모양 27종과 위치 7종 선택. 기본, 상자, 분홍 강조, 작게는 자막 언어에 맞는 설치된 글꼴, 나머지는 무료 글꼴(한국어 14종, 일본어 6종, 영어 4종, SIL Open Font License)을 처음 쓸 때 내려받아 사용(확인값 검사). 위치는 아래쪽, 아래 가장자리, 가운데, 위쪽과 Shorts의 방송 화면 바로 아래, 화면 위, 화면 바로 위. 크기는 한국 키리누키 기준(Shorts는 높이의 약 4.5%, 가로 클립은 약 5.5%)
문장 경계 맞춤	자막 파일이 있는 경우 구간의 시작과 끝을 자막 줄의 경계로 이동. 이동 한도 3초. 하이라이트 찾기 실행 시 적용
받아쓰기	자막이 없는 경우 음성에서 자막 생성(Whisper small, medium, large-v3-turbo, large-v3). large 계열은 GPU 권장. 파이썬 패키지에서 제공
번역	번역기 목록의 **번역기 파일 고르기**에서 `translators\openai_compatible.py` 선택 시 OpenAI 호환 주소(로컬 Ollama 등)로 번역
용어집	원문과 번역어 두 열로 구성한 CSV. 인명 등 고정 표기에 사용. 받아쓰기에는 이름 힌트로 쓰고, heard 열에 적은 잘못 들은 표기는 자막에서 이름으로 교정
원문 병기	원문과 번역문을 두 줄로 표시
자막 확인	내보낸 클립마다 읽기에 빠른 줄, 화면 폭을 넘는 줄, 짧게 지나가는 줄, 반복되는 줄, 받아쓰기가 무음에서 지어내기 쉬운 문구를 clips.json에 기록하고 개수 표시

- 번역 서비스는 내장되어 있지 않으며 사용자가 번역기를 연결
- 번역 품질은 연결한 모델에 따라 상이

7. OBS 리플레이 저장

OBS의 리플레이 버퍼에 보관된 최근 구간을 저장하여 클립으로 변환.

7.1 OBS 사전 설정

1. OBS 설정의 출력 메뉴에서 리플레이 버퍼 사용 설정. 신규 설치 시 기본값은 사용 안 함
2. OBS 도구 메뉴의 WebSocket 서버 설정에서 서버 사용 설정 및 비밀번호 확인
3. 방송 시작 시 OBS에서 리플레이 버퍼 시작

7.2 stream2clip 설정 및 사용

1. 런처의 **연결 설정**에 OBS 비밀번호 입력. 비밀번호는 저장하지 않으므로 실행 시마다 입력
2. **마지막** 항목에서 보존할 길이 지정 후 **리플레이 저장** 실행. 화면 녹화 설정의 형식과 크기로 저장

3. **F10** 선택 시 다른 프로그램 사용 중에도 `F10`으로 저장. 사용 중에는 다른 프로그램이 F10 입력을 받지 못함

8. 라이브 모드

방송 중 채팅 급증을 감지하여 OBS 리플레이 버퍼를 저장하고 해당 장면을 약 35초 길이의 클립으로 생성.

8.1 사전 조건

- OBS 리플레이 버퍼 실행 및 WebSocket 연결 설정 완료
- OBS 최대 리플레이 시간 60초 이상. 기본값 20초에서는 장면 앞부분이 누락

8.2 사용 절차

1. 런처의 라이브 모드 칸에 YouTube, Twitch, CHZZK, SOOP의 라이브 주소 입력
2. 민감도 선택. 높음은 작은 급증에도 저장, 낮음은 큰 급증에만 저장
3. **라이브 모드 시작** 실행. 채팅 급증 시 저장 폴더에 `live_날짜_시각` 형식의 클립 생성
4. **라이브 모드 중지**로 종료

8.3 동작 기준

- 5초 단위의 채팅량, 참여자 수, 반응 어휘, 후원 내역을 직전 15분과 비교
- 시작 후 2분간은 기준 수집만 수행하며 클립 간 최소 간격은 90초
- 급증 8초 전을 장면 시점으로 판단하여 앞 20초, 뒤 10초를 포함
- 스트리머 또는 모더레이터가 채팅에 `!clip` 입력 시 급증 여부와 무관하게 즉시 저장
- 지원 대상: YouTube 라이브 방송, Twitch, CHZZK, SOOP. CHZZK, SOOP의 성인 인증 및 비밀번호 방송은 제외

8.4 명령줄 실행

```
stream2clip.exe live "https://www.youtube.com/watch?v=VIDEO_ID" -o clips
```

옵션	기능
`--threshold`	발동 기준. 기본값 3.5. 높일수록 저장 횟수 감소
`--no-obs`	저장 없이 급증 발생만 출력
`--simulate`	종료된 방송의 채팅 파일로 발동 시점 확인
`--delay`	시청자 반응 지연 시간. 방송 송출자는 송출 지연만큼 증가시켜 사용

9. 방송 기록기

방송 중 채팅, 표시, OBS 장면, VTube Studio 아바타의 위치와 표정을 세션 파일로 기록하고 방송 후 클립 제작에 사용. 영상은 OBS가 녹화.

9.1 사용 절차

1. 시작 화면 클립 영역의 **방송 기록기** 실행
2. 본인 채널의 라이브 주소 입력, OBS와 VTube Studio 사용 여부 선택. VTube Studio는 설정에서 API를 켜고, 첫 연결 시 VTube Studio에 표시되는 플러그인 허용 창에서 허용
3. OBS에서 VTube Studio를 표시하는 소스 이름 입력. 클립 제작 시 아바타 영역 자동 지정에 사용
4. **기록 시작** 실행. 클립으로 남길 장면에서 **지금 표시** 또는 F11(사용 설정 시)
5. 방송 종료 후 **기록 중지**, **하이라이트 찾기**로 하이라이트 창에 세션 파일 입력

9.2 기록 내용과 용도

- 채팅과 표시: 채팅 다시보기를 받지 않고 하이라이트 검출. 표시한 장면은 항상 클립에 포함
- OBS 장면: 대기 화면, 엔딩 등 지정한 장면의 하이라이트 제외
- VTube Studio 외곽선과 OBS 배치: 세로 클립의 아바타 영역 지정
- 영상의 0초는 OBS 송출 시작 시점. 송출 기록이 없으면 녹화 시작 시점
- 시청자 투표: 시청자 3명이 30초 안에 `!clip` 입력 시 표시로 기록
- OBS 위젯: OBS 브라우저 소스로 `http://127.0.0.1:8790/widget` 추가 시 투표 진행과 표시를 방송 화면에 표시
- 채팅 작성자는 식별 불가능한 값으로만 기록하며 파일은 이 PC에만 저장

9.3 하이라이트 창 설정

- **세션** 칸에 세션 파일 입력. 채팅 칸에 넣거나 창에 끌어다 놓은 세션 파일도 세션 칸으로 이동. 채팅 칸이 비어 있으면 세션의 채팅과 표시로 검출
- **0초 기준**: 플랫폼 다시보기는 OBS 송출 시작, OBS 녹화 파일은 OBS 녹화 시작 선택. 영상이 기준보다 늦게 시작하면 그 차이를 초 단위로 입력
- **제외 장면**: 장면 이름의 일부를 쉼표로 구분해 입력. 절반 이상이 해당 장면에 속한 하이라이트 제외
- **아바타 움직임 (실험)**: 아바타 움직임이 큰 장면에 점수 추가
- 내보내기 설정 아바타 줄의 **세션에서**: 체크한 하이라이트 구간의 외곽선으로 아바타 영역 입력
- **오디오 트랙**: OBS에서 마이크를 별도 트랙으로 녹화한 경우 그 트랙 번호 선택. 자막 인식, 무음 구간, 음량 판정에 사용하며 클립의 소리는 변경 없음

9.4 명령줄 실행

```
stream2clip.exe session record --chat "https://chzzk.naver.com/live/CHANNEL_ID" --vts --vts-source "VTube Studio"
```

```
stream2clip.exe auto vod.mp4 -o clips --session session.jsonl --skip-scene 대기 --layout stack --avatar session
```

10. 명령줄 사용

앱의 모든 기능은 `stream2clip.exe`로 실행 가능. 다수의 방송을 일괄 처리할 때 사용.

명령	기능
`fetch`	방송 주소에서 방송 정보, 채팅, 영상, 자막 다운로드
`auto`	영상과 채팅 또는 방송 주소로 하이라이트 검출부터 클립 생성까지 수행
`highlights`, `clips`	검출과 내보내기를 개별 수행
`convert`	영상의 구간과 영역을 잘라 GIF, WebP, MP4로 변환
`record`, `capture`	화면 영역 녹화 및 캡쳐
`replay`, `live`	OBS 리플레이 저장, 라이브 모드
`check`	설명란을 기획사 가이드라인 항목과 대조

```
stream2clip.exe auto "방송 주소" --top 5 -o clips
```

```
stream2clip.exe auto vod\video.mp4 --chat vod\VIDEO_ID.live_chat.json --info vod\info.json -o clips
```

```
stream2clip.exe convert vod.mp4 --start 1:02:03 --end 1:02:18 --width 480 -o highlight.webp
```

전체 옵션은 각 명령에 `--help`를 붙여 확인. 추가 예시는 README 참조.

11. 문제 해결

증상	조치
프로그램 실행 차단	코드 서명 미적용으로 인한 차단. Smart App Control 사용 환경에서는 파이썬 패키지 설치 사용

증상	조치
새 버전 알림	시작 화면 상단에 표시. **내려받기 페이지 열기**로 새 설치 프로그램 다운로드, **알림 끄기**로 확인 중지. 확인은 하루 1회 이내이며 사용자 정보는 전송하지 않음
ffmpeg 미검출	앱 시작 화면의 **ffmpeg 내려받기** 버튼 사용. 명령줄에서는 `stream2clip setup --ffmpeg`. 직접 설치한 경우 **다시 확인** 버튼 또는 프로그램 재실행
검출 구간 없음	임계값을 낮춰 재실행. 채팅 파일이 해당 방송의 자료인지 확인
YouTube 로그인 요구	명령줄에서 `--cookies-from-browser chrome` 옵션으로 브라우저 쿠키 전달
리플레이 버퍼 미실행	OBS 설정의 출력 메뉴에서 리플레이 버퍼 사용 설정 후 버퍼 시작. 장면에 소스가 없으면 OBS가 버퍼를 시작하지 않음
OBS 연결 실패	WebSocket 서버 사용 여부, 비밀번호, 포트(기본값 4455) 확인
라이브 모드 클립의 앞부분 누락	OBS 최대 리플레이 시간을 60초 이상으로 변경
F8, F9 미동작	동일한 키를 사용하는 다른 프로그램 종료 후 재실행

기타 문제는 GitHub 이슈에 등록.

12. 저작권 및 가이드라인



- 클립은 원본 방송의 2차 창작물에 해당하며 원저작자와 소속 기획사의 가이드라인 적용 대상
- 내보내기 설정의 **가이드라인**에서 기획사 선택 시 원본 제목과 주소를 포함한 설명란 작성 및 점검 항목 확인
- 점검 항목: 멤버십 한정 방송 여부, 아카이브 공개 여부, 채널 등록 여부, AI 사용 표기
- 점검 결과는 참고 자료이며 법적 판단이 아님. 업로드 전 각 기획사와 방송인의 가이드라인 원문 확인 필요

Stream in. Clips out.