

jeongsunyong iss#65 특문제거

6de1617 on 17 Jan

2 contributors

167 lines (125 sloc) | 5.34 KB

Player

미디어에 대한 재생문제를 다루다 보면 미디어 종류에 따라서 재생할 수 있는 플레이어가 다르다는 것을 알 수 있습니다. VFX를 위한 미디어 재생 플레이어들을 알아보고 설치해 봅시다.

DJV View



오픈소스 플레이어입니다. Exr, Jpg 시퀀스를 빠르게 확인하기 좋습니다.

홈페이지 : <http://djv.sourceforge.net>

쉽게 설치하는 방법

OpenGL에러가 발생하면 <https://sourceforge.net/projects/djv/files/djv-stable/> 에서 너무 높지 않은 버전의 rpm파일을 다운로드 받습니다.

```
# yum install DJV-1.2.5-1.x86_64.rpm
```

프로그램 > 그래픽 > djv_view에 설치됩니다. 실제 설치경로는 /usr/local/DJV/bin 입니다.

터미널에서 실행하려면 .bashrc 파일에 아래 옵션을 추가해 줘야합니다.

```
export PATH=$PATH:/usr/local/djv-1.1.2-Linux-64/bin
export LD_LIBRARY_PATH=$LD_LIBRARY_PATH:/usr/local/djv-1.1.2-Linux-64/lib
```

OpenGL 체크

DJV 뷰어는 OpenGL3.3 이상이어야 작동합니다. 아래 명령어를 통해서 OpenGL 버전을 알 수 있습니다.

```
$ glxinfo | grep "OpenGL version"
```

컴파일

컴파일하고 설치할 때 djv view는 cmake 3.12 이상을 요구합니다.

Cmake를 먼저 설치합니다.

```
$ cd ~/app
$ git clone git://git.code.sf.net/p/djv/git-third-party djv-git-third-party
$ mkdir djv-git-third-party-Debug
$ mkdir djv-install-Debug
$ cd djv-git-third-party-Debug
$ ~/app/cmake3.13/bin/cmake ../djv-git-third-party \
  -DCMAKE_BUILD_TYPE=Debug \
  -DCMAKE_PREFIX_PATH=$HOME/app/djv-install-Debug \
  -DCMAKE_INSTALL_PREFIX=$HOME/app/djv-install-Debug
$ make
$ cd ~/app

$ export LD_LIBRARY_PATH=$LD_LIBRARY_PATH:$HOME/app/djv-install-Debug/lib

$ git clone git://git.code.sf.net/p/djv/git djv-git

$ mkdir djv-git-Debug
$ cd djv-git-Debug
$ ~/app/cmake3.13/bin/cmake ../djv-git -DCMAKE_BUILD_TYPE=Debug -DCMAKE_PREFIX_PATH=$HOME/app/djv-install-Debug
$ make
$ make install
```

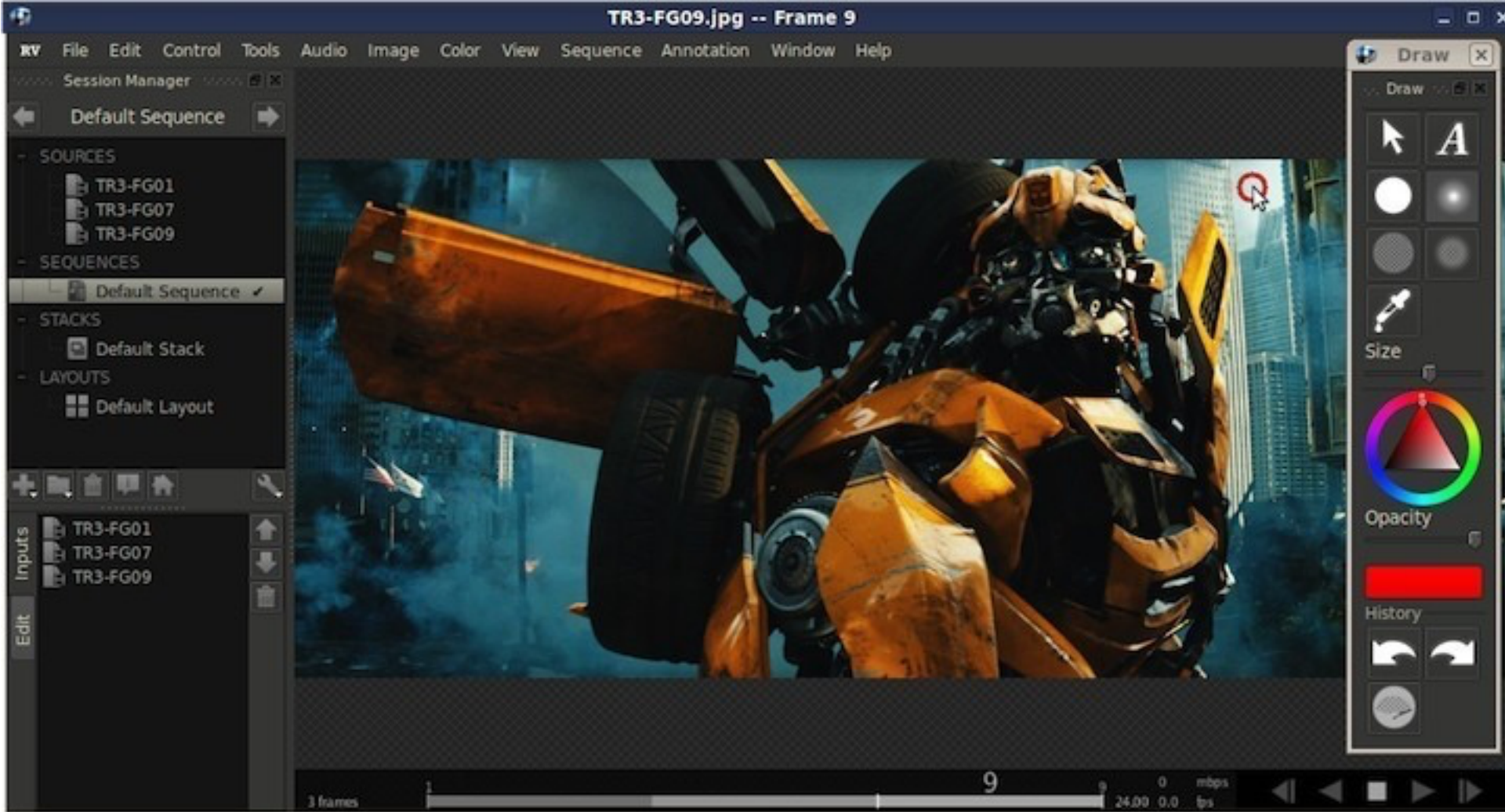
실행

djv 실행을 위해서 설치했던 lib 폴더를 LD_LIBRARY_PATH로 잡아준다.

```
$ export LD_LIBRARY_PATH=$LD_LIBRARY_PATH:$HOME/app/djv-install-Debug/lib
$ djv_view
```

그래픽카드 드라이버를 설치하고 OpenGL3.3 에러를 한번 관찰하기.

RV player



아마도 대부분의 VFX회사에서는 이 플레이어를 가장 많이 사용합니다. 샷건 파이프라인툴을 사용하면 이 플레이어를 무료로 사용할 수 있습니다. Python을 이용해서 RV 플레이어에 기능을 추가할 수 있습니다.

- <http://www.tweaksoftware.com>
- Download : <http://www.tweaksoftware.com/downloads/tweak-software-license-terms-and-conditions>

```
$ cd ~/app
$ wget http://www.tweaksoftware.com/static/downloads/rv-Linux-x86-64-7.3.0.tar.gz
$ tar -zxvf rv-Linux-x86-64-7.3.0.tar.gz
```

prores 코덱 활성화하기

주의! 이 방식을 사용하면 법적 책임이 발생할 수 있습니다. prores코덱을 재생하도록 배포시 RV 플레이어에서 기본적으로 지원하지 않는 이유는 아래 나열된 코덱은 독점권, 지적 재산권 하에서 보호되는 알고리즘이 사용되기 때문입니다. 이 코덱을 실제로 사용하기 위해서는 적절한 라이선스와 권리가지고 사용해야 하며 무단으로 사용하게 되면 그에 따른 책임이 발생할 수 있습니다. 만약 책임지기 싫다면 항목을 따라하지 말아주세요.

이 내용은 init.cpp 파일 상단 주석에도 표기되어 있습니다.

RV가 설치된 폴더로 이동합니다.

```
$ cd src/mio_ffmpeg
$ vim init.cpp
```

init.cpp 중 아래 항목에 대한 코드가 보입니다. 나열된 코드는 법적문제가 있는 코덱들입니다.

```
static const char* disallowedCodecsArray[] = {
    "mp3",
    "ac3",
    "hevc",
    "mpeg2video",
    "prores",
    "prores_aw",
    "prores_ks",
    "prores_lgpl",
    "svq1",
    "svq3",
    0 };
```

init.cpp내용중 위 코드를 아래처럼 바꾸어주세요.

```
static const char* disallowedCodecsArray[] = {
    0 };
```

저장후 빠져나와 아래 처럼 컴파일 합니다.

```
$ make
$ make install
```

컴파일시 아래와 같은 내용의 에러가 발생한다면..

```
warning: include path for stdlibc++ headers not found; pass '-std=libc++' on the command line to use
the libc++ standard library instead [-Wstdlibcxx-not-found]
```

Makefile 내용에서 아래항목을 다음 줄처럼 수정해주세요. 저는 macOS Mojave 에서 위에 해당하는 에러를 만났습니다.

Makefile에서 아래 줄을 찾아줍니다.

```
STD_CXXFLAGS = -std=gnu++98 -stdlib=libstdc++
```

위 내용을 아래 처럼 수정해줍니다.

```
STD_CXXFLAGS = -std=gnu++98 -stdlib=libc++
```

컴파일을 다시 해줍니다.

실습

- dlv_view를 환경변수에 추가해 줍니다.
- dlv_view를 app 폴더에 자동으로 설치 할 수 있도록 .sh 파일을 제작합니다.
- dlv_view 실행명령어도 alias로 추가해 줍니다.
- dlv_view alias가 출력되도록 echo 화면을 만들어 보시다.

Reference

- <https://www.uncleninja.com/uncategorized/2016-05-04/install-mpv-player-smplayer-centos-7/>