cineSpace™ v2.7 Documentation

Michael Anderson, Michael Chenery, Gina Chuang, Didier Elzinga,
Clive Haskins, Sam Keats & Jeremy Pollard

July 30, 2008

The cineSpace suite of software tools are created by Rising Sun Research (http://www.cinespace.rsrhq.com). For further information not contained in this document look in the cineSpace forums (http://cinespace.rsrhq.com/forum/).

cineSpace uses

- openSSL

Contents

Chapter 1

The cineSpace suite

1.1 What is cineSpace?

cineSpace is a digital colour management system (CMS) designed by Rising Sun Research (RSR). Using a hardware probe, cineSpace can accurately profile displays throughout a facility and then, using a simple application, the artist is able to

- match the display to any other display in the building
- match the display to any desired film output stock
- match the display to any video standard

Simple to use and built with film users in mind, cineSpace is designed to give you true "what you see is what you get" functionality on the computer screen without having to print to film – potentially saving thousands of dollars on every project.

Built on the basis of extensive experience in digital film, cineSpace draws upon this esoteric knowledge to make accurate calibration available to everyone, simplifying the colour pipeline process. Running on Linux, Windows 2000/XP and Mac OS X with licenses that are both cross-platform and floating, the cineSpace colour management suite provides world class tools to artists however they need them.

1.2 Why is cineSpace needed?

Increasingly, companies working with digital film are experiencing the need to establish a consistent representation of their images across all displays in their facilities. This means having confidence that what an artist is seeing at their workstation is exactly the same as what a production manager sees on their monitor when reviewing the images. Furthermore, companies require that their monitors are calibrated to the optimum settings for viewing the types of images they are commonly working with so the best possible use is made of the hardware investment.

In addition to having matched monitors, artists require an accurate representation of how their images will look when sent to the final output, whether it be film, HD, Digital Cinema or some other media. They may also wish to compare between possible film output stocks to determine which provides the best colour response for the current project.

In the past, the only way to check the final output result was through the expensive film-out process. A colour grading expert would have spent hours making adjustments to the frames to obtain approximately the right colour balance before transferring them to film, where the final result could be checked. In even the best cases, this process may have been repeated several times, and in most cases it would take 15 to 20 attempts to get everything right. The result was many hours and thousands of dollars wasted.

With the advent of cineSpace, all artists involved in a project are able to have a consistent view of the images they are working with, and that view can provide an accurate representation of the final output. No longer is it necessary to produce numerous film-outs to check that the colours are right – the correct colours are displayed right from the start. On average, the number of film-outs per project can be reduced to only one or two, representing a significant cost savings. Many hours are saved by eliminating the need to go back and forth between artists, producers and colour experts before obtaining the desired result. There is no discrepancy between what each artist views at their workstation and what the producer sees on the preview screen.

cineSpace also makes a reality the concept of digital approvals, where VFX supervisors, producers and other key team members can sign off on shots viewed on a calibrated digital projection system. Once again, this streamlines the approval process and can eliminate the cost of having to go our to film time and time again.

1.3 The cineSpace suite

The cineSpace colour management suite consists of a number of applications that work together to deliver accurate calibration across multiple systems. By providing a combination of stand alone tools, plug-ins and other application support, cineSpace can be used very effectively in virtually any colour pipeline and with all display types, including CRTs, LCDs and digital projectors.

The components of the cineSpace suite are

cineProfiler An application that utilises a hardware probe (such as a X-Rite i1 Display or Pro, LaCie Blue Eye 2 or Konica Minolta CS-200 or the Hubble) to *profile* the response of your display (CRT, LCD or digital projector). These profiles are stored as XML files that are used by the applications listed below to calibrate your display.

equalEyes A stand-alone tool that uses a monitor profile created by cineProfiler to match your monitor to another monitor, a video standard (like PAL, NTSC or HD) or to film. It works by modifying the gamma look-up table (LUT) of your graphics card.

cinePlugins A suite of plug-ins that provide the functionality of both equalEyes and cineCube within the parent application and also support multiple accelerated modes for dealing with 3D LUTs:

cineShake (*Linux & OS X only*) The plug-in to Shake, which supports cineSpace as a node within a tree or as a viewer LUT. Currently cineShake supports Shake versions 4.0 and 4.1.

cineFusion (*Windows only*) The plug-in to Digital Fusion, which supports cineSpace as a node within Fusion. Currently cineFusion supports Fusion version 5.0 and 5.2.

cineNuke (*Windows, Linux & OS X*) The plug-in to Nuke, which supports cineSpace as a node within Nuke. Currently cineNuke supports Nuke versions 4.7, 4.8 and 5.0.

cineFilmMaster (*Windows*) The plug-in to Digital Vision (Nucoda) FilmMaster, which supports cineSpace as a colour management module within Film Master. Currently cineFilmMaster supports FilmMaster versions 3.0.2 and 3.5.

cineCube Visual (*Windows & OS X only*) A graphical interface for generating 3D colour 'cubes' (LUTs) that can be used by real-time playback products from a variety of vendors, including ...

- ASSIMILATE (Scratch),
- Autodesk (Lustre, Flame, Inferno & others),
- CHROME Imaging (Matrix Compositing),
- IRIDAS (the SpeedGrade and FrameCycler lines of products),
- Quantel (iQ & eQ),
- S.G.O. (Mistika),
- Apple (Color),
- The Pixel Farm (PFPlay, PFClean & PFClip)
- and many others.

These products use the 'cubes' to provide a cineSpace calibrated display within their own application.

cineCube A command line tool for generating 3D colour 'cubes' (LUTs) that can be used by real-time playback products from a variety of vendors, as per cineCube Visual. These products use the 'cubes' to provide a cineSpace calibrated display within their own application.

probeServer A tool that allows a hardware probe to be installed and running on one computer and then used for profiling another computer's monitor.

> It is designed for using a single probe with a laptop to profile all monitors in a facility, or to profile a system where the workstation and display(s) are some distance apart.

Each of these individual applications that comprise the cineSpace suite will be covered in detail in the chapters to follow.

Chapter 2

Setting up cineSpace

2.1 Preparing ahead

The cineSpace colour management suite has been designed to be as flexible as possible, supporting a wide range of platforms, applications, displays and hardware probes. cineSpace has very low resource demands and will run on nearly all computers, requiring only a standard workstation or laptop with a USB port[1] and a modern graphics card. Before you begin, however, you should read through this chapter to ensure that you understand what you need in order to get up and running with cineSpace.

The main things you will need before starting out with cineSpace are:

- Computer workstation and display[2]
- cineSpace software package for your operating system
- Hardware probe
- cineSpace license

The supported platforms, probes and cineSpace licensing mechanisms are explained in further detail below.

2.1.1 Supported platforms

cineSpace will run on most common platforms (operating systems), including Windows, Mac OS X and Linux, making it an ideal solution for mixed platform environments. The applications and interfaces are consistent across all platforms to simplify the user experience, although the installation process is slightly different in each case.

[1]Systems without a USB port can also be accommodated using a special utility packaged with the software.

[2]The display can be a CRT or LCD monitor, or a digital projector.

The following table details the specific platforms that it has been designed for and tested on:

Operating system	*Designed for*	*Tested on*
Windows	Windows XP SP2 32 bit Windows 2000	Windows XP SP2 32 bit Windows 2000
Mac OS X	OS X 10.5 (Leopard) OS X 10.4 (Tiger) OS X 10.3 (Panther)	OS X 10.4 (Leopard) OS X 10.4 (Tiger) OS X 10.3 (Panther)
Linux	Fedora Core 1, 2, 3 & 4 (32bit) Red Hat Enterprise Linux 3 (32bit) gcc 3.2.2, 3.3.1	Fedora Core 2, 3 & 4 32 bit gcc 3.2.2, 3.3.1

The cineSpace suite is known to run on a number of Linux distributions not mentioned in the table above, including Debian, Red Hat Enterprise Linux and Suse, and tested with both the GNOME and KDE graphical interfaces. Since, however, there can be differences in the way that they are configured for a particular site, please contact the RSR team to discuss any potential issues. Currently, we only support the 32 bit architecture on Linux. We have had success with some 64 bit Linux systems, but at the current time we will not be supporting them.

Obtaining the cineSpace software packages

The cineSpace software packages for all platforms are distributed via FTP server, which is located at the following web address:

`ftp://ftp.rsrhq.com`

You will need a username and password to access the FTP server – please contact RSR to obtain these details if you do not already have them.

2.1.2 Hardware probes

cineSpace uses a *hardware probe* to capture accurate colour measurements from your displays. A probe is a device – often categorised as either a *colorimeter* or a *spectrophotometer* – that is sensitive to the wavelengths of light that make up colours, enabling those colours to be represented as numerical values [3]. These values are then used within the cineSpace software suite to determine the adjustments required to achieve accurate calibration.

cineSpace does not ship with a hardware probe and you will need to source a device to use with the software. The type of probe that you need will depend on what display types you are wanting to calibrate. Although most probes will work for CRT monitors, some older models are not suitable for LCD panels. If

[3] The colour measurements are described by tristimulus values defined in the CIE XYZ colour space.

you wish to calibrate digital projectors, you will need a more specialised – and more expensive – device that can take reflective readings from a screen.

We currently support the following hardware probes:

Manufacturer	*Model*	*Comments*
X-Rite *Supported Platforms: Windows XP (32bit), OS X & Linux (32bit)*	Eye-One Display2	Accurate and affordable, our recommended monitor probe.
	Eye-One Display LT	Same hardware as the Display2, but sold in a different bundle.
	Eye-One Display	Replaced by the improved Display2, but still supported by the manufacturer.
	Eye-One Pro	Highly versatile, can profile reflected-light displays (e.g. projected images) in addition to monitors.
X-Rite *Supported Platforms: Windows XP (32bit)*	Hubble	Improved repeatability over the Eye-One Pro and greater reduced noise in the blacks.
LaCie *Supported Platforms: Windows XP (32bit), OS X & Linux (32bit)*	blue eye 2	Same hardware as the Eye-One Display2 but re-badged with the LaCie branding.
Konica Minolta *Supported Platforms: Windows XP (32bit)*	CS-200	High-end spectrally based colorimeter.

There are three probes supported for doing Output Independent profiling (OIP):

- X-Rite **Eye-One Pro** and **Hubble**
- Konica Minolta **CS-200**

For information on OIP see section 5.8.

Probes can be obtained through local resellers or ordered online at a number of different web sites. A good starting point is the X-Rite online store, which contains information and details for the Eye-One series of devices:

`http://www.store.xrite.com`

One other device that is still used in some facilities, but no longer supported by the manufacturer, is the X-Rite DTP-92Q (available both in a serial and a USB version). Whilst this probe will generally work with cineSpace, it is not suitable for LCD monitors and is not listed as a supported device. Please consult the RSR team if you intend to use the DTP-92Q with cineSpace.

2.1.3 How cineSpace is licensed

cineSpace uses a standard *Reprise License Manager* (RLM) license architecture (v3.0 build:4 or higher). There are two primary ways in which the licensing may be handled:

1. Using a *node-locked* license file on each workstation running cineSpace applications; or
2. Using a *license server* that handles the license management for all workstations running cineSpace applications, allowing *floating* licenses to be used.

Node-locked licenses

Node-locked licenses bind the cineSpace software to a single workstation. They provide a way to run cineSpace where a machine will not always be connected to a network on which a license server resides, for example a laptop or stand-alone review station. This method of licensing is simpler to set up, since there is no need to install an RLM license server in addition to the cineSpace software. The limitation, however, is that the license cannot be shared (*floated*) across multiple machines, making it a less flexible option should your needs change.

License server

Floating licenses that run on the RLM server allow multiple users, across multiple workstations and platforms, to share access to the cineSpace software suite. The RLM server runs in the background on a computer on your network. Usage is restricted only by the number of concurrent seats allowed by the license file, making it the ideal option for facilities with a large number of users or changing production needs. Setting up a license server does, however, require an extra step during the cineSpace installation process.

For full details on license server concepts and running multiple applications under this architecture, please consult the RLM End User documentation:

`http://cinespace.rsrhq.com/docs/RLM`

2.2 The installation procedure

Whilst the installation procedure for cineSpace is fairly simple, there are a number of steps that you should follow to ensure that everything goes smoothly. This section outlines the major aspects of the procedure, with the subsequent sections providing platform-specific details.

There are two types of installation

- Quick - great for getting started quickly with a node-locked license.

- Advanced - for users who want to float licenses across multiple computers and operating systems. Involves installing a license server.

This section gives a brief overview of the steps involved in installing cineSpace.

Not all steps are required for all installations; see the following sections on quick (2.2.1) and advanced (2.2.2) installs to find out which steps are needed to complete your cineSpace installation.

For details on how to actually do each step see the Platform specific sections.

- Windows - section 2.3
- OS X - section 2.4
- Linux - section 2.5

Before you begin, however, you will need to ensure that you have the cineSpace software and RLM license server (if required) for the platforms you are running. The required software packages may be downloaded from the RSR FTP server:

`ftp://ftp.rsrhq.com`

You will need login details (username and password) to access the FTP server. Please contact Rising Sun Research if you do not already have the required login information.

2.2.1 Quick Install (node-locked licenses)

Users who are using a node-locked license do not require a license server and so can do a quick install. Please follow the following steps to do a "quick" cineSpace installation.

1. Obtain a license. Gather all the needed information and request a license from Rising Sun Research . See Section 2.2.3

2. Install cineSpace. See Section 2.2.6

3. Copy license file. Place the license file in the "licenses" directory inside the cineSpace applications directory. See Section 2.2.8

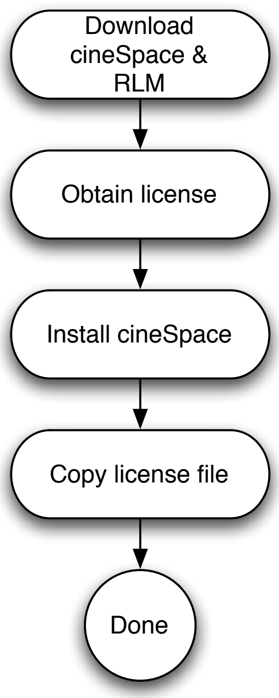


Figure 2.1: cineSpace quick install

2.2.2 Advanced Install (with license server)

For users who wish to use floating licenses an advanced install is required. Please follow the following steps to do an “advanced” cineSpace installation.

1. Obtain a license. Gather all the needed information and request a license from Rising Sun Research. See Section 2.2.3

2. Install the license server. See Section 2.2.4

3. Set license server to auto-start. See Section 2.2.5

4. Install cineSpace. See Section 2.2.6

5. Set environment variables. See Section 2.2.7

6. Edit cineSpace configuration file (rsr.conf). See Section 2.2.9

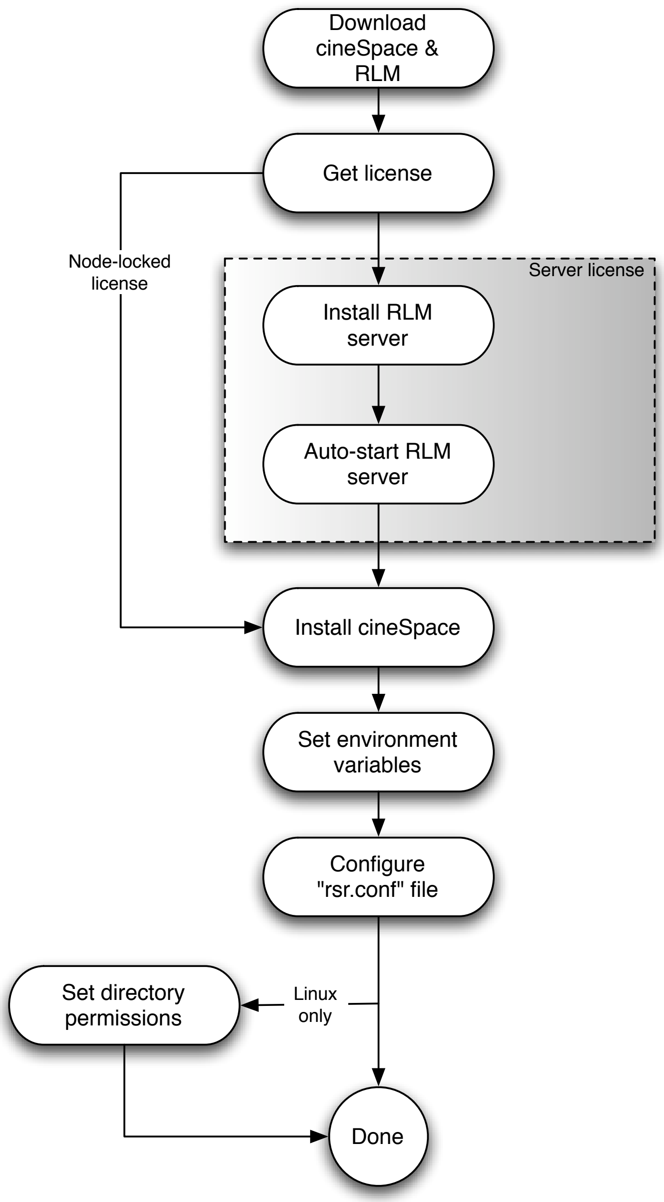


Figure 2.2: cineSpace advanced install

2.2.3 Obtaining a license

When requesting a license for cineSpace, you will need to provide the RSR team with some system information so that the correct license file can be generated:

- Your computer *hostname*
- Your computer *host ID*; and also
- Whether you require a *floating* or *node-locked* license

The *hostname* is the unique name which identifies your computer on your network and will be, in most cases, the same as the computer name. The *host ID* is a number that is unique to a specific computer, used to identify that machine for licensing purposes. In most cases, the host ID for your system will be the same as the *MAC address*[4] of your primary ethernet device (`en0`). The method for retrieving this information is dependent upon your operating system and the steps involved are described in the platform-specific sections that follow.

- If you will be using a node-locked license then the hostname and host ID should be obtained from all machines on which the cineSpace software will be running.

- If you are planning to use a license server, you should provide the hostname and host ID of the machine on which the RLM server will be installed - this does not necessarily have to be a computer that will be running cineSpace software.

2.2.4 Installing the RLM license server

Most cineSpace installations make use of the RLM server in order to utilise floating licenses across multiple machines (and platforms). If you will be using this sort of configuration, it is best to set up the license server prior to installing the cineSpace software package.

For details on how to set up the RLM server please see the relevant section in the platform specific notes; Windows (Sec.2.3.2), OS X (Sec.2.4.2), Linux (Sec.2.5.2).

For node-locked license configurations there is no need to install the RLM server.

2.2.5 Auto-starting your RLM server

In order to properly provide licenses, the RLM server needs to be running at all times. We recommend that you configure the RLM server to automatically start running when your computer boots up. This will ensure that if your license server machine is restarted, the cineSpace license server will also restart.

[4] The Media Access Control (MAC) address is a unique identifier for most forms of networking equipment found in your computer.

For details on how to configure the RLM server to start automatically see the relevant section in the platform specific notes; Windows (Sec.2.3.3), OS X (Sec.2.4.3), Linux (Sec.2.5.3).

2.2.6 Installing cineSpace

The main cineSpace installer package will install the following components of the cineSpace suite:

- cineProfiler
- equalEyes
- cineCube Visual
- cineCube
- probeServer
- profileTools
- Configuration files and colour space profiles

In most cases, you will be able to simply run the installer package for your platform, which will place the required software components into their correct locations. For details on how to run the installers see the relevant section in the platform specific notes; Windows (Sec.2.3.4), OS X (Sec.2.4.4), Linux (Sec.2.5.4).

For information about installing each of the cinePlugins, please refer to Chapter 3 - Installing the cinePlugins.

Note: You must install the cineSpace package before installing any of the plug-ins.

2.2.7 Setting Environment Variables

Environment variables are dynamic, system-wide values that can affect the way that running processes behave. To complete an advanced cineSpace installation, you need to set an environment variable (`$RSR_APP_DIR`) so that the cineSpace application suite knows where to find its configuration files. The way in which you set this environment variable depends upon the platform that you are using.

For details on how to set environment variables please see the relevant section in the platform specific notes; Windows (Sec.2.3.5), OS X (Sec.2.4.5), Linux (Sec.2.5.5).

2.2.8 Copy license file

For node-locked licenses place the license file in the licenses directory within the cineSpace application directory. cineSpace will look in this directory for node-locked licenses.

For floating licenses placing a copy of the license file in the location above will tell cineSpace where to look for the license server.

2.2.9 The rsr.conf file

You can also specify the location of the license server in the rsr.conf. This is done by editing the rsr.conf configuration file within the cineSpace distribution. In addition to the license setting, the rsr.conf file contains default values for many of the cineSpace suite settings, though these values can be overridden using environment variables.This allows you to use a single configuration file for your network (to simplify installation at large facilities) or individual settings for each computer. Where each setting (and description) is preceded by '`#`', it will be treated as a comment and ignored. To use the setting in your configuration, remove the preceding '`#`' and enter the desired parameter value.

You can also set which rsr.conf file cineSpace uses by setting the environment variable `$RSR_CONF_FILE` to point to the location of the configuration file. This means you can save a rsr.conf file for each project you may be working on.

For details on how to set environment variables please see the relevant section in the platform specific notes; Windows (Sec.2.3.7), OS X (Sec.2.4.7), Linux (Sec.2.5.7).

2.3 Windows (Platform Notes)

2.3.1 Obtaining a license

To find your hostname and hostid, you will need to open a *command prompt* and enter some commands that will return information about your system. To do this, go to Start → Programs → Accessories → Command Prompt.

1. To find the hostname of your computer, type the command:
   `hostname`

2. To find the hostid of your computer take the Physical Address of the first Ethernet adapter listed from the following command:
   `ipconfig /all`

Please send an email to cinespace-support@rsrhq.com with all of the information generated from these commands.

2.3.2 Installing the RLM license server

For node-locked license configurations there is no need to install the RLM server.

For floating license configurations, the RLM server needs to be installed and started on the computer that you have nominated as the license server for your network. You can follow these steps to install and run the RLM server:

1. Unzip `RLM_v3.0BL4_win32.zip` to `C:\`. This creates the folders and files required for RLM to run.
2. Navigate to `C:\rlm\rsun\` and copy your license file into this folder.
3. Open a command prompt and navigate to `C:\rlm\rsun\` and run RLM with the full path to the license file and log file. The log file will be created when rlm is run.

   ```
   rlm.exe -c C:\rlm\rsun\rsr_license.lic
           -dlog C:\rlm\rsun\rsr_log.txt
   ```

   You should get a message indicating that the server has started.
   Also included in the message should be a line indicating that the RLM web service has started. For example:

   ```
   09/11 14:07 (rlm) Web server starting on port 9000
   ```

   To check on the server status open a web browser and enter the hostname and port number indicated. For example:

   ```
   http://localhost:9000
   ```

   This will open the Reprise License Server Administration web page.

You can then set the `RSR_LICENSE` variable as described below to complete the license configuration.

2.3.3 Auto-starting your RLM server

On Microsoft Windows servers, you may want to install and run the RLM server as a Windows service process. A service process can start automatically at boot time and remain running as long as the system is up, regardless of user logins and logouts.

Installing RLM as a service is done in a command window. Once installed as a service, it remains installed until it is explicitly deleted as a service. Installing RLM as a service does not start RLM; services are started via the Windows Services control panel, and at boot time. RLM is installed using the rlm.exe program itself, with special arguments:

```
rlm.exe -install_service -dlog [+]logfile
       [-service_name sname] <rlm runtime args>
```

where:

- *logfile* is the pathname for the server debug log. This parameter is required. If preceded by the '+' character, the logfile will be appended, rather than created.
- *sname* is an optional name for the installed service. If not specified, *sname* defaults to "rlm". If sname contains embedded whitespace, it must be enclosed in double quotes.
- <rlm runtime args> are any other command line arguments to be passed to rlm when it is started.

Example:

```
rlm.exe -install_service -service_name rlm-rsr
        -dlog c:\rlm\rsun\rsr_log.txt
        -c c:\rlm\rsun\rsr_license.lic
```

This installs RLM as a service under the name "rlm-rsr". When started via the Services control panel or at boot time, rlm.exe will be passed the " `-c c:\rlm\rsun\rsr_license.lic`" args, and it will write its debuglog information to the file `c:\rlm\rsun\rsr_log.txt`

Installed RLM services are also deleted with the rlm.exe program. Services must be stopped via the service control panel before they can be deleted. Note that deleting a service deletes it from the Windows service database; it does not delete the rlm.exe or associated license file(s):

```
rlm.exe -delete_service [-service_name sname]
```

where:

- *sname* is an optional name for the installed service. If not specified, service_name defaults to "rlm". If service_name contains embedded whitespace, it must be enclosed in double quotes.

Notes:

- You should use the -c <license file> command line argument with RLM when installed as a service.
- Because the Service Controller on Windows invokes services under a special user account in a special default directory, it is necessary to use full paths:
  - for the -c <license file> argument on the rlm command line
  - for the -dlog debug_log argument on the command line

2.3.4 Installing cineSpace

Running the installer (.exe file) will install the cineSpace suite into a user specified directory and place shortcut icons in the appropriate places. The installer will also setup the required environment (registry) variables for you.

2.3.5 Setting Environment Variables

The cineSpace installation package for Windows will set your environment variables automatically. If you wish to manually perform this task, please follow these steps:

1. Open the *System* Control Panel in Windows. This can be done by selecting *Start → Control Panel → System.*
2. Select the *Advanced* tab.
3. Click on the *Environment Variables* button.
4. Inspect the *System Variables* list and see whether *RSR_APP_DIR* exists. If it does not, then:
   - Under *System Variables* click the *New* button
   - For the Variable Name, enter *RSR_APP_DIR*
   - For the Variable Value, enter the location of the cineSpace installation, e.g.
     `C:/Program Files/RisingSunResearch/cineSpace`
   - Click *OK*, and then *OK* again
5. Click *OK* to complete the process.

2.3.6 Copy license file

For node-locked licenses place the license file in the licenses directory within the cineSpace application directory. The cineSpace will look in this directory for node-locked licenses. For example:

```
C:\Program Files\RisingSunResearch\cineSpace\licenses\rsr_license.lic
```

For floating licenses placing a copy of the license file in the location above will tell cineSpace where to look for the license server.

2.3.7 The rsr.conf file

You can also specify the location of the license server in the rsr.conf. This is useful when administering large network based installations of cineSpace. You will find the rsr.conf file within the base directory of your cineSpace installation. You can also locate it via *Start → Program Files → cineSpace →* rsr.conf file. Open this file and scroll down to *Section 1: License Settings.* Un-comment the line (remove the preceding '#') and enter the port and path to your license server, e.g. `RSR_LICENSE = 2764@rlm_server.company.com`

When entering path names, an absolute path works best since relative paths are relative to the directory that the application was run from, not the directory where the application is located.

Please remember to update the correct license file. I.E. If `$RSR_CONF_FILE` has been set to point to a particular license file then it is that file that requires updating.

2.4 Mac OSX (Platform Notes)

2.4.1 Obtaining a license

To find your hostname and hostid, you will need to open a *Terminal* window and enter some commands that will return information about your system. To do this, open Finder and go to Applications → Utilities → Terminal.

1. To find the hostname of your computer, type the command:
   `hostname`

2. To find the hostid of your computer

   - Take the string to the right of the HWadd, and remove the colons from the output of the following command:
     `/sbin/ifconfig en0 | grep -i ether`

     or

   - Take the string under the Address, and remove the colons from the output of the following command:
     `netstat -I en0`

     or

   - Open the System Profiler application in `/Applications/Utilities`. Look in the Network overview of the System Profile, selecting *Built-in Ethernet* to find your system's MAC address.

Please send an email to cinespace-support@rsrhq.com with all of the information generated from these commands.

2.4.2 Installing the RLM license server

For node-locked license configurations there is no need to install the RLM server.

For floating license configurations, the RLM server needs to be installed and started on the computer that you have nominated as the license server for your network. You can follow these steps to install and run RLM:

1. Unpack the rsunServer archive for your platform and place the files in an appropriate location (e.g. `/usr/local/`). This creates the folders and files required for RLM to run.

2. Navigate to `/usr/local/rlm/rsun/` and copy your license file into this folder.

3. Open a Terminal window (`Applications` → `Utilities` → `Terminal`). From inside the Terminal navigate to `/usr/local/rlm/rsun/` and run RLM with the full path to the license file and log file. The log file will be created when rlm is run.

   ```
   rlm -c /usr/local/rlm/rsun/rsr_license.lic
       -dlog /usr/local/rlm/rsun/rsun_log.txt
   ```

   You should get a message indicating that the server has started. Also included in the message should be a line indicating that the RLM web service has started. For example:

   ```
   09/11 14:07 (rlm) Web server starting on port 9000
   ```

   To check on the server status open a web browser and enter the hostname and port number indicated. For example:

   ```
   http://localhost:9000
   ```

   This will open the Reprise License Server Administration web page.

You can then set the `RSR_LICENSE` variable as described below to complete the license configuration.

2.4.3 Auto-starting your RLM server

1. Open a Terminal Window, and become root
   `sudo su -`

2. Create a rlm folder in the StartupItems Directory
   `mkdir -p /Library/StartupItems/rlm`

3. In this folder, create a new script file called `rlm`. Copy the following into the script.

```
    #!/bin/sh
    . /etc/rc.common

    RLM_PATH=/usr/local/rlm
    LICENSEFILE=rsr_license.lic
    LOGFILE=rsun_log.txt

    run=(${RLM_PATH}/rsun/rlm -c ${RLM_PATH}/rsun/${LICENSEFILE} \
        -dlog ${RLM_PATH}/rsun/${LOGFILE})
    stop=(${RLM_PATH}/rsun/rlmdown -c ${RLM_PATH}/rsun/licenses/${LICENSEFILE} -q)


StartService ()
{
  ConsoleMessage "Starting rlm Server"
  if ! ps axco command | grep -q "rlm"
  then
    ${run[@]} &
  fi
}

StopService ()
{
  ConsoleMessage "Stopping rlm Server"
  ${stop[@]} > /dev/null &
}

RestartService ()
{
  ConsoleMessage "Restarting rlm Server"
  ${stop[@]} > /dev/null
  ${run[@]} &
}

RunService "$1"
```

4. Change the following variables in the script
   - `RLM_PATH` to be the location of your RLM base directory.
   - `LICENSEFILE` is the name of your license file
   - `LOGFILE` is the name of your log file
5. Make this script executable
   `chmod 755 /Library/StartupItems/rlm/rlm`
6. In this same folder, create a file called *StartupParameters.plist*. Copy the following in to this file.

```
{
  Description       = "cineSpace License Server";
  Provides          = ("rlm");
  Requires          = ("Network");
  Uses              = ("Network");
  OrderPreference   = "None";
}
```

The cineSpace RLM Server will start automatically next time you reboot. If you wish to test this setup, you can run the startup script with in the Terminal window:
`/Library/StartupItems/rlm/rlm start`

2.4.4 Installing cineSpace

The Mac OS X version of cineSpace comes as a disk image containing an installer. Open the disk image (.dmg file) and run the cineSpace installer (.pkg file) – it will be the only file in the disk image.

Running the installer and following the on-screen directions will install the cineSpace programs into the directory of your choice. The default directory is `/Applications/cineSpace`.

2.4.5 Setting Environment Variables

If you prefer to use a GUI (Graphical User Interface) to setup your environment variables, you may try using this program:
http://www.apple.com/downloads/macosx/development_tools/plisteditpro.html

To make environment variables available to Mac OS X GUI applications, these variables must be defined per user in their `~/.MacOSX/environment.plist`. Sometimes it is necessary to create the file named `~/.MacOSX/environment.plist` or, if the file already exists, you should just add the key/string lines to the 'dict' section.

The environment.plist is a hidden file that can only be seen from the command line in a "Terminal" window. To create/edit the environment.plist file please follow the following procedure:

1. Open a Terminal window (Applications → Utilities → Terminal)

2. Check to see if the directory .MacOSX exists. To do this copy and paste the following line to the Terminal window and press enter ...

   ls -la

   This will list all the folders contained in your home directory. If you can see a folder named .MacOSX then it already exists. If not then we need to create it. To do this copy and paste the following line to the Terminal window and press enter ...

   mkdir .MacOSX

3. Next we need to open / create the environment.plist using TextEdit. To do this copy and paste the following line to the Terminal window and press enter ...

   **/Applications/TextEdit.app/Contents/MacOS/TextEdit **
   ./.MacOSX/environment.plist

   This will either open the environment.plist file if it exists or start a new file if it doesn't exist.

4. Copy and paste the following into TextEdit ...

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE plist PUBLIC
"-//Apple Computer//DTD PLIST 1.0//EN"
"http://www.apple.com/DTDs/PropertyList-1.0.dtd">
<plist version="1.0">
<dict>
<key>RSR_APP_DIR</key>
<string>/Applications/cineSpace</string>
</dict>
</plist>
```

   If the file already exists just add the key/string lines to the 'dict' section.

5. Close TextEdit. TextEdit will ask whether you want to save the file. Choose 'Save'. TextEdit will then save the file as "environment.plist" in your .MacOSX directory and exit.

6. Close the Terminal window.

Now your environment.plist file should be set up correctly.

2.4.6 Copy license file

For node-locked licenses place the license file in the licenses directory within the cineSpace application directory. The cineSpace will look in this directory for node-locked licenses. For example:

```
/Applications/cineSpace/licenses/rsr_license.lic
```

For floating licenses placing a copy of the license file in the location above will tell cineSpace where to look for the license server.

2.4.7 The rsr.conf file

You can also specify the location of the license server in the rsr.conf file. This is useful when administering large network based installations of cineSpace. You will find the rsr.conf file within the base directory of your cineSpace installation. The default location for the file is `/Applications/cineSpace/rsr.conf`. Open this file and scroll down to *Section 1: License Settings*. Un-comment the line (remove the preceding '`#`') and enter the port and path to your license server, e.g. `RSR_LICENSE = 2764@rlm_server.company.com`

When entering path names, an absolute path works best since relative paths are relative to the directory that the application was run from, not the directory where the application is located.

Please remember to update the correct license file. I.E. If `$RSR_CONF_FILE` has been set to point to a particular license file then it is that file that requires updating.

2.5 Linux (Platform Notes)

2.5.1 Obtaining a license

To find your hostname and hostid, you will need to open a *Terminal* window and enter some commands that will return information about your system.

1. To find the hostname of your computer, type the command:
   `hostname`

2. To find the hostid of your computer

   - Take the string to the right of the HWadd, and remove the colons from the output of following command:
     `/sbin/ifconfig eth0 | grep -i hwadd`

Please send an email to cinespace-support@rsrhq.com with all of the information generated from these commands.

2.5.2 Installing the RLM license server

For node-locked license configurations there is no need to install the RLM server - please skip ahead to Section 2.5.4.

For floating license configurations, the RLM server needs to be installed and started on the computer that you have nominated as the license server for your network. You can follow these steps to install and run RLM:

1. Unpack the rsunServer archive for your platform and place the files in an appropriate location (e.g. `/usr/local/`).

2. Copy the license file into the `rlm/rsun/` directory.

3. The server program is called `rlm` and should be located in the `/usr/local/rlm/rsun/` directory. Run this program with the full path to the license file and the log file. The log file will be created when rlm is run.

   ```
   rlm -c /usr/local/rlm/rsun/rsr_license.lic
       -dlog /usr/local/rlm/rsun/rsun_log.txt
   ```

   You should get a message indicating that the server has started.
   Also included in the message should be a line indicating that the RLM web service has started. For example:

   `9/11 14:7` (rlm) Web server starting on port 9000

   To check on the server status open a web browser and enter the hostname and port number indicated. For example:

   `http://localhost:9000`

   This will open the Reprise License Server Administration web page.

You can then set the `RSR_LICENSE` variable as described below to complete the license configuration.

2.5.3 Auto-starting your RLM server

There are some different ways to start the RLM automatically, depending on what flavour of Linux you are running. The main boot scripts to look at are: `/etc/rc.boot, /etc/rc.local, /etc/rc2.d/Sxxx, /sbin/rc2.d/Sxxx, etc.` For this example, we will edit the file `rc.local` - this can be found in `/etc` or `/etc/rc.d`

1. Add the following line to this file (Note: you will need to have root access to be able to do this):

```
        sudo -u <username> <full path & name to rlm server> \
                     -c <full path & name to license file> \
                     -dlog <full path & name to log file> &

            e.g.\\
            \begin{verbatim}
                sudo -u rlmuser /usr/local/rlm/rsun/rlm \
                     -c /usr/local/rlm/rsun/rsr_license.lic \
                     -dlog /usr/local/rlm/rsun/rsun_log.txt &
```

   - It is important to note that you run this command under a *sudo* user. This sudo user (rlmuser in this example) is set up to have limited access to the system. This is to keep your system secure, as it is a security hazard to run the RLM server with root user privileges.

2. Now restart your machine. When Linux boots up it will now automatically run the RLM server. To check to see whether RLM has started up correctly, run the command:

   `ps -eouser,fname | grep rlm`

   This should return the result:

   `rlmuser rlm`

   This means that the RLM server is running correctly with the permissions of an rlmuser.

3. To check on the server status open a web browser and enter the hostname and port number. The port number should be indicated in the rsun_log.txt. For example:

   ```
   http://localhost:9000
   ```

   This will open the Reprise License Server Administration web page.

2.5.4 Installing cineSpace

The Linux cineSpace suite is packaged as a tarball archive. When you have downloaded this archive, you will need to 'untar' this package:

i.e. `tar xzvf [file_name]`
e.g. `tar xzvf cineSpace_v2.7.tgz`

This will generate a new directory called 'cineSpaceInstaller'. Within this directory are the setup files for the cineSpace suite. Run the installer script (setup.sh file):

```
cd cineSpaceInstaller
./setup.sh
```

This will install the cineSpace suite into a user specified directory and place shortcut icons in the appropriate places.

2.5.5 Setting Environment Variables

The way to set your environment variables is dependent upon the underlying shell on which your Linux system is based. The find out your shell, type the command:

```
echo $SHELL
```

This will return either

`/bin/tcsh`
or
`/bin/bash`

Please consult the relevant section below.

tsch shell

You may either set the environment variable for all users or just a single user. The process is the same - the only difference is the file you need to edit.

Type	Files
Global	`/etc/csh.cshrc` or `/etc/csh.login`
Single User	`~/.tcshrc` or `~/.cshrc`

Add the following line to the file of your choice:
```
setenv RSR_APP_DIR <cineSpace Location>
```

where `<cineSpace Location>` is the base directory of your cineSpace installation, e.g.
`setenv RSR_APP_DIR /usr/local/cineSpace`

bash shell

You may either set the environment variable for all users or just a single user. The process is the same - the only difference is the file you need to edit.

Type	Files
Global	/etc/bashrc
Single User	~/.bashrc

Add the following line to the file of your choice:
`export RSR_APP_DIR=<cineSpace Location>`

where `<cineSpace Location>` is the base directory of your cineSpace installation, e.g.
`export RSR_APP_DIR=/usr/local/cineSpace`

2.5.6 Copy license file

For node-locked licenses place the license file in the licenses directory within the cineSpace application directory. The cineSpace will look in this directory for node-locked licenses. For example:

```
/usr/local/cineSpace/licenses/rsr_license.lic
```

For floating licenses placing a copy of the license file in the location above will tell cineSpace where to look for the license server.

2.5.7 The rsr.conf file

You can also specify the location of the license server in the rsr.conf file. This is useful when administering large network based installations of cineSpace. You will find the rsr.conf file within the base directory of your cineSpace installation. Open this file and scroll down to *Section 1: License Settings*. Un-comment the line (remove the preceding '#') and enter the port and full path to your license server, e.g. `RSR_LICENSE = 2764@rlm_server.company.com`

When entering path names, an absolute path works best since relative paths are relative to the directory that the application was run from, not the directory where the application is located.

Please remember to update the correct license file. I.E. If `$RSR_CONF_FILE` has been set to point to a particular license file then it is that file that requires updating.

2.5.8 Set permissions for `monitor-profiles` directory

By default, the permissions on the `monitor-profiles` directory are set so that only the root user can write to this directory. If you wish, you may need to change the permissions on this directory so that all users who run cineProfiler will have write access to this directory.

For example, if you wish all users to have access to this directory, and you have installed cineSpace into `/usr/local/cineSpace`, then you should run:

```
chmod a+w /usr/local/cineSpace/monitor-profiles
```

2.6 Troubleshooting

The most common issues regarding licensing relate to the `RSR_LICENSE` setting in the rsr.conf file. If you are getting license errors then first check that you have the appropriate `RSR_LICENSE` setting for your configuration, i.e. the license server name is correct.

Depending on how your network is set up, you may need to specify the full machine path on the network (e.g. 'server.company.com') instead of just the computer name (e.g. 'server'). You may even need to specify an IP address if your network is having DNS resolution problems.

Check that the license server is running and that it is pointing to the correct license file. To check on the server status open a web browser and enter the hostname and port number. The port number can be found in the rlm server log. For example:

```
http://localhost:9000
```

This will open the Reprise License Server Administration web page.
For node-locked licenses running on stand-alone machines, check that you have typed the license file path correctly and that it is specified in absolute, rather than relative, terms. You can also check that you have the correct license type for your set-up by opening the license (`.lic` or `.dat`) file in a text editor. If the term `USE_SERVER` appears in the file then it must be used with a license server. In this case, either alter your set-up to use a server or request a different license for that particular computer.

A great site for learning about and troubleshooting your cineSpace suite is the cineSpace forum. We encourage all to sign up and participate on this online site, which can be found at:

http://cinespace.rsrhq.com/forum

If you are unable to find a solution in the forum, please contact the cineSpace support team at:

cineSpace-support@rsrhq.com

Chapter 3

Installing the cinePlugins

3.1 Preparing ahead

3.1.1 Obtaining the cinePlugins

You can obtain the cinePlugins for all platforms from the same RSR FTP server on which the main cineSpace software distributions are located:

ftp://ftp.rsrhq.com/

You will need login details (username and password) to access the FTP server. Please contact the RSR team if you do not already have the required login information.

3.1.2 Installing the cineSpace suite

Please ensure that you have installed the cineSpace suite prior to installing any of the cinePlugins (see Chapter 2 for more information). This will ensure that your system environment is correctly configured and licensed before you attempt to install the plug-in(s).

3.2 cineShake

cineShake is a plug-in to Shake that provides the functionality of both equalEyes and full 3D colour transforms as a node within Shake. cineShake will work under Mac OS X, and Linux Fedora Core (and presumably other glibc 2.3 based systems) with Shake 4.0 and 4.1.

3.2.1 Linux

The Linux distribution of cineShake is provided in two formats:

1. A *GUI installer package* containing all versions of the plug-in; and
2. Individual *tarball distributions* for each version of Shake.

In most situations, the GUI installer package is the recommended method for installing cineShake. If problems occur using the GUI installer, you may need to manually install cineShake using the appropriate tarball distribution.

GUI installer package

The Linux cineShake plug-in GUI installer package is distributed within a tarball archive. When you have downloaded this archive, you will need to 'untar' this package using the following command:

```
tar xzvf cineShakeInstaller_v2_7rc_Linux.tgz
```

This will generate a new directory called 'cineShakeInstaller'. Within this directory are the setup files for the cineShake plug-in. Run the setup script (setup.sh file), which will install the cineShake plug-in:

```
cd cineShakeInstaller
./setup.sh
```

Tarball distribution

The Linux cineShake plug-in is packaged as a tarball archive. Select the correct archive from the FTP point, depending upon which installation of Shake you are running. When you have downloaded this archive, you will need to copy this to your `/nreal` directory and 'untar' it using the following command:

```
cp cineShake_v2_7_shake40_Linux.tgz ~/nreal/.
cd ~/nreal
tar xzvf cineShake_v2_7_shake40_Linux.tgz
```

The cineShake plug-in installation is then complete.

3.2.2 Mac OS X

The Mac OS X cineShake plug-in *GUI installer package* is distributed as a `.tgz` archive. When you have downloaded this archive, you will need to 'unzip' this package using Finder, and run the installer. To do this:

- In Finder, double-click on the downloaded archive. This will create a new application in the same directory called *CineShakeInstaller*.
- Double-click on the *CineShakeInstaller* application.

This will open the cineShake installer GUI, which will allow you to install cineShake for Shake v4.0 or v4.1 [1]. The installer will prompt you to choose the version you wish to install, and next to each version will be a drop-down box with the following options:

Option	**Explanation**
Current User	Only the current user installing the plug-in can run cineShake
All Users	Anyone that uses this Shake application will be able to run cineShake
None	Will not install this version of cineShake

The cineShake installer will attempt to identify the correct install location for the cineShake plug-in. If you wish to override the default path, click the *Choose* button and select your desired location.

Press the *Install* button to install the cineShake plug-in. A window will pop up showing the progress of the installation. When the installation is complete, a green *OK* will be displayed next to the installed cineShake plug-in name. Close this window and the original cineShake installer window to finish.

3.3 cineFusion

cineFusion is a plug-in to Fusion that provides the functionality of both equalEyes and full 3D colour transforms as a node within Fusion.

cineFusion requires Fusion version 5 and is only available on the Windows platform.

3.3.1 Windows

To install this plug-in, simply double-click on the cineFusion installer file – this will open a wizard that will guide you through the installation. It will prompt you to enter the path to the location of your Fusion installation and then complete the cineFusion installation set-up.

3.4 cineNuke

cineNuke is a plug-in to Nuke that provides the functionality of both equalEyes and full 3D colour transforms as a node within Nuke.

There are versions of cineNuke available to support Nuke v4.7 and higher, for Linux, OSX and Windows.

3.4.1 Windows

To install this plug-in, first identify what version of Nuke you are currently using and then download the corresponding cineNuke installer package from the RSR

[1] Please note that you can only have one version installed at any one time

FTP server. Double-click on the cineNuke installer file – this will open a wizard which will guide you through the installation process. It will prompt you to enter the path to the location of your Nuke installation and then complete the cineNuke installation set-up.

3.4.2 Linux

The Linux cineNuke plug-in is packaged as a tarball archive. When you have downloaded this archive, you will need to 'untar' this package:

i.e. `tar xzvf [file_name]`
e.g. `tar xzvf cineNuke_v2.7_Linux.tgz`

This will generate a new directory called 'cineNukeInstaller_VERSION_Linux_REVISION_DATE'. Within this directory are the setup files for the cineNuke plug-in. Run the installer script (setup.sh):

```
cd cineNukeInstaller
./setup.sh
```

Select the version of Nuke you are using. This will install the cineNuke plug-in into a user specified directory.

3.4.3 Mac OS X

The Mac OS X cineNuke plug-in *GUI installer package* is distributed as a `.dmg` archive. When you have downloaded this archive, you will need to 'unzip' this package using Finder and follow the instructions in the *Readme.txt*. The instructions are:

- Locate the Nuke5.0v1 application bundle on your system (most likely in the /Applications directory)
- Ctrl click on the Nuke5.0v1 application bundle and select "Show Package Contents".
- Go into the Contents directory, then into the MacOS directory, and finally into the plugins directory. (Figure 3.1)
- Finally, drag the CineNuke.dylib file into the plugins directory

3.5 cineFilmMaster

cineFilmMaster is a plug-in to Digital Vision (Nucoda) Film Master that provides the functionality of both equalEyes and full 3D colour transforms as a CMS object within Film Master.

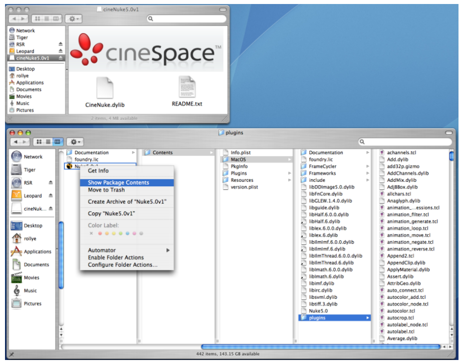


Figure 3.1: cinePlugin Nuke OSX

cineFilmMaster requires FilmMaster 3.02 or 3.5, and is only available on the Windows platform.

The cineFilmMaster plug-in uses the OpenFX (OFX) framework, with custom libraries provided by Digital Vision that extend the OpenFX API. This allows the interpolation of LUT data, which is rendered on the GPU in real time during playback.

3.5.1 Windows

To install this plug-in, simply double-click on the cineFilmMaster installer file – this will open a wizard that will guide you through the installation.

Chapter 4

cineSpace features

4.1 Overview of cineSpace features

The cineSpace colour management suite consists of multiple applications that enable you to achieve correct calibration within a wide range of industry tools. Across the cineSpace application suite, however, many of the features – and the colour transforms themselves – are the same, thus providing consistency throughout your colour pipeline.

This chapter provides an overview of the features provided by the cineSpace suite and some information about the uses and limitations of each, plus examples of situations in which they may be utilised. The chapters covering the individual cineSpace applications contain details about how to activate each of these features in that context.

cineSpace summary feature list:

- Monitor optimisation
- Display profiling
- Colour transforms (1D and 3D)
- Inverse transforms *(NOTE - Optional extra)*
- Scaling
- Printer lights
- Black point correction
- White point adaptation
- Pre-transforms (lin-to-log and custom)
- Out of gamut treatment

4.1.1 Definitions

Before explaining each of the features, we should clarify some important terms used in the context of colour calibration. Achieving correct calibration involves three broad steps that are carried out in turn: *Optimising*, *Profiling* and *Calibrating*. These terms are defined as follows:

Optimising Simple adjustments to the colour settings of your viewing device (using the provided hardware controls) to adjust the native gamut before profiling.

Profiling Recording how your viewing device behaves in response to different video colour signals.

Calibrating The application of a LUT (e.g. via equalEyes) to ensure that your viewing device matches a chosen standard.

4.2 Optimisation

NB: This is done in cineProfiler, and is only relevant to viewing devices that have separate RGB gain controls, and, ideally, separate RGB bias controls.

Optimisation involves setting the luminance and *chromaticity* of the viewing device's white point. The aim of this process is two-fold:

- To reduce the overall colour cast seen by users; and
- To maximise available dynamic range and improve accuracy.

Uses

By setting the monitor's native white point as close as possible to the target device's white point, the user can minimise white point shifts when applying or removing LUTs. These shifts can cause users to perceive colour casts (due to the large change in white point) that are not actually present, which can be a problem when making colour choices.

When the monitor's gamut is close to the intended target device's gamut, you maximise the dynamic range available when applying LUTs. This will give you the best result for matching the target device, and reduce the chance of banding occurring due to quantisation errors.

Limitations

Optimisation is typically only performed for CRT monitors that have RGB colour gain and, preferably, bias controls. Since LCD monitors and digital projectors adjust internal colour settings in a different manner, optimisation does not offer the same benefits as for CRT monitors.

Examples

- Adjusting a CRT monitor so that it is suitable for viewing film images at a different white point

4.3 Profiling

Profiles are files that describe the behaviour of different display devices, and the process of producing such files is known as *profiling.* A profile records the colours displayed by the device for a given set of input values (stimuli). The colours are represented in the device-independent *Commission Internationale de l'Eclairage* (CIE) XYZ tri-stimulus colour space.

There are two main profile types:

- Monitor profiles – generated by cineProfiler to describe the behaviour of your current viewing device (monitor or digital projector).
- Target profiles – measured or generated mathematically, describing the behaviour of the target device.

These profile files are written in XML format. The profile file will have some header meta data followed by a data section that contains the response of the device. The profile will look something like:

```
<profile>
<head>
   ⋮
</head>
<data>
<strip frames="61">
<red steps="15">
<patch step="1" frame="1">
<stimuli>
<red>0.0666666666667</red>
<green>0.0</green>
<blue>0.0</blue>
</stimuli>
<results>
<RGB type="non-linear">
<red>0.0056053916242</red>
<green>0.0</green>
<blue>0.0</blue>
</RGB>
<XYZ>
<X>0.184937944371</X>
<Y>0.0953586275663</Y>
<Z>0.00866896614239</Z>
</XYZ>
</results>
</patch>
   ⋮
</red>
```

```
<green>
    ⋮
</green>
    ⋮
</strip>
</data>
</profile>
```

The data section of the profile can either contain four 1D ramps (a *ramp profile*), or a 3D cube (a *cube profile*) of stim/result data. Please see the following sections for more information about the two profile types.

4.4 Ramp profiles

These profiles contain four 1D data ramps. The ramps contain information about the grey and primary colour response of the profiled device. The advantage of these profiles is that they contain significantly fewer sample points than a cube profile and thus are much quicker to create. All other points in the colour gamut are then obtained through software interpolation.

Uses

This is the recommended profile type to use when your display device *decouples* as it can be created quickly and still deliver accurate results. Please refer to Section E for information about decoupling devices.

Limitations

This profile type cannot accurately represent the full response of a display device that does not decouple, since it relies on the device adhering to specific models of colour response. This makes ramp profiles unsuitable for representing film colour as well as some monitor and projector types.

Examples

- sRGB (standard profiles)
- ITU_R_709
- Most CRT monitors [1]
- Some LCD monitors [2]

[1]Some older CRT monitors do not correctly decouple and may require a cube profile for accurate calibration.

[2]Please carefully observe the *decoupling information* generated by cineProfiler when attempting to use a ramp profile with an LCD monitor.

4.5 Cube profiles

These profiles contain a full 3D cube of data sampled evenly throughout the device's colour gamut. The advantage of these profiles is they contain many more data points than a ramp profile and thus provide a more accurate representation of the colour gamut.

Uses

This is the recommend profile type to use when your display device does *not* decouple since it makes no assumptions about the colour response of the device. Please refer to Section E for information about decoupling devices. To take full advantage of a cube profile, you should apply a *3D transform* (refer below for further information).

Limitations

The main limitation of this profile type is size – such profiles can take a long time to create as a large number of samples need to be taken.

Whether you can take advantage of a cube profile's accuracy is also dependent upon having a way to apply a 3D transform in your work flow. If you are only using equalEyes, for example, then you will not gain any benefits from capturing a cube profile from your display.

Examples

- Film profiles
- Most LCD monitors
- Digital projectors
- Older CRT monitors

4.6 Synthetic profiles

Synthetic profiles are generated in memory by the cineSpace applications and do not exist in a separate XML file. They may be adjusted "on-the-fly" by the user instead of being hard-coded with specific values.

Uses

This is the recommended profile type for when a user wants to specify a custom gamma and white point for the target profile.

Limitations

Synthetic profiles are currently only available in equalEyes.

Examples

If a user wishes to see how things would look on a display device with a gamma of 1.1, this can be achieved simply by setting this value for gamma on the Synthetic Target page of equalEyes.

Another example would be if you wish to see how a Cineon image would look when displayed on your in-house projector – just choose the Cineon option on the Synthetic Target page of equalEyes and choose your desired white point (to match your projector).

4.7 Colour transforms

Colour transforms are a critical component of achieving accurate calibration. They determine how to interpret the information stored in monitor and target profiles in order to generate a LUT that provides the correct mapping from one colour space to another. Since only a limited number of sample points can be captured in each profile, the colour transforms must accurately interpolate the remaining values.

cineSpace offers two main colour transform types: 1D and 3D. These are describe in further detail below.

4.7.1 1D transform

A 1D transform makes use of the monitor's primary ramp values to simulate the grey ramp behaviour of the target device.

Uses

A 1D transform may be used in situations where both the monitor and target are described by ramp (1D) profiles. Where the target profile is a cube (3D), the 1D transform will accurately reproduce the tonal curves and chromaticity of the target's grey ramp. It is the only transform type that can be applied to a computer screen through conventional 1D LUTs on a graphics card, using equalEyes.

Limitations

1D transforms cannot change the colour (chromaticity) of the primary ramps of the monitor (viewing device). They are not recommended for accurately matching targets that are non-decoupling – such as film, which exhibits cross-talk and desaturation that cannot be accurately mapped with this transform type. Such transforms are also not recommended for use on non-decoupling display devices, such as many digital projectors.

Examples

- Matching one CRT monitor to another
- Displaying the sRGB colour space on a CRT monitor
- Displaying a Cineon film profile on a CRT monitor

4.7.2 3D transform

A 3D transform makes use of the full monitor (viewing) device gamut – not just the grey ramp and primaries – to simulate the entire displayable gamut of the target device.[3]

Uses

The great accuracy achievable when using 3D transforms means that they are ideal for matching all types of target devices. A 3D transform can change the colour (chromaticity) of the primary colour ramps and accurately reproduce the colour and tonal curves of even non-decoupling target devices. Such transforms will also enable a non-decoupling viewing device to accurately match the chosen target colour space.

Two 3D transform options are available in the cineSpace application suite: *quick* and *accurate*. Because it can be processed more rapidly, the quick transform is used when you want to make adjustments to the transform parameters. Once you have finalised your settings, switch to the accurate mode for the best quality match.

3D – quick Fast set-up time and fast transform time, with better GUI interactivity when adjusting cinePlugin parameters, but may give slightly inaccurate results near gamut boundaries.

3D – accurate Accurate results throughout entire colour space, including gamut boundaries, but slower set-up time and poor GUI interactivity when adjusting cinePlugin parameters.

Limitations

Using 3D transforms in practice involves specific hardware and/or software, meaning they cannot be utilised within all applications. The 3D transform option is not available in equalEyes due to this limitation, but can be used in cineCube (to generate a 3D LUT) and the cinePlugins.

Examples

- Matching a digital projector to a film stock
- Accurately matching any LCD monitor to a video or film colour space

[3] *Displayable* in this sense means "able to be displayed on the monitor (viewing) device that you are using."

4.8 InverseTransform

An inverse transform occurs when the monitor (viewing) and target profiles are reversed. Inverse transforms can be used to apply a data transform to your images so that they reproduce correctly when sent to a different output medium or colour space. When applied, the raw output will no longer look correct on the current viewing device but will instead be correctly transformed for the output medium. In general, the transform will be baked into the material instead rather than being used during image viewing. This is essentially the opposite of normal colour transforms in cineSpace, which are viewing transforms used to obtain an accurate on-screen preview of how your material will look on the final output.

NOTE *This feature is not enabled in the standard cineSpace suite, but is available for purchase as an optional extra. If this feature is used without the correct license the processed image will be displayed with a strong green cast. Please contact cinespace-support@rsrhq.com if you require further information or wish to upgrade your license to include this feature.*

Uses

This type of transform is useful where material has already been graded in one colour space but needs to be delivered in another. The most common example is where images have been captured in HD and graded on a Rec. 709 reference monitor, but require delivery as log DPX frames for film out. It can also be useful, however, where material has been graded for a particular film out path (taking into account film stocks, recorder and lab) but needs to be processed in a different way, such as through another lab. By creating a custom film profile for each film out path, these can be used to generate an inverse transform that compensates for the differences, and this is then baked into the frames before film out.

Limitations

Film and video colour spaces have quite dissimilar gamut shapes and so there can be some issues with out-of-gamut colours when applying inverse transforms to achieve a video look on film. In most cases this can be addressed by using the transform scaling factor (see section below) to reduce the video gamut size, but this is somewhat scene-dependent and may require a small amount of manual re-grading in some cases. Where the profiles are both for video or both for film, gamut issues are usually minimal.

4.9 Scaling

Luminance *scaling* involves making use of the entire possible luminance range of a viewing device. The maximum luminance of the target profile is scaled to match the maximum achievable luminance of the viewing device.

More precisely, having scaling *on* means that, for normalised code values between 0 and 1, an input RGB triple of (1,1,1) (i.e. "white") will always produce an output RGB triple where at least one of the three values will be 1. Put another way, the transformed white value will be the brightest that the monitor device is capable of displaying without *clipping* any of the channels.

If, instead, we had scaling *off*, there is a chance that some output colour values may be *clipped* if the monitor device is not capable of reproducing the correct values, resulting in chromaticity shifts.

There are two main situations that we need to consider when deciding whether or not to make use of scaling. If we define the monitor (viewing) device's white point luminance as 'm' and the target device's white point luminance as 't', we then have the following alternatives:

- m > t: The monitor (viewing) device's white point luminance is greater than the target device's white point luminance.
- m < t: The monitor (viewing) device's white point luminance is less than the target device's white point luminance.

When the monitor (viewing) device's white point luminance is *greater* than the target's white point luminance, (i.e. m > t) you can work both with and without scaling without encountering any issues. However, scaling *on* will utilise the full dynamic range of the monitor (viewing) device.

When, on the other hand, the monitor (viewing) device's white point luminance is *less* than the target's white point luminance, (i.e. m < t) there will be clipping issues when scaling is off.

The table below highlights the effects of using scaling for a given input:

Case	*Input*	*"Correct" output*	*Scaling off*	*Scaling on*
m > t	[1.0,1.0,1.0]	[0.6,0.7,0.8]	[0.6,0.7,0.8]	[0.75,0.88,1.0]
m < t	[1.0,1.0,1.0]	[0.9,1.2,1.1]	[0.9,1.0,1.0]	[0.63,1.0,0.92]

Uses

Using scaling means that the full dynamic luminance range of the viewing device may be utilised, reducing any potential banding. This, combined with the use of a *relative colorimetric intent* when applying the colour transform, ensures that accurate chromaticity is maintained.

Scaling *on* is the default mode for the cineSpace application suite and is the recommended mode for most applications. It is suitable for all situations where the absolute luminance level does not need to precisely match the target luminance.

Limitations

Since scaling results in an output maximum white point luminance that is dependent on the viewing device, rather than being an absolute value, two calibrated monitors placed next to each other may exhibit different luminance levels. In situations where this may present an issue (e.g. needing a perfect match in both chromaticity and luminance between adjacent monitors), it may be appropriate to deactivate scaling. Scaling may also need to be deactivated when your output needs to precisely match a specific luminance level, e.g. for a chosen video standard.

With scaling *off*, you will achieve a precise match to the target in both chromaticity – using an *absolute colorimetric intent* – and luminance, but only if the viewing device is 'brighter' than the target. If, on the other hand, the target has a higher luminance, then the white point values will be clipped and the frame output will not match the target. In the case where the monitor is significantly 'brighter' than the target, only a small portion of the monitor's dynamic range will be utilised, thus reducing resolution and increasing the chance that banding may occur.

As such, we recommend leaving scaling on, unless you are an experienced user.

Examples

- Grading for film on a digital projector, using the full dynamic range of the display device
- Using an older CRT with low luminance levels to view NTSC video material

Scale Factor

Scale factor allows a user to ensure that target and monitor gamuts line up as optimally as possible. Using the scale factor feature allows a decrease in the overall brightness and an increase in the overall quality of the colours. This will therefore reduce the amount of out-of-gamut colours.

4.10 Printer lights

Printer light settings within cineSpace are designed to emulate the printer light controls available on film recorders, enabling tonal adjustments of the target profile. Film recorders vary the red, green and blue light levels in order to control the cyan, magenta and yellow densities of the output film. In cineSpace, this is emulated by offsetting the red, green and blue channels in RGB space. In effect, printer lights apply a primary grade to the target profile being used by cineSpace.

Uses

Traditionally, during the print process, the **trims** are used to neutralise the print. That is, we adjust the RGB trims so that an offset of 25 for each channel will produce an *on-aim* print. The **offsets** are then used to adjust/grade the print.

For simple printer light adjustments, only the printer light **offsets** are adjusted – this is what adjusting printer lights in equalEyes does. It provides a way to see what effect certain printer light settings applied at the lab will have on the look of the final print.

To emulate the behaviour of a known lab or 'look' the printer light **trims** can also be adjusted via the rsr.conf file. This enables the user to first apply the 'look' while the offsets are set neutrally at (25,25,25) and then adjust the offsets as needed to make alterations relative to the original starting 'look'.

By default, cineSpace uses eight (8) printer light points in one stop of exposure. Due to there being a number of other commonly used values, the number of printer light points per stop can be changed via the rsr.conf file.

Some background on the reasons why different values of printer light points per stop exist may be found in discussions on the cineSpace forum site:

http://cinespace.rsrhq.com/forum

Limitations

The printer light adjustments are only an *approximation* of the actual physical changes that occur in a film recorder. As such, whilst they can be very effective in compensating for subjective differences between the target profile being used and the actual film print output obtained, there may be some cases where the controls cannot achieve a precise match.

Examples

In situations where the lab output has shifted from the target film profile being used during grading, printer light settings may be used to adjust the displayed profile to compensate. This would be done by comparing a known reference frame (e.g. a 'Marcie') on-screen with the film out and then adjusting the printer lights until the images match.

Printer lights can also be used to obtain a preview of how the images will look with increased or decreased overall exposure, by adjusting all three values (R, G and B) up or down by the same amount.

4.11 Black point correction

Black point correction is a method for adjusting the response of colour transforms through the low luminance region of the colour gamut. By using a *relative colorimetric intent* when performing the transform, it provides a means for obtaining a good perceptual match even when there are significant differences between the low luminance regions of the monitor and target profiles.

The black point correction algorithm performs two duties:

- Reduces noise in the low luminance areas; and

- Ensures that the black point of the target is not darker than the monitor (viewing) device's black point.

Without this algorithm, trying to view a black darker than your viewing device's capability will result in clipping in the low luminance regions (often referred to as *crushed blacks*).

4.11.1 Correction methods

The default mode is 'Match Blacks', which will provide the best results in most if not all situations. In some very rare cases, one of the other modes may deliver improved results, especially for colours near gamut boundaries. However, we recommend leaving the black point correction mode at its default setting unless you are an experienced user.

Currently there are four black point correction methods provided by cineSpace though this list will be reduced in future releases:

- Match Blacks
- Along Grey Ramp
- Nearest Gamut Boundary
- Target Offset

Black point correction is *on* by default, but it will not actually modify the colour transform unless it is required. In cases where the monitor's black point is *lower* than that of the target profile, no correction will be performed. Where the monitor's black point is *higher* than that of the target profile, however, black point correction will be used to obtain the best possible perceptual match.

Users can force black point correction to always occur – even when not needed – by using the 'Force Correction' option.

Please refer to Appendix B for a detailed explanation of the various black point correction methods.

Uses

Current display technologies, including LCD monitors and digital projectors, are often unable to accurately reproduce the colour response of various film and video target spaces at very low luminance levels (these devices are said to have a *lifted* black point). By using black point correction, however, the user achieves a very good *perceptual* response on such devices. This greatly improves the level of detail that can be viewed in low luminance ("shadow") areas of images when working on devices that are not ideal for critical colour work (e.g. LCD monitors).

Limitations

Black point correction is very effective for almost all situations. There may be some rare cases, however, where you choose to disable black point correction so that the higher luminance colours precisely match the target in *absolute* value rather than *relative* value. An example may be a scene that has few dark areas and you wish to display the exact higher luminance colours, at the expense of shadow detail.

Examples

- Compositing, using an LCD monitor, with material that contains significant shadow detail
- Minimising "crushing" of blacks when using a digital projector that has a lifted black point

cineCube Visual

cineCube Visual is slightly different when using black point correction. It only uses the *Match Blacks* option and allows the following options -

`On` This is 'Force Correction', meaning that black point correction using *Match Blacks* will occur even if it not needed.

`Auto` Black point correction is *on*, but it will not actually modify the colour transform unless it is required. In cases where the monitor's black point is *lower* than that of the target profile, no correction will be performed. Where the monitor's black point is *higher* than that of the target profile, black point correction will be used to obtain the best possible perceptual match.

`Off` No black point correction will take place.

4.12 White point adaptation

White point adaptation is a feature that allows you to shift the white point chromaticity of your target profile from its measured value, effectively changing the *illuminant*. The effect on the displayed image is similar to changing the colour of the lamp in a film projector.

Uses

White point adaptation is useful if you want to simulate a particular illuminant, e.g. if you want all of your monitors to have the same white point as your digital projector. It can also help reduce perceived colour casts that may occur due to perceptual differences between the white point of your images and the native white point of your monitor (as displayed in other application windows).

There are three ways to set the desired target white point:

Monitor This sets the white point to the original monitor white point temperature. It ensures that the white point of the monitor is not shifted when the LUT is applied. This can help reduce potential banding problems by ensuring that you are making use of the entire dynamic range of your monitor.

Temperature This sets the white point using a *colour temperature* in degrees Kelvin or a standard CIE illuminant. It is useful when trying to match a known illuminant such as CIE D65 or 5400K.

Chromaticity This sets the white point using CIE chromaticity coordinates (x,y). It is useful when trying to match an illuminant with known chromaticity coordinates, such as a digital projector that has been measured to determine its white point chromaticity.

Limitations

When applying white point adaptation, the absolute values of displayed colours will not exactly match the "correct" target profile. Normally, the human eye compensates for changes in white point when there is no reference to compare it with (such as in a movie theater), but we still recommend doing a final review of your colour choices under ideal viewing conditions using the "true" (unadjusted) target profile.

Examples

- Reducing subjective colour casts that may be perceived due to differences between the image white point and the monitor's native white point
- Maximising available dynamic range to reduce potential banding

4.13 Pre-transforms

Pre-transforms are transforms that are applied *before* the cineSpace transform. They have the effect of modifying the input image data values that are fed into the cineSpace transform.

The transform steps performed by cineSpace can broadly be represented as follows:

$$input[R, G, B] \rightarrow \langle Pre{-}trans.\rangle \rightarrow [R_1, G_1, B_1] \rightarrow \langle cineSpace\rangle \rightarrow output[R_2, G_2, B_2] \tag{4.1}$$

Two types of pre-transform are currently supported by cineSpace:

- Lin-to-log
- Custom (LUT) file

Printer lights (Section 4.10) are technically also a pre-transform, but are treated slightly differently by cineSpace and so are not covered here.

4.13.1 Lin-to-log

The *lin-to-log* (also abbreviated as *lin2log*) pre-transform is used when the input image data is in linear (linear-to-light) space. The pre-transform will convert the data (image) into log (Cineon) space using the industry standard Cineon transform.

There are four parameters that may be used to adjust the output of the Cineon lin-to-log transform in special cases:

White point This is the reference white value for the Cineon standard (default 685). The Cineon curve has its shoulder above the reference white.

Black point This is the reference black value for the Cineon standard (default 95). The negative exposure curve has its toe beneath the reference black.

Display gamma This is the gamma of the display device (default 1.7).

Film gamma This is the mean slope value of the film curves between the black point and white point (default 0.6).

The values for the white and black point are in 10-bit Cineon space. In general, the standard transform works perfectly well for common situations and it is recommended that only an experienced user should deviate from the default values.

Uses

Usually when working with film images, they will be stored in log format (e.g. Cineon or DPX file format) as this is the format that film recorders expect. At times, however, you may need to work with them in linear format[4] as part of your work flow. To obtain linearised film images from the original Cineon or DPX files, a log-to-lin conversion is performed at some stage (often when loading the images into your working application). Before sending the images to the film recorder, the reverse transformation (i.e. lin-to-log) is applied to produce standard Cineon/DPX log files.

To compensate for this, cineSpace can apply the lin-to-log conversion to your linearised film images in addition to modifying the colour characteristics to suit the target profile. Thus regardless of which file type you prefer to work with, cineSpace will still display the images correctly.

Further discussion and comments on the use of linear and Cineon/DPX log images may be found in Appendix C.

Limitations

When working with linearised film images, you will need to ensure that you are using sufficient bit depth in your colour values to obtain the same tonal resolution as when using 10-bit log Cineon images. You also need to be able to

[4] *Linear*, in the context of cineSpace, means linear-to-light.

represent values outside of the 0 to 1 range to accommodate “super-whites”. In practice, this usually means using at least a 16-bit floating point colour pipeline.

Examples

The lin-to-log pre-transform would normally be used when your images are in linear-to-light space and you have the intention of performing a lin-to-log transform prior to sending them to the film recorder (Figure 4.1).

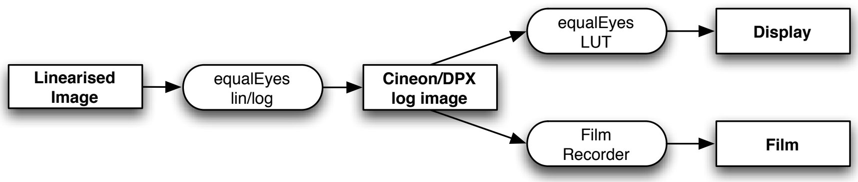


Figure 4.1: Working with linear-to-light images: lin-to-log pre-transform active

Where you are working with Cineon/DPX log images (Figure 4.2), remember to *deactivate* the lin-to-log pre-transform.

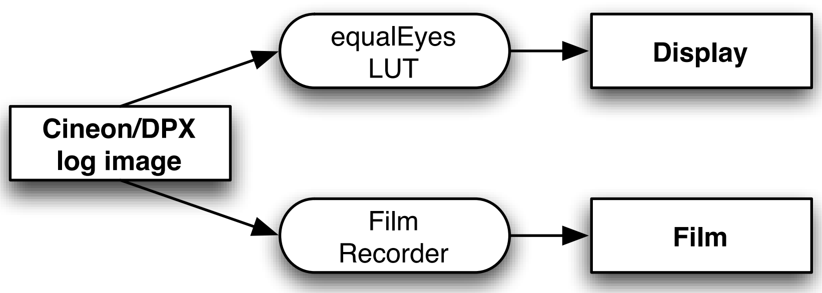


Figure 4.2: Working with Cineon/DPX log images

When working with normal video files (for, say, a computer animation) and viewing them using HD calibration (Figure 4.3), you do not need to apply the lin-to-log pre-transform – the images are already in their output format (for viewing on a monitor).

4.13.2 Custom (LUT)

This allows a user to custom design a 1D LUT that will be applied as a pre-transform prior to the cineSpace colour transform. The custom pre-transform

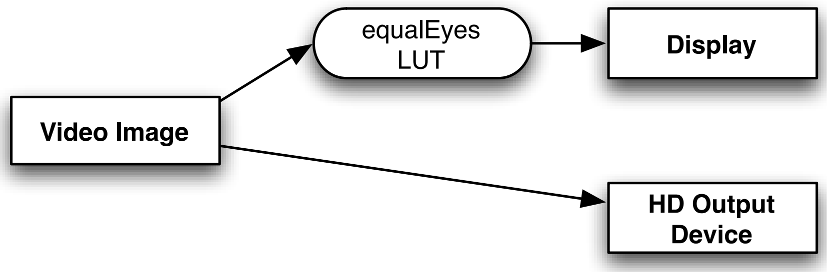


Figure 4.3: Working with video images

can thus apply any arbitrary conversion as required by the user.

Uses

Custom pre-transforms can be useful when the material being viewed has been acquired from a device with an unusual response curve, such as a digital camera. The user can design a LUT that will transform the images into standard Cineon/DPX log space and then load it as a custom pre-transform before applying the cineSpace calibration.

The custom pre-transform LUT format is very simple, containing basic header information and then the actual LUT data. The format and some example LUTs are detailed in Appendix D.

Limitations

When using custom pre-transforms, you must ensure that the pre-transform LUT you are applying exactly matches any transforms that you will apply to your images prior to sending them to the final output.

Examples

- Compensating for a digital camera's native sensor response curve when previewing images

4.14 Out-of-gamut treatment

Out-of-gamut treatment shows the user which colours within an image or data set are not able to be correctly represented on the viewing device (monitor). These are colours that exist in the target profile's gamut but not in the gamut of the monitor profile. In other words, they can be physically reproduced by the target device, but the monitor (viewing) device is incapable of reproducing them.

There are three possible out-of-gamut treatment modes:

None This is the default mode, in which any out-of-gamut colours are simply clipped to the monitor (viewing) device's colour gamut.

Green When set to 'Green', any colours that would be out-of-gamut for the monitor (viewing) device are displayed as bright green.

Alpha When set to 'Alpha', the alpha (luminance) channel of any colour that would be out-of-gamut for the monitor (viewing) device is set to zero. If the colour is inside the gamut of the viewing device, the alpha value is set to one (1).

Uses

The out-of-gamut feature is extremely helpful in identifying those colours in an image that may not be accurately reproduced upon final output. It is commonly used when colour grading for film using a video monitor, where there is a significant mismatch between the gamuts of the monitor and target profiles. By activating out-of-gamut treatment from time to time, the user can quickly identify regions of the image that may reproduce differently when going out to film.

Limitations

Whilst out-of-gamut treatment will indicate the colours in an image that are not being correctly displayed by the monitor (viewing) device, there may be some adjacent in-gamut colours that appear to be different on the final output that are not identified using this feature. This is due to the fact that our perception of a particular colour is affected by surrounding colours, some of which may be out-of-gamut and hence incorrect.

Examples

- Checking that critical colours in an image will reproduce precisely the same when transferring to film

Figure 4.4: Example of out-of-gamut treatment

Chapter 5

cineProfiler

cineProfiler is an application that utilises a *hardware probe* (covered in Section 2.1.2) to *profile* the response of your monitor. A profile describes how your monitor responds – in terms of the actual output from the screen – when presented with various colour stimuli. Profiles are stored as XML files that are used by the cineSpace applications to perform colour transforms – they are explained in further detail in Section 4.3.

cineProfiler also attempts to help you *optimise* your monitor so that you will obtain the best possible colour response from your display device (see Section 4.2 for more information).

Before optimising and profiling your monitor, we suggest that you review the section on profiling considerations (Section 5.9) to familiarise yourself with factors that can affect the process and how often you need to re-profile.

5.1 Initialising cineProfiler

When you start cineProfiler you will be presented with the cineProfiler initialisation window (Figure 5.1). This allows you to select the type of display you are profiling, which controls are available on that display, and the type of profile you would like to create. There is also the option to choose the advanced functionality of *output independent profiling*, and to use a *remote probe*.

5.1.1 Viewing Device

It is important to select the correct type of viewing device that you will be profiling. You can select from:

CRT Used for profiling normal CRT monitors ("computer" monitors).

LCD Used for profiling LCD/TFT monitors (e.g. IPS, S-IPS, PVA, MVA).

Broadcast CRT Used for profiling CRT broadcast monitors.

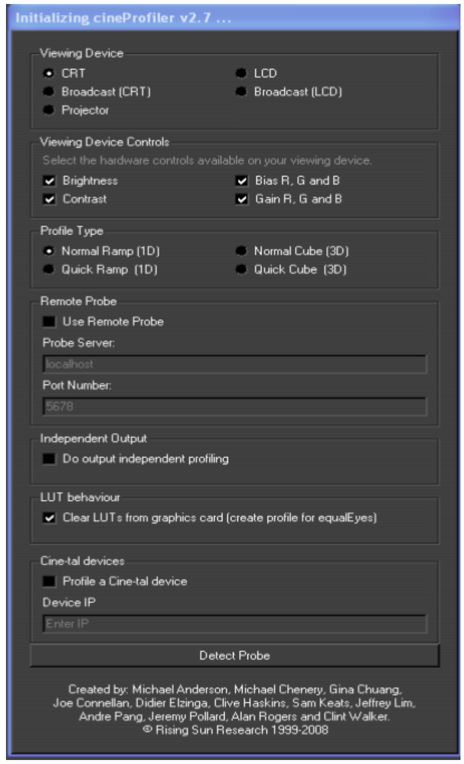


Figure 5.1: cineProfiler initialisation window

Broadcast LCD Used for profiling broadcast LCD monitors.

Projector Used for profiling digital projectors.

When you select your viewing device, cineProfiler will set the default options for the other controls available on the initialisation window. The following table indicates the default options set for each type of viewing device:

	CRT	*LCD*	*Broadcast (CRT)*	*Broadcast (LCD)*	*Projector*
Viewing Device Controls					
Brightness	•	•			•
Contrast	•	•			•
Bias R, G and B	•				
Gain R, G and B	•				
Profile Type					
Normal Ramp (1D)	•	•	•	•	
Quick Ramp (1D)					
Normal Cube (3D)					
Quick Cube (3D)					•
Remote Probe					
Use Remote Probe					
Independent Output					
Do output indep. profiling			•	•	
LUT behaviour					
Clear LUTs	•	•			
Cine-tal Devices					
Profile Device					

5.1.2 Viewing Device Controls

Select the controls that are available on your viewing device. This helps cine-Profiler present you with the best procedure for optimising your viewing device.

The options available are:

- Brightness
- Contrast
- Bias R, G and B
- Gain R, G and B

Note that the terms *bias* and *gain* (in reference to RGB controls) are sometimes referred to as *brightness* and *contrast* for each colour channel.

5.1.3 Profile Type

Next you need to choose the profile type you wish to create. The profile type is made up of two parts:

- Ramp (1D) or Cube (3D)
- Normal or Quick

The *Ramp (1D)* profile only samples the primary colour ramps and the grey ramp, which is fine for display devices that decouple. If your device does not decouple, we recommend that you use a *Cube (3D)*. This type of profile will sample an even distribution of the total available stim space for that device.

The *Quick* and *Normal* tags refer to how many samples will be measured during the profiling. The more data that is captured, the more accurate the final transform.

	Ramp (1D)	*3D Cube*
Quick	51 frames	1000 frames
Normal	205 frames	4096 frames

In general, you should choose *Normal Ramp (1D)* for devices that decouple. For non-decoupling devices, the *Quick Cube (3D)* option is the most practical choice so that you can complete profiling within a reasonable time frame, whilst still obtaining accurate results.

5.1.4 Remote Probe

Select this option if you will be utilising the probeServer utility to facilitate communication between cineProfiler and the hardware probe. This may be used in cases where your workstation hardware is located some distance away from your viewing device or (where it does not support USB devices) and you need to communicate with the probe via a laptop computer. Please refer to Chapter 10 for more information on using probeServer.

5.1.5 Independent Output

Select this option if you want to use *output independent profiling*. It enables you to profile devices that are unable to display the cineProfiler GUI, such as broadcast monitors connected to an SDI output. For more information on this process, please refer to Section 5.8.

5.1.6 LUT behaviour

Selecting this option will clear the LUTs from the graphics card prior to profiling. Use this option if the profile you will be generating is intended for use within equalEyes. Otherwise, this option should be unchecked.

Application	*Clear LUTs*
equalEyes	yes
cinePlugins	no
cineCube	no

When you have selected the appropriate options on the initialisation window, ensure that your probe is properly connected to the workstation and then click the *Detect Probe* button.

5.1.7 Cine-tal Devices

Selecting this option will allow cineProfiler to directly access a Cine-tal monitor for profiling. The monitor will display it's own patches for profiling eliminating the need to complete a profile via *output independent profiling*. Currently this feature is for either the eL-1000 or the Cinemage monitor.

5.1.8 Probe detection fails

If cineProfiler fails to detect your hardware probe, check that you have correctly installed the device and that you have the appropriate drivers if needed – consult the manufacturer's documentation for your probe.

The other possibility is that the current user does not have the correct read-/write permissions to access the device. Changing user or device permissions will remedy the situation.

5.2 Calibrate probe

The probe calibration procedure is dependent on the type of probe you are using and the type of viewing device that is being profiled. On-screen information will instruct you on how to calibrate your probe, but please refer to the following tables to identify which surface you should use.

CRT

Probe	*Surface to use when calibrating*
Eye-One Display1	Dark (opaque) surface
Eye-One Display2	White patch displayed on monitor
LaCie blue eye 2	White patch displayed on monitor
Eye-One Pro	Lens cover closed
Hubble	Lens cover on

LCD

Probe	*Surface to use when calibrating*
Eye-One Display1	Dark (opaque) surface
Eye-One Display2	Dark (opaque) surface
LaCie blue eye 2	Dark (opaque) surface
Eye-One Pro	White *reference tile* (supplied with probe)
Hubble	Lens cover on

Digital projector

Probe	*Surface to use when calibrating*
Eye-One Pro	White *reference tile* (supplied with probe)
Hubble	Lens cover on

Since there can be slight colour variations across the screen, ensure that you attach the probe to the area of your monitor that you use most frequently for critical viewing. You can then move the colour patch underneath the probe for optimising and profiling.

Click the *Calibrate* button to continue.

5.2.1 Probe calibration fails

Most probes need to be placed on an opaque surface or in their own holder during calibration. Some of the supported probes (e.g. the GretagMacbeth Eye-One Display2), however, are designed to be calibrated on a white patch on-screen when used with CRT monitors. Primarily, this white patch is used by the probe to detect the monitor's refresh rate in order to obtain the most accurate results. Check that you are using the correct calibration conditions (Section 5.2). for your probe and try again. Additionally, check that you have selected the correct monitor type when starting cineProfiler – if the probe is expecting a CRT refresh rate but you are using an LCD monitor then it will fail!

5.3 Pluge

When entering this stage, a *lighting information* box will pop-up to remind you about some important details regarding ambient lighting conditions. The key points to be aware of here are:

- When profiling, ensure that the ambient lighting conditions are exactly the same as they will be when you are viewing images on the display.
- Ideally, the lighting conditions should be very similar to the conditions in which work produced on the monitor will be finally viewed. For film work, ambient lighting should be as dark as possible, and for video/TV work it should be dim but not too dark.

Click *OK* to continue.

The *pluge* will appear on the screen (Figure 5.2), showing a series of grey bars on a black background. This image is a visual aid that allows you to set the black level of the monitor correctly for your ambient light conditions. If lighting conditions differ, the black point will perceptually appear different and the amount of detail visible in the low luminance colours will vary. Using the pluge, then, each artist can set the black point based upon their perception and surrounding lighting conditions. This will then guarantee that all artists see the same amount of detail in the low luminance colours.

The pluge is moveable to cater for on-screen display controls that cover the grey bars and for dual monitor configurations where the pluge appears on the wrong monitor (drag it to the correct location).

The way in which you make use of the pluge will depend on the type of viewing device you are using. In most cases, you will need to set the viewing device to some initial values before setting the black point using the pluge. If, however, you have calibrated this device before and are simply updating the profile, please skip this section.

Note: When adjusting your monitor settings, note that the terms bias *and* gain *(in reference to RGB controls) are sometimes referred to as* brightness *and* contrast *for each colour channel.*

Figure 5.2: Pluge window

CRT monitors

Follow the steps below to adjust your monitor into a state ready for optimisation. For CRT monitors that do not have separate colour bias and gain controls, you should just set the contrast and skip the remaining steps.

1. Set the monitor's contrast to 100%.
2. Set the *gain* for the Red, Green and Blue channels to approximately 50%.
3. Adjust each *gain* channel until white looks *neutral*, trying to keep them roughly balanced around 50%.
4. Set the *bias* for Red, Green and Blue channels to approximately 50% (if you have separate bias controls).
5. Adjust each *bias* channel until white looks *neutral*, trying to keep them roughly balanced around 50%.

Then, to set your black level correctly using the pluge, follow these steps:

1. Increase the brightness of your monitor until five vertical bars can be seen on the black background.
2. Reduce the brightness gradually until the two outer bars *just* blend into the background, leaving only three vertical bars visible.

When finished, click *Next*.

LCD monitors

Many LCD monitors lack separate colour bias and gain controls. For such monitors, you will skip the optimisation step in cineProfiler and proceed directly to profiling. To adjust your monitor into an appropriate state for profiling, you can choose from the following options:

Factory default Reset your LCD display to the factory default settings. Often, this native state can work quite well.

Colour temperature Select a *colour temperature* on your LCD's OSD that is close to your desired target space.

Colour space Select a pre-defined *colour space* that is available for your LCD (e.g. sRGB).

For LCD monitors that *do* have colour bias and gain controls, performing the optimisation step *may* yield better results, although there are some exceptions for particular models. If you are using an LCD that has these controls, please try optimising the display first and then, if you find that it is producing unusual results, repeat the process using one of the preset options listed above.

Follow the steps below to adjust your monitor into a state ready for optimisation.

1. Set the monitor's contrast to 100%.

2. Set the *gain* for the Red, Green and Blue channels to approximately 50%.

3. Adjust each *gain* channel until white looks *neutral*, trying to keep them roughly balanced around 50%.

4. Set the *bias* for Red, Green and Blue channels to approximately 50% (if you have separate bias controls).

5. Adjust each *bias* channel until white looks *neutral*, trying to keep them roughly balanced around 50%.

Then, to set your black level correctly using the pluge, follow these steps:[1]

1. Increase the brightness of your monitor until five vertical bars can be seen on the black background.
2. Reduce the brightness gradually until the two outer bars *just* blend into the background, leaving only three vertical bars visible.

When finished, click *Next*.

Broadcast monitors

You should not change any settings on a broadcast device. If your broadcast device needs adjusting, please contact your local video engineer.

Click *Next*.

[1] If your LCD monitor does not have a brightness control then you should skip these steps.

Digital projectors

Many digital projectors are intended to only be adjusted by the manufacturer and the colour controls available are often different from those found on monitors. In many cases, then, you will not change any settings on the projector and simply proceed directly to profiling.

In cases where you are able to make adjustments to the device colour settings, you can choose from the following options:

Factory default Reset your projector to the factory default settings. Often, this native state can work quite well.

Colour temperature Select a *colour temperature* on your projector's OSD that is close to your desired target space.

Colour space Select a pre-defined *colour space* that is available for your projector (e.g. sRGB).

Manual settings To obtain optimal results from your projector, you should adjust the brightness and contrast settings in the following manner:

1. Reset your projector to its default settings.
2. Set the colour temperature (if possible) to the desired target colour space (e.g. 5500K).
3. Set the brightness of your projector using the pluge:
    (a) Increase the brightness of your monitor until five vertical bars can be seen on the black background.
    (b) Reduce the brightness gradually until the two outer bars *just* blend into the background, leaving only three vertical bars visible.
4. Display the projector contrast test image (Figure 5.3) by selecting 'Test Images' from the 'Images' menu in cineProfiler. Then, increase the projector's contrast. Note how the white center of the colour circle will get larger as you increase the contrast. Then, reduce the contrast step-by-step and stop at the moment when this white center becomes a point.
5. The contrast control on a projector unfortunately often changes the position of the black point. You many need to repeat steps 3 and 4 until both look good.

When finished, click *Next*.

5.4 Optimisation

Note: This step in cineProfiler will only appear if you have checked the RGB bias and/or gain control boxes at the initialisation screen. By default, this will

Figure 5.3: Projector contrast test image

only occur for CRT computer monitors.

The optimisation process (Figure 5.4) sets the monitor's white point to be the same as either the target profile or a user defined specification, ensuring that the monitor is in the best possible state prior to measuring a profile. During optimisation, cineProfiler will also ensure that the overall brightness is correctly maintained at the level that was set when using the pluge. Please refer to Section 4.2 for more details regarding the benefits of optimisation.

On the left hand side, there is a drop-down menu with two options that allow a user to choose how they specify their target white point:

- custom data (default)
- cineSpace profile

5.4.1 Custom data

The *custom data* option enables you to choose a specific white point value – defined by its luminance and chromaticity – to use during optimisation.

White Point - Luminance

On the right-hand side of the screen is an area labelled *White Point - Luminance*. You can select the standard units to be displayed as either candela per square meter (`cd/m^2`) or foot-lamberts (`ft-L`). Clicking on the drop-down box will enable you to select from a range of standard luminance levels.

Standard level	`cd/m^2`	`ft-L`
Film	47.97	14.00
Film (open gate)	54.82	16.00
Video 80	80.00	23.35
Video 100	100.00	29.19

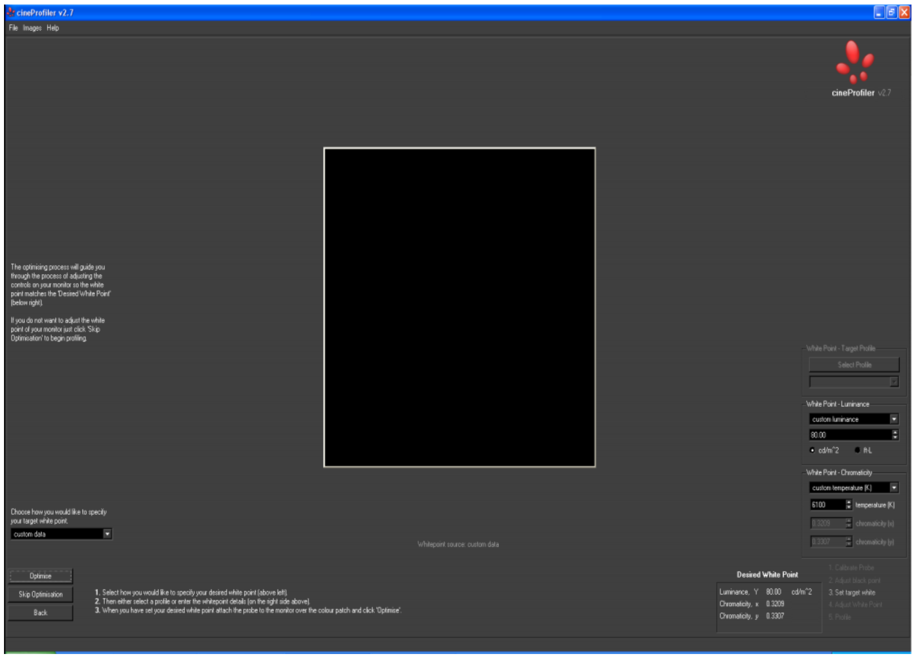


Figure 5.4: Optimisation Setup

If you wish to use a custom luminance value, select *Custom Luminance* from the drop-down box, and enter the value in the text box below.

White Point - Chromaticity

Also on the right-hand side of the screen is an area labelled *White Point - Chromaticity*. This option allows you to set the white point chromaticity as either a standard illuminant, a colour temperature (in degrees Kelvin) or chromaticity co-ordinates (x,y).

When choosing a standard illuminant, the following options are provided:

- A
- B
- C
- D50
- D55
- D65
- D75
- E
- F2
- F7
- F11

If you wish to use a custom chromaticity value, select from either:

- custom chromaticity (xy); or
- custom temperature (K)

You can then enter the desired value in the applicable text box(es).

5.4.2 cineSpace profile

This method allows the user to optimise the monitor (viewing) device's white point to be the same as an existing cineSpace profile. Selecting the *cineSpace profile* option for the target white point will open up a file browser window, allowing the user to select a cineSpace profile.

The target profile that you choose to optimise your monitor to will determine the colour and brightness of the white point of the monitor. For the best match from your monitor, you should always choose the target that you intend to use when working on the profiled monitor. For example, if you are about to do some film work and will be using the `kodak_2383.xml` target profile then select this profile as your target when profiling. This will set your monitor up to provide the best possible contrast and resolution when matching the target. If you will be using your monitor for a variety of work then we recommend that you choose the default target profile as it will adjust the monitor into a more general colour space better suited for use with a variety of target profiles.

When you have selected the target white point, observe the colour patch (initially black) at the center of the screen. If the patch is obscured by the OSD controls or is on the wrong screen in a dual monitor configuration then it can be moved. It can also be resized to make it easier to aim the probe when profiling projector displays, which may require a full-screen patch.

Attach the probe to the center of the patch, ensuring that it is sitting flat on the screen, and click *Optimise*. The screen will change (Figure 5.5) to display bias values on the left and gain values on the right [2].

Adjust the R, G and B *bias* on the monitor until the *bias* sliders all read 0 (between -2 and +2 is okay). Then adjust the R, G and B *gain* sliders on the monitor until the *gain* sliders all read 0 (between -2 and +2 is okay). If the bias levels move while doing the gain then redo the bias, and then the gain again – repeat until all sliders read 0 (between -2 and +2 is okay).

Note: If it is not possible to get all the *gain* sliders to 0 then ensure they are all the same and that the *bias* sliders are all set to 0 (between -2 and +2 is okay). Thus if the Red *gain* slider reads +23 and the monitor's gain is at 100% then just adjust the Green and Blue gains until both the Green and Blue sliders also read +23, or the text under the *gain* sliders reads *colour good* or *colour OK*. Turn off the OSD and check that the *gain* and *bias* sliders are still correct.

5.4.3 Tips on adjusting sliders

If you are having trouble getting all the sliders (*gain* and *bias*) close to zero, try adjusting the gain first. The gain setting on the monitor affects the *gain* sliders much more than the *bias* sliders, whilst the bias setting affects both sliders. Adjust the gain until the *gain* slider is at approximately the same value as the corresponding *bias* slider, then adjust the bias to bring both sliders close to zero – repeat for each slider (Red, Green and Blue).

[2] If your monitor does not have bias controls then a brightness slider will be displayed on the left instead of the bias values.

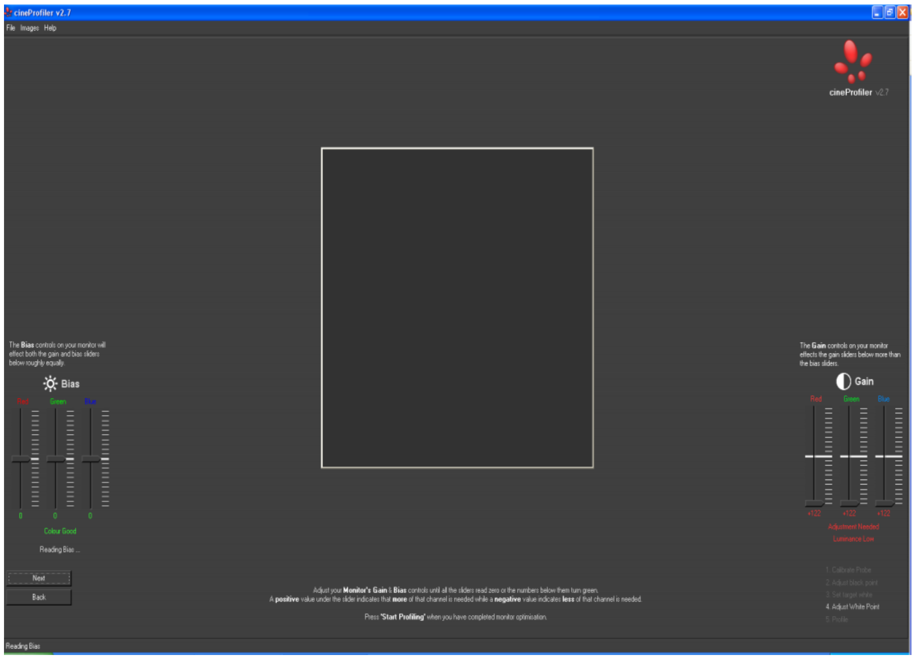


Figure 5.5: Optimising the viewing device

These steps will help reduce the time taken to reach ideal optimisation settings.

5.5 Profiling

Move the colour patch and probe to the screen area that you use most frequently for critical viewing and ensure that your screen-saver is disabled, then click the *Start Profiling* button. Sit back and relax while cineProfiler goes to work gathering information about how your monitor performs (Figure 5.6). This can take from 10 minutes (typical for a *Normal Ramp (1D)* profile) to 3 hours (*Full Cube (3D)*) depending upon the type of profile and the probe device being used. The progress bar on the bottom of the screen will show an approximation of how much time is remaining.

During profiling, ensure that the ambient lighting conditions remain consistent. Any changes to the lighting or probe position (or screen-saver activation) will most likely result in you having to repeat the profiling process.

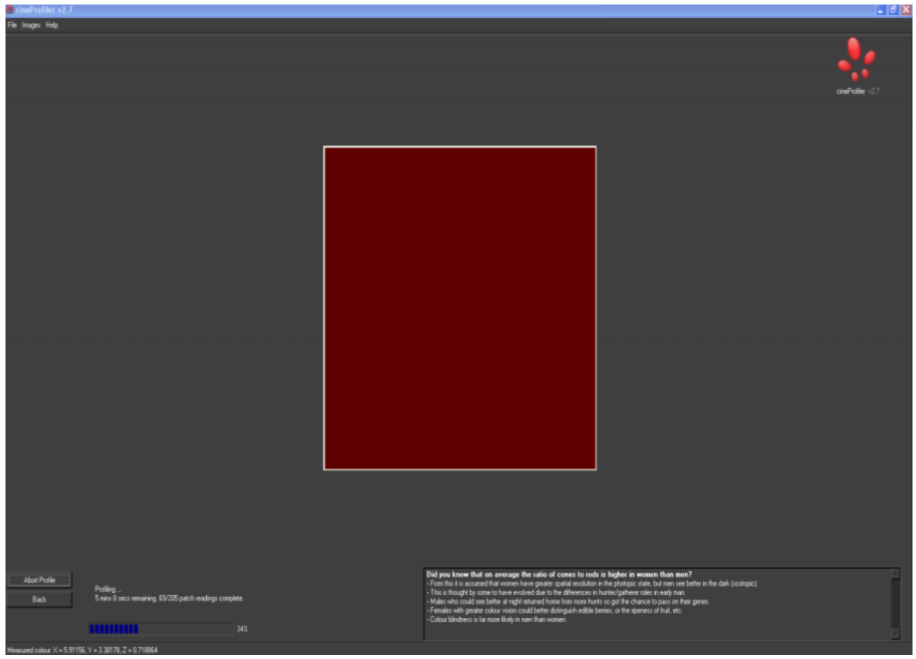


Figure 5.6: Profiling the viewing device

5.6 Decoupling information

When profiling is completed, the user will be presented with a graph (Figure 5.7) displaying the response of the primary curves and the decoupling error for increasing luminance.

If your display device does not decouple, this means that you may need to use 3D LUTs to achieve an accurate match to target profiles (due to the complexity of its colour response). We recommend profiling again in such cases, this time selecting a cube profile type, and using either cineCube or one of the cinePlugins when calibrating your display. Please refer to Appendix E for further details about decoupling error and non-decoupling devices.

5.7 Save profile

When cineProfiler has finished profiling the viewing device, it will prompt the user to name and save the profile (Figure 5.8). The default name should be the name of the machine being profiled. The default save location will be the `RSR_MONITOR_DIR` value set in the `rsr.conf` file. To make sure that equalEyes finds the profile automatically, we suggest you use the default name and location.

Ensure that no display settings are changed once you have completed optimi-

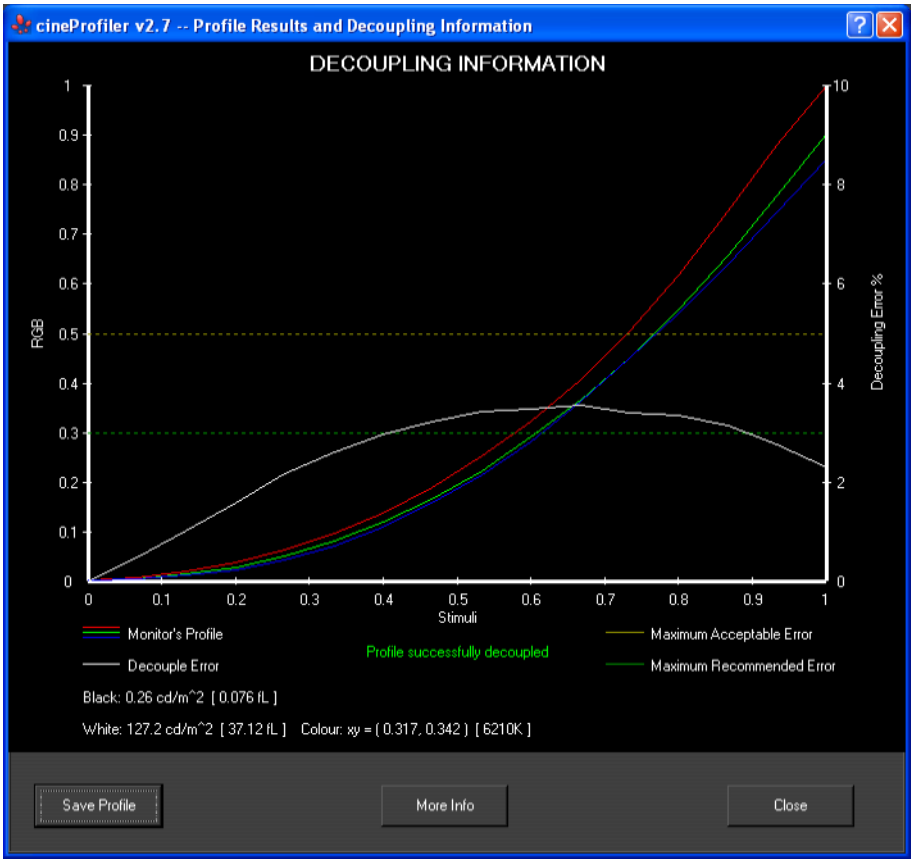


Figure 5.7: Decoupling information

sation and profiling, otherwise the device will need to be optimised and profiled again.

On completion, cineProfiler displays the name of the profile, and the path where it was saved (Figure 5.9). It also shows the time taken to profile the viewing device.

5.8 Output independent profiling

Output independent profiling (OIP) allows a user to profile devices that are unable to display the cineProfiler GUI. The most common case is when trying to profile broadcast monitors or projectors that derive their source from an SDI stream.

OIP enables you to send a series of test frames to the broadcast monitor via a third party application, with cineProfiler reading these patches by detecting

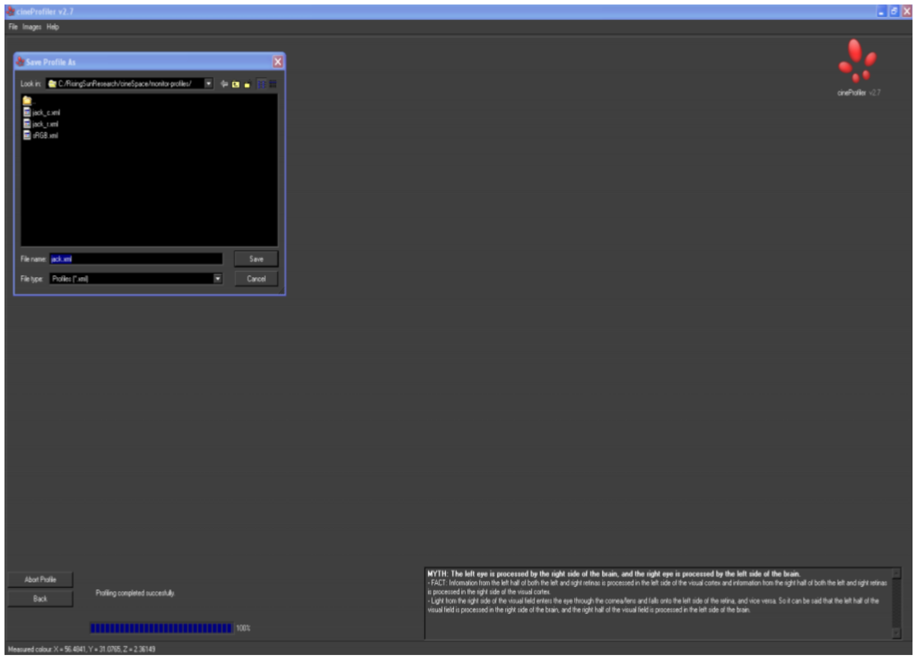


Figure 5.8: Save the profile

when they have changed colour, thus building a profile of the display device.

There are two available modes for OIP:

- Use timing
- Use error detection (default)

For most situations, you should be able to use the default *Use error detection* mode. If you have problems keeping the frames and cineProfiler in sync, then check the *Use timing* check box.

When performing OIP, follow these steps:

1. Decide what type (length) of profile you wish to create. We recommend doing a *Normal Ramp (1D)* profile when profiling a broadcast monitor, and use a *Quick Cube (3D)* profile for projectors.

2. Download the test frames for the length of profile you are creating. The frames are provided in TIFF, BMP and DPX format and can be obtained from:
`ftp.rsrhq.com`
`/images/OIPframes/OIPquickRamp_1D`
`/images/OIPframes/OIPnormalRamp_1D`
`/images/OIPframes/OIPquickCube_3D`

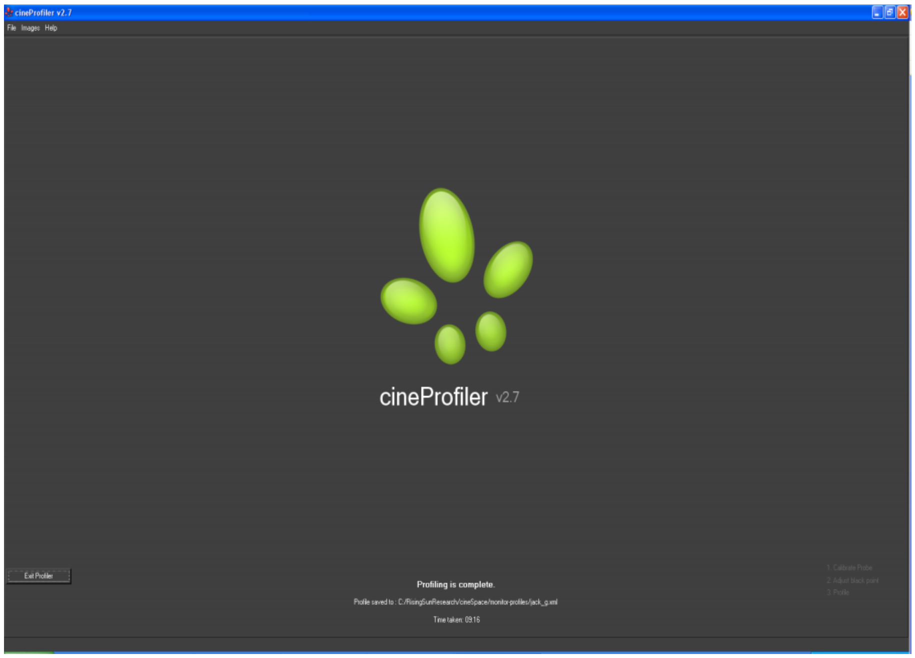


Figure 5.9: Profiling Complete

3. Unzip the test frames and load them into the application that you normally use to send frames to your broadcast monitor. You will need to set up the frames to play at a rate that will depend upon the type of probe you are using. Currently three probes are supported for doing OIP:

   - X-Rite **Eye-One Pro** and **Hubble**
   - Konica Minolta **CS-200**

   Refer to the following table to obtain the correct frame rate:

Probe	*Frames/second*	*Seconds/frame*
X-Rite Hubble	0.125	8
X-Rite Eye-One Pro	0.10	10
Konica Minolta CS-200	0.08	12

4. Once the frames are ready to play, run cineProfiler on another machine and select the *Output independent profiling* checkbox on the initialisation window.

5. Detect and calibrate the probe as usual.

6. For projectors, you will be given a chance to set your brightness and contrast. For broadcast devices, this step is skipped automatically.

7. You will then be presented with the *Setting up the test frames* page (Figure 5.10), which will confirm the instructions presented here in the manual. Enter the correct *Interval* corresponding to the frame rate (in seconds per frame) chosen in the application that will be playing the test patches, then click *Next* to continue.

8. Ensure that the first frame is being displayed (should be a black frame) and then place the probe on the device to be profiled. Click *Start Profiling*.

9. Once the probe starts taking measurements, press play on the application feeding the broadcast monitor to begin playing the test frames. cineProfiler will now measure the frames to create a profile of the broadcast monitor.

   Note: For each test patch displayed on the broadcast monitor, cineProfiler will display the corresponding patch in the application window. However, the patches will not *change at the same time – there will be a delay of a few seconds for each patch change. This is normal behaviour.*

10. When cineProfiler finishes, save the profile as usual.

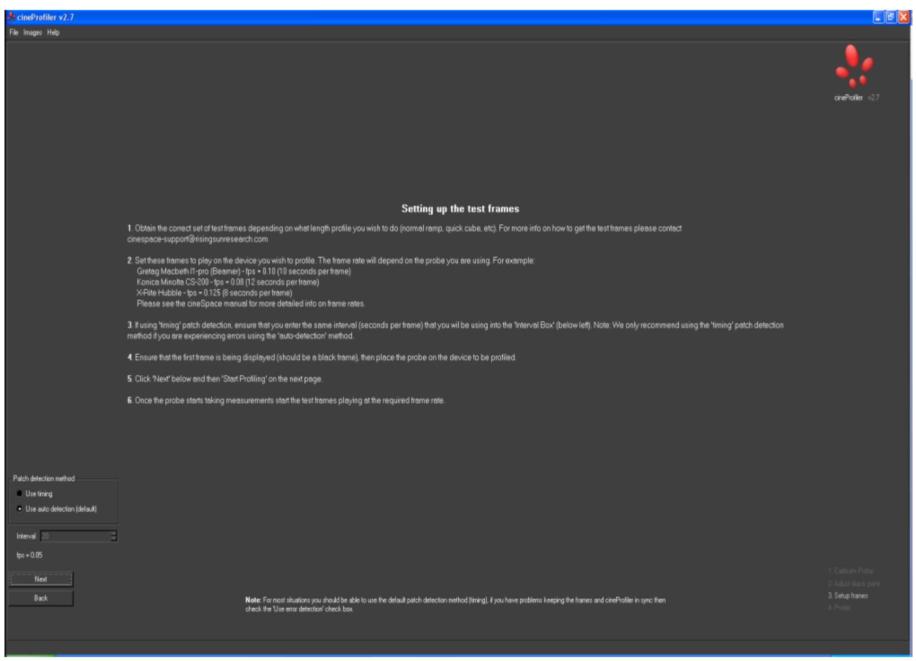


Figure 5.10: Output Independent Profiling

5.9 Profiling considerations

5.9.1 Environmental aspects

Ambient lighting

One of the most important parts of profiling is setting the ambient lighting conditions correctly. The ambient lighting should be set up to closely match the lighting conditions under which the work you will be doing on the monitor will finally be viewed. For TV/video/HD this is quite dim but not completely dark, for film it is just about completely dark.

Viewing angle

The colour, brightness and contrast of some monitors can vary significantly depending on viewing angle (particularly for LCD monitors). As such, you should ensure that the pluge is adjusted while viewing the monitor from a normal working position.

Warming up the viewing device

Many monitors, in particular CRTs, require time to warm up to a stable operating temperature. During this warm up phase there can be significant fluctuations in the colour response for an hour or more, thus when using a CRT monitor we recommend leaving it on for at least *two hours* prior to profiling.

Since the monitor's operating conditions change slightly every time it is switched on, you will need to *re-profile every time you turn off the power.* As such, you should disable any power-save feature on your monitor and leave it switched on permanently where possible. You may still use a screensaver to avoid burn-in, but disable it during profiling.

Whilst LCD monitors are less prone to such effects, we still recommend leaving them switched on at all times as for CRTs.

Projectors also require that they are warmed up prior to profiling. We recommend a minimum of one hour for warm-up, longer if possible.

Monitor movement

Since CRTs use magnetic fields to produce an image, they are affected by movement within the earth's magnetic field (or other localised fields, such as those produced by loudspeakers). Thus each time you move your monitor it can affect the colour balance and re-profiling will be required. Ensure you have your workspace organised so that your monitor does not need to be regularly moved.

5.9.2 When to re-profile

The recommended maximum age for a profile is dependent on the type of device you are using. The maximum length of time we recommend before re-profiling each type of device is given by the table below.

Device type	*Time*
CRT	14 days
LCD	28 days
Digital projector	3 months
Custom film print	6 months

You may modify the profile validity times from the recommended values by altering the relevant flags in your rsr.conf file (see Appendix A for more information). In addition to gradual changes, any of the factors discussed above can mean a need for re-profiling.

Any time you switch off or move your monitor or change the working ambient lighting conditions you should create a new profile.

5.9.3 Optimising

When using a monitor that has no physical controls for adjusting colour settings, you should skip the optimisation step of the process. Sometimes these monitors are controlled using the graphics card LUTs and since cineProfiler and equalEyes each clear the LUTs when they load, optimising this type of monitor will, in fact, ruin the profile that is created. Unless you know this isn't the case for your LCD don't try to adjust the brightness or contrast.

If you are optimising and find that your monitor's bias and/or gain cannot be adjusted correctly (e.g. you need to add more Red gain but your red gain is already at 100%) there are a few things you can try:

- If your monitor has a colour restore function, close cineProfiler, run *colour restore* and then try profiling again.
- While still in the optimisation page, adjust the brightness of the monitor in the direction you need (up if you need more Red gain) then adjust your *bias* sliders to 0 before adjusting the gain again.
  Note: As long as all the *bias* sliders are at 0 (between -2 and +2 is okay) then the black point you set on the pluge page will remain valid even if you have changed the brightness.
- Just set the *bias* sliders correctly and then adjust the R, G and B gains until the *gain* sliders are all level. Thus if the Red *gain* slider says +23 and your monitor's gain is at 100% then just adjust the Green and Blue gains until both the Green and Blue sliders also read +23, or the text under the *gain* sliders reads *colour good* or *colour OK*.
- If all else fails, it may be time to buy a new monitor!

5.9.4 Target luminance

Before commencing monitor optimisation, you are provided with the option of choosing the target luminance. This provides you with a way of modifying the overall brightness of your calibration to provide optimum viewing for the target profile or adjusting the display according to specific preferences. The luminance of your monitor can be set to the following options:

Film Sets a white point luminance of 47.96 cd/m^2 (14 ft-L)

Film Open Gate Sets a white point luminance of 54.82 cd/m^2 (16 ft-L)

Video Sets a white point luminance of 80 cd/m^2 (23.35 ft-L)

Video Sets a white point luminance of 100 cd/m^2 (29.19 ft-L)

Target Profile Sets the white point according to the target profile chosen when preparing the monitor for optimisation – this is the recommended option unless you have specific requirements.

Custom Allows you to specify the desired white point luminance (range 10 to 1000 cd/m^2).

There are three common scenarios regarding target luminance:

Match to *target profile* luminance

If, when profiling, you choose *cineSpace profile* as your target luminance and are able to adjust your monitor so that the gain sliders (in cineProfiler) all read zero (± 2) then it will be set to the white point of your target. If equalEyes or the cinePlugins are then used with *scaling off* then the white point of the monitor will be the same as that of the target profile. Thus if you want to match to the luminance of the target profile, this is the method to use.

Note that if you choose a target with a luminance greater than of the target used during profiling, clipping will occur and the white point will be completely inaccurate. If you are using a range of different target profiles, you should optimise your monitor using the target with the highest luminance.

Match *multiple monitors* or *video/HD*

If, when preparing the monitor, you choose *custom data* (and then leave the luminance setting at *Video 80* cd/m^2) and are able to adjust your monitor so that the gain sliders (in cineProfiler) all read zero (± 2) then it will be set to a white point of 80 cd/m^2. If equalEyes or the cinePlugins are then used with *scaling on*, the white point will be this bright no matter what target profile you use. Thus if you want to match brightness across monitors or do general video/HD work, this is the method to use. You can also work with *scaling off* if you want to view the comparative brightness of different targets.

Match to *standard film* luminance

If, when profiling, you choose *custom data* and select *Film 47.97* cd/m^2 as your target luminance and are able to adjust your monitor so that the gain sliders (in cineProfiler) all read zero (± 2) then it will be set to a white point with the same luminance as 'standard' film, i.e. 47.97 cd/m^2. If equalEyes or the cinePlugins are then used with *scaling on*, the white point will be this bright no matter what target profile you use. Thus if you want to match to standard film luminance, this is the method to use.

Chapter 6

equalEyes

equalEyes is an application for matching a display device (monitor) to a target device/medium (e.g. film or HD). It modifies the behaviour of the display device to resemble that of the target device so that when given an RGB stimulus the display responds with the same colour as the target.

This enables users to view images on their display device as they would appear when, for example, transferred to film. Without such correction the images prepared on your monitor will not look the same when viewed on the target device.

6.1 How equalEyes works

equalEyes uses two profiles to do its work. The first profile is the display device profile or *Monitor Profile*; the second is the target device profile or *Target Profile*. From these profiles, a set of look-up tables (LUTs) is generated and loaded onto the graphics card. These LUTs are calculated specifically to cause the output from the display device (monitor) to closely match the output from the target device/media for a given stimulus.

A set of standard profiles for different film stocks (Cineon, Kodak and Fuji) and video standards is included with cineSpace (see Appendix F). We can also create custom profiles for you as required by profiling a sample of your film printed using specific test frames. Please contact Rising Sun Research to arrange this service.

6.2 Limitations

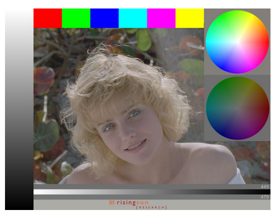
Because equalEyes uses a 1D transform and hence a 1D LUT for the intensity of each primary, differences in the chromaticities of the primaries (between display device and target device) cannot be accommodated, only changes in intensity. Similarly, cross-talk between the red, green and blue primaries (as occurs in film) cannot be simulated. In these cases the LUTs are chosen to match the grey ramps of both the display and target.

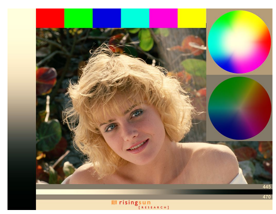
In general, this only affects users who would like to match to a film target; if you want to match to another monitor or one of the HD/video standards then using 1D LUTs is not an issue and equalEyes will give you an excellent match.

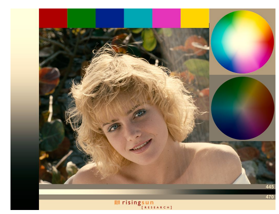
This limitation does, however, affect users working on LCD monitors and digital projectors, which are often non-decoupling devices. If the decoupling error measured during profiling is too high, it may not be possible to achieve an accurate match on your display device when using equalEyes. Please refer to Appendix E for further details on decoupling error.

The plug-ins section (Chapter 7) provides details on how to get more accurate results from within applications such as Shake and Digital Fusion by using our cinePlugins.

Examples with a normal Cineon file, the same file viewed with equalEyes and the file viewed through the cineEngine3D library used by the plug-ins can be seen below. For more information about the limitations of 1D LUTs, please see Section 4.7: Colour Transforms.


Unprocessed


equalEyes


cineEngine3D

6.3 Overview

All the commands in equalEyes can be accessed via different control pages. These pages can be accessed from the right-click menu, or by keyboard shortcuts.

6.3.1 Right-click menu

A right-click context menu is available on every page. To display the menu, simply right-click anywhere in the equalEyes application window (Figure 6.1).

Confidence Test This performs a test that the screen output is correct, for the current monitor/target profile settings, using a connected probe. For further details see Section 6.4.8, *Notes on Confidence Test*.

Opacity 50% (not in Linux) The application can be made semi-transparent to be less intrusive.

Minimise (not in OS X) Minimises the application.

EqualEyes Displays the main / front page.

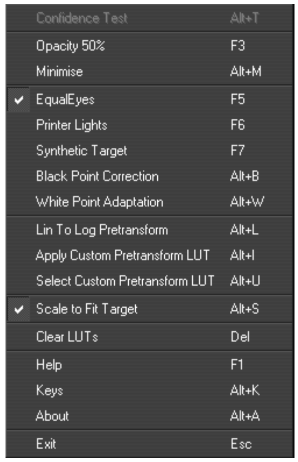


Figure 6.1: EqualEyes Right Click Menu

Printer Lights Displays the Printer Lights settings page. This allows you to grade the final look using printer lights. See Section 6.4.3, *Printer Lights* for more details.

Synthetic Target Displays the Synthetic Target settings page, allowing you to set a custom target profile. For further information, see Section 6.4.2, *Synthetic Profile*.

Black Point Correction Displays the Black Point Correction settings page. This is an advanced user page which allows precise control over the variables used in the black point correction algorithms. For information regarding these controls, see Section 6.4.4, *Black Point Correction*.

White Point Adaptation Displays the White Point Adaptation settings page. Here you can change the white point of your final display. See Section 6.4.5, *White Point Adaptation* for more details.

Lin To Log Pretransform Activates *Linear Mode* so that you can work with linearised film images (as opposed to Cineon/DPX log images). For information on the lin-to-log pre-transform, see Section 6.4.6.

Refer to the section on *lin-to-log transforms* (Section 4.13.1) for details on using this mode.

Apply Custom Pretransform LUT Allows you to apply a custom pre-transform LUT. For information regarding the custom pre-transform LUT, see Section 6.4.7, *Custom Pretransform LUT*.

Select Custom Pretransform LUT Allows you to select the LUT to use for the custom pre-transform. See Section 6.4.7, *Custom Pretransform LUT* for more information.

Scale to Fit Target When equalEyes is in scale mode (the default), the brightness of what you see is determined by the maximum brightness your monitor is capable of displaying. When scaling is off, the brightness is controlled by the brightness of the target profile. Usually you will want to use the full brightness range available for your monitor and thus have scaling on. However, when it is important that the *brightness* of nearby monitors match or to compare the brightness of target profiles you should use the non-scaling mode.

Note: Non-scaling mode can introduce gamut clipping artifacts – a pop-up warning will be displayed if your monitor is incapable of reproducing the target white point without clipping.

Clear LUTs Clears the LUTs (sets the LUTs to a gamma of 1) on the graphic's card. That is, the LUTs will have no effect on the video stream.

Help Opens the Help browser.

Keys Opens a window to display the keyboard short-cut keys.

About Shows the About page, displaying general information about the application and its configuration.

Exit Quits the application, and restores the graphic's card LUTs to the state they were in prior to running equalEyes.

6.3.2 Key Bindings

Confidence test	—	[Alt + T]	—
Opacity On	[F3]	—	—
Opacity Off	[F4]	—	—
Minimise	—	[Alt + M]	[Ctrl + M]
Show main page	[F5]	[Alt + E]	[Ctrl + E]
Show printer light page	[F6]	[Alt + P]	[Ctrl + P]
Show synthetic target page	[F7]	[Alt + G]	[Ctrl + G]
Black Point Correction	—	[Alt + B]	[Ctrl + B]
White Point Adaptation	—	[Alt + W]	[Ctrl + W]
Lin to Log Pretransform	—	[Alt + L]	[Ctrl + L]
Apply Custom Pretransform LUT	—	[Alt + I]	[Ctrl + I]
Select Custom Pretransform LUT	—	[Alt + U]	[Ctrl + U]
Scale to Fit Target	—	[Alt + S]	[Ctrl + S]
Clear LUTs	—	[Backspace]	[Delete]
Help	[F1]	[Alt + H]	[Ctrl + H]
Keys	—	[Alt + K]	[Ctrl + K]
About	—	[Alt + A]	[Ctrl + A]
Exit	[Esc]	—	—
Activate / Deactivate	[Space]	—	[Ctrl + Space]
Copy printer lights	—	[Alt + C]	[Ctrl + C]
Save LUT	[+ or =] then [1–9]		
Delete Saved LUT	[_ or -] then [1–9]		
Apply Saved LUT	[1 – 9]		
Return to last settings	0	—	—
Name the active saved LUT	[n]	—	—

Other key bindings may be defined by the operating system and window manager.

6.4 Controls

6.4.1 Front page

This page contains the primary functions of equalEyes. When you first load up equalEyes you will be presented with the main window (Figure 6.2).

Activating/Deactivating LUT

Press 'Spacebar' to toggle LUTs on and off.

The top button of equalEyes allows you to turn on and off monitor matching. The keyboard shortcut for toggling the LUTs is the 'space bar'.

Click to Apply LUT Loads the currently calculated LUT into the graphics card's LUT. If, at any point, you choose profiles or settings that your

Figure 6.2: Starting equalEyes

monitor is unable to display, a pop-up warning will appear and the display will revert to its original state.

Deactivate LUT Loads a uniform, linear LUT with gamma of 1.0 into the graphics card's LUT.

Note: When you exit equalEyes the LUT that was being used by the graphics card prior to running equalEyes is reloaded into the graphics card.

`on:` Monitor Profile

This is where you select the profile (generated by cineProfiler) of the monitor (viewing device) that you are using with equalEyes. When you click this button, equalEyes will attempt to guess the correct monitor profile for the display you are using. In some cases, however, it may need your help in locating the correct XML file. The default monitor profile can be set in the rsr.conf file. Information about the age of the profile being used will also be displayed after the profile name this button (see *profile validity* below).

`match:` Target Profile

This controls the current target profile. It is the profile of the desired target colour space that you want your display to match. If the current target is specified by an XML profile, this button will allow you to change the file used. If the current target is a *synthetic profile* (see below), clicking the button will take you to the synthetic profile page. The default settings for this button can be controlled by the rsr.conf file.

Profile validity

Both the monitor and target profile will have a coloured "light" to the right of their profile box. This provides a visual status indicator as to the current validity of the profile, with the different colours indicating the following:

Green The profile is valid.

Yellow Using this profile is not recommended – it is either too old or it does not decouple.
Red The profile is corrupt / invalid.

The recommended maximum age for a profile is dependent on the type of device you are using. The maximum length of time we recommend before re-profiling each type of device is given by the table below.

Device type	*Time*
CRT	14 days
LCD	28 days
Projector	3 months
Custom film print	6 months

You may modify the profile validity times from the recommended values by altering the relevant flags in your rsr.conf file. See Appendix A for more information.

6.4.2 Synthetic Profile page

Press F7 to open synthetic target settings

equalEyes provides a *synthetic profile* facility where you can specify the white point colour and tonal curve (gamma or Cineon) for your target rather than needing to load an XML file. Please refer to Section 4.6 for more information on synthetic profiles.

There are three ways to access the synthetic profile settings:

1. Click on the green "flask" (see Figure 6.3) and then press the target profile button;
2. Press F7; or
3. Right-click and select *Synthetic Profile*.

This will take you to the synthetic target settings pages.

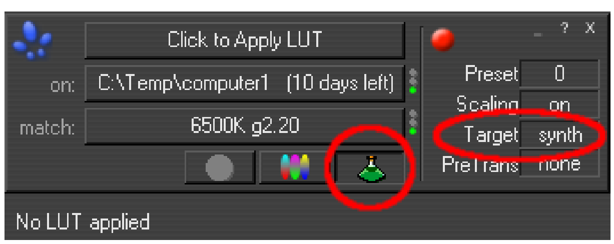


Figure 6.3: Opening Synthetic Profile Page

You can toggle between two synthetic target pages – Curves and White Point – using the drop-down menu on the upper left side of the window. The synthetic target options available on each page are explained in the sections below.

To go back to the main screen when you have completed your settings, either click the up-arrow, press F5 or right-click and select 'EqualEyes'.

Curves

You can select (Figure 6.4) between two tonal curve types:

- gamma curve
- Cineon curve

Figure 6.4: Synthetic Profiles Curves Page

When you select the gamma curve option, you can choose the synthetic target gamma value. The Cineon curve option generates a synthetic target based on a film-like ‘S-curve’.

White Point

This page allows you to specify your target white point colour in one of several different ways:

- As a chromaticity value (x,y);
- Using a correlated colour temperature (in degrees Kelvin);
- Using a standard CIE D-illuminant; or
- By matching the white point of your monitor (viewing) device.

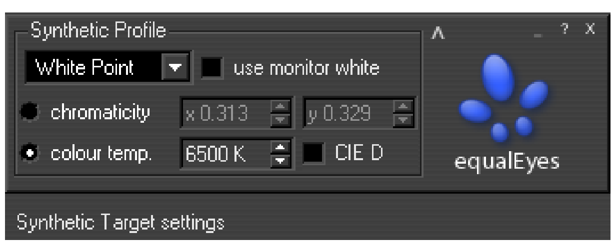


Figure 6.5: Synthetic Profiles White Point Page

Example

To set up a target profile of CIE D65 with gamma 2.2 in equalEyes:

- Open equalEyes and select your monitor profile (if not selected by default).
- Switch to the Synthetic Profile page by right-clicking and selecting *Synthetic Target* (or by pressing **F7**). You will be presented with Figure 6.4.
- Select the *Curves* page via the drop down box on the upper left hand side of the page.
- Check the *gamma curve* radio button.
- Enter a value of 2.2 into the gamma curve text field.
- Select the *White Point* page via the drop down box on the upper left hand side of the page.
- Ensure *use monitor white* checkbox is unchecked.
- Click the *colour temp* radio button.
- Check the *CIE D* checkbox.
- Enter 65 into the *colour temp* text field and press **enter**.

The monitor is now being matched to a CIE D65 illuminant with a gamma of 2.2. You can return to the equalEyes *main page* by using the right click menu or pressing **F5**.

Alternatively, you can set the synthetic profile values in the rsr.conf file so that you need only click the *Use Synthetic Target* button on the equalEyes main page to match to these settings. To do this, enter the following values for the corresponding cineSpace variables in the rsr.conf file:

```
RSR_SYNTH_CURVE_TYPE = GAMMA
RSR_SYNTH_GAMMA = 2.20
RSR_SYNTH_TEMP = 6500
RSR_SYNTH_WP_TYPE = temperature
RSR_SYNTH_TEMP_TYPE = CIE
```

6.4.3 Printer Lights

Press F6 for printer light settings.

The Printer Lights page provides controls that simulate the printer light settings on film recorders, so that you can make adjustments to the target profile if required (see Section 4.10 for further details). By default, there are eight (8) printer light points in one stop of exposure. The default values for these settings can be set in the rsr.conf file.

To open the printer lights page, you can click on the printer lights button on the main page (highlighted in Figure 6.6), press F6, or right-click and select *Printer Lights*.

The printer lights page is illustrated in Figure 6.7. You can adjust each printer light individually by clicking the up and down arrows next to each number. If you wish to maintain the overall brightness (densities) when adjusting the printer lights, click on the colour patches instead of the up and down arrows.

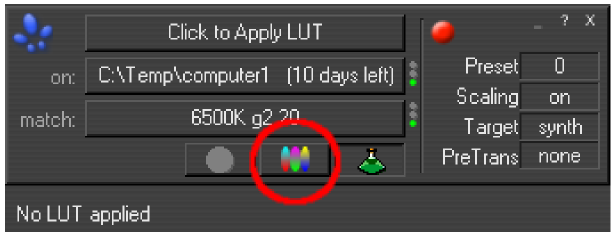


Figure 6.6: Opening Printer Lights page

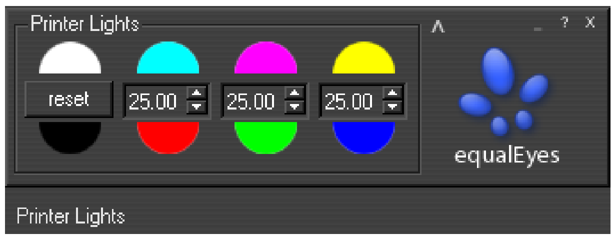


Figure 6.7: Printer Lights page

This will move the colour cast towards the colour of the patch, and automatically adjust the other numbers to maintain overall brightness.

If you wish to just adjust the overall brightness (exposure), you can do this by clicking on the black and white colour patches. Clicking on the reset button will restore the original printer light values.

Copy Printer Lights

It is possible to copy the printer light values, using Ctrl + C, and paste them into Shake to generate a node which will have the same effect as the printer lights.

6.4.4 Black Point Correction

Press ALT+B to open Black Point Correction Settings page.

The Black Point Correction page allows you to choose whether or not to activate the black point correction feature, and to choose the method (algorithm) used to perform the correction. For details on the black point correction feature, please refer Section 4.11.

To open the black point correction page, either right-click and select *Black Point Correction* from the menu, or press ALT-B. The equalEyes application will then change to show Figure 6.8.

By default, *On/Off* will be checked (this is recommended). This employs smoothing algorithms for the low luminance colours to reduce noise introduced

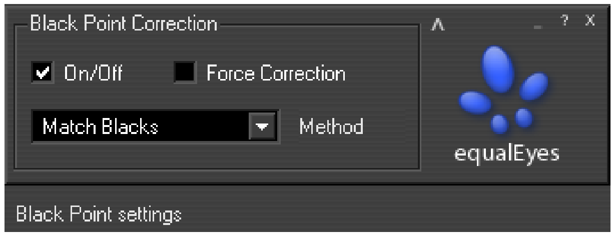


Figure 6.8: Black Point Correction settings page

from probe measurement errors, and compensates for the viewing device's lifted black point. There are four modes that may be used to perform black point correction:

- Match Blacks (default)
- Target Offset
- Nearest Gamut Edge
- Along Grey Ramp

The default mode is "Match Blacks", which will provide the best results in most situations. Appendix B provides a detailed explanation of the various black point correction methods and the situations in which you might select one of the other alternatives.

The *Always Correct* option causes the black point correction to be applied even if the target profile's black point is brighter than the monitor (viewing) device's black point. If this box is not checked, and the target's black point is brighter than the monitor's (viewing device), no correction is needed, and therefore no correction is done.

6.4.5 White Point Adaptation

Press ALT+W to open White Point Adaptation Settings page.

The White Point Adaptation page enables you to activate this feature and set your desired white point. For details on the white point adaptation feature, please refer Section 4.12.

Right-click and select *White Point Adaptation* or press ALT+W to get to the *White Point Adaptation* page (see Figure 6.9).

To activate white point adaptation, check the *Adjust target white point* box. You can then select your desired method of specifying the white point chromaticity from the option box below.

- Monitor: This sets the target's white point chromaticity to the monitor (viewing) device's white point chromaticity.

- Temperature: Choose this to specify the final white point in degrees Kelvin.

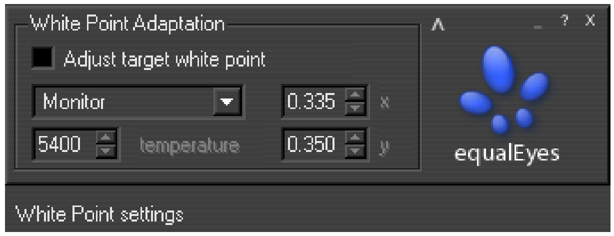


Figure 6.9: White Point Adaptation settings page

- Chromaticity: Allows you to specify the final white point using CIE chromaticity coordinates (x,y).

Press ALT+L to toggle lin-to-log pretransform mode.

6.4.6 Lin-to-Log Pretransform

To activate the lin-to-log pretransform, either right-click and select *Lin to Log Pretransform*, or press "ALT+L". When selected, the PreTrans area on the main-page will show "lin2log" as highlighted in figure Figure 6.10.

Figure 6.10: Lin-to-log pretransform selected

Note that the lin-to-log parameters can be set in Section 5 of the rsr.conf file.

Press ALT+I to toggle custom pretransform LUT

6.4.7 Custom Pretransform LUT

equalEyes allows you to specify a custom pretransform LUT that can be used in a similar manner to the lin-to-log pretransform. To use the custom pretransform LUT:

1. Select the LUT by pressing ALT+U or clicking *Select Custom Pretransform LUT* from the right-click menu.
2. Choose your custom pretransform LUT file from the file browser.
3. Apply the LUT by pressing ALT+I or right-clicking and selecting *Apply Custom Pretransform LUT*.

Figure 6.11: Custom pretransform LUT on main page

When the custom pretransform LUT is being used, the PreTrans area on the main page of equalEyes will display "lut" as shown in Figure 6.11.

For detailed information about creating your own Custom Pretransform LUT, see Appendix D. cinLUT

6.4.8 Confidence Test

The Confidence Test provides a quick method for checking the validity of your current monitor profile. It compares the expected values for each colour with the actual values being displayed and calculates an error measurement (CIE Lab94 error).

To start the Confidence Test, you need to ensure that you are currently applying the LUT. You will then be able to click the "target " button as highlighted in Figure 6.12) to start the test.

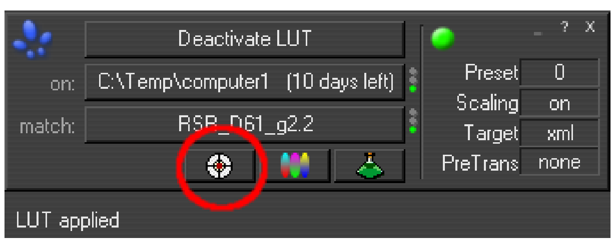


Figure 6.12: Confidence Test button

When starting the Confidence Test, a new window will open that will prompt you to enter your monitor (viewing display) type and probe server details if needed (see Figure 6.13).

Press the *Start* button once you have selected your settings. *Detect* the probe, then *Calibrate* the probe following the instructions on the screen. Next hit the *Start Test* button to start the confidence test. This will take approximately a minute.

At the end of the confidence test you will be presented with your monitor

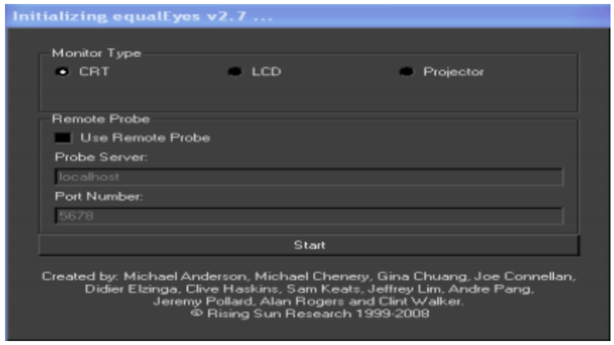


Figure 6.13: Confidence Test start-up

(viewing) device's confidence test results. This includes a graph that shows the CIE Lab94 error (ΔE) for different points along the grey ramp. A statement at the bottom of this chart will inform you whether the monitor passed (green writing) or failed (red writing) the test.

If the confidence test passes, this means that your monitor profile is valid.

If the confidence test fails, try re-profiling your monitor with cineProfiler and then running the confidence test again in equalEyes. If the test still fails, please contact Rising Sun Research by sending an email to cinespace-support@rsrhq.com with your monitor profile.

6.4.9 Presets (saved states)

You can save the current state of equalEyes and restore it at a later time.

Save a preset Press the +/= key followed by the number [1–9] of the preset you want to store the current settings in.

Load a preset Press the number key of the preset you would like to load.

Rename a preset While a preset is loaded press the 'N' key then enter the name you would like to give to the preset and click the 'Set Name' button.

Unload a preset While a preset is loaded press the '0' key to unload the preset and return to the previous settings.

Delete a preset Press the _/- key followed by the number [1–9] of the preset you want to delete/clear.

Saved states will stay set even when equalEyes is shutdown and restarted.

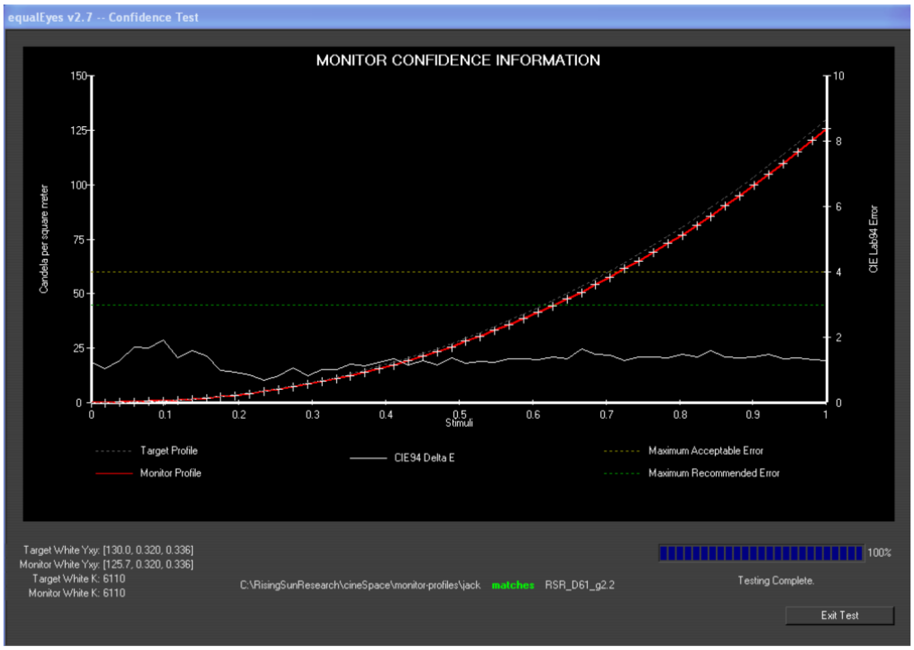


Figure 6.14: Confidence Test results

6.4.10 Clipping Error

A *Clipping Error* warning (Figure 6.15) occurs when some values in the target profile are brighter than the monitor can display. There are two ways to address this situation, each with its own pros and cons.

1. With clipping, any values in the target that are brighter than the monitor can display instead use the brightest displayable value. This results in all values below this threshold being correct, but brighter colours being distorted.

or

2. With scaling, the brightness of the target is reduced so that the relative brightness of all output colours are correct, but the absolute values will be darker than the target profile (see Section 4.9 for further information on scaling).

Often this problem occurs when the monitor physically can't display colours as bright as requested (such as trying to match an LCD display with a CRT). Sometimes increasing the monitor brightness and re-profiling can fix this.

In particular this can occur when using a dark target profile when optimising with cineProfiler. What happens is that cineProfiler tells you to reduce the

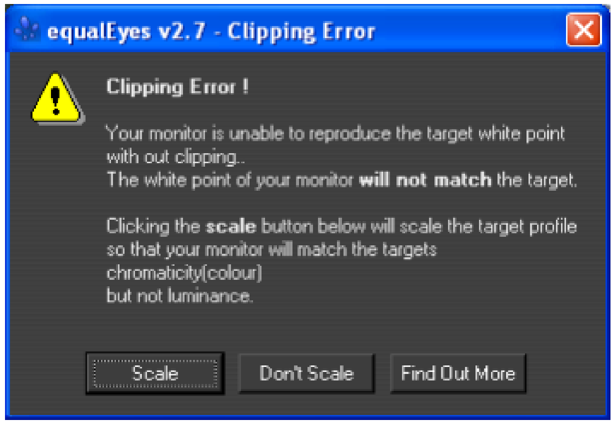


Figure 6.15: Clipping Error warning

global brightness of the screen down to match the target requested. If the screen is re-optimised using a brighter target profile (or so that the brightness levels match at higher than 0 in the profiler) these clipping errors may go away.

6.4.11 Decoupling Warning

If you load a monitor (viewing device) profile that does not decouple, equalEyes will present you with a *Monitor Decouple Warning* (Figure 6.16). This means that your monitor may not be able to match the target device using 1D LUTs only.

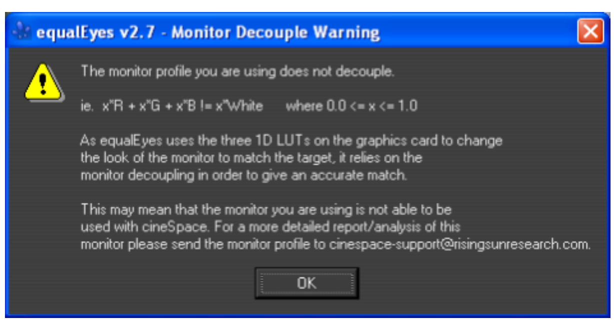


Figure 6.16: Decoupling warning

To test whether your monitor can match the target device, please run the confidence test. If the confidence test passes, then your monitor can match the intended target. If, however, the confidence test fails then you may need to capture a cube profile and use either the cinePlugins or cineCube in order to obtain an accurate match.

Please refer to Appendix E for details on decoupling errors and non-decoupling devices.

6.4.12 Disable GUI

Some facilities want to set up equalEyes through the rsr.conf file and, when it is running, stop the artists from being able to change the settings. This can be easily done by finding the following section in your rsr.conf file and setting it to true.

```
# RSR_EE_INACTIVE_GUI: If TRUE sets whether Users can interact with the
# EqualEyes gui.  This is for users who would like to have all EqualEyes
# settings set by the rsr.conf file and /or presets and does not want
# other users to be able to change any of the settings.
# Note:  If this option is set to TRUE then RSR_EE_INIT_SIZE will default to
# MINIMISED and RSR_EE_INIT_STATE will default to ACTIVE.
# Range TRUE or FALSE. Defaults to FALSE if not set.
#
RSR_EE_INACTIVE_GUI = TRUE
```

6.5 Dual monitors

equalEyes can be used with dual monitor configurations, but there are some limitations to bear in mind. Due to the way in which most systems support dual monitors, there are two possible scenarios:

- A single LUT is used for both monitors with one copy of equalEyes running (using a single monitor profile). Applications and windows can be dragged between and spread across both monitors, but only one monitor will be correctly matched to the target profile.
- Two separate LUTs are used (one for each monitor) with two copies of equalEyes running (using the correct monitor profile for each display). Both monitors will be correctly matched to the target profile, but applications and windows cannot be dragged between or spread across both monitors.

We currently recommend the first option, with one monitor being used for viewing images and the other for displaying a "tools palette" (anything that does not require accurate colour). Both monitors will be optimised for viewing the type of target you are using, but only one will precisely match the colour response.

The following steps outline the process:

1. Run cineProfiler on the left monitor and save the monitor profile.
2. Run cineProfiler on the right monitor and save the monitor profile.
3. Start equalEyes and choose the correct target profile. Select the monitor profile for the left monitor and activate the LUT.
4. Create a preset for the left monitor (e.g. '1').
5. Select the monitor profile for the right monitor and create a second preset (e.g. '2').

6. Layout your workspace with your images on one monitor and tools on the other. Choose the equalEyes preset corresponding to the monitor displaying your images so that the colours will be correctly displayed. If you wish to switch display matching to the other monitor, simply use the other preset.

6.6 Troubleshooting

6.6.1 Linux

When switching applications under Linux the LUTs are reset sometimes. This can occur when there is a graphics configuration issue with your XF86Config file. To fix this, edit your XF86Config file to correspond closely with the following:

```
Section "Device"
        #VideoRam    131072
        Identifier  "Videocard0"
        Driver      "nvidia"
        VendorName  "Videocard vendor"
        BoardName   "NVIDIA Quadro FX (generic)"
        Option      "NvAGP" "3"
        #Don't use overlay. It wll cause cineSpace to fail if Maya is open
        #Option      "Overlay" "1"
        #Option      "CIOverlay" "1"
EndSection
```

(Thanks to Jami Levesque for this fix.)

Chapter 7

cinePlugins

The cineShake, cineFusion, cineNuke and cineFilmMaster plug-ins provide a way to access the features of equalEyes, as well as the complete 3D colour transforms required for reproducing film behaviour, from within their respective applications. The ability to apply 3D transforms means it is possible to achieve an accurate match when using non-decoupling monitor or target profiles. This makes the cinePlugins ideal for working with film images or for use with LCD monitors and digital projectors.

7.1 Correct usage

When using equalEyes, it first clears the LUTs before loading the appropriate values for your chosen monitor and target profiles. This provides some certainty regarding the state of your display. Without equalEyes, it is possible that another application is controlling the LUT values (e.g. system display settings on Mac OS X). Since the plug-ins assume that the LUTs are flat (i.e. no correction) when calculating the required colour values, there is a potential for errors to arise.

7.1.1 Using equalEyes with cinePlugins

One way to remedy this situation is to run equalEyes while using the plug-ins (unless you are *certain* that the LUTs are flat):

1. In equalEyes, select your monitor profile and a standard video profile (such as sRGB) for the target profile.
2. In the plug-in, set the monitor profile as the standard profile you chose above (e.g. sRGB) and select the desired target profile.

Using equalEyes in this way ensures that the LUT values are known. However, it does require two separate sets of calculations to determine the colour values. As such, if you are certain that your LUTs are flat then it is preferable to use the plug-in on its own.

7.1.2 Using cinePlugins without equalEyes

Without equalEyes running, there are no guarantees that the LUTs are cleared when using the compositing application with the cinePlugin. Where this becomes a problem is that in general, when your monitor is profiled using cineProfiler, the LUTs are cleared prior to profiling.

The solution to this problem is to uncheck the *Clear LUTs from graphics card* box on the initialisation screen when starting cineProfiler, which prevents the LUTs from being cleared prior to profiling. You can then safely use monitor profiles generated from cineProfiler with this newly adjusted flag, as it encapsulates any LUTs that are present during default use of your computer.

7.2 cinePlugin parameters

The cinePlugins provide similar settings to those found in equalEyes[1], with the addition of the ability to perform full 3D colour transforms. The following sections outline the various settings available within the cinePlugins.

7.2.1 Profiles

MonitorProfile

Select the profile for your monitor (viewing) device.

TargetProfile

Select the profile of the device / colour space you would like to match.

Invert

Select to produce an inverse transform where the cineSpace plugin has internally switched the monitor and target profiles. Inverse transforms can be used to apply a data transform to your images so that they reproduce correctly when sent to a different output medium or colour space.

(NOTE: This feature is not enabled in the standard cineSpace suite, but is available for purchase as an optional extra. If this feature is used without the correct license the processed image will be displayed with a strong green cast.)

7.2.2 cineEngine

cineEngineType

Default: 3D Quick
⟨*1D / 3D Quick / 3D Accurate*⟩

Select the cineEngine transform type that you wish to use (refer to Section 4.7).

- **1D:** Uses a 1D transform, producing the same result as equalEyes.
- **3D Quick:** Uses a lower resolution transform which is quicker, but less accurate.
- **3D Accurate:** Uses a high resolution transform which is slower, but accurate.

[1] Due to the way in which the various cineSpace applications are implemented, the feature set is not perfectly consistent across the entire suite.

scalingType

Default: Scaling
⟨Scaling / None⟩

Select whether or not to perform luminance scaling. For more information see Section 4.9: *Scaling*.

- **None:** No scaling will be done. This will match the luminance as well as the chromaticity of the target white point.
- **Scaling:** This will scale the target profile to use the full dynamic range of the monitor (viewing device). We recommend leaving this active, unless you are an experienced user.

scaleFactor

Default: 1

This double value is used as a scaling factor to ensure that the target and monitor gamuts line up as optimal as possible. Decreasing this value decreases the overall brightness, but increases the overall quality of the colours (reduces the amount of out-of-gamut colours). Recommend values are in range from 0.3 to 1.0

gamutTreatment

Default: None
⟨None / Alpha / Green⟩

Highlights the colours in an image that cannot be correctly represented on the monitor (viewing) device (refer to Section 4.14).

- **None:** No out-of-gamut colours will be highlighted.
- **Alpha:** The pixel of any colour that would be out of gamut has its alpha channel set to zero.
- **Green:** Any colours that would be out of gamut are displayed as green.

Pretransform

Default:None
⟨None / LinToLog / LUT⟩

Applies a pretransform prior to the main cineEngine colour transform (refer to Section 4.13).

- **None:** No pretransform is applied.
- **LinToLog:** Select this for working with linear-to-light images while using film (log) target profiles.
- **LUT:** Select this option to apply the selected custom pretransform LUT file.

7.2.3 Advanced - White Point Adaptation

Do white point transform

Default: Off
⟨Off / On⟩

Toggle white point adaptation on or off. For more information, see Section 4.12.

- **Off:** No white point adaptation is done.
- **On:** White point adaptation occurs.

Target

Default: Monitor
⟨Monitor / Colour Temperature (K) / Chromaticity (xy)⟩

Set the desired target white point.

- **Monitor:** Sets the white point to the monitor (viewing) device's white point temperature.
- **Colour Temperature (K):** Set the final white point temperature in degrees Kelvin.
- **Chromaticity (xy):** Set the final white point in CIE chromaticity coordinates (x,y).

Temperature (K)

Default: 5400K
⟨4000K⟷9000K⟩

Set the desired white point colour temperature (in degrees Kelvin) for the target.

Chromaticity (x)

Default: 0.3349
⟨0.0⟷1.0⟩

Set the desired white point CIE x chromaticity coordinate.

Chromaticity (y)

Default: 0.3497
⟨0.0⟷1.0⟩

Set the desired white point CIE y chromaticity coordinate.

7.2.4 Advanced - Black Point Correction

Do black point correction

Default: On
⟨On / Off⟩

Toggle black point correction on and off. For more information, see Section 4.11.

- **On:** Black point correction is done.
- **Off:** No black point correction occurs.

Method

Default: Match Blacks
⟨*Match Blacks / Along Grey Ramp / Nearest Gamut Boundary / Target Offset*⟩

Select the black point algorithm to be used. For more information, see Section 4.11.1.

- **Match Blacks:** The target's black point will be mapped to monitor's black point.
- **Along Grey Ramp:** Offset the target profile along the grey ramp, to the edge of the monitor (viewing) device's gamut.
- **Nearest Gamut Boundary:** Offsets the target profile to the nearest point in the monitor (viewing) device's gamut.
- **Target Offset:** The distance and direction the target profile is away from 'real black' will be moved the same distance and direction from the monitor (viewing) device's black point.

Force Correction

Default: Off
⟨*On / Off*⟩

Toggle whether black point correction is to be done, even if not required.

- **Off:** Black point correction works normally (recommended).
- **On:** Forces black point correction to be done, even if not required - the target profile's black point is brighter than the monitor (viewing) device's black point.

7.3 cinePlugin warnings

The cinePlugins provide feedback on error conditions by changing the colour of the image.

7.3.1 Blue image

If a cinePlugin outputs a *blue* frame (Figure 7.1) it means that the cinePlugin cannot find a license. Check that your `RSR_APP_DIR` environment variable is pointing to the directory that contains the `rsr.conf` file. Check that the `RSR_LICENSE` setting in the conf file is set to the full path and name of the license server (or file name of the node-locked license file).

Figure 7.1: cinePlugin blue image

7.3.2 Red image

If a cinePlugin outputs a *red* frame (Figure 7.2) it means that the cinePlugin does not have two valid profiles (monitor and target) needed to perform a colour transform. Ensure that both your monitor and target profiles are valid.

Figure 7.2: cinePlugin red image

7.3.3 Green image

If a cinePlugin outputs a *green* frame (Figure 7.3) it means that the cinePlugin is trying to perform an invalid transform, for example trying to complete an invert transform without the correct license. Ensure that if you are using a cube profile for your monitor profile it was generated by cineProfiler. For more information about this warning, please contact RSR.

Figure 7.3: cinePlugin green image

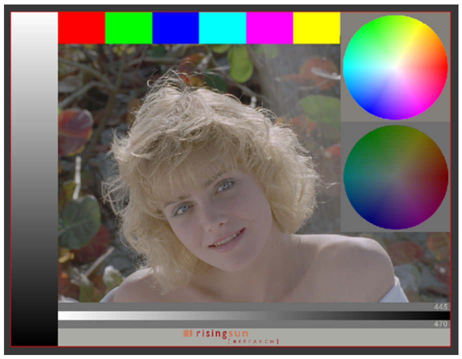
7.3.4 Yellow image

If a cinePlugin outputs a *yellow* frame (Figure 7.4) it means that the cinePlugin either is unable find the pretransform (E.G. incorrect path) or it is trying to load an invalid file.

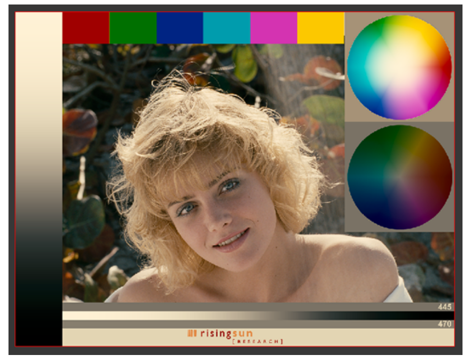
Figure 7.4: cinePlugin yellow image

7.4 cinePlugin results

A typical result for the plug-ins is shown below:



Original



cineSpace calibrated

7.5 cineShake

cineShake is a plug-in to Shake that provides the functionality of both equalEyes and full 3D colour transforms as either a node in the Shake tree or as a viewer script. cineShake will work under Mac OS X, and Linux Fedora Core (and some other glibc 2.3 based systems) with Shake 4.0 and 4.1.

7.5.1 Using cineShake

Using cineShake as a node

- Open Shake.
- Start cineShake (Figure 7.5) by selecting the RSR tab and clicking on the cineShake icon.

Figure 7.5: Starting cineShake

- Load an image and connect to the cineShake node (Figure 7.6).

Figure 7.6: cineShake node

- Select the profiles you would like to use (Figure 7.7). Use a recent monitor profile generated from cineProfiler for your monitor and select target profile of the colour space you would like to match.

Using cineShake as a viewer script

- Open Shake

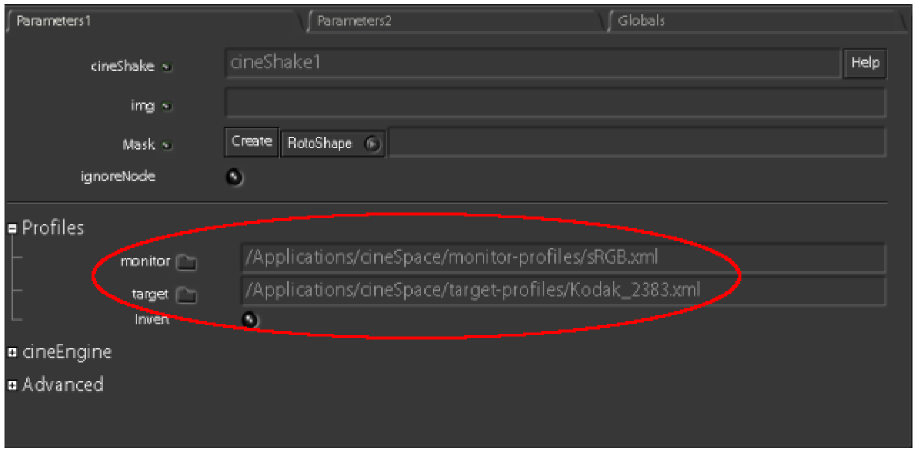


Figure 7.7: Select monitor and target profiles

- Activate cineShake by selecting the cineSpace icon from the viewer script menu (Figure 7.8). This will apply the cineSpace 3D LUT to all images loaded in the viewer window.

Figure 7.8: Start cineShake viewer script

7.5.2 Control Panel

Whether you use cineShake as a node within your Shake tree or as a viewer script, the control settings (Figure 7.9) have the same effect. For detailed information about how these controls work, please see Section 7.2.

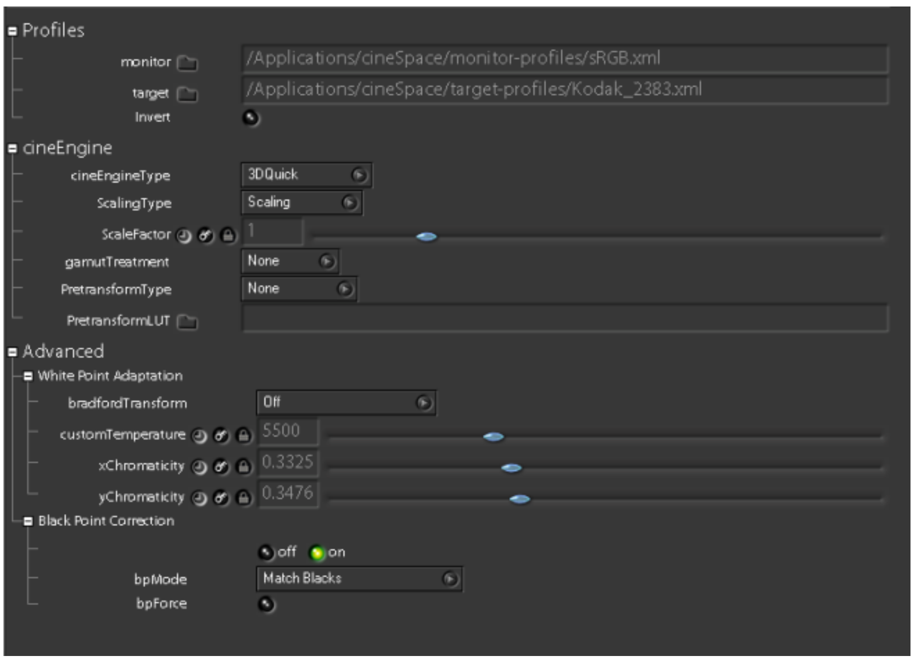


Figure 7.9: cineShake control panel

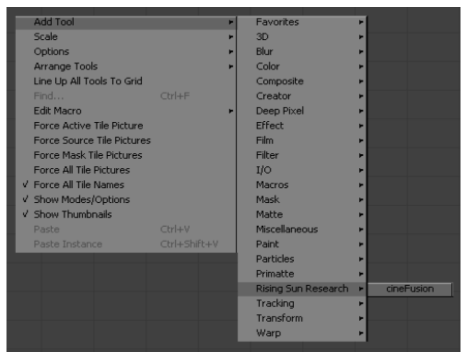
7.6 cineFusion

cineFusion is a plug-in to Digital Fusion that provides the functionality of both equalEyes and full 3D colour transforms as a node within Fusion.

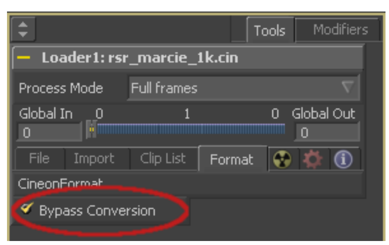
There are versions of cineFusion that support Digital Fusion versions 5.0 and 5.2. These are only available on the Windows platform.

7.6.1 Using cineFusion

- Open Digital Fusion v5+
- Load cineFusion tool: `Tools>Rising Sun Research>cineFusion`



- Load an image and connect it to the cineFusion node. If you are using Cineon images be sure to check the *Bypass Conversion* (Fusion 5) checkbox under the Format tab in the loader.



- Select the profiles you would like to use. Use a recent monitor profile generated from cineProfiler for your monitor and select a target profile of the colour space you would like to match.

7.6.2 Controls

Below is a quick snap-shot of the cineFusion control panel. For detailed information about how these controls work, please see Section 7.2.

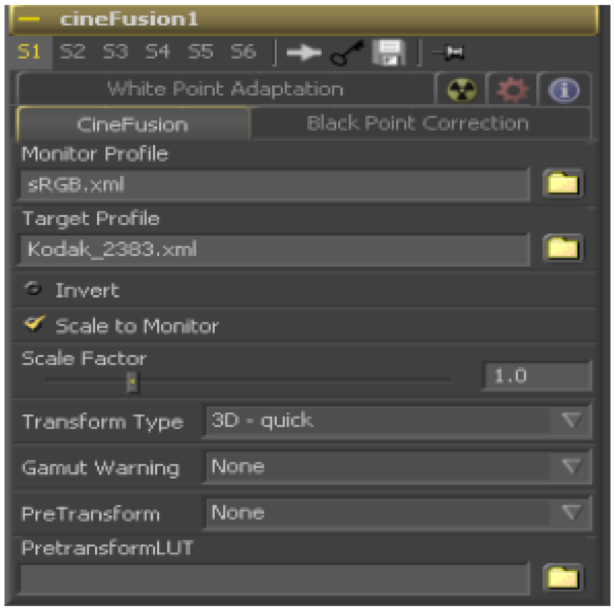


Figure 7.10: cineFusion main control panel

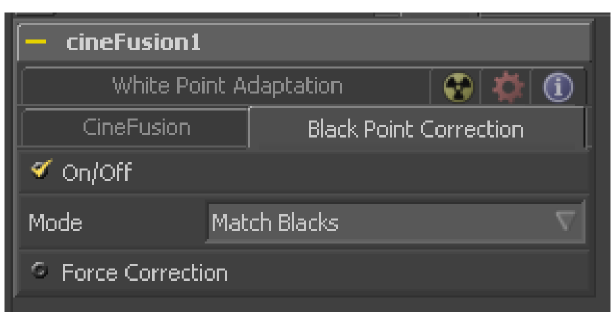


Figure 7.11: cineFusion black point control panel

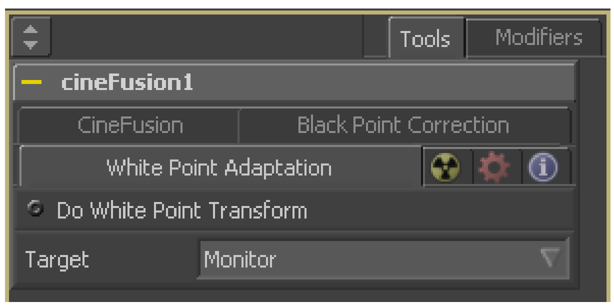


Figure 7.12: cineFusion white point adaptation control panel

7.7 cineNuke

cineNuke is a plug-in to Nuke that provides the functionality of both equalEyes and full 3D colour transforms as a node within Nuke.

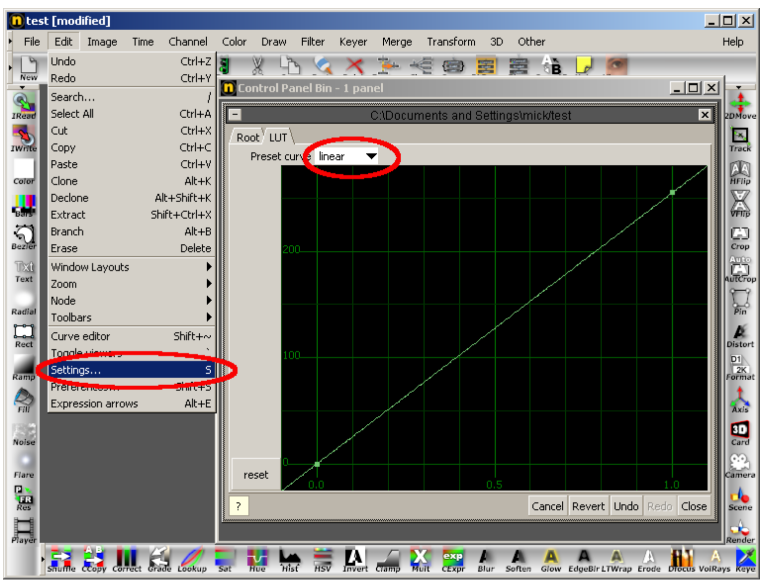
There are versions of cineNuke available to support Nuke v4.7, Nuke v4.8 and Nuke v5.0 for Linux, Windows and OSX.

7.7.1 Using cineNuke

Nuke applies an sRGB LUT by default. To use cineNuke this LUT needs to be disabled. To do this depends which Nuke version is running.

Nuke v4.7

- Click edit → settings
- Select the 'LUT' tab
- Change the 'Preset curve' from sRGB to linear



Nuke v4.8

- Click edit → settings
- Select the 'LUT' tab
- Change the 'Viewer' from sRGB to linear

(Figure 7.13)

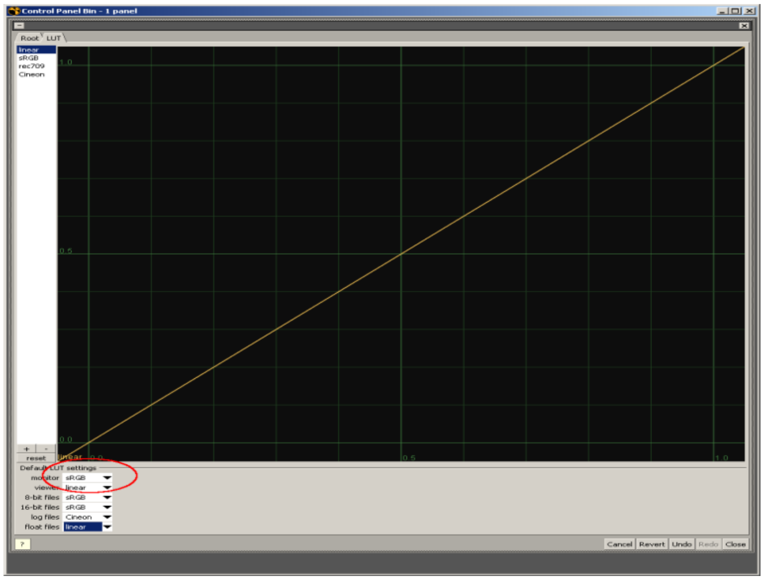
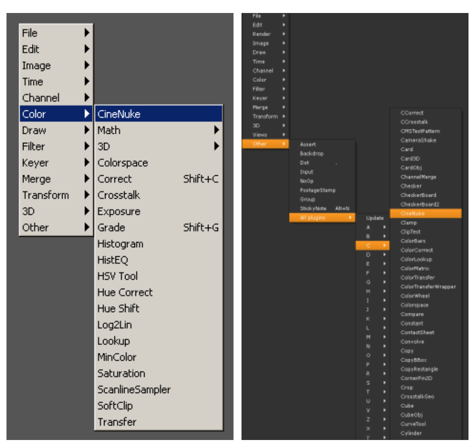


Figure 7.13: cineNuke set Nuke4.8 View LUT

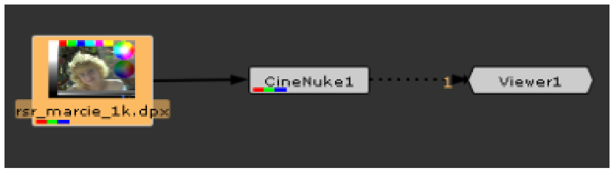
Nuke v5.0

Select 'linear' from the drop down box on the tool bar.

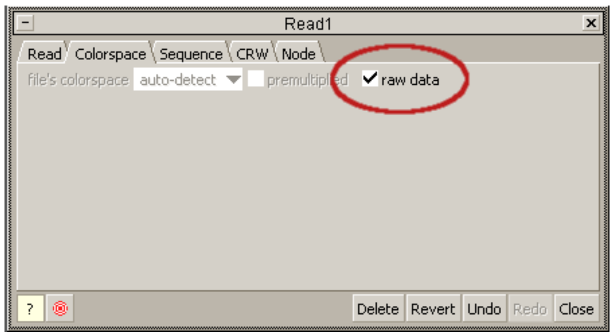
- Load the cineNuke plug-in. For version 4.7 and 4.8 right-click on the background:`Color>CineNuke` For version 5.0:`Other>Allplugins>C>CineNuke`

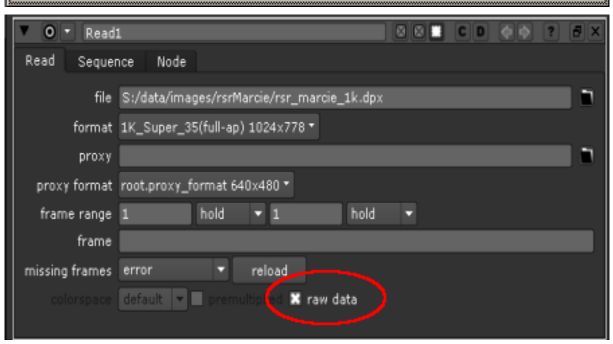


- Load an image and connect it to the cineNuke node. Then connect a viewer node to the output of the cineNuke node.



- If you are using Cineon images, be sure to check the *raw data* checkbox under the 'Colorspace' tab in the Read node in version 4.7 and 4.8. In version 5.0 it is in the 'Read' tab.





- Select a recent monitor profile generated from cineProfiler for your monitor and select the target profile of the colour space you would like to match.

7.7.2 Controls

Below are some quick snap-shots of the cineNuke control panel. For detailed information about how these controls work, please see Section 7.2.

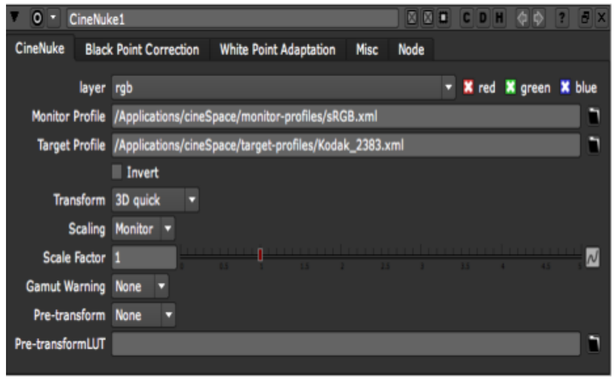


Figure 7.14: cineNuke main control panel

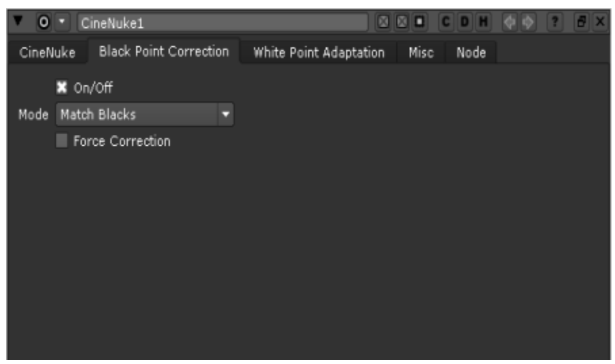


Figure 7.15: cineNuke black point control panel

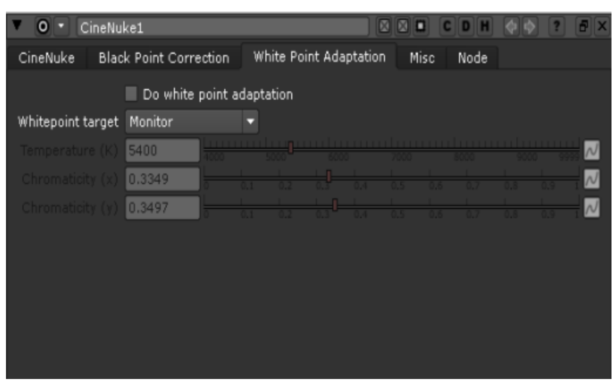


Figure 7.16: cineNuke white point adaptation control panel

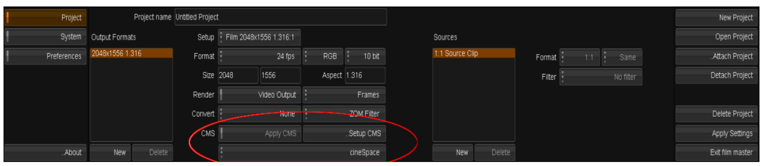
7.8 cineFilmMaster

cineFilmMaster is a plug-in to Digital Vision (Nucoda) Film Master that provides the functionality of both equalEyes and full 3D colour transforms as a colour management module within Film Master.

cineFilmMaster uses the OpenFX framework with custom libraries provided by Digital Vision (Nucoda) that extend the OpenFX API. This allows the rendering of LUT data in real time on the GPU during playback.

7.8.1 Loading cineFilmMaster

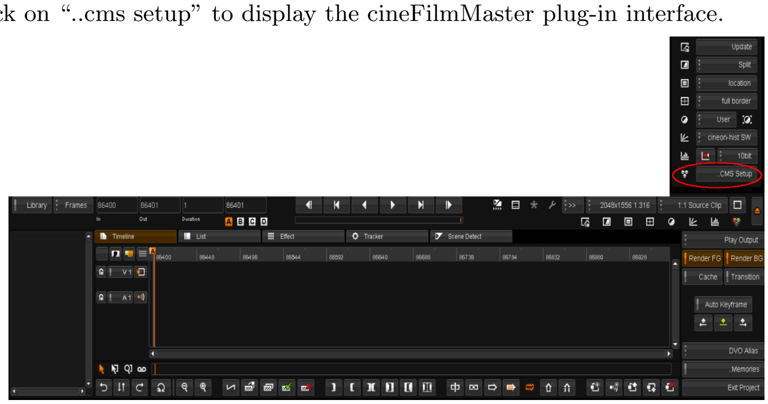
- Open Film Master.
- Select "cineSpace" from the CMS options.



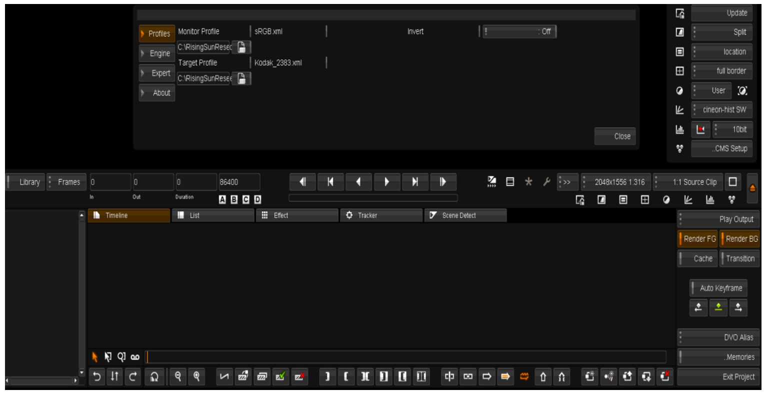
- Select "New project" or "Open project".

- Open the monitor tools menu by clicking on the small arrow on the far right of the player toolbar.

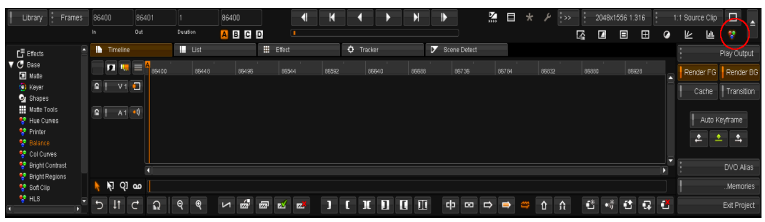
- Click on "..cms setup" to display the cineFilmMaster plug-in interface.



- The options for controlling the plug-in parameters will now be displayed.



- To activate the cineSpace Engine, press the "CMS" button.



7.8.2 Controls

cineFilmMaster lets the user control the cineEngine parameters used for the colour space transform via the use of the graphical user interface panel. For detailed information about how these controls work, please see Section 7.2.

7.8.3 Additional controls

cineFilmMaster provides some additional controls not found in the other cinePlugins so that you can adjust a number of advanced options.

gamutTreatment

Default: None
⟨*None / Alpha / Red / Green / Blue / desaturate in / desaturate out*⟩

Highlights the colours in an image that cannot be correctly represented on the monitor (viewing) device (refer to Section 4.14).

- **None:** No out-of-gamut colours will be highlighted.
- **Alpha:** The pixel of any colour that would be out of gamut has its alpha channel set to zero.
- **Red:** Any colours that would be out of gamut are displayed as red.
- **Green:** Any colours that would be out of gamut are displayed as green.

- **Blue:** Any colours that would be out of gamut are displayed as blue.
- **desaturate in:** Any colours that would be in gamut are desaturated.
- **desaturate out:** Any colours that would be out of gamut are desaturated.

LUT Resolution

Default: Low
⟨*Low / High*⟩

Select the size of the cube that the cineEngine will generate. This is loaded into the video card LUT by Film Master and is used to render the displayed image.

- **Low:** This is recommended for real time playback.
- **High:** This is the most accurate transform.

Lin To Log Parameters

These parameters only take effect when the selected pretransform is LinToLog. The values for the white and black point are in 10-bit Cineon space. It is recommended that only an experienced user should deviate from the default values. The available parameters are:

- White Point
- Black Point
- Display Gamma
- Film Gamma

For more information about these parameters, please see Section 4.13.1.

Printer lights

These controls simulate the printer light settings on film recorders, so that you can make adjustments to the target profile if required (see Section 4.10 for further details).

Figure 7.17: cineFilmMaster profiles control panel

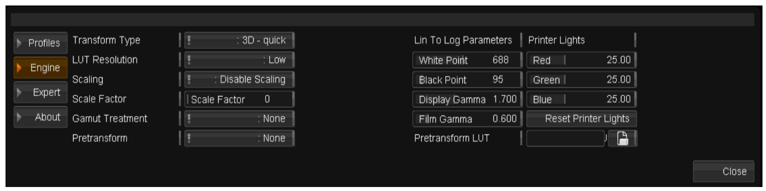


Figure 7.18: cineFilmMaster engine control panel

Figure 7.19: cineFilmMaster expert control panel

Chapter 8

cineCube Visual

8.1 cineCube Visual Overview

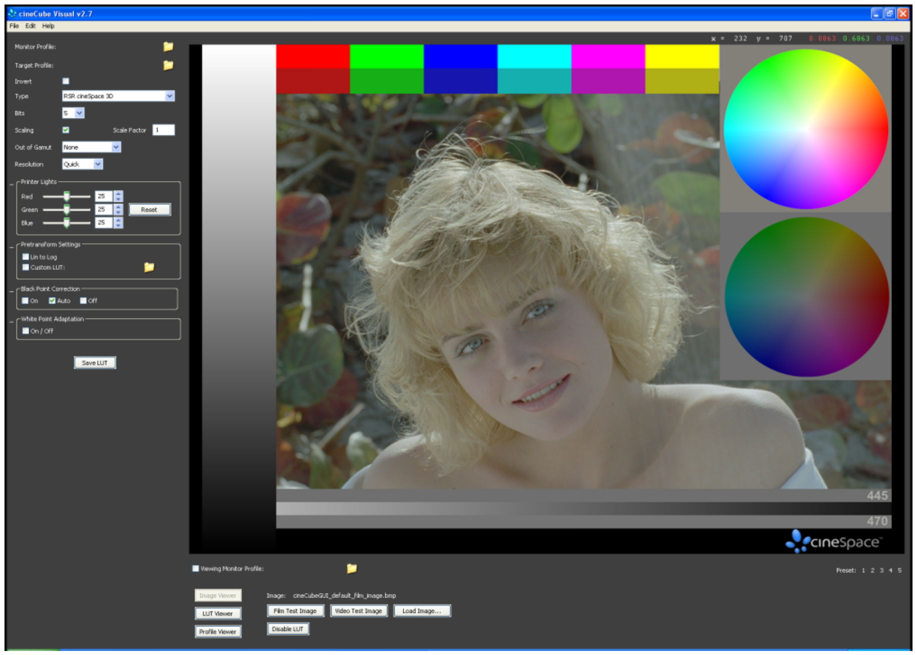
cineCube Visual is an application for generating 3D LUTs (cubes[1]) that can be used within a wide range of industry applications developed by partner vendors. These products load the cubes onto the graphics card to provide a cineSpace calibrated image within their applications.

Depending on the application you are using, you may also be able to render out with the cube "baked in" to your images. This feature can be particularly useful when needing to produce video deliverables in addition to a film grade, or for exchanging "looks" between facilities.

cineCube Visual allows you to analyse the colour gamuts of the profiles you are using, and to inspect the effects of the LUT before it is created. You can preview an image with the LUT applied and even see how the LUT will map the input image data.

NOTE *This feature is only avaiable for Windows and Mac OS X*

[1]The terms *3D LUT* and *cube* may be used interchangeably.



8.2 cineCube Visual features

cineCube Visual generates 3D look-up tables (cubes) from cineSpace target and monitor profiles.

8.2.1 General Features

`Monitor profile` Select the monitor profile (profile of device you will be working / viewing on). Must be a valid cineSpace v2.x `.xml` profile.

`Target profile` Select the target profile (profile of device you want to match). Must be a valid cineSpace v2.x `.xml` profile.

`Invert` Select to produce an inverse transform where the cineCube Visual has internally switched the monitor and target profiles. Inverse transforms can be used to apply a data transform to your images so that they reproduce correctly when sent to a different output medium or colour space.
NOTE *This feature is not enabled in the standard cineSpace suite, but is available for purchase as an optional extra. If this feature is used without the correct license the processed image will be displayed with a strong green cast.*

`Type` Type of cube you want to generate.

Academy / ASC	Adobe Photoshop 1D
AJA Kona 1D	Apple Color
Apple FinalTouch v1.0.8 and older	Apple FinalTouch v1.0.9 and newer
Apple Shake 1D	Arri
Assimilate Scratch	Autodesk Fire
Autodesk Flame	Autodesk Flint
Autodesk Inferno	Autodesk Lustre
Autodesk Smoke	Autodesk Toxik
Barco D-Cine	Blackmagic Decklink 1D
Blackmagic HDLink Pro	Chrome Imaging
Cine-tal Cinemage	Cyborg
DaVinci Resolve	Digital Ordnance Frame Thrower
Digital Vision Nucoda v1	Digital Vision Nucoda v2
DVS Clipster	Grass Valley LUTher
Iridas	IT
Pixel Farm	Quantel 1D
Quantel 3D	Rip
RSR cineSpace 1D	RSR cineSpace 3D
SGO Mistika	SideFX Houdini
TCube	

`Bits` Resolution of generated cube 4..9 (Recommend using 5, i.e. a 32x32x32 cube). Some cube types will not allow certain sizes and therefore will not be avaiable.

`Scaling` Uncheck this if you do not want the target and monitor spaces normalised. Unchecking this option will give greater accuracy as long as the monitor is 'brighter' than the target. If the monitor is not as bright as the target clipping will occur and the created cube will be corrupted.

`Scale factor` This value is used as a scaling factor to ensure that the target and monitor gamuts line up as optimally as possible. Decreasing this value decreases the overall brightness, but increases the overall quality of the colours (reduces the amount of out-of-gamut colours). Recommended values are in the range from 0.3 to 1.0. Refer to Section 4.9

`Out of Gamut` Use this option to set all the out-of-gamut colours to either red, blue, green or desaturate colour in or out.

`Resolution` *Quick* or *Accurate* transform. Refer to Section 4.7.2

(See Figure 8.1 for the features listed above.)

`Viewing monitor profile` The *viewing monitor profile* (see Figure 8.2) allows you to select a second monitor profile. (Must be a valid cineSpace v2.x `.xml` profile.)
In many cases the display used with cineCube Visual will be different from the one for which the LUT is being created. The *Viewing monitor profile* will be the profile of the device used when running cineCube Visual, while the *Monitor profile* and *Target profile* will be chosen based on the LUT you wish to create.

`Presets` You can save the current state of cineCube Visual and restore it at a later time.

Save a preset - Press the +/= key followed by the number [1–5] of the preset you want to store the current settings in. This will be indicated above the requested number

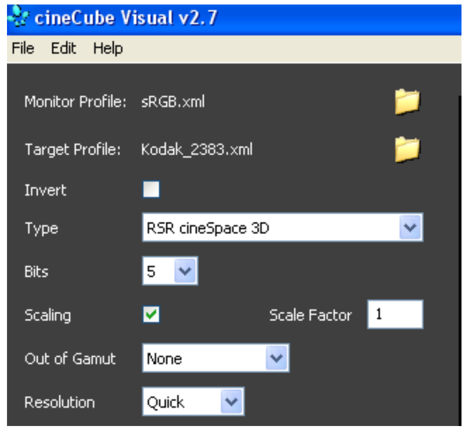


Figure 8.1: General features

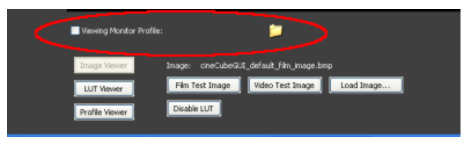


Figure 8.2: Viewing monitor profile

Load a preset - Press the number key of the preset you would like to load. The loaded preset number will be green

Delete a preset - Press the _/- key followed by the number [1–5] of the preset you want to delete/clear.

So in Figure 8.3 two presets are saved (1–2), and number 1 is loaded on the screen.

Saved states will NOT be saved when cineCube Visual is shutdown.

Figure 8.3: Presets

`Co-ordinates` This is for information a gives the x / y value of the mouse over a loaded image and the Red / Green / Blue value. (See Figure 8.4) This information is the colour value of the image (before / after a transform is applied) but ignoring the *viewing monitor* transform.

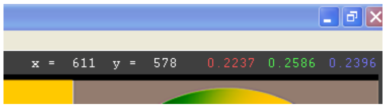


Figure 8.4: co-ordinates

`Save LUT` Select to save the LUT.

8.2.2 Printer Lights

`Printer lights` Enable printer light usage. Refer to Section 4.10 for an explanation of printer light settings and their uses.

`Red` The red printer light value (default value set in rsr.conf or 25 if not set).

`Green` The green printer light value (default value set in rsr.conf or 25 if not set).

`Blue` The blue printer light value (default value set in rsr.conf or 25 if not set).

`Reset` Resets all three printer light values back (default value set in rsr.conf or 25 if not set).

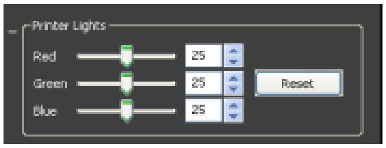


Figure 8.5: printer lights

8.2.3 Pretransform

`lintolog` Set this when working with linearised film images so that a lin-to-log pretransform is performed before applying the 3D colour transform. Refer to Section 4.13.1 for details on using this conversion.

`Custom LUT` Select to load a custom pretransform LUT that will be used in a similar manner to the lin-to-log pretransform.

Figure 8.6: pretransform

8.2.4 Black Point Correction

Select the required state -

`On` Force black point correction on using 'Match Blacks', even if it is not needed.

`Auto` (Default) Black point correction using 'Match Blacks' is completed, if required.

`Off` Set this to disable black point correction.

Refer to to Section 4.11.1 for an explanation of black point correction.

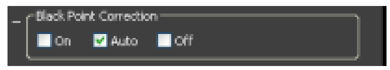


Figure 8.7: blackpoint correction

8.2.5 White Point Adaptation

`On` This will cause white point adaptation (Section 4.12) to occur

`Off` (Default) White point adaptation is not applied.

`Method` Select one of the following -

- `monitor` - Causes the white point produced by the resulting cube to be the same as that of the monitor device.
- `chromaticity` - Set the x and y value to a certain chromaticity coordinate.
- `temperature` - Set the white point to a certain colour temperature (in Kelvin).

Only once the 'Method' has been selected will you be able to set either the chromaticity or temperature value.

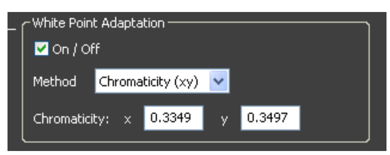


Figure 8.8: Whitepoint adaptation

NOTE - All four sections - Printer lights, Pretransform, Blackpoint correction and White point Adaptation are collapsible by clicking on the +/- on each section. (See Figure 8.9)

8.3 Functions

cineCube Visual allows you to view various aspects of the LUT that is being created before saving it. (Figure 8.10)

`Image Viewer` Select this to preview an image in the viewer.

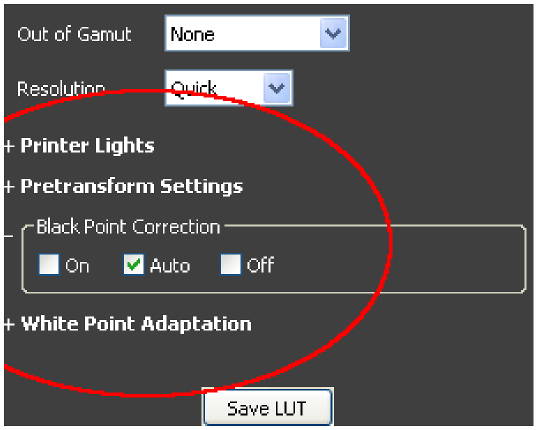


Figure 8.9: Collapsible sections

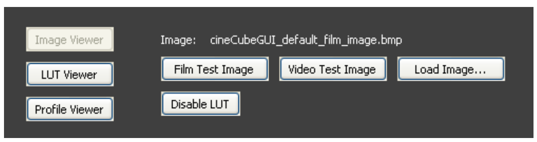


Figure 8.10: Functions

`LUT Viewer` Select this to view how the LUT is mapped to the data. (See Figure 8.11)
The viewer controls allow you to change the way in which the data for both your LUT and the comparison *identity LUT* is displayed.

- `None` - No data shown
- `Points` - All points shown
- `Edge` - Outline shown
- `Frame` - Framework shown
- `Surface` - Solid surface shown

`Profile Viewer` Select this to view the monitor and target colour gamuts in 3D space. (See Figure 8.12)
The controls in the viewer allow you to change the way in which the data for the monitor and target profile is displayed. This is very useful for identifying potential out-of-gamut colour issues and when making scaling factor adjustments.

- `None` - No data shown
- `Points` - All points shown
- `Edge` - Outline shown
- `Frame` - Framework shown
- `Surface` - Solid surface shown

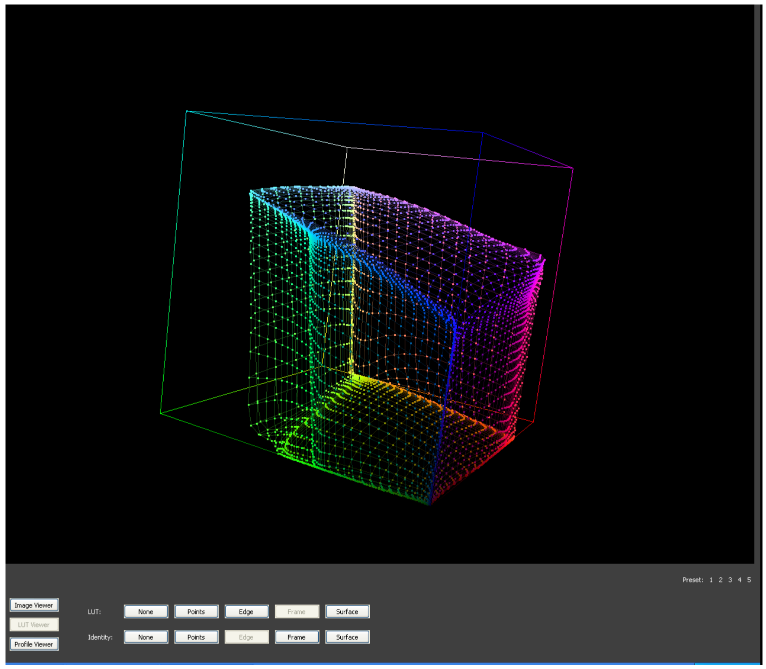


Figure 8.11: LUT viewer

Film Test Image Load default film test image that is installed with cineCube Visual. This default image can be replaced by over writing it with a new image. Image location is:
./images/cineCubeVisual_default_film_image.cin

Video Test Image Load default video test image that is installed with cineCube Visual. This default image can be replaced by over writing it with a new image. Image location is:
./images/cineCubeVisual_default_video_image.png

Load Image Select this to load an image other than the default images. Image formats that can be loaded are: .BMP/.CIN/.DPX/.GIF/.JPG/.PNG

Apply / Disable LUT Turns the LUT on or off.

8.4 Advanced Options

The advanced options are avaible from the top toolbar (edit/advanced settings). (See Figure 8.13) This allows LintoLog, Printer lights, LUT input, Transform resolution, Background display and Viewing monitor features to be set. (See Figure 8.14)

Lin to Log

- `white` Sets film white point for log/lin conversions.

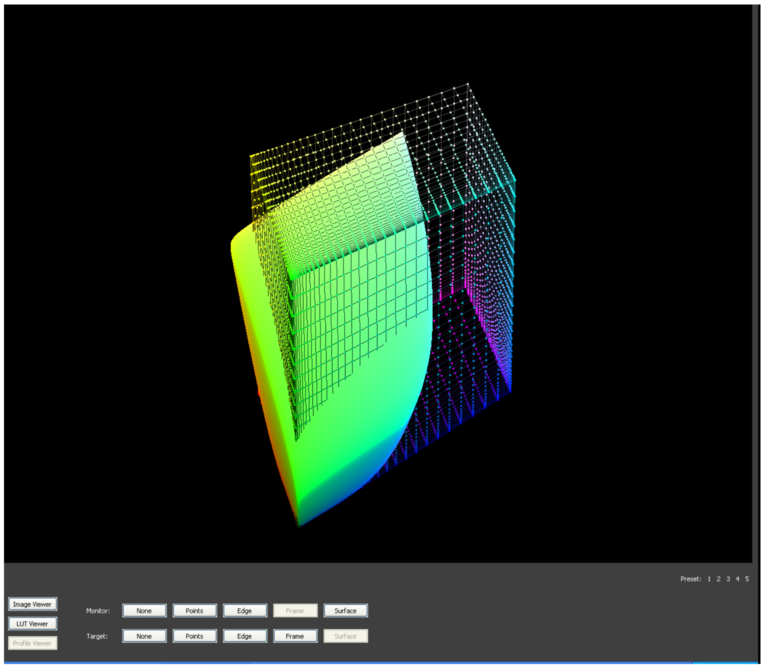


Figure 8.12: Profile viewer

- `black` Sets film black point for log/lin conversions.
- `display gamma` Sets conversion/device gamma for log/lin conversions.
- `film gamma` Sets film gamma for log/lin conversions.

Printer Lights

`red-trim` The red trim value (default value set in the rsr.conf or 25 if not set).

`blue-trim` The blue trim value (default value set in the rsr.conf or 25 if not set).

`green-trim` The green trim value (default value set in the rsr.conf or 25 if not set).

`printer lights per stop` Sets the approximate number of printerlight points per stop.

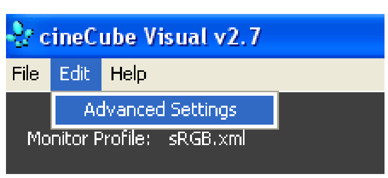


Figure 8.13: Advanced toolbar

LUTs

max stim input Set this option when creating cineSpace (.csp) formatted LUTs to specify an extended (or reduced) range of input values to your LUT. This enables the use of values above 1.0, to avoid clipping when using floating-point images. Defaults to 1.0 if not set.

auto optimise Uncheck this option when creating cineSpace (.csp) formatted LUTs to disable automatic optimisation. By default an optimised 1D pre-LUT will be created when generating cineSpace format LUTs, minimising potential banding artifacts arising from LUT interpolation. This may be disabled where consistency compared with other LUT types is more important than absolute accuracy.

gamma Can be set to change the gamma value for Scratch LUTs. Defaut is 1.

Engine

transform resolution This integer represents the internal interpolator cube side length that is used in generating the final LUT. Increasing this number increases the time taken to create a LUT but also increases the overall quality of the final output. Recommended resolution > 10 (defaults to 40).

Display

background colour Change the background colour as required on the viewers.

Viewing Monitor

scaling Turn scaling on or off for the viewing monitor transform.

blackpoint correction Sets the black point correction for the viewing monitor transform. (On/Auto/Off) See Section 8.2.4.

transform resolution Change the transform resolution for the viewing monitor transform. This integer represents the internal interpolator cube side length that is used in generating the final LUT.

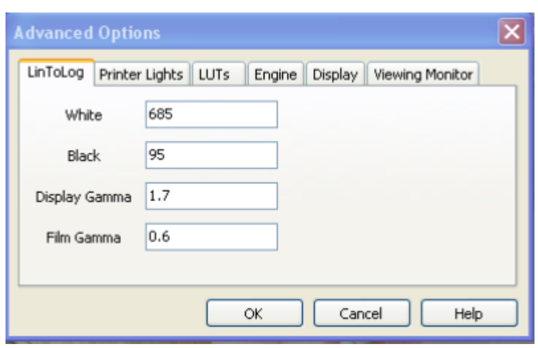


Figure 8.14: Advanced options

8.5 Key Bindings

Save LUT	[+ or =] then [1–5]
Delete Saved LUT	[_ or -] then [1–5]
Apply Saved LUT	[1 – 5]
Activate / Deactivate LUT	[Space]

Other key bindings may be defined by the operating system and window manager.

8.6 Mouse Control

Task	*Control*	*Viewer*
Move image	Left click then move mouse	Image / LUT and profile viewer
Zoom	Mouse wheel	Image viewer only

The zoom will occur at the point of the mouse pointer on the image.

Chapter 9

cineCube

cineCube is a command line tool for generating 3D LUTs (cubes[1]) that can be used within a wide range of industry applications developed by partner vendors. These products load the cubes onto the graphics card to provide a cineSpace calibrated image within their applications. For a full list of supported applications, please refer to the usage information below.

Depending on the application you are using, you may also be able to render out with the cube "baked in" to your images. This feature can be particularly useful when needing to produce video deliverables in addition to a film grade, or for exchanging "looks" between facilities.

Being a command line tool, you will need to open a command prompt on your operating system in order to run cineCube. You can then navigate to the cineSpace installation folder and run the 'cineCube' command with a range of options as outlined below. If you run cineCube without specifying any options, it will display the usage information, including all supported cube formats.

9.1 cineCube usage

Usage:

`cineCube` `-target` ⟨*target profile*⟩`.xml -monitor` ⟨*monitor profile*⟩`.xml`
`-bits` ⟨*4..9*⟩ `-type` ⟨*...*⟩
`[-noscale] [-lintolog] [-printerlights]`

Details: cineCube generates 3D look-up tables (cubes) from cineSpace target and monitor profiles.

`-target` Full path to target profile (profile of device you want to match). Must be a valid cineSpace v2.x `.xml` profile.

`-monitor` Full path to monitor profile (profile of device you will be working / viewing on). Must be a valid cineSpace v2.x `.xml` profile.

`-bits` Resolution of generated cube 4..9 (Recommend using 5, i.e. a 32x32x32 cube).

[1] The terms *3D LUT* and *cube* may be used interchangeably.

`-type` Type of cube you want to generate.

arri	(.ari) Arri 3D LUT file.
asc	(.asc) Academy / ASC 3D LUT file.
barco	(.LUT-CLUT) Barco D-Cine 3D LUT file.
blackmagic1d	(.txt) Blackmagic Decklink 1D LUT file.
cinemage	(.mga) Cine-tal Cinemage monitor LUT file.
chrome	(.ctp) Chrome Imaging 3D LUT file.
clipster	(.ari) DVS Clipster 3D LUT file.
cinespace	(.csp) Rising Sun Research cineSpace 3D LUT file.
cinespace1d	(.csp) Rising Sun Research cineSpace 1D LUT file.
color	(.mga) Apple Color 3D LUT file.
cyborg	(.lut) Cyborg5D lut file.
finaltouch1	(.mga) SiliconColor FinalTouch v1.0.8 and older 3D LUT file.
finaltouch2	(.mga) SiliconColor FinalTouch v1.0.9 and newer 3D LUT file.
fire	(.3dl) Autodesk Fire 3D LUT file.
flame	(.3dl) Autodesk Flame 3D LUT file.
flint	(.3dl) Autodesk Flint 3D LUT file.
framethrower	(.ftlut) Digital Ordnance Frame Thrower 3D LUT file.
houdini	(.lut) SideFX Houdini 3D LUT file.
inferno	(.3dl) Autodesk Inferno 3D LUT file.
iridas	(.itx) IRIDAS 3D LUT file.
it	(.lut) IT LUT file.
kona1d	(.txt) AJA Kona 1D LUT file.
lustre	(.lut) Autodesk Lustre 3D LUT file.
luther	(.txt) Grass Valley LUTher 3D LUT file.
mistika	(.itx) S.G.O. Mistika 3D LUT file.
nucodaold	(.lut) Digital Vision Nucoda 3D LUT v1 file.
nucoda	(.lut) Digital Vision Nucoda 3D LUT v2 file.
photoshop	(.acv) Adobe Photoshop 1D curves file.
pixelfarm	(.pfl) The Pixel Farm 3D LUT file.
quantel1d	(.txt) Quantel 1D LUT file.
quantel3d	(.txt) Quantel 3D LUT file.
resolve	(.dat) da Vinci Resolve 3D LUT file.
rip	(.lut) Rip 3D LUT file.
scratch	(.3dl) ASSIMILATE Scratch 3D LUT file.
shake1d	(.lut) Apple Shake 1D LUT file.
smoke	(.3dl) Autodesk Smoke 3D LUT file.
tcube	(.L3d) TCube 3D LUT file.
toxik	(.lut) Autodesk Toxik 3D LUT file.

`-noscale` Set this if you do not want the target and monitor spaces normalised. Setting this option will give greater accuracy as long as the monitor is ‘brighter’ than the target. If the monitor is not as bright as the target clipping will occur and the created cube will be corrupted.

`-lintolog` Set this when working with linearised film images so that a lin-to-log pretransform is performed before applying the 3D colour transform. Refer to Section 4.13.1 for details on using this conversion.

`-pretransform` Full path to the custom pretransform file (if applicable). This allows you to specify a custom pretransform LUT that will be used in a similar manner to the lin-to-log pretransform. Note: Enabling this functionality will disable the lin-to-log option, even if explicitly set.

-noblackcorrection Set this to disable black point correction. If this parameter is not used then black point correction is done by default, using the black point options below (`bp_mode`, `force_bp_correction`). If the black point options are not set then the default values in the rsr.conf file are used.

-bp_mode This sets the mode (method) used to do the black point correction. There are four possible arguments:

- `mb` - Match Blacks (default unless the rsr.conf file specifies otherwise)
- `agr` - Along Grey Ramp
- `ngb` - Nearest Gamut Boundary
- `to` - Target Offset

Refer to Appendix B for an explanation of the different black point correction modes.

-force_bp_correction Force black point correction even if it is not needed. If this option is not used then, by default, black point correction will only occur if the target black point is darker than the monitor black point.

-white-adjust This will cause white point adaptation (Section 4.12) to occur and has three possible arguments:

- `monitor` - Causes the white point produced by the resulting cube to be the same as that of the monitor device.
- `chromaticity` - Sets the white point to a certain chromaticity coordinate. This can be set using the `white-x` and `white-y` options, otherwise it defaults to the values set in the `rsr.conf` file.
- `temperature` - Sets the white point to a certain colour temperature (in Kelvin). This can be set using the white-temp option, otherwise it defaults to the values set in the `rsr.conf` file.

-white-x If `-white-adjust=chromaticity` is set then this option specifies the x chromaticity coordinate. If this parameter is not set, then the RSR_WHITE_X value from the rsr.conf file is used.

-white-y If `-white-adjust=chromaticity` is set then this option specifies the y chromaticity coordinate. If this parameter is not set, then the RSR_WHITE_Y value from the rsr.conf file is used.

-white-temp If `-white-adjust=temperature` is set then this option specifies the colour temperature. If this parameter is not set, then the RSR_WHITE_TEMP value from the rsr.conf file is used.

-printerlights Enable printer light usage. Refer to Section 4.10 for an explanation of printer light settings and their uses.

-red-off The red printer light value (default value set in rsr.conf or 25 if not set).

-blue-off The blue printer light value (default value set in rsr.conf or 25 if not set).

-green-off The green printer light value (default value set in rsr.conf or 25 if not set).

-red-trim The red trim value (default value set in the rsr.conf or 25 if not set).

-blue-trim The blue trim value (default value set in the rsr.conf or 25 if not set).

-green-trim The green trim value (default value set in the rsr.conf or 25 if not set).

If the offsets are the same as the trims there is no effect – only the difference between them is used.

-maxinput Set this option when creating cineSpace (.csp) formatted LUTs to specify an extended (or reduced) range of input values to your LUT. This enables the use of values above 1.0, to avoid clipping when using floating-point images. Defaults to 1.0 if not set.

-noopt Set this option when creating cineSpace (.csp) formatted LUTs to disable automatic optimisation. By default an optimised 1D pre-LUT will be created when generating cineSpace format LUTs, minimising potential banding artifacts arising from LUT interpolation. This may be disabled where consistency compared with other LUT types is more important than absolute accuracy.

-gamut ⟨*red/green/blue/invert*⟩ Use this option to set all the out-of-gamut colours to either red, blue, green or invert colour.

Advanced:

--transform_resolution # This integer represents the internal interpolator cube side length that is used in generating the final LUT. Increasing this number increases the time taken to create a LUT but also increases the overall quality of the final output. Recommended resolution > 10 (defaults to 40).

--target_scale_factor # This double value is used as a scaling factor to ensure that the target and monitor gamuts line up as optimally as possible. Decreasing this value decreases the overall brightness, but increases the overall quality of the colours (reduces the amount of out-of-gamut colours). Recommended values are in the range from 0.3 to 1.0.

cineCube currently does no file writing itself so you will need to 'pipe' or 'direct' the standard output to a file as required (see example below).

Example:

```
cineCube.exe -target target-profiles\Kodak_2383.xml -monitor monitor-profiles\sRGB.xml
-bits 5 -type iridas > 2383_sRGB.itx
```

9.2 Using cineSpace cubes

After creating a profile for your monitor and choosing your target profile, run cineCube to generate a cube for the application you are using. Copy the cube into a folder where the other cubes for that application are stored. If necessary, restart the application and then load the new cineSpace cube. Refer to the documentation for your application if you are unfamiliar with using cubes.

Two examples of using cineSpace cubes with common tools are listed below:

9.2.1 Example: Using cineCube with IRIDAS tools

1. Profile your monitor, which generates `host.xml`.

2. Run cineCube with the appropriate target profile.
   `cineCube -target ./target-profiles/film.xml -monitor ./monitor-profiles/host.xml -bits 5 -type iridas > cube.itx`
3. Copy the resultant `cube.itx` into the LUTs directory of your FrameCycler (or other IRIDAS application) installation.
4. The cube should then appear in the calibration tab when viewing an image sequence (restart FrameCycler if necessary).

9.2.2 Example: Using cineCube with Digital Vision (Nucoda) tools

1. Profile your monitor, which generates `host.xml`.
2. Run cineCube with the appropriate target profile.
   `cineCube.exe -target target-profiles\film.xml -monitor monitor-profiles\host.xml -bits 5 -type nucoda > cube.cms`
3. Copy the resultant `cube.cms` into a folder accessible by the Digital Vision (Nucoda) system.
4. Create a project in Data Conform or Film Master and specify userCMS.
5. Click on the CMS Tab on the upper right section of the interface to browse to the cube.

Chapter 10

probeServer

probeServer is a tool that allows a hardware probe to be installed and running on one computer and then used for profiling another computer's monitor. It is particularly useful where the probe is running on a laptop (with, for example, a wireless network connection) that is then moved around to profile all monitors in a facility. It is also useful for using certain hardware probes with a computer that does not have the appropriate ports – a nearby computer can be used to run the probe instead.

10.1 Using probeServer

probeServer is included with the other applications during installation. Running the server is a simple matter:

- Connect the hardware probe to your computer and install any required drivers.
- Start probeServer from the menu or command line.
- Enter the port number (or leave default). Ensure that the chosen port is not being used for anything else and that no firewalls are blocking the port.
- Click *Start*.

The probeServer is now running and can be used as a remote probe to profile any computer on your network. When running cineProfiler, simply check the *Use Remote Probe* option in the initial settings window and enter the IP address of the computer running probeServer and the port number that you have chosen.

If you have any problems with cineProfiler detecting the probe, restart probeServer and check that the server and port settings you have entered are correct. Also, ensure that you are using an unoccupied port with no firewalls blocking it – you may need to change your firewall settings if this is a problem.

Chapter 11

profileTools

The profileTools are a collection of command-line applications that aid in the parsing, generation and manipulation of cineSpace XML profiles. They can be useful when needing to capture data by some other means than using cineProfiler or when analysing data stored in cineSpace profiles.

11.1 xml2txt

This application will parse a cineSpace XML profile and generate a plain text file that contains the CIE XYZ values corresponding to each stim value.

Usage:
`xml2txt -xml` ⟨*source directory*⟩ `-txt` ⟨*target directory*⟩ ⟨*-cube|-dump*⟩
Converts each XML profile (`.xml`) file stored in ⟨*source directory*⟩ to a space delimited text (`.txt`) file in ⟨*target directory*⟩. Only the red, green, blue and grey ramps are written out by default.

`-cube` Will cause the whole profile to be written out.

`-dump` Will dump all environment variables to screen and to the log file if logging is enabled (`RSR_OUTPUT=LOG`) and verbose (`RSR_LOG_LEVEL=4`). Check your rsr.conf file.

11.2 txt2xml

This application will parse a space-delimited text file containing CIE XYZ values and generate a cineSpace XML profile that can be used within the applications in the cineSpace suite.

Usage:

`txt2xml -txt` ⟨*source directory*⟩ `-xml` ⟨*target directory*⟩ ⟨*-dump*⟩

Converts each space delimited text file (`.txt`) stored in ⟨*source directory*⟩ to an XML profile (`.xml`) file in ⟨*target directory*⟩. Only the red, green blue and grey ramps are used.

-dump Will dump all environment variables to screen and to the log file if logging is enabled (RSR_OUTPUT=LOG) and verbose (RSR_LOG_LEVEL=4). Check your rsr.conf file.

11.3 Text file format

There are two types of text (`.txt`) file format: one for Ramp (1D) profiles; the other for Cube (3D) profiles. These are both space-delimited text files.

11.3.1 Ramp profile

This is a list of the CIE XYZ response for increasing luminance. The length of the data is usually either 10, 17 or 52.

```
stim redX redY redZ greenX greenY greenZ blueX blueY blueZ greyX greyY greyZ
0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000
0.066 0.488 0.252 0.022 0.423 0.847 0.141 0.213 0.085 1.126 1.126 1.185 1.290
0.133 1.043 0.538 0.048 0.905 1.810 0.301 0.456 0.182 2.405 2.405 2.531 2.756
   ...
   ...
0.933 28.70 14.81 1.356 24.80 49.78 8.297 12.56 5.024 66.16 66.16 69.61 75.80
1.000 32.99 17.04 1.541 28.76 57.21 9.535 14.43 5.774 76.03 76.03 80.00 87.11
```

11.3.2 Cube profile

This is a list of the CIE XYZ responses for a full cube gamut, the order of the values is determined by:

```
for red = 0 to cubeLength;
   for green = 0 to cubeLength;
      for blue = 0 to cubeLength;
         output[red,blue,green];
      end;
   end;
end;
```

File format:

```
0.0000 0.0000 0.0000
0.0667 0.0384 0.4839
1.3455 1.2111 1.3384
   ...
73.4743 78.2822 75.2222
76.3848 80.0000 78.3848
```

Chapter 12

Frequently asked questions

The most up-to-date information, discussions and common questions about the cineSpace colour management suite can be found at the cineSpace forums:

http://cinespace.rsrhq.com/forum

Please visit, join and participate in the forums to learn more about the cineSpace suite and colour management in general. If you are unable to find the answer you are looking for, you can either post a new topic on the forums or contact our support team:

cineSpace-support@rsrhq.com

Appendix A

Environment variables and configuration files

Here we examine how the cineSpace libraries use environment variables and configuration files to determine default behaviour at runtime.

A.1 Philosophy

cineSpace uses a combination of configuration files and environment variables. Variables can be defined in either a global configuration file, a local configuration file or using environment variables.

A.2 Details

The cineSpace applications search for the configuration files in three locations:

```
$RSR_APP_DIR\basename
$HOME\basename
.\basename
```

If more than one of these files is found then they are read in that order, with entries in later files overwriting earlier entries. Similarly, repeated entries in the configuration files only take the last value in the file. Finally, environment variables can override each of these values.

Thus in order of loading we have

1. The RSR application directory (`$RSR_APP_DIR`) (if defined in the environment);
2. The user's home directory ($HOME) (if defined in the environment);
3. The directory from which the application was started (./); and
4. Environment variables.

The following sections detail the cineSpace variables available to the user and the effect that they have on the application suite.

A.3 Configuration File Setting

RSR_CONF_FILE = file
Sets the path to which configuration file cineSpace should use.
RSR_CONF_FILE = /user/local/RisingSunResearch/rsr.conf

A.4 License Settings

RSR_LICENSE = RLM path
Sets the path to your license file or server. An absolute path works best as relative paths are relative to the directory that the application was run from, not the directory where the application is located (unless the `RSR_APP_DIR` above is set).

RSR_LICENSE = 27000@rlm_server.domian.com

A.5 Global Environment

Settings that affect most cineSpace applications and plug-ins.

A.5.1 Directory Settings

RSR_MONITOR_DIR = directory
Sets the location of your monitor profiles.
RSR_MONITOR_DIR = /usr/local/RisingSunResearch/monitor-profiles/

RSR_TARGET_DIR = directory
Sets the location of your target profiles.
RSR_TARGET_DIR = /usr/local/RisingSunResearch/monitor-profiles/

RSR_APP_DIR = directory
A last resort place to look for shared RSR resources. Enter the absolute (not relative) path to the directory that has the application executable in it.
RSR_APP_DIR = /usr/local/RisingSunResearch/

A.5.2 Default Profile Settings

RSR_MONITOR_REGEXP = regexp
RSR applications will look in `RSR_MONITOR_DIR` for profiles matched to the regexp `RSR_MONITOR_REGEXP` to use as the default monitor profile, if more than one match is found the file with the most recent creation/modification date is used. More detail is provided in section A.14.
 # Pick out the most recent file containing the host name ending in .xml
RSR_MONITOR_REGEXP = (%HOST).\.xml$*

RSR_TARGET_REGEXP = regexp
RSR applications will look in `RSR_TARGET_DIR` for profiles matched to the regexp `RSR_TARGET_REGEXP` to use as the default target profile, if more than one match is found the file with the most recent creation/modification date is used. More detail is provided in section A.14.
 # Pick out the RSR_ideal_monitor profile
RSR_TARGET_REGEXP = ^RSR_D61_g2.2\.xml$

RSR_PRETRANSFORM_LUT = file
Sets the default LUT file to use as a pretransform. See the cineSpace documentation for more information on pretransforms and the LUT file format. When using this option you must use the full path to the LUT file. Defaults to no LUT file and the user will be prompted to find one if they try to load a LUT as a pretransform.
RSR_PRETRANSFORM_LUT = fullpath/lutfile

A.5.3 IO Settings

Settings that generally affect console output from `RSR_OUTPUT` and `RSR_LOG_LEVEL` options:

`RSR_OUTPUT`	`RSR_LOG_LEVEL`	What you actually get
NOTHING		Nothing
QUIET	0	Fatal errors
ERROR	1	Non-fatal errors + QUIET
WARNING	2	Warnings + ERROR
VERBOSE	3	Other comments + WARNING

RSR_OUTPUT = NOTHING, QUIET, ERROR, WARNING, VERBOSE, LOG
If RSR_OUTPUT = LOG then `RSR_LOG_LEVEL` and `RSR_LOG_FILE` should be set. (see below)
RSR_OUTPUT = QUIET

RSR_LOG_LEVEL = 0, 1, 2, 3
Sets level of log output
RSR_LOG_LEVEL = 0

RSR_LOG_FILE = filename
Sets path to file where logged output should be placed.
RSR_LOG_FILE = rsr.log

A.6 Profile lifetime options

RSR_CRT_LIFETIME = int
Sets the lifetime of CRT profiles. The lifetime of a profile is the number of days after the profile was created that the profile is considered to be valid for. Range -1 to 9999. Defaults to (14) if not set. The special value of -1 makes the profile permanently valid.
RSR_CRT_LIFETIME = 14

RSR_LCD_LIFETIME = int
Same as RSR_CRT_LIFETIME above. Range -1 to 9999. Defaults to (28) if not set. The special value of -1 makes the profile permanently valid.
RSR_LCD_LIFETIME = 28

RSR_PROJECTOR_LIFETIME = int
Same as RSR_CRT_LIFETIME above. Range -1 to 9999. Defaults to (90) if not set. The special value of -1 makes the profile permanently valid.
RSR_PROJECTOR_LIFETIME = 90

A.7 cineEngine Settings

Settings that change the cineSpace engine parameters for transforming data for most RSR applications and plugins

A.7.1 Printer Lights Options

RSR_PRINTERLIGHT_TRIM = float,float,float
Sets the initial trims for the printer lights. These will be the R, G and B (three doubles separated by commas) values displayed by the printer lights settings when no adjustments are being made to the currently generated LUT. Range 0.0 – 50.0. Defaults to (25.0,25.0,25.0) if not set.
RSR_PRINTERLIGHT_TRIM = 25.0,25.0,25.0

RSR_PRINTERLIGHT_OFFSET = float,float,float
Sets the initial offsets for the printer lights. These will be the initial R, G and B (three doubles separated by commas) offsets applied by the printer lights. Range 0.0 – 50.0. Defaults to (25.0,25.0,25.0) if not set.
RSR_PRINTERLIGHT_OFFSET = 25.0,25.0,25.0

RSR_PRINTERLIGHTS_PER_STOP = integer
Sets the number of printer light points per stop of exposure. Allows adjustment to cater for variations in film recorder standards. Range 2 – 12. Defaults to (8) if not set.
RSR_PRINTERLIGHTS_PER_STOP = 8

A.7.2 Log / Lin Settings

RSR_FILMDATA_WHITE = int
Sets film white point for log/lin conversions. Range 0 – 1023. Defaults to (685) if not set. For backward compatibility, 'FILMDATAENV_WHITE' is also accepted.
RSR_FILMDATA_WHITE = 685

RSR_FILMDATA_BLACK = int
Sets film black point for log/lin conversions. Range 0 – 1023. Defaults to (95) if not set. For backward compatibility, 'FILMDATAENV_BLACK' is also accepted.
RSR_FILMDATA_BLACK = 95

RSR_FILMDATA_D_GAMMA = float
Sets conversion/device gamma for log/lin conversions. Range 0.001 – 10.0. Defaults to (1.7) if not set. For backward compatibility, 'FILMDATAENV_D_GAMMA' is also accepted.
RSR_FILMDATA_D_GAMMA = 1.7

RSR_FILMDATA_F_GAMMA = float
Sets film gamma for log/lin conversions. Range 0.001 – 10.0. Defaults to (0.6) if not set. For backward compatibility, 'FILMDATAENV_FILM_GAMMA' is also accepted.
RSR_FILMDATA_F_GAMMA = 0.6

RSR_FILMDATA_SOFT_CLIP = int
Sets the Lin/Log white point soft-clipping range. Soft clipping occurs over the range WHITE − SOFT_CLIP to WHITE + 4 × SOFT_CLIP. Range 0 – 1023. Defaults to (0) if not set. For backward compatibility, 'FILMDATAENV_SOFT_CLIP' is also accepted.
RSR_FILMDATA_SOFT_CLIP = 0

A.7.3 Black Point Correction

RSR_BLACK_DEFAULT_ON = true, false
Sets whether or not black point correction is done by default. Range (true, false). Defaults to (true) if not set.
RSR_BLACK_DEFAULT_ON = true

RSR_BLACKPOINT_FORCE_CORRECT = true, false
If black point correction is on this option sets whether or not black point correction is done at all times or if set to false, only when the monitor device is unable to reproduce the black point of the target device. Range (true, false). Defaults to (false) if not set.
RSR_BLACKPOINT_FORCE_CORRECT = false

A.7.4 White Point Adaptation

RSR_WHITE_DEFAULT_ON = true, false
Sets whether or not white point adaptation is done by default. Range TRUE - FALSE. Defaults to FALSE if not set.
RSR_WHITE_DEFAULT_ON = FALSE

RSR_WHITE_TARGET_TYPE = int
If white point adaptation is being used this sets the type of target white point for cineSpace to use options are:
0 : Use the white point of the monitor profile being used.
1 : Use a the colour temperature specified by RSR_WHITE_TEMP in Kelvin (K)
2 : Use chromaticity coordinates specified in RSR_WHITE_X and RSR_WHITE_Y.
Range 0, 1, 2. Defaults to (0) if not set.
RSR_WHITE_TARGET_TYPE = 0

RSR_WHITE_TEMP = int
If white point adaptation is being used this sets the white point temperature of the target profile being used. Range 4000 - 9999. Defaults to (5400) if not set.
RSR_WHITE_TEMP = 5400

RSR_WHITE_X = float
If white point adaptation is being used this sets the (x) chromaticity coordinate of the white point of the target profile being used. Range 0.0 - 1.0. Defaults to (0.3349) (ie. 5400K) if not set.
RSR_WHITE_X = 0.3349

RSR_WHITE_Y = float
If white point adaptation is being used this sets the (y) chromaticity coordinate of the white point of the target profile being used. Range 0.0 - 1.0. Defaults to (0.3497) (ie. 5400K) if not set.
RSR_WHITE_Y = 0.3497

A.8 equalEyes Options

A.8.1 Start-up Options

RSR_EE_INIT_STATE = inactive, active
Sets whether equalEyes starts active or inactive. Defaults to INACTIVE if not set.
RSR_EE_INIT_STATE = active

RSR_EE_INIT_SIZE = normal, minimised
Sets whether equalEyes starts minimised or not. Defaults to NORMAL if not set.
RSR_EE_INIT_SIZE = normal

RSR_EE_INIT_TARGET = xml, synthetic
Sets whether equalEyes starts with an XML or synthetic profile. Defaults to xml if not set.
RSR_EE_INIT_TARGET = xml

RSR_EE_INACTIVE_GUI = true, false
If TRUE sets whether users can interact with the EqualEyes gui. This is for users who would like to have all EqualEyes settings set by the rsr.conf file and /or presets and does not want other users to be able to change any of the settings. Note: If this option is set to TRUE then RSR_EE_INIT_SIZE will default to minimised and RSR_EE_INIT_STATE will default to active. Range (true, false). Defaults to (false) if not set.
RSR_EE_INACTIVE_GUI = false

RSR_EE_INIT_TARGET = xml, synthetic
Sets whether equalEyes starts with an XML or synthetic profile. Defaults to xml if not set.
RSR_EE_INIT_TARGET = xml

RSR_SHOW_DECOUPLING_WARNING = TRUE, FALSE
Sets whether or not a warning dialogue is displayed when the monitor profile does not decouple. Defaults to TRUE if not set.
RSR_SHOW_DECOUPLING_WARNING = TRUE

RSR_START_PRETRANS = none, log2lin, lut
Sets pretransform to be active on startup. Range none, log2lin or lut. Defaults to 'none' (the same as not setting any transform).
RSR_START_PRETRANS = none

A.8.2 General Options

RSR_EE_SCALING = true, false
Sets whether or not scaling is desired as default by the user. Range (true , false) Defaults to true if not set.
RSR_EE_SCALING = true

RSR_EE_CONFIDENCE_DE_MAX = float
Sets maximum allowable error values in the confidence test. Errors are CIE(1994)L*a*b where values greater than 2 are considered just perceivably different. Range 1.0 – 10.0 Defaults to (4.0) if not set.
RSR_EE_CONFIDENCE_DE_MAX = 4.0

RSR_EE_CONFIDENCE_DE_AVG = float
Sets average allowable error values in the confidence test. Errors are CIE(1994)L*a*b where values greater than 2 are considered just perceivably different. Range 0.1 – 5.0 Defaults to (3.0) if not set.
RSR_EE_CONFIDENCE_DE_AVG = 3.0

RSR_EE_CONFIRM_EXIT = true, false
Sets whether or not to check that the user really wants to exit when equalEyes is closed by the user. Range (true , false) Defaults to true (user is prompted)if not set.
RSR_EE_CONFIRM_EXIT = true

RSR_EE_MONPROF_LIST = filter
Provides a way for users to add extra filters to the Monitor Profile file Dialogue. The default filter is "Profiles (*.xml)" which will display all files with the .xml extension. You can add further filters here by specifying each new filter separated by ';;', e.g. "Synth Profiles (*synth.xml);;Host Profiles (%HOST*.xml)"
#The filter below will list all files containing the hostname and ending in .xml
RSR_EE_MONPROF_LIST = "Host Profiles (%HOST.xml)"*

A.8.3 Synthetic Profile Settings

RSR_SYNTH_CURVE_TYPE = gamma, cineon
Sets the default type of tonal curve to use for the synthetic profile. Range (gamma, cineon). Defaults to gamma if not set.
RSR_SYNTH_CURVE_TYPE = gamma

RSR_SYNTH_GAMMA = float
Default gamma for synthetic profiles. Range 0.001 – 9.999. Defaults to 2.20 if not set.
RSR_SYNTH_GAMMA = 2.20

RSR_SYNTH_WP_TYPE = chromaticity, temperature, monitor
Default white point type for synthetic profiles. Defaults to TEMPERATURE.
RSR_SYNTH_WP_TYPE = temperature

RSR_SYNTH_TEMP_TYPE = BLACKBODY, CIE
Default white correlated colour temperature (CCT) type for synthetic profiles (If RSR_SYNTH_WP_TYPE = TEMPERATURE). Defaults to CIE. (CIE D illuminants are nearly always used when specifying white point colour.)
RSR_SYNTH_TEMP_TYPE = CIE

RSR_SYNTH_TEMP = int
Default white point correlated colour temperature (CCT) for synthetic profiles (If RSR_SYNTH_WP_TYPE = TEMPERATURE). Range 4000 – 25000. Defaults to 6500 if not set.
RSR_SYNTH_TEMP = 6500

RSR_SYNTH_X = float
Default white point CIE x-chromaticity co-ordinate for synthetic profiles (If RSR_SYNTH_WP_TYPE = CHROMATICITY). Range 0.001 – 0.999. Defaults to 0.3127 (CIE D65) if not set.
RSR_SYNTH_X = 0.3127

RSR_SYNTH_Y = float
Default white point CIE y-chromaticity co-ordinate for synthetic profiles (If RSR_SYNTH_WP_TYPE = CHROMATICITY). Range 0.001 – 0.999. Defaults to 0.3290 (CIE D65) if not set.
RSR_SYNTH_Y = 0.3290

A.9 cineCube Visual Options

A.9.1 Default Startup Options

RSR_CC_LUT_TYPE = LUT type
Sets the default LUT type to be generated. Defaults to cineSpace 3D if not set.
RSR_CC_LUT_TYPE = RSR cineSpace 3D

RSR_CC_BITS = int
Sets the default bit value for the generated LUT. Range (4..9). Defaults to 5 if not set.
RSR_CC_BITS = 5

RSR_CC_SCALING = true, false
Sets whether or not scaling is desired as default by the user. Range (true , false) Defaults to true if not set.
RSR_CC_SCALING = true

RSR_CC_SCALE_FACTOR = float
Sets the default scale factor to be used when scaling is on. Range (0.0..5.0). Defaults to 1.0 if not set.
RSR_CC_SCALE_FACTOR = 1.0

RSR_CC_PREVIEW_RESOLUTION = quick, accurate
Sets the default preview resolution. Range (Quick / Accurate). Defaults to Quick if not set.
RSR_CC_PREVIEW_RESOLUTION = Quick

RSR_CC_START_PRETRANS = none, lin2log, lut
Sets pretransform to be active on startup. Range none, lin2log or lut. Defaults to none.
RSR_CC_START_PRETRANS = none

RSR_CC_FILM_TEST_IMAGE = file
Sets the path and full-name to the default film test image.
RSR_CC_FILM_TEST_IMAGE=./images/cineCubeVisual_default_film_image.cin

RSR_CC_VIDEO_TEST_IMAGE = file
Sets the path and full-name to the default film test image.
RSR_CC_FILM_TEST_IMAGE=./images/cineCubeVisual_default_video_image.png

A.9.2 Optional Override Settings

RSR_CC_XXXX overrides RSR_XXXX

RSR_CC_MONITOR_REGEXP = regexp
RSR applications will look in `RSR_MONITOR_DIR` for profiles matched to the regexp `RSR_MONITOR_REGEXP` to use as the default monitor profile, if more than one match is found the file with the most recent creation/modification date is used. More detail is provided in section A.14.
 # Pick out the most recent file containing the host name ending in .xml
RSR_CC_MONITOR_REGEXP = (%HOST).\.xml$*

RSR_CC_TARGET_REGEXP = regexp
RSR applications will look in `RSR_TARGET_DIR` for profiles matched to the regexp `RSR_TARGET_REGEXP` to use as the default target profile, if more than one match is found the file with the most recent creation/modification date is used. More detail is provided in section A.14.
 # Pick out the RSR_ideal_monitor profile
RSR_CC_TARGET_REGEXP = ^RSR_D61_g2.2\.xml$

RSR_CC_PRETRANSFORM_LUT = file
Sets the default LUT file to use as a pretransform. See the cineSpace documentation for more information on pretransforms and the LUT file format. When using this option you must use the full path to the LUT file. Defaults to no LUT file and the user will be prompted to find one if they try to load a LUT as a pretransform.
RSR_CC_PRETRANSFORM_LUT = fullpath/lutfile

RSR_CC_PRINTERLIGHT_TRIM = float,float,float
Sets the initial trims for the printer lights. These will be the R, G and B (three doubles separated by commas) values displayed by the printer lights settings when no adjustments are being made to the currently generated LUT. Range 0.0 – 50.0. Defaults to (25.0,25.0,25.0) if not set.
RSR_CC_PRINTERLIGHT_TRIM = 25.0,25.0,25.0

RSR_CC_PRINTERLIGHT_OFFSET = float,float,float
Sets the initial offsets for the printer lights. These will be the initial R, G and B (three doubles separated by commas) offsets applied by the printer lights. Range 0.0 – 50.0. Defaults to (25.0,25.0,25.0) if not set.
RSR_CC_PRINTERLIGHT_OFFSET = 25.0,25.0,25.0

RSR_CC_PRINTERLIGHTS_PER_STOP = integer
Sets the number of printer light points per stop of exposure. Allows adjustment to cater for variations in film recorder standards. Range 2 – 12. Defaults to (8) if not set.
RSR_CC_PRINTERLIGHTS_PER_STOP = 8

RSR_CC_FILMDATA_WHITE = int
Sets film white point for log/lin conversions. Range 0 – 1023. Defaults to (685) if not set. For backward compatibility, 'FILMDATAENV_WHITE' is also accepted.
RSR_CC_FILMDATA_WHITE = 685

RSR_CC_FILMDATA_BLACK = int
Sets film black point for log/lin conversions. Range 0 – 1023. Defaults to (95) if not set. For backward compatibility, 'FILMDATAENV_BLACK' is also accepted.
RSR_CC_FILMDATA_BLACK = 95

RSR_CC_FILMDATA_D_GAMMA = float
Sets conversion/device gamma for log/lin conversions. Range 0.001 – 10.0. Defaults to (1.7) if not set. For backward compatibility, 'FILMDATAENV_D_GAMMA' is also accepted.
RSR_CC_FILMDATA_D_GAMMA = 1.7

RSR_CC_FILMDATA_F_GAMMA = float
Sets film gamma for log/lin conversions. Range 0.001 – 10.0. Defaults to (0.6) if not set. For backward compatibility, 'FILMDATAENV_FILM_GAMMA' is also accepted.
RSR_CC_FILMDATA_F_GAMMA = 0.6

RSR_CC_WHITE_DEFAULT_ON = true, false
Sets whether or not white point adaptation is done by default. Range TRUE - FALSE. Defaults to FALSE if not set.
RSR_CC_WHITE_DEFAULT_ON = FALSE

RSR_CC_WHITE_TARGET_TYPE = int
If white point adaptation is being used this sets the type of target white point for cineSpace to use options are:
0 : Use the white point of the monitor profile being used.
1 : Use a the colour temperature specified by RSR_WHITE_TEMP in Kelvin (K)
2 : Use chromaticity coordinates specified in RSR_WHITE_X and RSR_WHITE_Y.
Range 0, 1, 2. Defaults to (0) if not set.
RSR_CC_WHITE_TARGET_TYPE = 0

RSR_CC_WHITE_TEMP = int
If white point adaptation is being used this sets the white point temperature of the target profile being used. Range 4000 - 9999. Defaults to (5400) if not set.
RSR_CC_WHITE_TEMP = 5400

RSR_CC_WHITE_X = float
If white point adaptation is being used this sets the (x) chromaticity coordinate of the white point of the target profile being used. Range 0.0 - 1.0. Defaults to (0.3349) (ie. 5400K) if not set.
RSR_CC_WHITE_X = 0.3349

RSR_CC_WHITE_Y = float
If white point adaptation is being used this sets the (y) chromaticity coordinate of the white point of the target profile being used. Range 0.0 - 1.0. Defaults to (0.3497) (ie. 5400K) if not set.
RSR_CC_WHITE_Y = 0.3497

RSR_CC_BLACK_DEFAULT = on, auto, off
This overrides the variables *RSR_BLACK_DEFAULT_ON* which sets whether or not black point correction is done by default and *RSR_BLACKPOINT_FORCE_CORRECT* which if black point correction is on this option sets whether or not black point correction is done at all times or if set to false, only when the monitor device is unable to reproduce the black point of the target device. Range (on, auto, off). Defaults to auto.
RSR_CC_BLACK_DEFAULT = auto

A.9.3 Viewing Transform Settings

RSR_CC_VIEW_MONITOR_REGEXP = regexp
RSR applications will look in *RSR_MONITOR_DIR* for profiles matched to the regexp *RSR_CC_VIEW_MONITOR_REGEXP* to use as the default monitor profile for the viewing monitor profile. Defaults to *RSR_CC_VIEW_MONITOR_REGEXP* if not set.
RSR_CC_VIEW_MONITOR_REGEXP = (%HOST).*\.xml$

RSR_CC_VIEW_SCALING = true, false
Sets whether or not scaling is desired as default by the user. Range (true , false) Defaults to true if not set.
RSR_CC_VIEW_SCALING = true

RSR_CC_VIEW_BLACK_DEFAULT = on, auto, off
Sets the view monitor transform default black point correction behaviour.
RSR_CC_VIEW_BLACK_DEFAULT = auto

RSR_CC_VIEW_TRANSFORM_RESOLUTION = int
Sets the default value of the internal interpolator cube length which is used in generating the final output LUT. Range (10..60). Defaults to 40 if not set.
RSR_CC_VIEW_TRANSFORM_RESOLUTION = 40

A.9.4 Advanced Settings

RSR_CC_LUT_MAX_STIM_INPUT = int
Sets the maximum stim input for a cineSpace or ASC generated LUT. Range (0.1..100.0). Defaults to 1 if not set.
RSR_CC_LUT_MAX_STIM_INPUT = 1.0

RSR_CC_LUT_AUTO_OPTIMISE = true, false
Set this to true to automatically optimise a cineSpace or ASC generated LUT.
RSR_CC_LUT_AUTO_OPTIMISE = true

RSR_CC_LUT_SCRATCH_GAMMA = float
Sets the default gamma used when generating an Assimilate Scratch LUT. Range (0.1..10.0). Defaults to 1.0 if not set.
RSR_CC_LUT_SCRATCH_GAMMA = 1.0

RSR_CC_ENGINE_MAX_TRANSFORM_RESOLUTION = int
Sets the default value of the internal interpolator cube length which is used in generating the final output LUT. Range (10..60). Defaults to 40 if not set.
RSR_CC_ENGINE_MAX_TRANSFORM_RESOLUTION = 40

RSR_CC_GUI_BACKGROUND_COLOUR = black,dark grey,light grey,white
Set the default background colour. Range (black, dark grey, light grey, white). Defaults to black if not set.
RSR_CC_GUI_BACKGROUND_COLOUR = black

A.10 cineProfiler Options

RSR_CP_PROFILE_TYPE = normal_ramp, short_ramp, normal_cube, short_cube
Sets the default number of samples the order they are used to make a profile. The depreciated option RSR_CP_PROFILE_LENGTH is also supported. Effectively this option sets the default state of the "Profile Type" button group on the startup dialog of cineProfiler.
RSR_CP_PROFILE_TYPE = normal_ramp

RSR_CP_MONITOR_TYPE = crt, lcd, projector, broadcast_crt, broadcast_lcd
Sets the default monitor type when starting cineProfiler. The depreciated option RSR_MONITOR_TYPE is also supported. Effectively this option sets the default state of the "Profile Type" button group on the startup dialog of cineProfiler. Note: This option also sets the default monitor type when doing a confidence test in equalEyes.
RSR_CP_MONITOR_TYPE = crt

RSR_CP_CLEAR_GAMMA = true, false
If set to (false) then cineProfiler will not clear the LUTs while profiling. Defaults to (true) if not set. This option should be set to false when profiling a monitor that one of the cineSpace plug-ins will be used on, as the plug-ins do not clear the LUTs like equalEyes does.
RSR_CP_CLEAR_GAMMA = true

RSR_CP_GAIN = true, false
Sets the default "Gain R, G and B" option. Range (true or false). If not set, the gain option will default to the recommened default for the *RSR_CP_MONITOR_TYPE*
RSR_CP_GAIN = true

RSR_CP_BIAS = true, false
Sets the default "Bias R, G and B" option. Range (true or false). If not set, the bias option will default to the recommened default for the *RSR_CP_MONITOR_TYPE*
RSR_CP_BIAS = true

RSR_CP_OP_WHITE_SOURCE = PROFILE, CUSTOM
Sets how the target white point that is used during optimising is set. PROFILE means the white point info is gathered from a profile, CUSTOM means that the Luminance and either colour temperature or chromaticity of the white point need to be explicitly set, via rsr.conf options or the cineProfiler GUI.
RSR_CP_OP_WHITE_SOURCE = CUSTOM

RSR_CP_OP_PROFILE = xml profile
Sets the target profile to be used as a target when setting the white point of a display device, during the optimising stage of cineProfiler. Defaults to the what ever profile is selected via the RSR_TARGET_REGEXP option set in Section 2b above.
RSR_CP_OP_PROFILE = C:/profileDir/target.xml

RSR_CP_AUTOSAVE = true, false
When set to set true, cineProfiler will automatically save the monitor profile with the signature "%HOST%_%DATE%.xml" at the end of the profiling procedure. Defaults to false if not set.
RSR_CP_AUTOSAVE = false

RSR_CP_OP_LUMINANCE_UNITS = CANDELA, FOOTLAMBERTS
Sets the units used when setting the target white point Luminance during the optimisation stage of cineProfiler.
RSR_CP_OP_LUMINANCE_UNITS = CANDELA

RSR_CP_OP_WP_LUMINANCE = float
Target white point Luminance to use if using custom settings as a target during the optimisation stage of cineProfiler.
RSR_CP_OP_WP_LUMINANCE = 80.0

RSR_CP_OP_COLOUR_TYPE = TEMPERATURE, CHROMATICITY
Sets how the target white point colour is to be specified during the optimisation stage of cineProfiler.
RSR_CP_OP_COLOUR_TYPE = TEMPERATURE

RSR_CP_OP_WP_TEMPERATURE = 4000 - 9999
Sets the Correlated Colour Temperature (K) to use when using custom settings as a target during the optimisation stage of cineProfiler.
RSR_CP_OP_WP_TEMPERATURE = 6100

RSR_CP_OP_WP_CHROMATICITY_X = float
Sets the x chromaticity coordinate to use when using custom settings as a target during the optimisation stage of cineProfiler.
RSR_CP_OP_WP_CHROMATICITY_X = 0.3209

RSR_CP_OP_WP_CHROMATICITY_Y = float
Sets the y chromaticity coordinate to use when using custom settings as a target during the optimisation stage of cineProfiler.
RSR_CP_OP_WP_CHROMATICITY_X = 0.3307

RSR_CP_PLUGE_CENTER = int
Sets the grey value (8bit) of the centre bar of the pluge.
RSR_CP_PLUGE_CENTER = 50

RSR_CP_PLUGE_INNER = int
Sets the grey value (8bit) of the inner bars of the pluge.
RSR_CP_PLUGE_INNER = 16

RSR_CP_PLUGE_OUTER = int
Sets the grey value (8bit) of the outer bar of the pluge.
RSR_CP_PLUGE_OUTER = 4

RSR_CP_PLUGE_BLACK = int
Sets the grey value (8bit) of the black background of the pluge.
RSR_CP_PLUGE_BLACK = 0

RSR_CP_PLUGE_WHITE = int
Sets the 90% white value (8bit) of the white horizontal bars of the pluge.
RSR_CP_PLUGE_WHITE = 230

RSR_CP_GREY_PATCH_OFFSET = int
Sets an offset for the dark grey patch used when optimising. Normally this patch is the same brightness as the centre bar of the pluge, but sometimes this can be too dark and so this offset can be used to adjust the brightness if needed. Range -50 – 205. Defaults to (0) if not set.
NB: If this value is set too low then the patch becomes too dark for the probe to read reliably and it may be very difficult to optimise the monitor. If it is set too high then the gain controls on the monitor will affect the bias sliders in cineProfiler too much also making it very difficult to optimise.
RSR_CP_GREY_PATCH_OFFSET = 0

RSR_CP_ERROR_MEDIAN_DEVIATIONS = float
Sets how much a sample must be in error for it to be reread during the error checking phase. Testing has found a value of 30.0 to be low enough to catch problems but high enough to allow unaviodable probe noise through. Adjust this value upwards if you find that a lot of patches are being reread when they dont need to be, or adjust it down if you find errors being allowed through that you think should have been caught.
RSR_CP_ERROR_MEDIAN_DEVIATIONS = 30.0

RSR_CP_ERROR_RETRIES = int
Sets the number of attempts to fix failed samples during the error checking phase. If set to a high number of retries cineProfiler will be able to correct more errors but will take longer to do so, and up to a point you would be better off simply running the profile again. Range 0 - 100. Defaults to (3) if not set.
RSR_CP_ERROR_RETRIES = 3

RSR_CP_ERROR_RETRIES_PROJECTOR = int
Same as RSR_CP_ERROR_RETRIES above except this value will be used if profiling a projector. Projectors tend to have more noise in their measurements so to prevent profiling taking to long this defaults to (0).
RSR_CP_ERROR_RETRIES_PROJECTOR = 0

RSR_CP_OIP_FRAME_CHANGE_THRESHOLD = float
Sets the CIE 1994 DeltaE value that needs to be exceeded for cineProfiler to recognise a new colour patch while doing output independent profiling. Range 0.0 to 10. Defaults to (3.0) if not set.
RSR_CP_OIP_FRAME_CHANGE_THRESHOLD = 3.0

RSR_CP_PATCH_TIME = int
Sets time delay between when a colour patch is shown and when a measurement is made. THis may be used to create a longer delay if the device being profiled takes to long to stabilize after the patch changes colour. Range 500 to 9999 Defaults to (500) if not set.
RSR_CP_PATCH_TIME = 500

A.10.1 Probes (Advanced)

NB: The default ports checked for probes are:

LINUX	/dev/ttyS0 ... /dev/ttyS7 , USB ports
OSX	USB ports only
WIN32	com0 ... com7 , USB ports

RSR_SERIAL_PROBES_EXTRA =
Sets extra ports to check for a probe. Useful if the probe is on an unusual port and is not found by equalEyes / cineProfiler. Defaults to “ ” if not set (i.e. no extra ports are checked for a probe).
RSR_SERIAL_PROBES_EXTRA = /hw/ttys/ttyd1

RSR_SERIAL_PROBES_ONLY =
Sets the only port to check for a probe. Useful if the probe is on an unusual port or the user does not want equalEyes / cineProfiler checking all common ports for a probe. Defaults to “ ” if not set (i.e. only the default ports are checked for a probe).
RSR_SERIAL_PROBES_ONLY = /hw/ttys/ttyd1

RSR_USE_REMOTE_PROBE = true, false
Sets cineSpace to use a remote probe. Defaults to ‘false’ if not set. cineProfiler and equalEyes are able to connect to a probe that is being served by the rsrProbe-Server.
RSR_USE_REMOTE_PROBE = false

RSR_PROBE_SERVER = hostname
Sets the IP address or name of the machine hosting the rsrProbeServer. Defaults to ‘localhost’ if not set.
RSR_PROBE_SERVER = 10.5.0.30

RSR_PROBE_SOCKET = int
Sets the socket/port the server will listen for connections on. Defaults to (5678) if not set.
RSR_PROBE_SOCKET = 5678

A.11 GUI options

This affects cineProfiler, equalEyes and the probeServer

A.11.1 Font size

RSR_FONT_SIZE = int
Font size in points (all platforms). Defaults to platform defaults below if not set. NB: This setting is overridden by the platform specific options below if they are set.
RSR_FONT_SIZE = 9

RSR_FONT_SIZE_LINUX = int
Font size in points (Linux only). Enables users to choose a different font size for each platform if needed. Defaults to (9) if not set.
RSR_FONT_SIZE_LINUX = 9

RSR_FONT_SIZE_OSX = int
Font size in points (Mac OS X only). Enables users to choose a different font size for each platform if needed. Defaults to (12) if not set.
RSR_FONT_SIZE_OSX = 12

RSR_FONT_SIZE_WIN32 = int
Font size in points (Windows only). Enables users to choose a different font size for each platform if needed. Defaults to (9) if not set.
RSR_FONT_SIZE_WIN32 = 9

A.12 Plug-in Specific Options

RSR_PLUGIN_TARGET = xml profile
Sets the default target profile used when creating an new instance of the plugin. Defaults to Kodak_2383.xml if not set NB: The plugin will look for this profile in the directory specified in RSR_TARGET_DIR ie. RSR_APP_DIR/RSR_TARGET_DIR/
RSR_PLUGIN_TARGET = Kodak_2383.xml

RSR_PLUGIN_MONITOR = xml profile
Sets the default monitor profile used when creating an new instance of the plugin. Defaults to sRGB.xml if not set NB: The plugin will look for this profile in the directory specified in RSR_MONITOR_DIR ie. RSR_APP_DIR/RSR_MONITOR_DIR/
RSR_PLUGIN_MONITOR = sRGB.xml

RSR_PLUGIN_3D_QUICK_RES = int
Sets the internal cube resolution used when doing 3D transforms. A lower value means a quicker but less accurate transform. This is usefull when adjusting other options as the GUI is more responsive. A Higher value means a slower but more accurate transform. Range 5 - 20. Defaults to (10) if not set.
RSR_PLUGIN_3D_QUICK_RES = 10

RSR_PLUGIN_3D_ACCURATE_RES = int
Sets the internal cube resolution used when doing 3D transforms. A lower value means a quicker but less accurate transform. A Higher value means a slower but more accurate transform. Range 20 - 50. Defaults to (40) if not set.
RSR_PLUGIN_3D_ACCURATE_RES = 40

A.13 Deprecated Variables

RSR_LINEAR_MODE = grey, RGB
Sets whether channels can be seperately adjusted when doing a lin to log pre-transform. Range (grey, RGB). Defaults to (grey) if not set. Note: When set to RGB the RBG lin2log settings below can be used instead of the singel channel grey options below.
RSR_LINEAR_MODE = grey

RSR_PRETRANSFORM_LUT_INTERP_TYPE = constrainedcatmullrom, cubicbezier, linear

Sets the type of interpolation used to do lookups on the LUT file specified in RSR_PRETRANSFORM_LUT. Range constrainedcatmullrom, cubicbezier or linear. Defaults to constrainedcatmullrom.
RSR_PRETRANSFORM_LUT_INTERP_TYPE = constrainedcatmullrom

RSR_BLACKPOINT_METHOD =

MATCHBLACKS,TARGETOFFSET,NEARESTBOUNDARY,ALONGGREY
Sets the method used to do the black point correction. Almost always the default method will the best method to use but it is unsatisfactory for a given set of profiles one of the aother methods can be used to get a more accurate result. Range (MATCHBLACKS, TARGETOFFSET, NEARESTBOUNDARY, ALONGGREY). Defaults to (MATCHBLACKS) if not set.
RSR_BLACKPOINT_METHOD = MATCHBLACKS

RSR_WHITE_TEMP_CRT = int
The CRT, LCD or PROJ extension means that the option will be applied to CRT, LCD or Projector monitors only (as determined from the monitor profile that is used). If not set these all default to the RSR_WHITE_TEMP value.
RSR_WHITE_TEMP_CRT = 5400

RSR_WHITE_TEMP_LCD = int
The CRT, LCD or PROJ extension means that the option will be applied to CRT, LCD or Projector monitors only (as determined from the monitor profile that is used). If not set these all default to the RSR_WHITE_TEMP value.
RSR_WHITE_TEMP_LCD = 5400

RSR_WHITE_TEMP_PROJ = int
The CRT, LCD or PROJ extension means that the option will be applied to CRT, LCD or Projector monitors only (as determined from the monitor profile that is used). If not set these all default to the RSR_WHITE_TEMP value.
RSR_WHITE_TEMP_PROJ = 5400

RSR_WHITE_X_CRT = float
The CRT, LCD or PROJ extension means that the option will be applied to CRT, LCD or Projector monitors only (as determined from the monitor profile that is used). If not set these all default to the RSR_WHITE_X value.
RSR_WHITE_X_CRT = 0.3349

RSR_WHITE_X_LCD = float
The CRT, LCD or PROJ extension means that the option will be applied to CRT, LCD or Projector monitors only (as determined from the monitor profile that is used). If not set these all default to the RSR_WHITE_X value.
RSR_WHITE_X_LCD = 0.3349

RSR_WHITE_X_PROJ = float
The CRT, LCD or PROJ extension means that the option will be applied to CRT, LCD or Projector monitors only (as determined from the monitor profile that is used). If not set these all default to the RSR_WHITE_X value.
RSR_WHITE_X_PROJ = 0.3349

RSR_WHITE_Y_CRT = float
The CRT, LCD or PROJ extension means that the option will be applied to CRT, LCD or Projector monitors only (as determined from the monitor profile that is used). If not set these all default to the RSR_WHITE_Y value.
RSR_WHITE_Y_CRT = 0.3497

RSR_WHITE_Y_LCD = float
The CRT, LCD or PROJ extension means that the option will be applied to CRT, LCD or Projector monitors only (as determined from the monitor profile that is used). If not set these all default to the RSR_WHITE_Y value.
RSR_WHITE_Y_LCD = 0.3497

RSR_WHITE_Y_PROJ = float
The CRT, LCD or PROJ extension means that the option will be applied to CRT, LCD or Projector monitors only (as determined from the monitor profile that is used). If not set these all default to the RSR_WHITE_Y value.
RSR_WHITE_Y_PROJ = 0.3497

RSR_WHITE_TARGET_TYPE_CRT = int
The CRT, LCD or PROJ extension means that the option will be applied to CRT, LCD or Projector monitors only (as determined from the monitor profile that is used). If not set these all default to the RSR_WHITE_TARGET_TYPE value.
RSR_WHITE_TARGET_TYPE_CRT = 0

RSR_WHITE_TARGET_TYPE_LCD = int
The CRT, LCD or PROJ extension means that the option will be applied to CRT, LCD or Projector monitors only (as determined from the monitor profile that is used). If not set these all default to the RSR_WHITE_TARGET_TYPE value.
RSR_WHITE_TARGET_TYPE_LCD = 0

RSR_WHITE_TARGET_TYPE_PROJ = int
The CRT, LCD or PROJ extension means that the option will be applied to CRT, LCD or Projector monitors only (as determined from the monitor profile that is used). If not set these all default to the RSR_WHITE_TARGET_TYPE value.
RSR_WHITE_TARGET_TYPE_PROJ = 0

RSR_WHITE_DEFAULT_ON_CRT = true, false
The CRT, LCD or PROJ extension means that the option will be applied to CRT, LCD or Projector monitors only (as determined from the monitor profile that is used). If not set these all default to the RSR_WHITE_DEFAULT_ON value.
RSR_WHITE_DEFAULT_ON_CRT = FALSE

RSR_WHITE_DEFAULT_ON_LCD = true, false
The CRT, LCD or PROJ extension means that the option will be applied to CRT, LCD or Projector monitors only (as determined from the monitor profile that is used). If not set these all default to the RSR_WHITE_DEFAULT_ON value.
RSR_WHITE_DEFAULT_ON_LCD = FALSE

RSR_WHITE_DEFAULT_ON_PROJ = true, false
The CRT, LCD or PROJ extension means that the option will be applied to CRT, LCD or Projector monitors only (as determined from the monitor profile that is used). If not set these all default to the RSR_WHITE_DEFAULT_ON value.
RSR_WHITE_DEFAULT_ON_PROJ = FALSE

A.14 Regular Expressions

Instead of selecting a monitor using name alone we can match the most recent version of the file matching a given regular regular expression using RSR_MONITOR_REGEXP. From all matching filenames the one chosen is

1. The one with the highest (sorted alphabetically) last capture.
2. If more than one have same highest last capture, then most recently modified of these.

For example: Example: regexp = `^file(\d\d\d\d)a` is interpreted as the beginning of the string: `[^]file(\d\d\d\d)a`,
followed by the string "file": `^[file](\d\d\d\d)a`,
followed by a capture: `^file[(]\d\d\d\d[)]a`
of four digits: `^file([\d\d\d\d])a`
finally followed by an "a": `^file(\d\d\d\d)[a]`.

That is the regexp matches the files file0000a, file0000abc, file1234abc and file1234avc but not other_file or file0000b or file1234 or file000x. Of the matches both file1234abc and file1234avc match the last capture, `(\d\d\d\d)`, with "1234" which is greater than the "0000" of the other files. Thus the file chosen will be the most recently modified of these two files.

A.15 Example

```
# RSR cineSpace v2.7
# Configuration file (11-July-2008)
#
# RSR applications read configuration files in the following places. (in the following order)
# 1) The RSR application directory (if defined in the environment)   ( $RSR_APP_DIR )
# 2) The users home directory (if defined in the environment)        ( $HOME )
# 3) The directory the application was started from                  ( ./)
#
# with variables set in the later files overwriting those set in earlier files.
#
# Finally the values defined in the environment are loaded and overwrite any of these variables
#

# CONTENTS
#
# Section 1: License Settings
# Section 2: Global Environment
#   Section 2a: Directory Settings
#   Section 2b: Default Profile Settings
#   Section 2c: IO settings
# Section 3: Global cineEngine Settings
#   Section 3a: Printer Lights options
#   Section 3b: Lin / Log settings
#   Section 3c: Black Point Correction
#   Section 3d: White Point Adaptation
# Section 4: equalEyes options
#   Section 4a: Start up options
#   Section 4b: General options
#   Section 4c: Synthetic profile options
# Section 5: cineCube options
#   Section 5a: Startup options
#   Section 5b: Optional Override Settings
#   Section 5c: Viewing Transform Settings
#   Section 5d: Advanced Settings
# Section 6: cineProfiler options
# Section 7: Probes (advanced)
# Section 8: GUI options
# Section 9: Plugin specific options
# Section 10: Depricated


### Section 1: License Settings
```

```
#

#
# RSR_LICENSE: Sets the path to your license file or server.
# An absolute path works best as relative paths are relative to the
# directory that the application was run from not the directory where
# the application is located (unless the RSR_APP_DIR below is set).
#
#RSR_LICENSE = ./rsun.dat




### Section 2: Global Environment
#
# Settings that effect most RSR applications and plugins
#


### Section 2a: Directory Settings
#

#
# RSR_MONITOR_DIR: Sets the location of your monitor profiles.
#
#RSR_MONITOR_DIR = /Applications/CineSpace/monitor-profiles

#
# RSR_TARGET_DIR: Sets the location of your target profiles.
#
#RSR_TARGET_DIR = /Applications/CineSpace/target-profiles

#
# RSR_APP_DIR: A last resort place to look for shared RSR resources.
# Enter the absolute (not relative) path to the directory that has
# the application executable in it.
#
#RSR_APP_DIR = ./


### Section 2b: Default Profile Settings
#

#
# RSR_MONITOR_REGEXP: RSR applications will look in RSR_MONITOR_DIR
# for profiles matched to the regexp RSR_MONITOR_REGEXP to use as the
# default monitor profile, if more than one match is found the
# file with the most recent creation/modification date is used.
#
# Pick out the most recent file containing the hostname ending in .xml
#
RSR_MONITOR_REGEXP = (%{HOST}).*\.xml$

#
# RSR_TARGET_REGEXP: RSR applications will look in RSR_TARGET_DIR
# for profiles matched to the regexp RSR_TARGET_REGEXP to use as the
# default target profile, if more than one match is found the
# file with the most recent creation/modification date is used.
#
# Pick out the RSR_D61_g2.2 profile
# (white point: 6100K, gamma: 2.2 ).
RSR_TARGET_REGEXP = ^RSR_D61_g2\.2\.xml$


#
# RSR_PRETRANSFORM_LUT: Sets the default LUT file to use as a
# pretransform. See the cineSpace documentation for more information
# on pretransforms and the LUT file format.
```

```
# When using this option you must use the full path to the lut file.
# Defaults to no lut file and the user will be prompted to find one
# if they try to load a LUT as a pretransform.
#
#RSR_PRETRANSFORM_LUT = full/path/to/lutfile

#
# RSR_CRT_LIFETIME: Sets the lifetime of CRT profiles.
# The lifetime of a profile is the number of days after the profile was created
# that the profile is considered to be valid for.
# Range -1 <-> 9999. Defaults to (14) if not set.
# The special value of -1 makes the profile permanently valid.
#
RSR_CRT_LIFETIME = 14

#
# RSR_LCD_LIFETIME: Same as RSR_CRT_LIFETIME above.
# Range -1 <-> 9999. Defaults to (28) if not set.
# The special value of -1 makes the profile permanently valid.
#
RSR_LCD_LIFETIME = 28

#
# RSR_PROJECTOR_LIFETIME: Same as RSR_CRT_LIFETIME above.
# Range -1 <-> 9999. Defaults to (90) if not set.
# The special value of -1 makes the profile permanently valid.
#
RSR_PROJECTOR_LIFETIME = 90



### Section 2c: IO settings
#
# Settings that generally effect console output from
# RSR_OUTPUT and RSR_LOG_LEVEL options:
#
# [RSR_OUTPUT] [RSR_LOG_LEVEL] [What you actually get]
#
#   NOTHING                        Nothing
#   QUIET            0             Fatal errors
#   ERROR            1             Non-fatal errors + QUIET
#   WARNING          2             Warnings         + ERROR
#   VERBOSE          3             Other comments   + WARNING
#

#
# RSR_OUTPUT: {NOTHING, QUIET, ERROR, WARNING, VERBOSE, LOG}
# If RSR_OUTPUT = LOG then RSR_LOG_LEVEL and RSR_LOG_FILE
# should be set. (see below)
#
RSR_OUTPUT = QUIET

#
# RSR_LOG_LEVEL: Sets level of log output {0, 1, 2, 3, 4}
#
RSR_LOG_LEVEL = 0

#
# RSR_LOG_FILE: Sets path to file where logged output should be placed.
# If not set the log file will have the same name as the application and
# will be located in the directory the application was run from.
#
#RSR_LOG_FILE = rsr.log



### Section 3: Global cineEngine Settings
```

```
#
# Settings that change the cineSpace engine parameters
# for transforming data for most RSR applications and plugins
#

### Section 3a: Printer Lights options
#
#
# RSR_PRINTERLIGHT_TRIM: Sets the initial trims for the printer lights.
# Theses will be the R, G and B (three doubles separated by commas) values
# displayed by the printer lights settings when no adjustments are being
# made to the currently generated LUT.
# Range 0.0 <-> 50.0. Defaults to (25.0,25.0,25.0) if not set.
#
RSR_PRINTERLIGHT_TRIM = 25.0,25.0,25.0

#
# RSR_PRINTERLIGHT_OFFSET: Sets the initial offsets for the printer lights.
# Theses will be the initial R, G and B (three doubles separated by commas)
# offsets applied by the printer lights.
# Range 0.0 <-> 50.0. Defaults to (25.0,25.0,25.0) if not set.
#
RSR_PRINTERLIGHT_OFFSET = 25.0,25.0,25.0

#
# RSR_PRINTERLIGHTS_PER_STOP: Sets the approximate number of printerlight
# points per stop. (double)
# Range 2 <-> 20. Defaults to (8.0) if not set.
#
RSR_PRINTERLIGHTS_PER_STOP = 8.0


### Section 3b: Lin / Log settings
#

#
# grey lin/log options
#
# RSR_FILMDATA_WHITE: Sets film white point for log/lin conversions.
# Range 0 <-> 1023. Defaults to (685) if not set.
# For backward compatibility 'FILMDATAENV_WHITE' is also accepted.
#
RSR_FILMDATA_WHITE = 685

#
# RSR_FILMDATA_BLACK: Sets film black point for log/lin conversions.
# Range 0 <-> 1023. Defaults to (95) if not set.
# For backward compatibility 'FILMDATAENV_BLACK' is also accepted.
#
RSR_FILMDATA_BLACK = 95

#
# RSR_FILMDATA_D_GAMMA: Sets conversion/device gamma for log/lin conversions.
# Range 0.001 <-> 10.0. Defaults to (1.7) if not set.
# For backward compatibility 'FILMDATAENV_D_GAMMA' is also accepted.
#
RSR_FILMDATA_D_GAMMA = 1.7

#
# RSR_FILMDATA_F_GAMMA: Sets film gamma for log/lin conversions.
# Range 0.001 <-> 10.0. Defaults to (0.6) if not set.
# For backward compatibility 'FILMDATAENV_FILM_GAMMA' is also accepted.
#
RSR_FILMDATA_F_GAMMA = 0.6

#
# RSR_FILMDATA_SOFT_CLIP: Sets soft-clip threshold for log/lin conversions.
# Range 0 <-> 1023. Defaults to (0) if not set.
```

```
# For backward compatibility 'FILMDATAENV_SOFT_CLIP' is also accepted.
#
RSR_FILMDATA_SOFT_CLIP = 0


### Section 3c: Black Point Correction
#
#
#
# RSR_BLACK_DEFAULT_ON: Sets whether or not black point correction is done by default.
# Range (true, false). Defaults to (true) if not set.
#
RSR_BLACK_DEFAULT_ON = true

#
# RSR_BLACKPOINT_FORCE_CORRECT: If black point correction is on this option sets
# whether or not black point correction is done at all times or if set to false, only
# when the monitor device is unable to reproduce the black point of the target device.
# Range (true, false). Defaults to (false) if not set.
#
RSR_BLACKPOINT_FORCE_CORRECT = false


### Section 3d: White Point Adaptation
#
#
# RSR_WHITE_DEFAULT_ON: Sets whether or not white point adaptation is done by default.
# Range TRUE <-> FALSE. Defaults to FALSE if not set.
#
RSR_WHITE_DEFAULT_ON = FALSE

#
# RSR_WHITE_TARGET_TYPE: If white point adaptation is being used this sets the type
# of target white point for cineSpace to use options are:
# 0 : Use the white point of the monitor profile being used.
# 1 : Use a the colour temperature specified by RSR_WHITE_TEMP in Kelvin (K)
# 2 : Use chromaticity coordinates specified in RSR_WHITE_X and RSR_WHITE_Y.
# Range 0, 1, 2. Defaults to (0) if not set.
#
RSR_WHITE_TARGET_TYPE = 0

#
# RSR_WHITE_TEMP: If white point adaptation is being used this sets the white point
# temperature of the target profile being used.
# Range 4000 <-> 9999. Defaults to (5400) if not set.
#
RSR_WHITE_TEMP = 5400

#
# RSR_WHITE_X: If white point adaptation is being used this sets the (x) chromaticity
# coordinate of the white point of the target profile being used.
# Range 0.0 <-> 1.0. Defaults to (0.3349) (ie. 5400K) if not set.
#
RSR_WHITE_X = 0.3349

#
# RSR_WHITE_Y: If white point adaptation is being used this sets the (y) chromaticity
# coordinate of the white point of the target profile being used.
# Range 0.0 <-> 1.0. Defaults to (0.3497) (ie. 5400K) if not set.
```

```
#
RSR_WHITE_Y = 0.3497




### Section 4: equalEyes options
#

### Section 4a: Start up options
#
#
# RSR_EE_INIT_STATE: Sets whether EqualEyes starts active or inactive.
# Range (inactive, active). Defaults to (inactive) if not set.
#
RSR_EE_INIT_STATE = inactive

#
# RSR_EE_INIT_SIZE: Sets whether EqualEyes starts minimised or not.
# Range (normal, minimised). Defaults to (normal) if not set.
#
RSR_EE_INIT_SIZE = normal

#
# RSR_EE_INACTIVE_GUI:Sets whether users can interact with the
# EqualEyes gui. This is for users who would like to have all EqualEyes
# settings set by the rsr.conf file and /or presets and does not want
# other users to be able to change any of the settings.
# Note: If this option is set to TRUE then RSR_EE_INIT_SIZE will default to
# minimised and RSR_EE_INIT_STATE will default to active.
# Range (true, false). Defaults to (false) if not set.
#
RSR_EE_INACTIVE_GUI = false

#
# RSR_EE_INIT_TARGET: Sets whether EqualEyes starts with an xml or synthetic profile.
# Range (xml, synthetic). Defaults to (xml) if not set.
#
RSR_EE_INIT_TARGET = xml

#
# RSR_SHOW_DECOUPLING_WARNING: Sets whether or not a warning dialog
# is displayed when the monitor profile does not decouple.
# ie. x*R + x*G + x*B != x*White        (where 0.0 <= x <= 1.0)
# Defaults to TRUE if not set.
#
RSR_SHOW_DECOUPLING_WARNING = TRUE

#
# RSR_START_PRETRANS: Sets pretransform to be active on startup.
# Range none, lin2log or lut. Defaults to 'none' (the same as
# not setting any transform).
#
RSR_START_PRETRANS = none


### Section 4b: General options
#
#
# RSR_EE_SCALING: Sets whether or not scaling is desired as default by the user
# Range (true , false) Defaults to true if not set.
#
RSR_EE_SCALING = true

#
```

```
# RSR_EE_CONFIDENCE_DE_MAX: Sets maximum allowable error values in
# the confidence test. Errors are CIE(1994)L*a*b where values greater
# than 2 are considered just perceivably different.
# Range 1.0 <-> 10.0 Defaults to (4.0) if not set.
#
RSR_EE_CONFIDENCE_DE_MAX = 4.0

#
# RSR_EE_CONFIDENCE_DE_AVG: Sets average allowable error values in
# the confidence test. Errors are CIE(1994)L*a*b where values greater
# than 2 are considered just perceivably different.
# Range 0.1 <-> 5.0 Defaults to (3.0) if not set.
#
RSR_EE_CONFIDENCE_DE_AVG = 3.0

#
# RSR_EE_CONFIRM_EXIT: Sets whether or not to check that the
# user really wants to exit when equalEyes is closed by the user.
# Range (true , false) Defaults to true (user is prompted)if not set.
#
RSR_EE_CONFIRM_EXIT = true

#
# RSR_EE_MONPROF_LIST: Provides a way for users to add extra filters
# to the Monitor Profile file Dialogs. The default filter is
# "Profiles (*.xml)" which will display all files with the .xml extension.
# You can add further filters here by specifying each new filter separated
# by ";;" eg. "Synth Profiles (*synth.xml);;Host Profiles (%{HOST}*.xml)"
#
# The filter below will list all files containing the hostname and ending in .xml
#
RSR_EE_MONPROF_LIST = "Host␣Profiles␣(%{HOST}*.xml)"


### Section 4c: Synthetic profile options
#
#
# RSR_SYNTH_CURVE_TYPE: Sets the default type of tonal curve to
# use for the synthetic profile.
# Range (gamma, cineon). Defaults to gamma if not set.
#
RSR_SYNTH_CURVE_TYPE = gamma

#
# RSR_SYNTH_GAMMA: Default gamma for synthetic profiles.
# Range 0.001 <-> 9.999. Defaults to 2.20 if not set.
#
RSR_SYNTH_GAMMA = 2.20

#
# RSR_SYNTH_WP_TYPE: Default white point type for synthetic profiles.
# Range (chromaticity, temperature, monitor). Defaults to (temperature)
# if not set.
#
RSR_SYNTH_WP_TYPE = temperature

#
# RSR_SYNTH_TEMP_TYPE: Default white Correlated Colour Temperature (CCT)
# type for synthetic profiles (If RSR_SYNTH_WP_TYPE = TEMPERATURE).
# Range BLACKBODY or CIE. Defaults to (CIE).
# (CIE D Illuminants nearly always used when specifying white-point colour)
#
RSR_SYNTH_TEMP_TYPE = CIE

#
# RSR_SYNTH_TEMP: Default white point Correlated Colour Temperature (CCT)
# for synthetic profiles.
```

```
# Range 4000 <-> 9999. Defaults to (6500) if not set.
#
RSR_SYNTH_TEMP = 6500

#
# RSR_SYNTH_X: Default white point CIE x-chromaticity co-ordinate
# for synthetic profiles.
# Range 0.001 <-> 0.999. Defaults to (0.3127) (CIE D65) if not set.
#
RSR_SYNTH_X = 0.3127

#
# RSR_SYNTH_Y: Default white point CIE y-chromaticity co-ordinate
# for synthetic profiles.
# Range 0.001 <-> 0.999. Defaults to (0.3290) (CIE D65) if not set.
#
RSR_SYNTH_Y = 0.3290


### Section 5: cineCube Visual options
#

### Section 5a: Default Startup Settings
#

#
# RSR_CC_LUT_TYPE: Sets the default LUT type to be generated.
# Defaults to cineSpace 3D if not set.
#
#RSR_CC_LUT_TYPE = "RSR cineSpace 3D"

#
# RSR_CC_BITS: Sets the default bit value for the generated LUT.
# Range ( 4 .. 9 ). Defaults to 5 if not set.
#
#RSR_CC_BITS = 5

#
# RSR_CC_SCALING: Sets whether or not scaling is desired as default by the
user
# Range (true , false) Defaults to true if not set.
#
#RSR_CC_SCALING = true

#
# RSR_CC_SCALE_FACTOR: Sets the default scale factor to be used when
# scaling is on. Range (0.0 .. 5.0). Defaults to 1.0 if not set.
#
#RSR_CC_SCALE_FACTOR = 1.0

#
# RSR_CC_PREVIEW_RESOLUTION: Sets the default preview resolution.
# Range ( Quick / Accurate ). Defaults to Quick if not set.
#
#RSR_CC_PREVIEW_RESOLUTION = Quick

#
# RSR_CC_START_PRETRANS: Sets pretransform to be active on startup.
# Range none, lin2log or lut. Defaults to none.
#
#RSR_CC_START_PRETRANS = none

#
# RSR_CC_FILM_TEST_IMAGE: Sets the path and full-name to
# the default film test image.
#
#RSR_CC_FILM_TEST_IMAGE=./images/cineCubeVisual_default_film_image.bmp

```

```
#
# RSR_CC_VIDEO_TEST_IMAGE: Sets the path and full-name to
# the default video test image.
#
#RSR_CC_VIDEO_TEST_IMAGE=./images/cineCubeVisual_default_video_image.bmp


### Section 5b: Optional Override Settings
#
# RSR_CC_XXXX overrides RSR_XXXX
#

#RSR_CC_MONITOR_REGEXP = (%{HOST}).*\.xml$
#RSR_CC_TARGET_REGEXP = ^RSR_D61_g2\.2\.xml$
#RSR_CC_PRETRANSFORM_LUT = full/path/to/lutfile
#RSR_CC_PRINTERLIGHT_TRIM = 25.0,25.0,25.0
#RSR_CC_PRINTERLIGHT_OFFSET = 25.0,25.0,25.0
#RSR_CC_PRINTERLIGHTS_PER_STOP = 8.0
#RSR_CC_FILMDATA_WHITE = 685
#RSR_CC_FILMDATA_BLACK = 95
#RSR_CC_FILMDATA_D_GAMMA = 1.7
#RSR_CC_FILMDATA_F_GAMMA = 0.6
#RSR_CC_WHITE_DEFAULT_ON = FALSE
#RSR_CC_WHITE_TARGET_TYPE = 0
#RSR_CC_WHITE_TEMP = 5400
#RSR_CC_WHITE_X = 0.3349
#RSR_CC_WHITE_Y = 0.3497

#
# RSR_CC_BLACK_DEFAULT: This overrides the variables
# RSR_BLACK_DEFAULT_ON and RSR_BLACKPOINT_FORCE_CORRECT.
#
# Range ( on / auto / off ). Defaults to auto.
#
#RSR_CC_BLACK_DEFAULT = auto


### Section 5c: Viewing Transform Settings
#

# RSR_CC_VIEW_MONITOR_REGEXP: RSR applications will look in RSR_MONITOR_DIR
# for profiles matched to the regexp RSR_CC_VIEW_MONITOR_REGEXP to use as the
# default monitor profile for the viewing monitor profile.
#
# Defaults to RSR_CC_MONITOR_REGEXP if not set.
#
#RSR_CC_VIEW_MONITOR_REGEXP = (%{HOST}).*\.xml$

#
# RSR_CC_VIEW_SCALING: Sets whether or not scaling is desired as default
# for the view monitor transform.
# Range (true , false) Defaults to true if not set.
#
#RSR_CC_VIEW_SCALING = true

#
# RSR_CC_VIEW_BLACK_DEFAULT: Sets the view monitor transform default
# black point correction behaviour.
# Range ( on, auto, off ). Defaults to auto.
#
#RSR_CC_VIEW_BLACK_DEFAULT = auto

#
# RSR_CC_VIEW_TRANSFORM_RESOLUTION: Sets the default value
# of the internal interpolator cube length which is used in
# generating the viewing transform.
# Range (10 .. 60). Defaults to 40 if not set.
#RSR_CC_VIEW_TRANSFORM_RESOLUTION = 40
```

```

### Section 5d: Advanced Settings
#

#
# RSR_CC_LUT_MAX_STIM_INPUT: Sets the maximum stim input for
# a cineSpace or ASC generated LUT.
# Range (0.1 .. 100.0 ). Defaults to 1 if not set.
#
#RSR_CC_LUT_MAX_STIM_INPUT = 1.0

#
# RSR_CC_LUT_AUTO_OPTIMISE: Set this to true to automatically
# optimise a cineSpace or ASC generated LUT.
# Range (true, false). Defaults to true.
#
#RSR_CC_LUT_AUTO_OPTIMISE = true

#
# RSR_CC_LUT_SCRATCH_GAMMA: Sets the default gamma used when
# generating an Assimilate Scratch LUT.
# Range (0.1 .. 10.0). Defaults to 1.0 if not set.
#
#RSR_CC_LUT_SCRATCH_GAMMA = 1.0

#
# RSR_CC_ENGINE_TRANSFORM_RESOLUTION: Sets the default value
# of the internal interpolator cube length which is used in
# generating the final output LUT.
# Range (10 .. 60). Defaults to 40 if not set.
#
#RSR_CC_ENGINE_TRANSFORM_RESOLUTION = 40

#
# RSR_CC_GUI_BACKGROUND_COLOUR: Set the default background colour.
# Range (black, dark grey, light grey, white). Defaults to black if not set.
#
#RSR_CC_GUI_BACKGROUND_COLOUR = black.



### Section 6: cineProfiler options
#

#
# RSR_CP_PROFILE_TYPE: Sets the default number of samples the order they
# are used to make a profile.
# The deprecated option RSR_CP_PROFILE_LENGTH is also supported.
# Range (normal_ramp, short_ramp, normal_cube, short_cube).
# Defaults to (normal_ramp) if not set.
# Effectively this option sets the default state of the "Profile Type" button
  group
# on the startup dialog of cineProfiler.
#
RSR_CP_PROFILE_TYPE = normal_ramp

#
# RSR_CP_MONITOR_TYPE: Sets the default monitor type when starting
cineProfiler.
# The deprecated option RSR_MONITOR_TYPE is also supported.
# Range (crt , lcd, projector, broadcast_crt, broadcast_lcd).
# Defaults to (crt) if not set.
# Effectively this option sets the default state of the "Profile Type" button
  group
# on the startup dialog of cineProfiler.
# Note: This option also sets the default monitor type when doing a
confidence test
# in equalEyes.
```

```
#
RSR_CP_MONITOR_TYPE = crt

#
# RSR_CP_CLEAR_GAMMA: If set to 'false' then cineProfiler wont clear the LUTs while profiling.
# Range (true , false). Defaults to (true) if not set.
# This option should be set to false when profiling a monitor that one of the cineSpace
# plugins is going to be used on. As the plugins do not clear the LUTs like equalEyes does.
#
RSR_CP_CLEAR_GAMMA = true

#
# RSR_CP_GAIN: Sets the default "Gain R, G and B" option.
# Range ( true or false)
# If not set, the gain option will default to the recommened default for the RSR_CP_MONITOR_TYPE.
#
# RSR_CP_GAIN = true

#
# RSR_CP_BIAS: Sets the default "Bias R, G and B" option.
# Range ( true or false)
# If not set, the bias option will default to the recommened default for the RSR_CP_MONITOR_TYPE.
#
# RSR_CP_BIAS = true

#
# RSR_CP_OP_WHITE_SOURCE: Sets how the target white-point that is used during optimising
# is set.
# Range (PROFILE or CUSTOM). Defaults to CUSTOM if not set.
# PROFILE means the white point info is gathered from a profile, CUSTOM means that the
# Luminance and either colour temperature or chromaticity of the white point need to be
# explicitly set, via rsr.conf options or the cineProfiler GUI.
#
RSR_CP_OP_WHITE_SOURCE = CUSTOM

#
# RSR_CP_OP_PROFILE: Sets the target profile to be used as a target when setting the
# white point of a display device, during the optimising stage of cineProfiler.
# The default directory is set to RSR_TARGET_DIR.
#
RSR_CP_OP_PROFILE = RSR_D61_g2.2.xml


#
# RSR_CP_AUTOSAVE: When to set true, cineProfiler will automatically save the
# monitor profile with the signature "%HOST%_%DATE%.xml". The options for this
# variable are true or false. This defaults to false if not set.
#
RSR_CP_AUTOSAVE = false


#
# RSR_CP_OP_LUMINANCE_UNITS: Sets the units used when setting the target white point Luminance
# during the optimisation stage of cineProfiler.
# Range(CANDELA or FOOTLAMBERTS). Defaults to CANDELA (cd/m^2) if not set.
#
```

```
RSR_CP_OP_LUMINANCE_UNITS = CANDELA

#
# RSR_CP_OP_WP_LUMINANCE: Target white point Luminance to use if using custom settings as a
# target during the optimisation stage of cineProfiler.
# Range(20.0 - 500.0). Defaults to (80.0) if not set.
#
RSR_CP_OP_WP_LUMINANCE = 80.0

#
# RSR_CP_OP_COLOUR_TYPE: Sets how the target white point colour is to be specified
# during the optimisation stage of cineProfiler.
# Range(TEMPERATURE or CHROMATICITY). Defaults to (TEMPERATURE) if not set.
#
RSR_CP_OP_COLOUR_TYPE = TEMPERATURE

#
# RSR_CP_OP_WP_TEMPERATURE: Sets the Correlated Colour Temperature (K) to use when using
# custom settings as a target during the optimisation stage of cineProfiler.
# Range(4000 - 9999). Defaults to (6100) if not set.
#
RSR_CP_OP_WP_TEMPERATURE = 6100

#
# RSR_CP_OP_WP_CHROMATICITY_X: Sets the x chromaticity coordinate to use when using
# custom settings as a target during the optimisation stage of cineProfiler.
# Range(0.1 - 0.9). Defaults to (0.3209) if not set. Note: This is the the same as 6100K.
#
RSR_CP_OP_WP_CHROMATICITY_X = 0.3209

#
# RSR_CP_OP_WP_CHROMATICITY_Y: Sets the y chromaticity coordinate to use when using
# custom settings as a target during the optimisation stage of cineProfiler.
# Range(0.1 - 0.9). Defaults to (0.3307) if not set. Note: This is the the same as 6100K.
#
RSR_CP_OP_WP_CHROMATICITY_Y = 0.3307

#
# RSR_CP_PLUGE_CENTER: Sets the grey value (8bit) of the center bar of the pluge.
# Range (0 , 255). Defaults to (50) if not set.
#
RSR_CP_PLUGE_CENTER = 50

#
# RSR_CP_PLUGE_INNER: Sets the grey value (8bit) of the inner bars of the pluge.
# Range (0 , 255). Defaults to (16) if not set.
#
RSR_CP_PLUGE_INNER = 16

#
# RSR_CP_PLUGE_OUTER: Sets the grey value (8bit) of the outer bar of the pluge.
# Range (0 , 255). Defaults to (4) if not set.
#
RSR_CP_PLUGE_OUTER = 4

#
# RSR_CP_PLUGE_BLACK: Sets the grey value (8bit) of the black background of the pluge.
```

```
# Range (0 , 255). Defaults to (0) if not set.
#
RSR_CP_PLUGE_BLACK = 0

#
# RSR_CP_PLUGE_WHITE: Sets the 90% white value (8bit) of the white horizontal
# bars of the pluge.
# Range (0 , 255). Defaults to (230) if not set.
#
RSR_CP_PLUGE_WHITE = 230

#
# RSR_CP_GREY_PATCH_OFFSET: Sets an offset for the dark grey patch
# used when optimising. Normally this patch is the same brightness
# as the center bar of the pluge but sometimes this can be too dark,
# so this offset can be used to change the brightness if needed.
# Range -50 <-> 205. Defaults to (0) if not set.
#
# NB: If this value is set to low then the patch becomes to dark for the probe
# to read reliably and it may be very difficult to optimise the monitor. If it
# is set to high then the gain controls on the monitor will effect the bias sliders
# in cineProfiler to much also making it very difficult to optimise.
#
RSR_CP_GREY_PATCH_OFFSET = 0

#
# RSR_CP_ERROR_MEDIAN_DEVIATIONS: Sets how much a sample must be in error
# for it to be reread during the error checking phase. Testing has found
# a value of 30.0 to be low enough to catch problems but high enough
# to allow unavoidable probe noise through. Adjust this value upwards if you
# find that a lot of patches are being reread when they don't need to be, or
# adjust it down if you find errors being allowed through that you think should
# have been caught.
#
RSR_CP_ERROR_MEDIAN_DEVIATIONS = 30.0

#
# RSR_CP_ERROR_RETRIES: Sets the number of attempts to fix failed samples
# during the error checking phase. If set to a high number of retries
# cineProfiler will be able to correct more errors but will take longer to
# do so, and up to a point you would be better off simply running the profile again.
# Range 0 -> 100. Defaults to (3) if not set.
#
RSR_CP_ERROR_RETRIES = 3

#
# RSR_CP_ERROR_RETRIES_PROJECTOR: Same as RSR_CP_ERROR_RETRIES above except
# this value will be used if profiling a projector. Projectors tend to have
# more noise in their measurements so to prevent profiling taking to long this
# defaults to (0).
#
RSR_CP_ERROR_RETRIES_PROJECTOR = 0

#
# RSR_CP_OIP_FRAME_CHANGE_THRESHOLD: Sets the CIE 1994 DeltaE value that needs
# to be exceeded for cineProfiler to recognise a new colour patch while doing
# output independent profiling.
# Range 0.0 <-> 10. Defaults to (3.0) if not set.
#
RSR_CP_OIP_FRAME_CHANGE_THRESHOLD = 3.0
```

```
#
# RSR_CP_PATCH_TIME: Sets time delay between when a colour patch is shown and when
# a measurement is made. THis may be used to create a longer delay if the device
# being profiled takes to long to stabilize after the patch changes colour.
# Range 500 <-> 9999 Defaults to (500) if not set.
#
RSR_CP_PATCH_TIME = 500


### Section 7: Probes (advanced)
#
#
# NB: The default ports checked for probes are:
#
# IRIX  :        /dev/ttyd1 ... /dev/ttyd8
# LINUX :        /dev/ttyS0 ... /dev/ttyS7 , USB ports
# OSX            :       USB ports only
# WIN32 :        com0 ... com7 , USB ports
#

#
# RSR_SERIAL_PROBES_EXTRA: Sets extra ports to check for a probe.
# Useful if the probe is on an unusual port and is not found by equalEyes/cineProfiler.
# Defaults to " " if not set (ie. no extra ports are checked for a probe).
#
#RSR_SERIAL_PROBES_EXTRA = /hw/ttys/ttyd1

#
# RSR_SERIAL_PROBES_ONLY: Sets the only port to check for a probe.
# Useful if the probe is on an unusual port or the user does not want equalEyes/cineProfiler
# checking all common ports for a probe.
# Defaults to " " if not set (ie. only the default ports are checked for a probe).
#
#RSR_SERIAL_PROBES_ONLY = /hw/ttys/ttyd1

#
# RSR_USE_REMOTE_PROBE: Sets cineSpace to use a remote probe.
# Range (true, false). Defaults to false if not set.
# cineProfiler and equalEyes are ale to connect to a probe that is being served
# by the rsrProbeServer.
#
RSR_USE_REMOTE_PROBE = false

#
# RSR_PROBE_SERVER: Sets the IP address or name of the machine hosting the rsrProbeServer.
# Defaults to localhost if not set.
#
RSR_PROBE_SERVER = localhost

#
# RSR_PROBE_SOCKET: Sets the socket /port the server will listen for connections on.
# Defaults to 5678 if not set.
#
RSR_PROBE_SOCKET = 5678



### Section 8: GUI options
#
# This affects cineProfiler, equalEyes and the probeServer
```

```

### Section 8a: Font Size
#
#
# RSR_FONT_SIZE: Font size in points (all platforms).
# Defaults to platform defaults below if not set.
# NB: This setting is overridden by the platform specific options
# below if they are set
#
#RSR_FONT_SIZE = 9

#
# RSR_FONT_SIZE_LINUX: Font size in points (Linux only).
# Enables users to choose a different font size for each platform if needed.
# Defaults to 9 if not set.
#
#RSR_FONT_SIZE_LINUX = 9

#
# RSR_FONT_SIZE_OSX: Font size in points (MacOSX only).
# Enables users to choose a different font size for each platform if needed.
# Defaults to 12 if not set.
#
#RSR_FONT_SIZE_OSX = 12

#
# RSR_FONT_SIZE_WIN32: Font size in points (Windows only).
# Enables users to choose a different font size for each platform if needed.
# Defaults to 9 if not set.
#
#RSR_FONT_SIZE_WIN32 = 9



### Section 9: Plugin specific options
#

#
# RSR_PLUGIN_TARGET: Sets the default target profile used
# when creating an new instance of the plugin.
# Defaults to Kodak_2383.xml if not set
# NB: The plugin will look for this profile in the directory specified in
# RSR_TARGET_DIR ie. RSR_APP_DIR/RSR_TARGET_DIR/
#
#RSR_PLUGIN_TARGET = Kodak_2383.xml

#
# RSR_PLUGIN_MONITOR: Sets the default monitor profile used
# when creating an new instance of the plugin.
# Defaults to sRGB.xml if not set
# NB: The plugin will look for this profile in the directory specified in
# RSR_MONITOR_DIR ie. RSR_APP_DIR/RSR_MONITOR_DIR/
#
#RSR_PLUGIN_MONITOR = sRGB.xml

#
# RSR_PLUGIN_3D_QUICK_RES: Sets the internal cube resolution used when doing
# 3D transforms. A lower value means a quicker but less accurate transform.
# This is useful when adjusting other options as the GUI is more responsive.
# A Higher value means a slower but more accurate transform.
# Range 5 <-> 20. Defaults to (10) if not set.
#
RSR_PLUGIN_3D_QUICK_RES = 10

#
# RSR_PLUGIN_3D_ACCURATE_RES: Sets the internal cube resolution used when doing
# 3D transforms. A lower value means a quicker but less accurate transform.
```

```
# A Higher value means a slower but more accurate transform.
# Range 20 <-> 50. Defaults to (40) if not set.
#
RSR_PLUGIN_3D_ACCURATE_RES = 40




### Section 10: Depricated
#
# RSR_PRETRANSFORM_LUT_INTERP_TYPE: Sets the type of interpolation
# used to do lookups on the LUT file specified in RSR_PRETRANSFORM_LUT.
# Range linear, constrainedcatmullrom, catmullrom.
# Defaults to constrainedcatmullrom.
#
# Used by:
# cineCube, cineFilmMaster, cineFusion, cineNuke, cineShake, equalEyes
#
#RSR_PRETRANSFORM_LUT_INTERP_TYPE = constrainedcatmullrom

#
# RSR_LINEAR_MODE: Sets which channels can be separately adjusted when doing
# a lin2log pretransform.
# Range (grey, RGB). Defaults to (grey) if not set.
# Note: When set to RGB the RGB lin2log settings below can be used instead of
# the single channel grey options below.
#
# Used by:
# cineFusion, cineNuke, equalEyes
#
#RSR_LINEAR_MODE = grey

#
# RGB lin2log options (see above for details)
#
# Used by:
# cineFusion, equalEyes
#
#RSR_FILMDATA_WHITE_R = 685
#RSR_FILMDATA_WHITE_G = 685
#RSR_FILMDATA_WHITE_B = 685
#RSR_FILMDATA_BLACK_R = 95
#RSR_FILMDATA_BLACK_G = 95
#RSR_FILMDATA_BLACK_B = 95
#RSR_FILMDATA_D_GAMMA_R = 1.7
#RSR_FILMDATA_D_GAMMA_G = 1.7
#RSR_FILMDATA_D_GAMMA_B = 1.7
#RSR_FILMDATA_F_GAMMA_R = 0.6
#RSR_FILMDATA_F_GAMMA_G = 0.6
#RSR_FILMDATA_F_GAMMA_B = 0.6
#RSR_FILMDATA_SOFT_CLIP_R = 0
#RSR_FILMDATA_SOFT_CLIP_G = 0
#RSR_FILMDATA_SOFT_CLIP_B = 0


#
# RSR_BLACKPOINT_METHOD: Sets the method used to do the black point correction. Almost
# always the default method will the best method to use but it is unsatisfactory
# for a given set of profiles one of the other methods can be used to get a more accurate result.
# Range (MATCHBLACKS, TARGETOFFSET, NEARESTBOUNDARY, ALONGGREY). Defaults to (MATCHBLACKS) if not set.
#
# Used by:
# cineCube, cineFilmMaster, cineFusion, cineNuke, cineShake, equalEyes
#
```

```
#RSR_BLACKPOINT_METHOD = MATCHBLACKS


#
# The CRT, LCD or PROJ extension means that the option will be applied to
# CRT, LCD or Projector monitors only (as determined from the monitor profile
# that is used). If not set these all default to the RSR_WHITE_TEMP value.
#
# Used by:
# cineCube, cineFusion, equalEyes
#
#RSR_WHITE_TEMP_CRT = 5400
#RSR_WHITE_TEMP_LCD = 5400
#RSR_WHITE_TEMP_PROJ = 5400


#
# The CRT, LCD or PROJ extension means that the option will be applied to
# CRT, LCD or Projector monitors only (as determined from the monitor profile
# that is used). If not set these all default to the RSR_WHITE_X value.
#
# Used by:
# cineCube, cineFusion, equalEyes
#
#RSR_WHITE_X_CRT = 0.3349
#RSR_WHITE_X_LCD = 0.3349
#RSR_WHITE_X_PROJ = 0.3349


#
# The CRT, LCD or PROJ extension means that the option will be applied to
# CRT, LCD or Projector monitors only (as determined from the monitor profile
# that is used). If not set these all default to the RSR_WHITE_Y value.
#
# Used by:
# cineCube, cineFusion, equalEyes
#
#RSR_WHITE_Y_CRT = 0.3497
#RSR_WHITE_Y_LCD = 0.3497
#RSR_WHITE_Y_PROJ = 0.3497


#
# The CRT, LCD or PROJ extension means that the option will be applied to
# CRT, LCD or Projector monitors only (as determined from the monitor profile
# that is used). If not set these all default to the RSR_WHITE_TARGET_TYPE value.
#
# Used by:
# cineCube, cineFusion, equalEyes
#
#RSR_WHITE_TARGET_TYPE_CRT = 0
#RSR_WHITE_TARGET_TYPE_LCD = 0
#RSR_WHITE_TARGET_TYPE_PROJ = 0


#
# The CRT, LCD or PROJ extension means that the option will be applied to
# CRT, LCD or Projector monitors only (as determined from the monitor profile
# that is used). If not set these all default to the RSR_WHITE_DEFAULT_ON value.
#
# Used by:
# cineCube, cineFusion, equalEyes
#
#RSR_WHITE_DEFAULT_ON_CRT = FALSE
#RSR_WHITE_DEFAULT_ON_LCD = FALSE
#RSR_WHITE_DEFAULT_ON_PROJ = FALSE
```

1068

Appendix B

Black point correction

B.1 Preamble

Whenever you are performing colour transforms, there are three possible scenarios for the position of your monitor and target gamuts:

(a) Target gamut outside monitor gamut (Figure B.1).

This situation occurs when the target black point luminance is lower than the monitor (viewing) device's black point luminance. Black point correction is required in this situation to prevent crushing (clipping) in the low luminance colours.

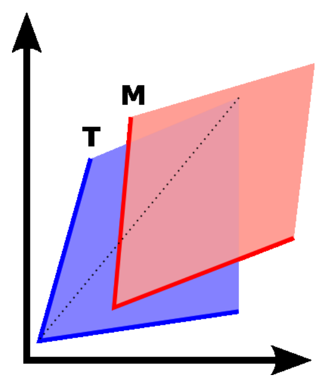


Figure B.1: Target outside monitor

(b) Target gamut crosses monitor gamut (Figure B.2).

This situation occurs when the target and monitor black points are similar in luminance but have different chromaticities. Black point correction is recommended to reduce unwanted colour casts in the low luminance colours.

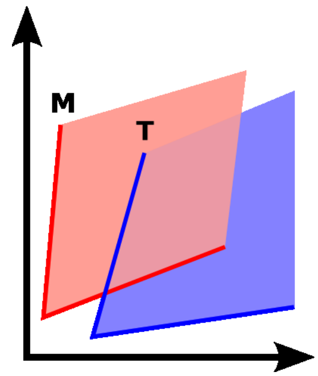


Figure B.2: Target crosses monitor

(c) Target gamut inside monitor gamut (Figure B.3).
This situation occurs when the target's black point luminance is higher than the monitor's black point luminance, and also falls within the monitor's gamut. No black point correction is needed. Even if black point correction is turned on, no correction will occur unless *force correction* is on.

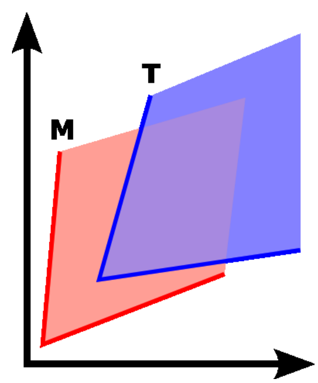


Figure B.3: Target inside monitor

B.2 Correction modes

Black point correction in the cineSpace suite can be performed using a number of different *modes*. The various modes and the effect that they have on the target profile in each case are described below. For each mode, the figures illustrate the effect that black point correction will have for the three possible monitor and target gamut scenarios (a, b and c) described above.

Explanation of symbols used in following sections:

- **M** Monitor gamut
- **T** Target gamut *before* black point correction
- **T′** Target gamut *after* black point correction

B.2.1 Match Blacks

'Match Blacks' will offset the entire target profile such that its black point corresponds to the same value as the monitor's (viewing device) black point. This means that when correction is occurring the resulting black value will be [0,0,0].

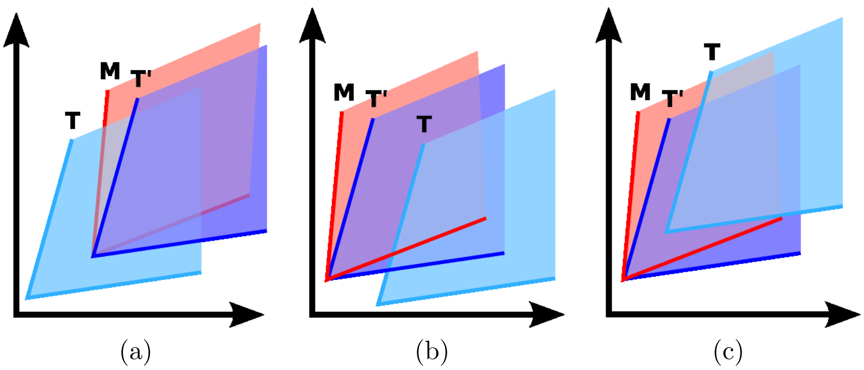


Figure B.4: 'Match Blacks' scenarios

B.2.2 Along Grey Ramp

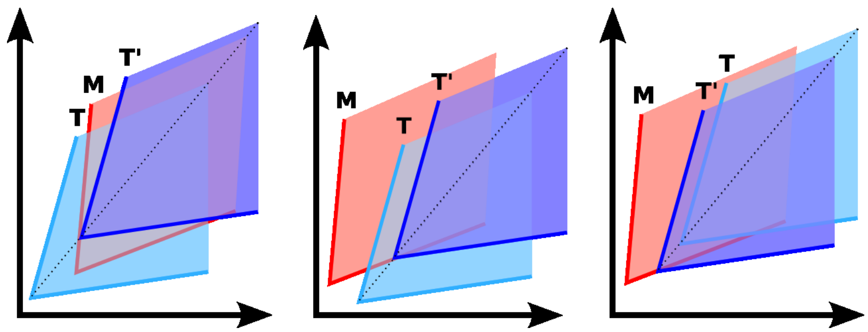
'Along Grey Ramp' will offset the target profile along its grey ramp until it reaches an edge of the monitor profile's gamut, i.e. until the target profile's black point *just* falls into the monitor profile's gamut.

B.2.3 Nearest Gamut Boundary

'Nearest Gamut Boundary' will offset the target profile so that its black point is moved to the nearest point in the monitor profile's gamut.

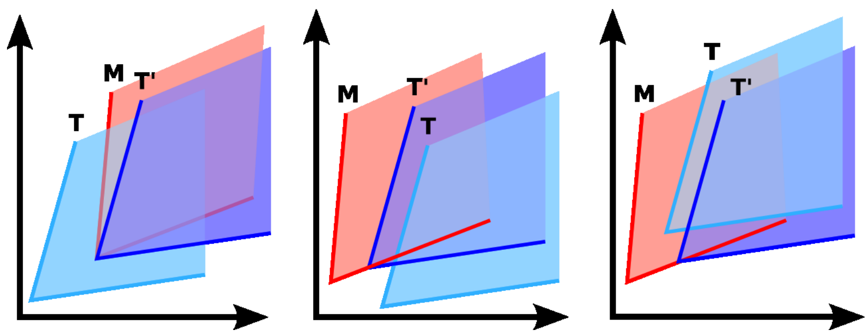
B.2.4 Target Offset

The distance and direction (vector) the target's black point is away from 'real black' will be mapped to the same vector away from the monitor's black point.



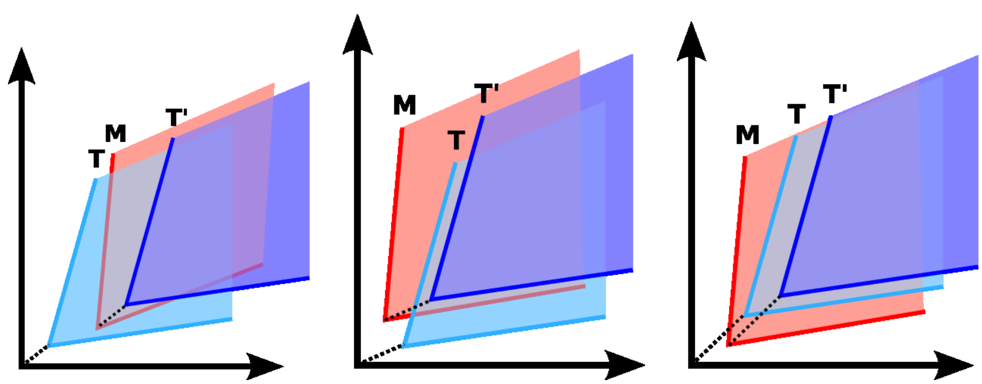
(a) (b) (c)

Figure B.5: ‘Along Grey Ramp’ scenarios



(a) (b) (c)

Figure B.6: ‘Nearest Gamut Boundary’ scenarios



(a) (b) (c)

Figure B.7: ‘Target Offset’ scenarios

Appendix C

Lin-to-log transforms

Typical 8-bit image files used for video images store each colour channel using code values from 0 to 255 to represent the linear light intensity. The darkest part of the image (black) is represented by 0, while the brightest part (white) is represented by 255. Now while this encoding may work well for images displayed on a monitor, things are not quite as simple when using film.

Density in a film emulsion exhibits a logarithmic response when exposed to light. If we were to encode that response using a linear model, we would lose much of the fine image detail – particularly in the darker areas of the image, where the shadows become "crushed". Encoding the image using a logarithmic curve provides a far better representation of the subtle nuances of film. Additionally, when compared with linear light encoding, logarithmic encoding more closely resembles human colour perception.

There are also limitations in using an 8-bit representation of film images, since a certain amount (large enough to be perceptible) of detail is lost during quantisation. The 8-bit format also does not cater for creative exposure adjustments or for the extreme highlights and shadow levels that can be present on film negatives. Although these may not be seen on the final print, they are important in the image manipulation process.

Due to the above factors, the industry standard for motion picture images has become logarithmically-encoded 10-bit files – primarily the Cineon and DPX file formats. Each colour channel is represented using code values from 0 to 1023. Nominal 'black' is represented by code value 95 and nominal 'white' is represented by code value 685. These values were chosen to allow some headroom for extreme shadows and highlights (e.g. direct light and specular reflections) that can be present on film. It also means that some exposure adjustment can be performed during digital manipulation without losing subtle details.

A complication arises when considering image manipulations in the digital domain. Due to the computations involved, it is easiest to use linearly-encoded data and so the log files must be converted into a linear format. The linear images are typically stored in 16-bit to ensure that fine details are not lost when converting from 10-bit log format. This emphasises one other benefit of log images: significantly superior storage efficiency.

Various algorithms and models are used throughout the industry to perform the required log-to-lin (and lin-to-log) conversions, which is one reason why the transform variables can be customised. The settings used in cineSpace must precisely match those used when converting your linear images to log format for sending to a film recorder.

Appendix D

cineSpace LUT format (.csp) v1.0

D.1 Format

The cineSpace LUT format contains three main sections.

Header

This section contains the LUT identifier and the LUT type, 3D or 1D.

It is made up of the first two (2) valid lines in the file. See Notes below for the definition of a valid line.

Examples

- (3D LUT) header:

```
CSPLUTV100
3D
```

- (1D LUT) header:

```
CSPLUTV100
1D
```

1D preLUT data

This section is designed to allow for unevenly spaced data and also to accommodate input data that maybe outside the 0.0 <-> 1.0 range. Each primary channel, red, green and blue has each own 3 line entry. The first line is the number of preLUT data entries for that channel. The second line is the input and the third line is the mapped output that will then become the input for the LUT data section.

It is made up of valid lines 3 to 11 in the LUT. See Notes below for the definition of a valid line.

Examples

- Map extended input (max. 4.0) into top 10% of LUT

```
11
0.0 0.1 0.2 0.3 0.4 0.5 0.6 0.7 0.8 0.9 4.0
0.0 0.1 0.2 0.3 0.4 0.5 0.6 0.7 0.8 0.9 1.0
11
0.0 0.1 0.2 0.3 0.4 0.5 0.6 0.7 0.8 0.9 4.0
0.0 0.1 0.2 0.3 0.4 0.5 0.6 0.7 0.8 0.9 1.0
11
0.0 0.1 0.2 0.3 0.4 0.5 0.6 0.7 0.8 0.9 4.0
0.0 0.1 0.2 0.3 0.4 0.5 0.6 0.7 0.8 0.9 1.0
```

- Access LUT data via a gamma lookup
  Red channel has gamma 2.0
  Green channel has gamma 3.0 but also has fewer points
  Blue channel has gamma 2.0 but also has fewer points

```
11
0.0 0.1 0.2 0.3 0.4 0.5 0.6 0.7 0.8 0.9 1.0
0.0 0.01 0.04 0.09 0.16 0.25 0.36 0.49 0.64 0.81 1.0
6
0.0 0.2 0.4 0.6  0.8 1.0
0.0 0.008 0.064 0.216  0.512 1.0
6
0.0 0.2 0.4 0.6  0.8 1.0
0.0 0.04 0.16 0.36  0.64 1.0
```

LUT data

This section contains the LUT data. The input stimuli for the LUT data is evenly spaced and normalised between 0.0 and 1.0. All data entries are space delimited floats. For 3D LUTs the data is red fastest.

It is made up of the valid lines 12 and onwards in the LUT. See Notes below for the definition of a valid line.

Examples

- Linear 3D LUT with cube sides R,G,B = 2,3,4 (ie. a 2x3x4 data set)

```
2 3 4
0.0 0.0 0.0
1.0 0.0 0.0
0.0 0.5 0.0
1.0 0.5 0.0
0.0 1.0 0.0
```

```
1.0 1.0 0.0
0.0 0.0 0.33
1.0 0.0 0.33
0.0 0.5 0.33
1.0 0.5 0.33
0.0 1.0 0.33
1.0 1.0 0.33
0.0 0.0 0.66
1.0 0.0 0.66
0.0 0.5 0.66
1.0 0.5 0.66
0.0 1.0 0.66
1.0 1.0 0.66
0.0 0.0 1.0
1.0 0.0 1.0
0.0 0.5 1.0
1.0 0.5 1.0
0.0 1.0 1.0
1.0 1.0 1.0
```

- Linear 1D LUT with length 1024

```
1024
0.015640 0.015640 0.015640
0.016618 0.016618 0.016618
0.017595 0.017595 0.017595
0.018573 0.018573 0.018573
0.019550 0.019550 0.019550
0.020528 0.020528 0.020528
0.021505 0.021505 0.021505
0.022483 0.022483 0.022483
0.023460 0.023460 0.023460
0.024438 0.024438 0.024438
0.025415 0.025415 0.025415
0.026393 0.026393 0.026393
0.027370 0.027370 0.027370
0.028348 0.028348 0.028348
0.029326 0.029326 0.029326
0.030303 0.030303 0.030303

              .
              .
              .

0.988270 0.988270 0.988270
0.989247 0.989247 0.989247
0.990225 0.990225 0.990225
0.991202 0.991202 0.991202
0.992180 0.992180 0.992180
0.993157 0.993157 0.993157
0.994135 0.994135 0.994135
```

```
0.995112 0.995112 0.995112
0.996090 0.996090 0.996090
0.997067 0.997067 0.997067
0.998045 0.998045 0.998045
0.999022 0.999022 0.999022
1.000000 1.000000 1.000000
```

D.1.1 Notes

- All lines starting with white space are considered not valid and are ignored.
- Lines can be escaped to the next line with "\".
- All values on a single line are space delimited.
- "BEGIN METADATA" indicates the start of the meta data section.
  "END METADATA" indicates the end of the meta data section.
  The above two lines and everything in between them are not considered valid lines are ignored by the parser.
- The first line must contain the LUT type and version identifier "CSPLUTV100"
- The second line must contain either "3D" or "1D".
- The third valid line is the number of entries in the red 1D preLUT. It is an integer.
- The fourth valid line contains the input entries for the red 1D preLUT. These are floats and the range is not limited. The number of entries must be equal to the value on the third valid line.
- The fifth valid line contains the output entries for the red 1D preLUT. These are floats and the range is limited to 0.0 ⟨-⟩ 1.0. The number of entries must be equal to the value on the third valid line.
- The sixth valid line is the number of entries in the green 1D preLUT. It is an integer.
- The seventh valid line contains the input entries for the green 1D preLUT. These are floats and the range is not limited. The number of entries must be equal to the value on the sixth valid line.
- The eighth valid line contains the output entries for the green 1D preLUT. These are floats and the range is limited to 0.0 ⟨-⟩ 1.0. The number of entries must be equal to the value on the sixth valid line.
- The ninth valid line is the number of entries in the blue 1D preLUT. It is an integer.
- The tenth valid line contains the input entries for the blue 1D preLUT. These are floats and the range is not limited. The number of entries must be equal to the value on the ninth valid line.
- The eleventh valid line contains the output entries for the blue 1D preLUT. These are floats and the range is limited to 0.0 ⟨-⟩ 1.0. The number of entries must be equal to the value on the ninth valid line.
- The twelfth valid line in a "3D" LUT contains the axis lengths of the 3D data cube in R G B order.
- The twelfth valid line in a "1D" LUT contains the 1D LUT length

- The thirteenth valid line and onwards contain the LUT data. For 3D LUTs the order is red fastest. The data values are floats and are not range limited. The data is evenly spaced.
- The LUT file should be named with the extension .csp

D.1.2 Open Source Project

Code for reading cineSpace LUTs is available from:

http://sourceforge.net/projects/cinespacelutlib/

(Please note that this is still Beta software but might be useful in implementing cineSpace LUTs in your own colour pipeline.)

Appendix E

Decoupling information

It is important to understand the concepts of *decoupling error* and *non-decoupling* devices, and the effects and implications of using non-decoupling devices within a colour managed pipeline. If used incorrectly, the end results may be undesirable. So, what does the term "decoupling" actually mean?

E.1 Decoupling error

A given display device can be characterised in terms of a set of *primaries* (usually three). When mixed in various combinations, these primaries allow a wide range of colours to be generated. The choice of primaries will determine the *gamut*, or range of colours, that can be represented by the device.

Typical CRT monitors use red, green and blue phosphors to generate displayed colours and so the primaries are defined as R, G and B. However, the actual colour of each primary will be determined by the chosen phosphors and so each CRT monitor will have its own unique colour characteristics.

In a CRT monitor, the primaries exhibit *independence*, whereby a change in the intensity of one colour channel does not affect either of the other two channels for the displayed colour. Thus it can be said that the primaries *decouple*. A consequence of this is that the sum of all three primaries at maximum intensity will produce white. This also holds for any fraction of maximum intensity: the sum of the three components will equal the corresponding grey scale value, i.e.

$$x.R + x.G + x.B = x.W \qquad \text{where } 0 \leq x \leq 1$$

Looking at it another way, an input containing a value for only one primary, will produce an output colour containing only the corresponding primary, e.g.

$$[r\ 0\ 0] \rightarrow [r_R\ 0\ 0]$$

Similarly:

$$[0\ g\ 0] \rightarrow [0\ g_G\ 0] \text{ and } [0\ 0\ b] \rightarrow [0\ 0\ b_B]$$

For a device whose primaries do *not* decouple (i.e. a *non-decoupling* device), we would obtain:

$$[r\ 0\ 0] \rightarrow [r_R\ r_G\ r_B]$$

For a given intensity level, we can compare the difference between the sum of the primary colour outputs and the grey scale output to obtain a measured *decoupling error*. The decoupling error, δ_{DE} (measured as a percentage), for each intensity level x between 0 and 1, may be defined as:

$$\delta_{DE} = \frac{\text{grey}[x] - (\text{red}[x] + \text{green}[x] + \text{blue}[x])}{\text{grey}[x] + 1} \times 100$$

By sampling the grey scale and primary output values at different intensity levels, we can assess the overall decoupling behaviour of the device that we are profiling. Then, by setting thresholds for the maximum tolerable average and maximum decoupling error over all intensity levels, we can reach a conclusion on whether or not the device correctly decouples.

E.2 Non-decoupling devices

If cases where the decoupling error is large, the property of decoupling primaries no longer holds and the simple mappings described above are not valid. Instead, our mapping of input values to output colour components now becomes:

$$\begin{bmatrix} r & g & b \end{bmatrix} \rightarrow \begin{bmatrix} r_R & r_G & r_B \\ g_R & g_G & g_B \\ b_R & b_G & b_B \end{bmatrix}$$

Since it is not possible to apply such a transform using 1D LUTs, we require a 3D LUT to obtain the correct mapping. Were 1D LUTs used on a non-decoupling device, there would be no guarantees about the accuracy of the displayed colour. The chromaticity and luminance would not be accurately modelled, producing uncertain results for colours not along the primary ramps. Furthermore, even the primary colour ramps could not be guaranteed due to the inability to accurately model desaturation or chromatic changes using 1D LUTs.

Clearly, being able to assume that the device decouple results in much simpler modelling and allows us to accurately match colour using 1D LUTs. Since this corresponds to the way standard video hardware determines the correct output values to display, we can implement any colour correction at the operating system level rather than in specialised applications or hardware.

For all practical purposes, we can assume that the primaries of most CRT monitors decouple correctly. Digital projectors, however, and also many LCD monitors, exhibit more complex characteristics and so the R, G and B primaries frequently do not sum to white in such devices.

Going a step further in complexity, film emulsions demonstrate significant *cross-talk* between dye layers, meaning that exposure to a single colour channel can produce changes in the recorded intensity of all three primaries. Also, whilst monitors and projectors produce colour using an *additive* process, film relies on *subtractive* colour based on the selective absorption of colour components. These factors make matching between monitors and film inherently more complex.

For these reasons, it is necessary to use 3D transforms (3D LUTs) when matching to certain targets such as film stocks or viewing material on digital projectors and LCD monitors. However, the use of 3D transforms is computationally intensive and requires either specialised hardware or software, rather than being performed at the operating system level.

E.3 Practical implications

When creating a monitor (viewing) device profile in cineProfiler, pay careful attention to the reported decoupling information displayed at the completion of profiling. If the profile successfully decouples then you will be able to use the monitor profile in all cineSpace applications (including equalEyes) to obtain accurate results.

If, however, the profile does *not* successfully decouple, you will most likely need to create a *Cube (3D)* profile in order to correctly capture the device's behaviour. This cube profile must then be used within cineCube or the cinePlugins in order to obtain accurate results. When used in equalEyes, it will produce the same results as a *Ramp (1D)* profile.

This does, of course, limit the flexibility of your work flow when using certain display types in your colour pipeline. However, the ability to measure decoupling error, and the fact that cineSpace will report possible issues when using some displays, means that you can be confident in the accuracy of your calibration and that the software is not hiding any approximations that may cause problems later.

Appendix F

Standard Profiles

F.1 Film

AGFA_CP30
Empirically measured from Agfa CP30 film stock
FUJIFILM_3510.xml
Empirically measured from FUJIFILM F-CP 3510 film stock (ETERNA-RDI 8511 negative)
FUJIFILM_3513DI.xml
Empirically measured from FUJIFILM ETERNA-CP 3513DI film stock (ETERNA-RDI 8511 negative)
FUJIFILM_3521XD.xml
Empirically measured from FUJIFILM ETERNA-CP 3521XD film stock (ETERNA-RDI 8511 negative)
Kodak_2383
Empirically measured from Kodak 2383 Vision film stock
Kodak_2383_average_2008
Average measured result from several Kodak 2383 Vision film stocks
Kodak_2393
Empirically measured from Kodak 2393 Premiere film stock

cineon_D55
Based on Kodak Cineon curve specification with a CIE D55 white point
cineon_D61
Based on Kodak Cineon curve specification with a CIE D61 white point
cineon_D65
Based on Kodak Cineon curve specification with a CIE D65 white point

F.2 Video

All Video profiles have a nominal white point Luminance $Y = 80cd/m^2$

AdobeRGB1998
chromaticity:

white:	xy[0.3127, 0.3290](D65)
red:	xy[0.6400, 0.3300]
green	xy[0.2100, 0.7100]
blue	xy[0.1500, 0.0600]

gamma/response:

$$s_{nl} = s^{563/256} = s^{2.19921875}$$

ITU-R_709_g2.2
chromaticity:

white:	xy[0.3127, 0.3290](D65)
red:	xy[0.6400, 0.3300]
green	xy[0.3000, 0.6000]
blue	xy[0.1500, 0.0600]

gamma/response:

$$s_{nl} = s^{2.2}$$

ITU-R_709_HD (ITU-Rec 709)
chromaticity:

white:	xy[0.3127, 0.3290](D65)
red:	xy[0.6400, 0.3300]
green	xy[0.3000, 0.6000]
blue	xy[0.1500, 0.0600]

gamma/response:

$$s_{nl} = \begin{cases} ((0.099 + s)/1.099)^{(1.0/0.45)} & \text{for } s > 0.081 \\ s/4.5 & \text{otherwise} \end{cases}$$

NTSC_601_g2.2 (CCIR 601-1)
The old NTSC standard
chromaticity:

white:	xy[0.3101, 0.3162](C)
red:	xy[0.6700, 0.3300]
green	xy[0.2100, 0.7100]
blue	xy[0.1400, 0.0800]

gamma/response:

$$s_{nl} = s^{2.2}$$

NTSC_601_g2.5 (CCIR 601-1)
The old NTSC standard
chromaticity:

white:	xy[0.3101, 0.3162](C)
red:	xy[0.6700, 0.3300]
green	xy[0.2100, 0.7100]
blue	xy[0.1400, 0.0800]

gamma/response:

$$s_{nl} = s^{2.5}$$

NTSC_SMPTE_g2.2
The current NTSC standard.
chromaticity:

white:	xy[0.3127, 0.3290](D65)
red:	xy[0.6300, 0.3400]
green	xy[0.3100, 0.5950]
blue	xy[0.1550, 0.0700]

gamma/response:

$$s_{nl} = s^{2.2}$$

NTSC_SMPTE_g2.5
The current NTSC standard.
chromaticity:

white:	xy[0.3127, 0.3290](D65)
red:	xy[0.6300, 0.3400]
green	xy[0.3100, 0.5950]
blue	xy[0.1550, 0.0700]

gamma/response:

$$s_{nl} = s^{2.5}$$

sRGB
chromaticity:

white:	xy[0.3127, 0.3290](D65)
red:	xy[0.6400, 0.3300]
green	xy[0.3000, 0.6000]
blue	xy[0.1500, 0.0600]

gamma/response:

$$s_{nl} = \begin{cases} ((0.055 + s)/1.055)^{2.4} & \text{for } s > 0.03928 \\ s/12.92 & \text{otherwise} \end{cases}$$

PAL_EBU_g2.2
chromaticity:

white:	xy[0.3127, 0.3290](D65)
red:	xy[0.6400, 0.3300]
green	xy[0.2900, 0.6000]
blue	xy[0.1500, 0.0600]

gamma/response:

$$s_{nl} = s^{2.2}$$

PAL_EBU_g2.5
chromaticity:

white:	xy[0.3127, 0.3290](D65)
red:	xy[0.6400, 0.3300]
green	xy[0.2900, 0.6000]
blue	xy[0.1500, 0.0600]

gamma/response:

$$s_{nl} = s^{2.5}$$

PAL_EBU_g2.8
chromaticity:

white:	xy[0.3127, 0.3290](D65)
red:	xy[0.6400, 0.3300]
green	xy[0.2900, 0.6000]
blue	xy[0.1500, 0.0600]

gamma/response:

$$s_{nl} = s^{2.8}$$

p3
chromaticity:

white:	xy[0.3140, 0.3510]
red:	xy[0.6800, 0.3200]
green	xy[0.2650, 0.6900]
blue	xy[0.1500, 0.0600]

gamma/response:

$$s_{nl} = s^{2.6}$$

p7v2
chromaticity:

white:	xy[0.3140, 0.3510]
red:	xy[0.6800, 0.3200]
green	xy[0.2650, 0.6900]
blue	xy[0.1500, 0.0600]
cyan:	xy[0.2048, 0.3602]
magenta	xy[0.3424, 0.1544]
yellow	xy[0.4248, 0.5476]

gamma/response:

$$s_{nl} = s^{2.6}$$

p7v2_reduced_gamut
chromaticity:

white:	xy[0.3140, 0.3510]
red:	xy[0.5980, 0.3269]
green	xy[0.2884, 0.5282]
blue	xy[0.1664, 0.0891]
cyan:	xy[0.2409, 0.3572]
magenta	xy[0.3382, 0.1838]
yellow	xy[0.3973, 0.4989]

gamma/response:

$$s_{nl} = s^{2.6}$$

RSR_D54_1.0
chromaticity:

white:	xy[0.3349, 0.3497](D54)
red:	xy[0.6400, 0.3300]
green	xy[0.3000, 0.6000]
blue	xy[0.1500, 0.0600]

gamma/response:

$$s_{nl} = s^{1.0}$$

RSR_D54_2.2
chromaticity:

white:	xy[0.3349, 0.3497](D54)
red:	xy[0.6400, 0.3300]
green	xy[0.3000, 0.6000]
blue	xy[0.1500, 0.0600]

gamma/response:

$$s_{nl} = s^{2.2}$$

RSR_D61_1.0
chromaticity:

white:	xy[0.3197, 0.3359](D61)
red:	xy[0.6400, 0.3300]
green	xy[0.3000, 0.6000]
blue	xy[0.1500, 0.0600]

gamma/response:

$$s_{nl} = s^{1.0}$$

RSR_D61_2.2
chromaticity:

white:	xy[0.3197, 0.3359](D61)
red:	xy[0.6400, 0.3300]
green	xy[0.3000, 0.6000]
blue	xy[0.1500, 0.0600]

gamma/response:

$$s_{nl} = s^{2.2}$$

RSR_D65_1.0

chromaticity:

white:	xy[0.3127, 0.3290](D65)
red:	xy[0.6400, 0.3300]
green	xy[0.3000, 0.6000]
blue	xy[0.1500, 0.0600]

gamma/response:

$$s_{nl} = s^{1.0}$$

RSR_D65_2.2

chromaticity:

white:	xy[0.3127, 0.3290](D65)
red:	xy[0.6400, 0.3300]
green	xy[0.3000, 0.6000]
blue	xy[0.1500, 0.0600]

gamma/response:

$$s_{nl} = s^{2.2}$$

Appendix G

Glossary

Some of the terminology used when describing colour spaces can cause confusion. This section outlines the definitions of terms that we use when discussing the cineSpace suite.

1D transform When calculating the output colour for a given input colour, each of the three components (R, G and B) is mapped separately:

R $\mapsto$ R', G $\mapsto$ G', B $\mapsto$ B'

This requires a look-up table of only $3 \times 256 = 768$ values (for 8 bit colour), which can be implemented on a standard video graphics card. However, it cannot represent transforms where, for example, the input R component would require changes to the output G' and/or B' components. This makes it unsuitable where the input and output primaries are different in chromaticity or where there is cross-talk between the colours (as occurs in film).

3D LUT See *cube*.

3D transform When calculating the output colour for a given input colour, every input RGB triple is mapped to an output RGB triple:

[R, G, B] $\mapsto$ [R', G', B']

This requires a look-up table of $256 \times 256 \times 256 = 16,777,216$ RGB triples (for 8 bit colour). Video graphics cards are not designed to cater for 3D transforms using the LUTs and so it must be implemented within specific applications (e.g. using the cineShake plug-in).

Although there is a greater processing overhead when using 3D transforms, it allows for very accurate representation of target profiles. This is of particular advantage when using film profiles, due to the required interaction between the R, G and B components.

calibrate To make adjustments to a system or device while comparing it with a known reference, with the intent of ensuring that the system or device matches a chosen standard. In the context of colour spaces, this usually means adjusting a viewing device to match either a mathematically-defined standard or an empirically measured output.

chromaticity The measurement of a colour that includes its dominant wavelength and purity, but not its absolute brightness.

cube (specifically, **colour cube**; also referred to as **3D LUT**) A file containing a 3D look-up table generated from cineSpace monitor and target profiles. The file can be loaded into various applications and used to determine the correct colour output for a given input. See also *look-up table*.

display device A medium capable of displaying images, such as a computer monitor, HD video screen or a projected film image.

gamma (γ) A parameter that describes the shape of the transfer function for one or more stages in an imaging pipeline. The transfer function is of the form $output = input^{\gamma}$, where both input and output are scaled to the range 0 to 1. Informally, gamma is a measure of the brightness of mid-level tones in an image.

gamut The range of different colours that can be interpreted by a colour model or generated by a specific device.

identity LUT A LUT that doesn't modify the input data. If you apply this LUT to an image, the image is not changed at all.

illuminant In general, a source of light. In the context of colour science, it refers to a particular spectral distribution such as the radiation from a *black body* at a certain temperature (in degrees Kelvin, e.g. 5400K) or one of the standards defined by the CIE (e.g. D65 or F11).

linear (lin) In the context of image files, refers to images where an incremental change in pixel value produces a relative change in the output brightness, i.e. doubling the value will double the brightness.

logarithmic (log) In the context of image files, refers to images where there in a non-linear relationship between pixel value and output brightness, i.e. there is a logarithmic relationship between the two. This is most commonly seen in film image file formats such as Cineon and DPX.

look-up table (LUT) An array of data values in the computer's frame buffer, used to relate the data value of each pixel in an image to the intensity of the corresponding screen pixel. A look-up table provides a *relative* mapping of one colour space to another and does not define an absolute colour space.

luminance Perceived brightness, or grey scale level, of a colour. Luminance and chromaticity together fully define a perceived colour.

LUT See *look-up table*.

optimise To adjust the settings (on your display device, e.g. CRT or LCD) obtain the maximum gamut possible and the optimal configuration for displaying the possible range of input data. This involves modifying the contrast and brightness, and individual colour bias and gain. The optimal settings will vary depending upon the intended output profile (e.g. film) you wish to display.

profile (*noun*) A file that describes the behaviour of a display device in response to RGB stimuli, by mapping the absolute response of the device (the colour produced) to each of the given stimuli. This file can be generated by *profiling* the display device through empirical measurement or by mathematically modelling the response according to a given specification.

profile (*verb*) The process of measuring the colour produced by a display device in response to a set of RGB stimuli in order to generate a description of the device's behaviour – this process is called *profiling*.

Appendix H

Change log

H.1 Changes in v2.7

H.1.1 New Features

- New application cineCube Visual added to the suite.
- Invert (reverse transform) functionality added to all applications.
- Ability to control scaling via a scale factor control.

H.1.2 cineProfiler

- Cosmetic GUI changes to cineProfiler.

H.1.3 equalEyes

- Fixed advanced command-line settings not being picked up from command line.
- Cosmetic GUI changes to equalEyes.

H.1.4 cineCube

- Added support for the ASC LUT format.
- Added the RSR_CONF_FILE information to debugging license errors.

H.1.5 cineNuke

- New build of cineNuke for OS X, Windows and Linux for Nuke 4.8vx
- New build of cineNuke for OS X, Windows and Linux for Nuke 5.0v1
- New build of cineNuke for OS X, WIndows and Linux for Nuke 5.0v2

H.1.6 cineShake

- Fixed license issue for OS X build on Leopard.

H.1.7 Misc

- Added new Kodak 2383 Generic Profile (Kodak_2383_average_2008) to Target Profile Library.
- Added new FujiFilm 3510 Generic Profile to Target Profile Library.
- Specify the location of the rsr.conf file via the RSR_CONF_FILE environment variable.
- Can specify the RSR_PRINTER_LIGHTS_PER_STOP as a float.
- Optimised the cineSpace libraries for speed rather than size.
- Updated and corrected minor bugs in documentation.

H.1.8 Drivers

- Updated the X-Rite drivers for Windows XP from v2.30 to v2.40

H.1.9 Licensing

- Added rlmdown utility to the RLM Server packages.
- Reduced total number of calls from applications to license server.
- Fixed the setLicensePath function for the cineSpace API.

H.2 Changes in v2.6.2

H.2.1 cineProfiler

- Added X-Rite Hubble Support.
- Modified some of the hint text.
- Fixed GUI focus box issue.
- Fixed GUI text cropping on Decoupling Graph.
- Now adds MAC addresses to profiles created on Irix.

H.2.2 equalEyes

- Scaling on / off now supported in rsr.conf file via RSR_EE_SCALING.
- Fixed GUI text cropping on Confidence Chart.
- Corrected formatting on equalEyes 'About' box.
- Corrected wording on Synthetic Profiles tool tip.

H.2.3 cineCube

- Added Digital Ordinance Frame Thrower LUT support.
- Can specify the printer lights per stop using the variable RSR_PRINTERLIGHTS_PER_STOP (defaults to 8).
- Fixed bug where cineCube could not find license directory.
- Can now load cineSpace 1D LUTs as pre-transform files.
- Fixed licensing so that cineCube_nlock can use default monitor-profile types.

H.2.4 xml2txt

- Reports extended messages on license failure to help with debugging.

H.2.5 txt2xml

- Reports extended messages on license failure to help with debugging.

H.2.6 cineNuke

- Attempts to auto-detect cineSpace installation if RSR_APP_DIR not set.
- Updated to use RSR_PLUGIN_RES to specify internal cineEngine resolution for quick / accurate transforms.
- Corrected name / header on cineNuke installation package.
- Corrected formatting on the cineNuke 'Help' page.

H.2.7 cineFusion

- Attempts to auto-detect cineSpace installation if RSR_APP_DIR not set.
- Added ViewLUT support. May access the cineFusion transform via the "LUT" button on the ViewWindow. Select "cineFusion", then select "Edit" to change the parameters of the cineFusion ViewLUT.
- Updated to use RSR_PLUGIN_RES to specify internal cineEngine resolution for quick / accurate transforms.

H.2.8 cineShake

- Attempts to auto-detect cineSpace installation if RSR_APP_DIR not set.
- Fixed cineShake from crashing on input of NaNs.
- Updated to use RSR_PLUGIN_RES to specify internal cineEngine resolution for quick / accurate transforms.
- Updated cineShake Viewer Controls to be same as cineShake Node.

H.2.9 Misc

- Added P3 and P7v2 profiles to the target-profiles folder.
- Changed permissions on OS X installation folders to allow non-admin users ability to copy licenses and monitor profiles.
- Fixed upgrade functionality for Linux cineSpace installations.
- Updated Linux installer to work on most RHEL 64 bit systems.

H.3 Changes in v2.6.1

H.3.1 cineSpace

- Silent installation now possible for Windows XP
- Improved the speed of cineSpace transforms when using a custom pretransform
- Fixed bug where Windows cineSpace installer was not creating a license directory

H.3.2 cineProfiler

- New gamma calculation algorithms for increased speed performance when saving profiles

- A user may control the default values for RGB Gain by setting the RSR_CP_GAIN environment variable
- A user may control the default values for RGB Bias by setting the RSR_CP_BIAS environment variable
- No longer requires libmdd.dll library
- Updated packaged X-Rite probe drivers to v3.4.0
- Cleaned up cineProfiler hint text
- Correct MAC address now always added to cube profiles on all operating systems
- Removed uneeded debugging messages

H.3.3 equalEyes

- Presets are now correctly removed from memory with the '-' key
- Removed uneeded debugging messages

H.3.4 cineCube

- Generate Cine-tal Cinemage LUTs
- Generate cineSpace 1D LUTs
- Generate Blackmagic Decklink 1D LUTs
- Generate TCube LUTs
- Generate AJA Kona 1D LUTs
- Displays more useful debug information on license failure
- Updated cineCube usage command

H.3.5 probeServer

- Now picks up RSR_PROBE_SOCKET variable defined in the rsr.conf file and as a command line option

H.3.6 cineNuke

- New OS X build for Nuke v4.7x
- Fixed bug where cineNuke was reporting wrong version number
- Fixed bug where license failure was being reported as invalid profile
- Fixed bug where cineFusion was not finding default license directory

H.3.7 cineFusion

- Fixed bug for cineFusion_render_token licensing
- Fixed bug where cineFusion would not re-try for a license

H.3.8 cineShake

- Can now handle monochrome input correctly
- Better support for Linux 64 bit machines
- Now prints out version and build numbers on startup

H.3.9 cineSpace API

- Now under OSX Tiger, a Universal Binary is generated rather than separate i386/ppc binaries
- Changed examples to build without warnings under Visual Studio 2005

H.3.10 Misc.

- Ensured generic target-profiles have luminance consistant with custom profiles
- Fixed thread related crash bug in various plugins and cineSpace API

H.4 Changes in v2.6

H.4.1 Licensing

- Changed licensing from FlexLM to Reprise Licensing Manager.

H.4.2 cineSpace

- Added Intel Mac Support.
- Includes Tiger Universal Binary build.
- Can now load cineSpace 1D LUTs as a custom pre-transform.

H.4.3 cineProfiler

- Monitor profile now auto-saves with file name %HOST%_%DATE%.xml if the variable RSR_CP_AUTOSAVE is set to true.
- Resolved intermittent hanging issues when profiling LCD monitors on OS X 10.4.
- Updated instructions on how to run Output Independent Profiling.
- Updated Output Independent Profiling instructions to correctly state supported probes.
- Updated GUI on IRIX to correctly display the initialisation window.
- Added the RSR_CP_AUTOSAVE variable as an option to automatically save the monitor profile with signature
- Added support for GretagMacbeth Eye-One LT probe on OS X.
- Provides feedback information during patch re-reading.
- Changed default Output Independent Profiling method to be, 'Use error detection.'
- Added a 'More info' button to decoupling information results.
- Updated cineProfiler decoupling chart to include more information about the monitor:
  - Black : x cd/m^2 [y fL]
  - White : x cd/m^2 [y fL] colour xy=(x,y) [temp K]
- Fixed bug, was incorrectly passing non-decoupling devices.
- Fixed bug, Display2 white calibration patch disappears if user presses the 'back' button.
- Fixed bug, generated profiles were missing decoupling information.

H.4.4 equalEyes

- The rsr.conf file variable RSR_START_PRETRANS has been fixed.
- A minor GUI change to correctly display the monitor name (without directory).
- Removed dependencies on legacy variable RSR_BLACK_KNEE - no longer in use.
- Fixed bug, problems selecting lin2log via the command line.
- Fixed bug, problems when setting synthetic target temperature via the command line.
- Fixed bug, RSR_EE_INACTIVE_GUI not activating the LUT.

H.4.5 cineCube

- Resolved issue causing incorrect Quantel LUT format.
- Added support for the cineSpace (.csp) 3D LUT format.
- Support for Barco LUT format has been corrected.
- Added support for Apple Color 3D LUT generation.
- Added support for CHROME Imaging 3D LUT format.
- Added support for the Houdini 3D LUT format.
- Added custom pre-transform LUT support.

H.4.6 cineShake

- Added custom pre-transform LUT support.
- Added support for Intel Mac.
- New Tiger Universal Build.

H.4.7 cineNuke

- Default white point adaptation settings (off) now consistent with the other cineSpace applications.
- New cineNuke plug-in for Nuke v4.7 for Linux.
- New cineNuke plug-in for Nuke v4.7 for Windows XP.
- New Linux Installation package for cineNuke that contains builds for both Nuke 4.3, 4.5 and 4.7.
- New Linux build for Nuke 4.5.27.
- New Windows XP 32 bit build for Nuke 4.5.27.
- Added custom pre-transform LUT support.
- Linux Installer will update Nuke's menu, cineNuke is found in Color -¿ cineNuke.
- Fixed bug, missing cineNuke license crashes Nuke 4.5.

H.4.8 cineFusion

- Added custom pre-transform LUT support.

H.4.9 cineFilmMaster

- Added printer light functionality within GUI interface (only works with FilmMaster 3.0.2 or higher).
- Added custom pre-transform LUT support.

- Added new out-of-gamut selection colours. Options are:
  - none
  - green
  - red
  - blue
  - desaturate in
  - desaturate out
- New GUI installer for installing cineFilmMaster.
- Fixed bug, could not find a license.

H.4.10 cineSpace API

- Changed win32 cineSpace API to be a dynamic library.
- Added an example that demonstrates how to use the out-of-gamut feature.
- Support for Fedora Core 4.
- Added Out-of-Gamut selection. The options are:
  - none
  - green
  - red
  - green
  - blue
  - desaturate
- Added custom pre-transform LUT support.
- Exposed Interpolator types used for the pre-transform LUT. The options are:
  - linear
  - constrained catmullrom
  - catmullrom

H.4.11 cineEngine

- New method used to find the nearest gamut edge for an out-of-gamut colour. Note that this does not affect any colours that are in gamut.
- New unconstrained CatmullRom 1D Interpolator (default 1D interpolator).
- Removed 1D Interpolator Cubic Bezier.
- Removed 1D Interpolator Constrained Cubic Bezier.
- Lin2Log transform now has correct behaviour for non-default values of DGamma.

H.5 Changes in v2.5

H.5.1 cineSpace

cineEngine

- publicAPI available
- New rsr.conf file parameters
- New improved 3D transform algorithms
  - 3D quick
  - 3D accurate

XML profiles

- Updated meta information
- New distributed target profiles

Licensing

- Feature names have changed and been added for the cineSpace suite
  - cineEngine3D has been replaced by cineShake, cineNuke, cineFilmMaster, cineFusion
  - cineCube has been replaced by cineCube_nlock, cineCube_float
  - cineSpaceAPI is a new feature for the public API

H.5.2 cineSpace suite

New components

- C public API for cineEngine
- C++ public API for cineEngine

cineProfiler

- New profile support - can do cube (3D) profiles
- Output independent profiling
- Improved LCD support
- Broadcast monitor support
- Projector support
- New monitor optimisation abilities:
  - Set via target profile
  - Set by CIE correlated temperature
  - Set by chromaticity coordinates (xy)
  - Set luminance
- Decoupling graphs / response for profiled device
- Improved error detection algorithms
- Will support display sizes down to 1024x768

equalEyes

- Can disable GUI via rsr.conf file (RSR_EE_INACTIVE_GUI)
- Confidence Check
  - Displays monitor's white point and chromaticity
- Custom pre-transform LUTs
- Black point correction

  - On / off
  - Force correction
  - Four modes:
    * Match Blacks
    * Target Offset
    * Nearest Gamut Edge
    * Along Grey Ramp
- White point adaptation
  - On / off
  - Three methods to select white point
    * Use monitor's (viewing device) white point
    * Set via CIE correlated temperature (K)
    * Set via chromaticity coordinates (xy)
- LUTs will clear when exiting with ctrl-c
- LUTs will be applied to monitors attached to Mac PowerBooks
- Profile validity date is now updated regularly, rather than at profile loading
- Cancelling probe server will not crash equalEyes

cineCube

- New LUT support for:
  - arri (.ari) Arri 3D LUT file
  - barco (.itx) Barco D-Cine 3D LUT file
  - clipster (.ari) DVS Clipster 3D LUT file
  - cyborg (.lut) Cyborg5D LUT file
  - finaltouch1 (.mga) Silicon Color Final Touch v1.0.8 and older 3D LUT file
  - finaltouch2 (.mga) Silicon Color Final Touch v1.0.9 and newer 3D LUT file
  - fire (.3dl) Autodesk Fire 3D LUT file
  - flame (.3dl) Autodesk Flame 3D LUT file
  - flint (.3dl) Autodesk Flint 3D LUT file
  - houdini (.itx) SideFX Houdini 3D LUT file
  - inferno (.3dl) Autodesk Inferno 3D LUT file
  - it (.lut) IT LUT file
  - luther (.txt) Grass Valley LUTher 3D LUT file
  - mistika (.itx) S.G.O. Mistika 3D LUT file
  - nucoda (.lut) Nucoda 3D LUT v2 file
  - photoshop (.acv) Adobe Photoshop 1D curves file
  - pixelfarm (.pfl) The Pixel Farm 3D LUT file

  - quantel3d (.txt) Quantel 3D LUT file
  - resolve (.dat) da Vinci Resolve 3D LUT file
  - rip (.lut) Rip 3D LUT file
  - smoke (.3dl) Autodesk Smoke 3D LUT file
  - toxik (.lut) Autodesk Toxik 3D LUT file
- Can change printer light settings in rsr.conf file
- New out-of-gamut options:
  - red, blue, green, invert

probeServer

- Reduced CPU usage

cinePlugins

- Plug-in for FilmMaster win32 – cineFilmMaster
- Plug-in for Shake v4.0 Linux – cineShake
- Plug-in for Shake v4.0 OS X – cineShake
- Plug-in for Shake v4.1 OS X – cineShake
- Plug-in for Nuke v4.5 – cineNuke
- Plug-in for Fusion v5.0 – cineFusion
- Network install package for cineShake OS X
- Black point correction
  - On / off
  - Force correction
  - Four modes:
    - Match Blacks
    - Target Offset
    - Nearest Gamut Edge
    - Along Grey Ramp
- White point adaptation
  - On / off
  - Three methods to select white point
    - Use monitor's (viewing device) white point
    - Set via CIE correlated temperature (K)
    - Set via chromaticity coordinates (xy)
- Won't crash client when loading invalid profiles

H.5.3 Operating systems

- Dropped support for gcc2.96 (Red Hat 7.3)
- Installer for Linux

H.5.4 Partners

- da Vinci Resolve integration

H.6 Changes in v2.1

H.6.1 New features

- New probeServer utility that allows a probe connected to one machine to be used for profiling other machines. This is helpful when using a USB probe with an Irix machine and also when using a laptop and probe to profile all workstations in a facility.
- The pluge and colour patch in cineProfiler are now movable and resizable to enable easier profiling of dual monitor configurations, digital projectors and monitors with large OSD controls.
- Support for the new GretagMacbeth i1 Display2 probe has been added.
- The cinePlugins now have a lin2log option.
- A PanalogToLog pre-transform has been added to all cineShake and equalEyes. This is analogous to the LinToLog used for viewing linear images.
- cineNuke has been updated for Linux and now also supports Windows.
- cineShake includes a new viewer script option in addition to cineSpace nodes.
- cineCube now has a lin2log option and support for new file formats (Scratch, Shake 1D LUT, Lustre, Final Touch).
- cineCube now uses default values for printer light and LinToLog parameters.
- New utilities have been added to convert profiles to and from plain text format for graphing and tweaking profiles (xml2txt and txt2xml).
- The target luminance used when running cineProfiler can now be customised.
- When the equalEyes license check fails, it now provides diagnostic information to aid troubleshooting.
- Changed order of probe detection in all OSes. This makes detecting Gretag or USB X-Rite probes significantly faster. Also allows probes to work in winXP on laptops without serial ports.
- A range of additional user-configurable settings have been added using variables in the rsr.conf file:
  - `RSR_PRINTERLIGHTS_PER_STOP` allows adjustment of the number of points per stop
  - `RSR_START_LINTOLOG` enables equalEyes to start in Linear Mode by default
  - `RSR_CP_PROFILE_LENGTH` sets Normal or Quick profile as the default option
  - `RSR_EE_MONPROF_LIST` adds extra user-specified filters to the Monitor Profile dialog box
  - `RSR_PLUGIN_MONITOR` and `RSR_PLUGIN_TARGET` specify the initial profiles used in cineShake
  - `RSR_CP_GAMMA_CLEAR` prevents cineProfiler clearing the LUTs when profiling (for using cinePlugins & cineCube when equalEyes is not used)

  - RSR_CP_PLUGE_CENTER, RSR_CP_PLUGE_INNER, RSR_CP_PLUGE_OUTER and RSR_CP_PLUGE_BLACK allow the pluge grey values to be changed
  - RSR_BLACK_KNEE_CRT, RSR_BLACK_KNEE_LCD, RSR_BLACK_CURVE_CRT, RSR_BLACK_CURVE_LCD So that different black-knee behaviour for LCDS and CRTS. The applications can tell which is needed from the monitor profile being used from the monitor profile.
- Switched to dynamic version of the Gretag probe library, resulting in smaller executables and several bug fixes.

H.6.2 Bug fixes

- The 'Confidence Test' was sometimes not passing even immediately after profiling – this has been fixed with improved bottom-end LUT behaviour in equalEyes.
- Issues causing a "Reading invalid error" in cineProfiler have been addressed.
- An issue with equalEyes presets not working correctly has been resolved.
- equalEyes no longer crashes on exit in OS X.
- Warning messages appearing when installing and running cineShake have been suppressed.
- Font issues occurring in Irix and Win32 have now been fixed.
- Clicking on the equalEyes application window was not bringing it to the front – this has been resolved.
- cineProfiler no longer takes an excessively long time on FC3.
- A bug causing the colour patch to disappear behind the main window under Linux RH7.3 has been fixed.
- Problems with setting RSR_SYTH_TEMP_TYPE have been addressed.
- Hardware probes now work correctly when using a USB serial adapter. (in most cases).
- Unnecessary Clipping that occurred in the log2lin pre-transforms have been fixed.
- Fixes to apply the LUTs to multiple screens in OS X.
- equalEyes confidence test window can now be moved when using gnome, by dragging in the interior.
- Fixed equalEyes crash when moving the application while the confidence test window was open.
- Fixed equalEyes confidence test focus problems under Linux FC2.
- Fixed equalEyes crash after confidence test was closure.
- equalEyes confidence test fixed so user can access the probe pop-up dialog properly.
- equalEyes now handles slashes in profile paths properly under win32, fixing presets not always working under win32.
- Fixed cases in equalEyes and cineProfiler where windows were unnecessarily modal, including the help-window.

- The cineProfiler Pluge is now correctly positioned on OS X.
- Internal improvements to memory management.
- Probe interface can now handle where old X-Rite probes would return an invalid string causing the application to behave incorrectly.
- Fixed crash on exit in cineProfiler on OS X.
- Fixed missing lifetime attribute in some profiles.
- Fixed serial probe problems on slow read.

H.7 Changes in v2.04

H.7.1 New features

- The availability of both a stable and a development version. It is intended that only serious bug fixes will be done to the stable version while feature additions will occur on the unstable development version.

H.7.2 Bug fixes

- Better accuracy for dark colours and reduced banding effects. This was considered important enough to get a release soon after v2.03.

H.8 Changes in v2.03

H.8.1 New features

- The help browser now stores names of locations in addition to locations. The history and bookmark files are now stored in the users `.rsr_settings` directory.
- The font sizes used in equalEyes and cineProfiler are now user specifiable through `RSR_FONT_SIZE`. Each operating system can override these values using `RSR_FONT_SIZE_LINUX`, `RSR_FONT_SIZE_IRIX`, `RSR_FONT_SIZE_OSX` or `RSR_FONT_SIZE_WIN32`.
- There are now indicators for the state of equalEyes on its front page.
- cineCube now has printer lights available using `-printerlights` with additional setting specified via `-red-off 26` or `-green-trim 24` type commands. All unspecified trims and offsets default to 25.

H.8.2 Bug fixes

- The help browser history no longer clashes with tcsh history file. Additionally the help files are limited in size, removing several crashes and performance issues.
- equalEyes and cineProfiler were showing in a reduced bit depth in Irix if the desktop was not in true color mode. The applications now always try to draw in true color mode. Also warns on all platforms if it is not drawing with enough bit-depth.
- The interpolation methods for 1D transforms have been changed and interpolation near 0 and 1 has been improved, resulting in smoother ramps and better reproduction of dark colours.

- A bug in the black point correction causing a light band to occur part way up the grey ramp in some cases has been fixed.
- Permanent profiles are now recognised correctly.
- In Linux equalEyes now obtains mouse focus correctly, and comes to the front when clicked.
- The Log-Lin button was in the wrong state.
- cineCube no longer complains about correct arguments being invalid.
- Changes to cineShake to support multi-threading broke it for float input – this has been fixed.
- A problem where equalEyes was being drawn way too small on some Fedora Core 2 Linux boxes in KDE window manager has been fixed.
- The machine name is now found correctly under Windows, rather than using the user domain.

H.9 Changes in v2.01

H.9.1 New features

- equalEyes has a splash page while waiting to obtain a license, which can take a while. (Previously it did not respond at all.)
- All programs allow specification of the configuration file to use from the command line. `--config=`*filename* will use the configuration file *filename*, while `--config-extra=`*filename* will load *filename* after the normal configuration files. Both options can be used on the same line.
- Serial devices can now be used through serial to USB converters. The port names to check can be given by `RSR_SERIAL_PROBES_ONLY` or `RSR_SERIAL_PROBES_EXTRA`. The first of these replaces the list of usual ports examined with that specified. The second appends the ports specified.
- The file that logging information goes to (when the `RSR_OUTPUT` is set to logging mode) can now be specified for equalEyes and cineProfiler using the `RSR_EE_LOG_FILE` and `RSR_CP_LOG_FILE`. When these are not specified the old method of writing to `RSR_LOG_FILE` is used.
- The age at which profiles are considered expired can be set for equalEyes. The variable `RSR_MONITOR_LIFETIME` defaults to 14 (days) and `RSR_FILM_LIFETIME` defaults to 180 (days).
- Configuration variables can now be set from the command line for most applications. For example to enable logging in cineProfiler we can run the program as `cineProfiler --rsr-output=LOG`. The variables are prefixed with `--` and are case insensitive. Also, where a variable name uses an underscore, the command-line version can use either an underscore or a single dash.
- Multi-threading has been enabled in the cineShake and cineFusion nodes.
- There is now better handling of profiles that contain only primary and grey ramps. (This has removed some of the color shifts observed in the cineon profile.) These profiles also support film-like desaturation of the primaries when the primaries add to less than the white point. (Only observable in applications using full 3D transforms such as cineShake, cineFusion or cineCube.)

- Extra navigation buttons added to equalEyes front page.
- Added `RSP_EE_NOCONFIRM_EXIT` variable which makes equalEyes quit without confirmation. This may be useful if you can't get the window manager to shut down while equalEyes is running.

H.9.2 Bug fixes

- Added functions for ensuring monotonicity/smoothness of monitor profiling.
- Fixed crash in cineFusion when user changes settings while plug-in is processing.
- Plug-ins no longer check out a license for each node.
- The Windows installer no longer sets shortcuts that override other applications.
- Fixed a string-related crash in the base library that occurred in unusual cases.
- The conflicting messages in the equalEyes dialog regarding profile age has been fixed.
- Inadvertently active button in equalEyes made inactive.
- Logging information is now generated by cineProfiler correctly.
- Various banding observable in the grey ramp has been fixed.