KEYLIGHT

User Guide for Keylight 2.2 for Shake

21st June 2006

The Foundry Plug-in Visual Effects

User Guide for Keylight 2.2 for Shake

Software engineering Bruno Nicoletti, Simon Robinson, Ben Kent.

Product testing Jack Binks, Jonathan Barson.

Writing and layout design Jonathan Barson and Dr. Bill Collis using LaTeX

Proof reading Jack Binks and Jonathan Barson.

Contents

Introduction

Welcome to this User Guide for Keylight on Shake.

Keylight has been refined over a number of years to make keying quicker and easier while providing a depth to the tools that will tackle even the most challenging shots. Keylight was first developed by The Computer Film Company to help with difficult keys feature films. Over the years Keylight has been refined in production on hundreds of films. This pioneering work on digital compositing was honoured with a Technical Academy Award in 1996.

Keylight has been included with Shake for a number of years (Figure 1.2). Our version of Keylight (Figure 1.1), although based on the same algorithm by CFC, has been refined over the past few years and is now included as a seperate plug-in for Shake alongside the builtin version.

Figure 1.1 Foundry's Keylight.

Figure 1.2 Shake's Keylight.

Keylight 1/Keylight 2

If you are used to Shake's Keylight 1 and want to know what's changed in this version see Changes on page 21.

About this User Guide

Use the Quick Key chapter to see how a simple key is pulled using Keylight. The Basic Keying Chapter goes over the most common parameters you'll need to pull a variety of keys. The Advanced Keying Chapter explains how to tackle more difficult keys.

What's New?

For information on system requirements, new features, improvements, fixed bugs and known bugs/workarounds, see Appendix A on page 65.

Example Images

Example images are provided for use with Keylight. You can download these images from our web site www.thefoundry.co.uk and try Keylight out on them.

Notation

In this User Guide we will refer to machines running Keylight and Shake as clients and machines that are running the Foundry FLEXlm Tools software as servers.

When we refer to blue screens throughout the text we mean, of course, blue or green screens.

Installing Keylight

Keylight is available as a download from our web site www.thefoundry.co.uk. The downloads are in compressed tar format (tgz) for Linux, and dmg format for Mac OS X machines. Keylight should be installed on the client machines. The plug-ins are licensed with FLEXlm. We supply a suite of tools to manage and monitor floating licenses running on a server across a network of machines. These tools are called Foundry FLEXlm Tools (FFT) and can be downloaded free of charge from our web site. The Foundry FLEXlm Tools should be installed on the server.

Note *Commands in these instructions may be shown wrapped over more than one line, with subsequent lines being indented to indicate the continuation. Regardless, these should be typed on a single line. So*

```
a command that wraps around from one line
   to the next line
```

should be typed like this:

```
a command that wraps around from one line to the next line
```

Keylight on Linux

Follow these instructions if you wish to install Keylight on a Linux machine running Shake.

1. Download the file from our web site (www.thefoundry.co.uk)
2. Move the download file to /usr/nreal

   ```
   mv Keylight2.2v2_shake4.10-linux-x86-release-32.tgz
     /usr/nreal
   ```

3. Change directory to /usr/nreal

   ```
   cd /usr/nreal
   ```

4. Extract the files from the archive using the command:

   ```
   tar xzvf Keylight2.2v2_shake4.10-linux-x86-release-32.tgz
   ```

5. A directory structure is created.

   ```
   /usr/nreal/Keylight2.2v2_shake4.10-linux-x86-release-32/
      include
   /usr/nreal/Keylight2.2v2_shake4.10-linux-x86-release-32/
      icons
   /usr/nreal/Keylight2.2v2_shake4.10-linux-x86-release-32/
      docs
   ```

6. You need to tell Shake where to look for the plug-ins. There are several ways of doing this and you should refer to your Shake documentation. However, to use the Nothing Real environment variable method you should add the following environment variables to your .login file using a text editor (vi). These environment variables enable Shake to locate the plug-ins and icons.

   ```
   setenv NR_INCLUDE_PATH /usr/nreal/
      Keylight2.2v2_shake4.10-linux-x86-release-32/
      include
   setenv NR_ICON_PATH /usr/nreal/
      Keylight2.2v2_shake4.10-linux-x86-release-32/
      icons
   ```

   If you already have these variables set you can add to them by putting a colon between the paths, for example:

```
setenv NR_INCLUDE_PATH /usr/nreal/
   Keylight2.2v2_shake4.10-linux-x86-release-32/
   include:/usr/nreal/
   Tinder2.0v1shake4.10-linux-x86-release-32/
   include
```

7. Proceed to "Licensing Keylight" on the following page.

Keylight on Mac OS X

Follow these instructions if you wish to install Keylight on a Mac OS X machine running Shake.

Default Installation

This will install our plug-ins into the Shake directory structure so that the plug-ins are picked up automatically. The disadvantage of this method is that only one version of Furnace can be used. To have the flexibility of choosing the version of Furnace when running Shake use the "Customised Installation" method on the current page.

1. Download the file from our web site (www.thefoundry.co.uk).
2. Double click on the downloaded dmg file.

   ```
   Keylight2.2v2_shake4.10-mac-ppc-release-32.dmg
   ```

3. Double click on the pkg file that is created.

   ```
   Keylight2.2v2_shake4.10-mac-ppc-release-32.pkg
   ```

4. Follow the onscreen instructions to install Keylight directly into the Shake 4.10 directory.
5. Proceed to "Licensing Keylight" on the following page.

Customised Installation

This will install our plug-ins to a directory of your choice. This gives you the flexibility of choosing different versions of Furnace on starting Shake, but requires the use of environment variables to pick up the plug-ins. To get the plug-ins to work you will have to tell Shake where to find them using the NR_INCLUDE_PATH and NR_ICON_PATH environment variables.

1. Download the file from our web site (www.thefoundry.co.uk)
2. Create a directory into which the plug-ins will be installed. We recommend the following:

   ```
   mkdir -p /Documents/
     Keylight2.2v2_shake4.10-mac-ppc-release-32
   ```

3. Double click on the downloaded dmg file.

```
Keylight2.2v2_shake4.10-mac-ppc-release-32.dmg
```

4. Double click on the pkg file that is created.

```
Keylight2.2v2_shake4.10-mac-ppc-release-32.pkg
```

5. Follow the onscreen instructions. When you get to Select Destination click on Choose (not Continue) and browse to the directory you wish to install the plug-ins. For example, select /Documents/Keylight2.2v2_shake4.10-mac-ppc-release-32.
6. Continue with the installation.
7. Quit.
8. Start a terminal and set the environment variables as follows:

```
setenv NR_INCLUDE_PATH /Documents/
   Keylight2.2v2_shake4.10-mac-ppc-release-32/
   shake.app/Contents/PlugIns

setenv NR_ICON_PATH /Documents/
   Keylight2.2v2_shake4.10-mac-ppc-release-32/
   shake.app/Contents/Resources/icons
```

9. Proceed to "Licensing Keylight" below.

Licensing Keylight

Keylight uses FLEXlm encryption in the license keys. For information on licensing Keylight, setting up a floating license server, adding new license keys and managing license usage across a network you should read the Foundry FLEXlm Tools (FFT) User Guide which can be downloaded from our web site.

Documentation

Keylight comes with two sets of documentation, this pdf and a set of html files which connect to the Help button in Shake. This button will work immediately if you have installed to the default location, however, if you installed elsewhere the software must know where to look for the documentation. To do this, set the environment variable KEYLIGHT_DOCS_DIR to point directly at the docs directory. Multiple paths, separated by a colon, are not supported at this time.

About Keylight

After installation, Keylight can be found in the Key tab panel as shown in Figure 1.3.

Figure 1.3 Key panel and Keylight node.

Other Foundry Products

The Foundry is a leading developer of plug-in visual effects for film and video post production. Its products include Furnace, Tinder, Tinderbox, Keylight and Anvil and run on a variety of compositing platforms including Discreet Flame, Discreet Flint, Discreet Fire, Discreet Inferno and Discreet Smoke, Shake, Avid|DS, After Effects, and a variety of OFX compliant hosts. For the full list of products and supported platforms see our web site www.thefoundry.co.uk

Tinder and Tinderbox are collections of image processing effects including blurs, distortion effects, background generators, colour tools, wipes, matte tools, paint effects, lens flares and much more.

Anvil is a collection of colour correction and colour manipulation tools originally developed by First Art.

Keylight is an award winning blue/green screen keyer giving results that look photographed not composited. The Keylight algorithm was developed by the Computer Film Company who were honoured with a technical achievement award for digital compositing from the Academy of Motion Picture Arts and Sciences.

Visit The Foundry's web site at www.thefoundry.co.uk for further details.

Getting Started

Introduction

Select the **screenColour** and pick a range of colours from the blue or green screen. Try the Average, Min and Max from the sampled colours and judge which colour works best. You can change the **output** parameter to 'status' to give you more information about the quality of the key. White is pure foreground, black pure background and any mix between them is shown in grey.

One important difference between Keylight and other keyers is that picking multiple colours does not 'key more'.

To remove blue/green spill from the foreground select the **alphaBias** and pick skin tones from the foreground.

Use the **clipMin** and **clipMax** parameters sparingly to improve the screen matte. Use the Screen Processing tools to further manipulate the matte or connect external mattes from QuickShape or other Keylight nodes to the third and fourth inputs to stop print through and clean up the background.

Quick Key

Consider this shot from The Saint, pictures courtesy of CFC and Paramount British Pictures Ltd.

Figure 2.1 Blue Screen.

Figure 2.1 is the blue screen foreground that should be composited over the background shown in Figure 2.2.

Step by Step

1. Connect the foreground and background clips to Keylight.
2. Pick the Screen Colour using the color picker tools.

Figure 2.2 Background.

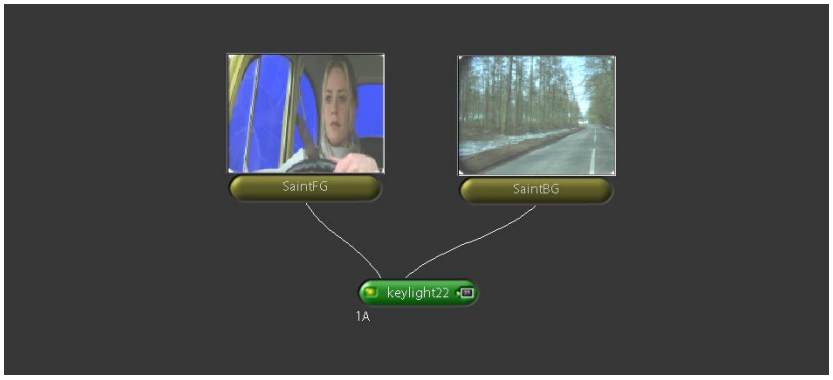

Figure 2.3 Node connections.

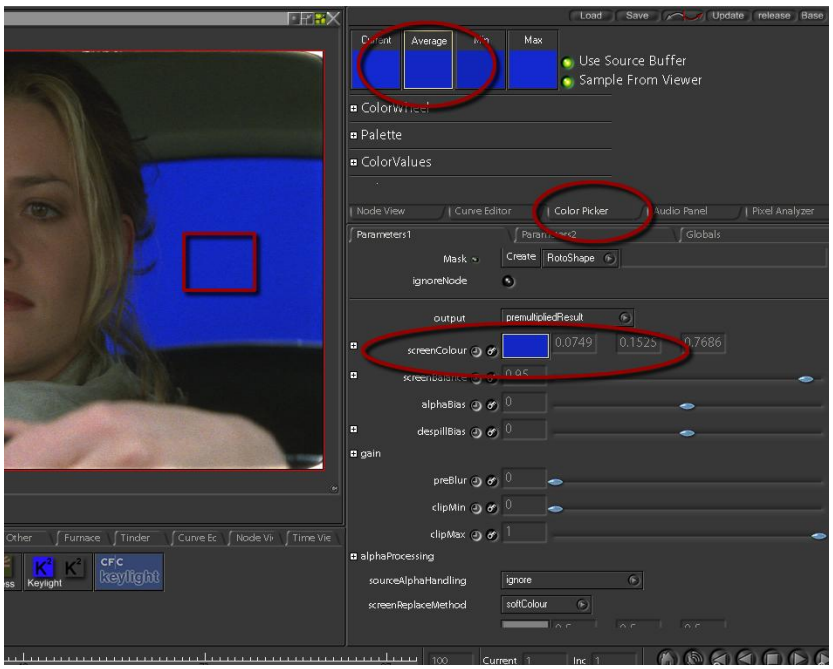

Figure 2.4 Picking the blue screen using Color Picker.

3. There is a tiny amount of blue spill that could be removed. Simply select the despillBias and pick skin tones from the image.

4. That's it. In many cases this is all you will need to do to perform a key. Switch output from premultipliedResult to composite to see the foreground keyed over the background. The final composite is shown in

Figure 2.5 Final composite.

If you want to have a go of this shot, you can! The images can be downloaded from our web site and this quick key is also covered in the Tutorial Chapter. See Tutorial 1: Simple Key on page 48.

Just picking the screen colour may be enough for a lot of keys, but there are many more tools within Keylight that can be used to tackle more complicated shots. These are described in later chapters.

Basic Keying

The following section describes the parameters you need to do basic keying. This will give you enough to tackle most simple keys. A discussion of advanced parameters to fine tune keys and tackle complex keys can be found in the next chapter.

Screen Colour

The screenColour is probably the most important parameter and you should always pick the screen colour before doing anything else. It should be set to the colour of the green or blue curtain behind the foreground object. Select the parameter and pick the screen colour directly from the Viewer by dragging an area.

Figure 3.1 Switch these on.

Make sure you have Use Source Buffer switched on so that the colour values are taken from the parent node and not the Keylight node so that we pick colour values off the prekeyed image. It's also important to do this if you are using a LUT to view your images.

Figure 3.2 Average colour.

Start by using the average colour (Figure 3.2).

Screen Matte

Setting the screen colour will pull a key or in other words, create a matte. This is shown in the alpha channel of the premultiplied output or composite, or viewable by selecting output Screen Matte and viewing the RGB channels. Setting the screen colour will also despill the foreground although you can use the despillBias to remove more spill. In some cases just picking the screen colour is enough to get a decent key. For more information on Screen Colour see page 26. For more information on using the Color Picker see your Shake Reference Guide.

Figure 3.3 shows a well lit blue screen behind an actor.

Figure 3.3 Blue Screen.

You should note that repeatedly picking colours does not add to previous selections and key more of the image with each click. To key more of the image, in other words to improve the background you should use the Clip Min parameter. See Keying More on the facing page.

Viewing the Key

After picking the screen colour you have created a matte (the screen matte) and despilled the foreground. The result can be displayed in a number of different ways, Figure 3.4 on the next page. You can output the final composite of the foreground over the background as an rgba, or you can output the premultiplied or unpremultiplied foreground for compositing elsewhere in the tree. The screen matte and the status view are the other two options which are useful in fine tuning the key rather than as an output image in their own right.

The Status is one of the options in the output menu and shows an exaggerated view of the key so that you can make a more informed decision when tuning the key.

Figure 3.6 shows the Status display after the screen colour has been picked from the image shown in Figure 3.5. Three colours

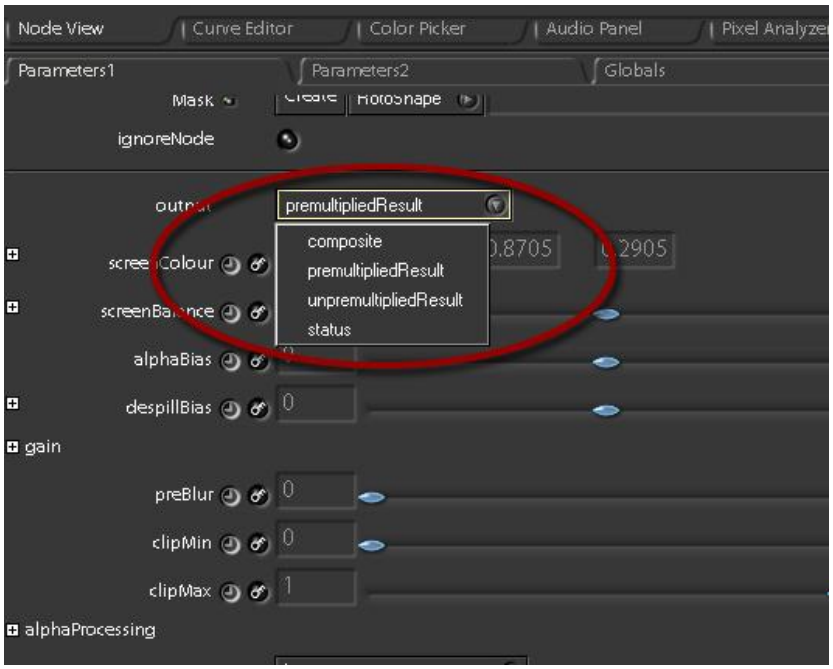

Figure 3.4 Ouput Menu.

are displayed. Black pixels show areas that will be pure background in the final composite. White pixels show areas that will be pure foreground. Midgrey pixels will be a blend of foreground and background pixels in the final composite. You need grey pixels around the edge of the foreground to get a good key at the foreground edge.

Pixels that are a blend between the foreground and background are shown in just one shade of grey. This is done to highlight potential problems with the key. These grey pixels may represent a foreground/background blend of 50/50 or 99/1. No distinction is made as to this ratio.

You may occasionally see other colours in the Status View and these are covered on page 22 in the Advanced Keying Chapter.

Keying More

To improve the key by firming up the foreground so the background doesn't show through you should adjust the Clip Max parameter. To key more of the foreground so that the background is clearer you should use the Clip Min parameter. Look at the alpha channel and the composite while you're doing this. Don't overdo either of these or the edges between foreground and background will become hard.

Figure 3.5 Green Screen.

Figure 3.6 Status.

Despilling the Key

Although the foreground is despilled automatically, you may find the need to pull out a little more of the screen colour after picking from the image. You can do this with the despillBias control (similar to the fgBias control in Keylight 1). Just select this and pick from the foreground. Trial and error is best here, so just pick different areas until you get a good result. If you prefer you can open up the colour controls and, if you wish to remove blue spill, decrease the despillBias.B parameter until the blue disappears.

Figure 3.7 Exaggeratted blue spill.

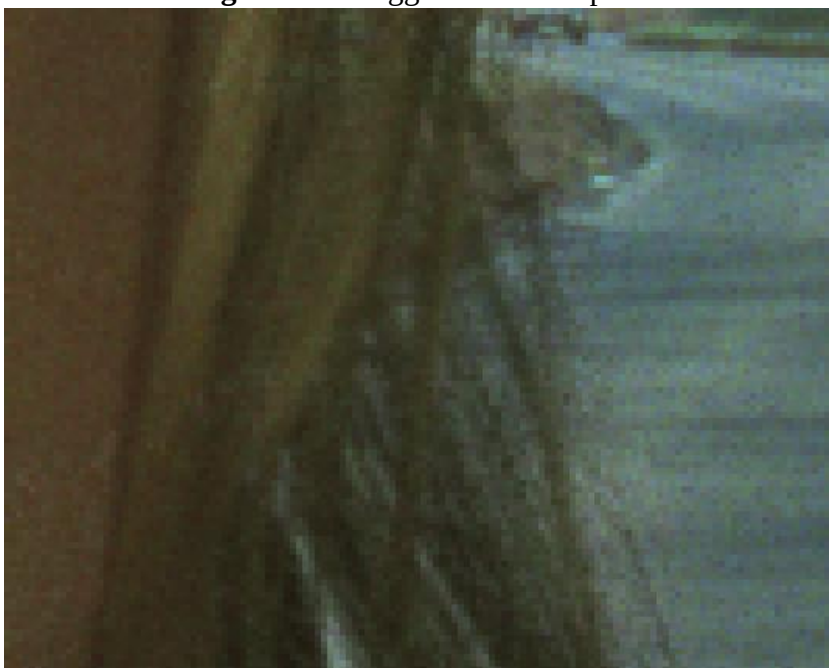

Figure 3.8 Despill Bias used to remove the blue spill.

Advanced Keying

The following section describes the parameters you need to fine tune keys and get the most out of Keylight. Basic parameters covered in the previous chapter may also be covered here in more detail.

Changes

If you are already familiar with Keylight 1 in Shake it's worth pointing out the following label mapping. In particular Keylight 1 screenRange is not equivalent to Keylight 2 screenGain.

Keylight 1	Keylight 2
output	output
screenColour	screenColour
screenRange	clipMin
fgBias	despillBias & alphaBias
*Balance	screenBalance
*Gain	*Gain
midTonesAt	midTonesAt
replaceColour	screenReplaceColour
fgMult	-
fgGamma	-
colourSpace	-

Output

The output parameter allows Keylight to render the final composite of the foreground over the background, or the premultiplied/unpremultiplied foregrounds for compositing further down the tree.

Two options, screen and status, are for viewing the key rather than as an output. The options are:

- composite this renders the foreground composited over the background using all mattes, spill and colour corrections.
- premultipliedResult this creates a premultiplied RGBA foreground.
- unpremultipliedResult this creates an unpremultiplied RGBA foreground.
- screen this is the matte created from picking the screen colour. It does not include any inside or outside mattes.

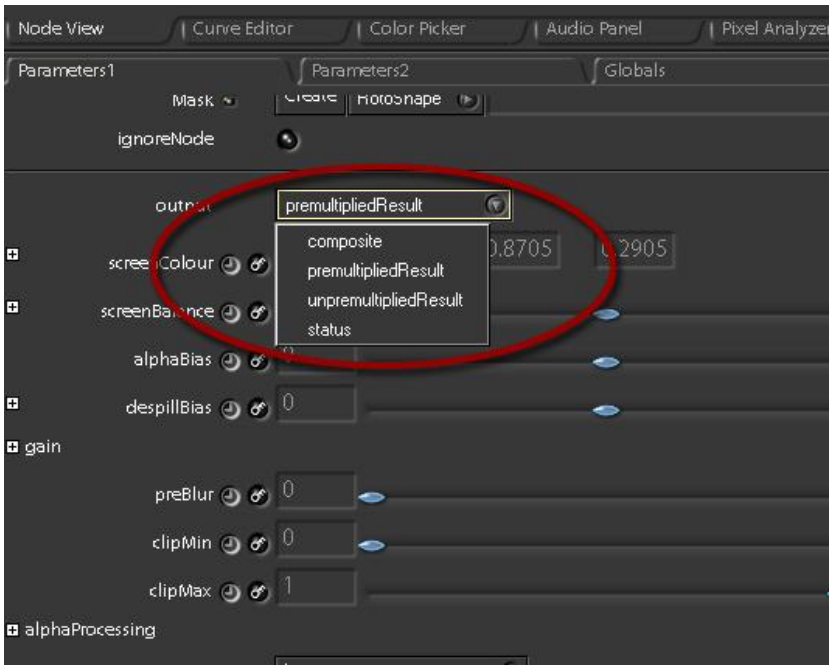

Figure 4.1 Ouput Menu.

- status this renders an exaggeratted view of the key so that minor problems are shown clearly.

Status is one of the options in the output menu and shows an exaggerated view of the key so that you can make a more informed decision when fine tuning the composite.

Figure 4.3 shows the Status after the screen colour has been picked from the image shown in Figure 4.2 on the next page. Three colours are displayed. Black pixels represent pure background in the final composite. White pixels are pure foreground and grey pixels are a blend of the foreground and background pixels. The grey is just one colour to highlight any areas that are not pure foreground or background. Grey pixels do not mean the key is poor the final composite may be fine.

You may occasionally see other colours in the Status View.

Figure 4.4 shows black, white, grey and green pixels. The green pixels are a warning. They show you the parts of the alpha that have changed through processing the alpha channel (clipped, softened or eroded). These areas have had the correct amount of spill removed, but the alpha has subsequently changed and the composite may no longer look right. This can be corrected using the Replace Colour to put back colour in these areas. Figure 4.5 on page 24 is an extreme example to illustrate the point. The Replace Colour has been set to pure red and you can see that this mirrors the green pixels in the Status View.

Similarly you may see blue pixels in the Status.

Figure 4.2 Green Screen.

Figure 4.3 Status.

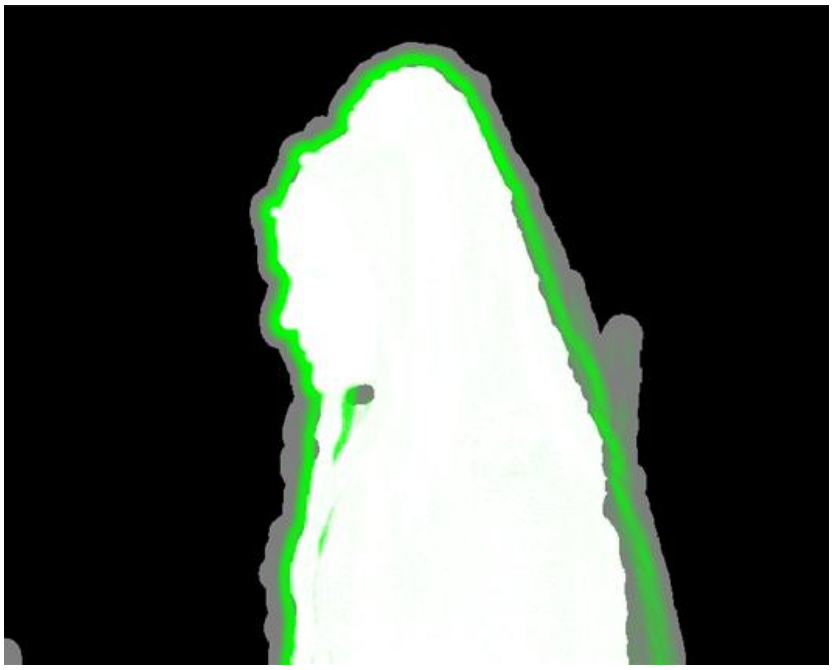

Figure 4.4 Status showing processing of the alpha channel.

Figure 4.5 Composite showing Screen Replace Colour.

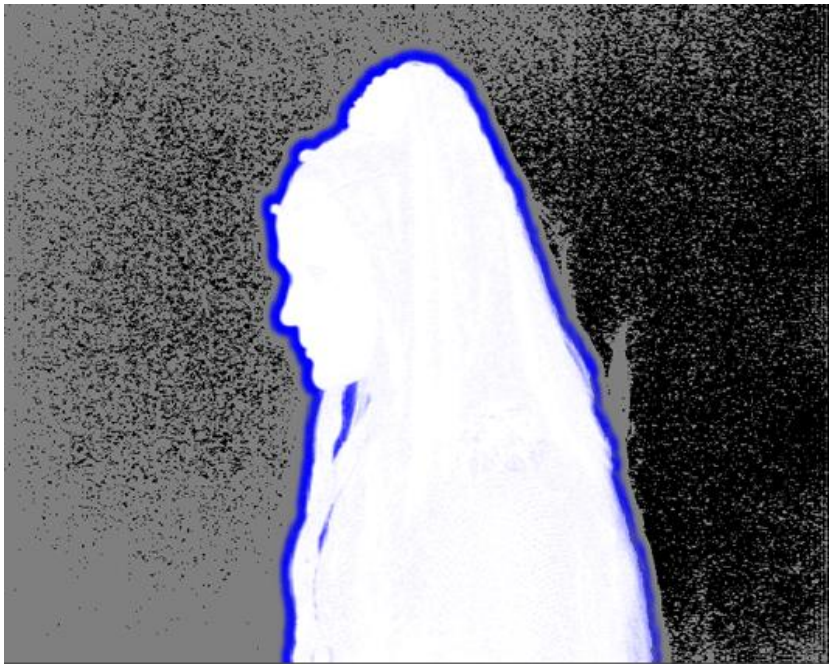

Figure 4.6 Status showing how the inside matte will affect the foreground.

Figure 4.7 Composite showing the inside replace colour.

These represent processed pixels in the inside matte that affect the despill of the foreground. The inside replace colour will be used to modify these pixels. Another extreme example is shown in Figure 4.7. The insideReplaceColour is set to pure yellow and the insideReplaceMethod is hardColour.

You may also see dark red pixels which indicate areas where an outside matte has been used to reduce the transparency of the image.

Screen Colour

The screen colour represents the colour of the pure blue (or green) screen. The first thing you should do when pulling a key is pick the Screen Colour. This single colour has a primary component, blue or green, and that has a saturation. Once the screen colour has been picked, Keylight analyses all the pixels in the image and compares the saturation of the primary component in each of these pixels with the corresponding saturation of the screen colour, setting the alpha and modifying the colour accordingly.

Tip

It's worth sampling a selection of screen (blue or green) colours and then swapping between the average, min and max and viewing the result. Picking different colours will give different results and using the average doesn't always give the best results.

Background Pixel

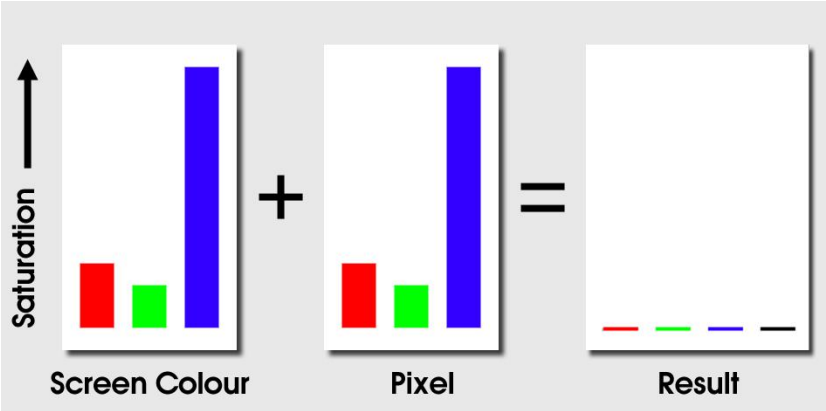

Figure 4.8 Blue screen pixel set alpha to zero.

If the saturation of the pixel in the image is as strong, or greater than the screen colour, then it'll be a pixel from the blue screen background, and that pixel will be set to completely transparent and black. See Figure 4.8.

Edge Pixel

If the saturation of the pixel is less than the screen colour, then it'll be the edge of the foreground object, and we subtract some of the screen colour from the pixel (de spilling) and set

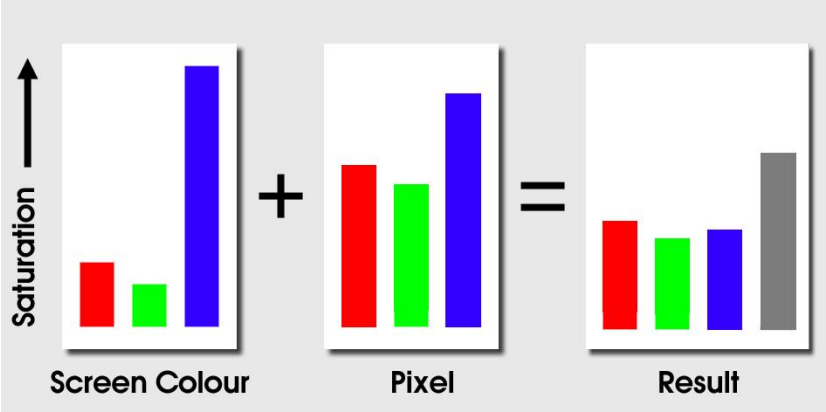

Figure 4.9 Edge pixel gives partial alpha.

the image to semiopaque. See Figure 4.9.

Foreground Pixel

If the primary component in the pixel is not the same as the primary component of the screen colour we have a foreground

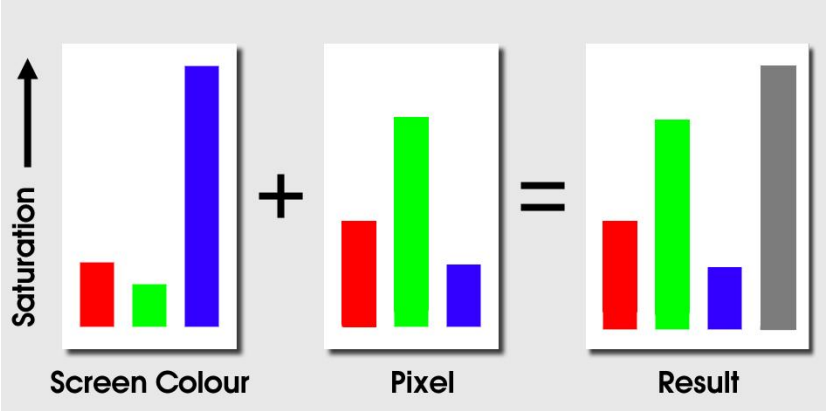

Figure 4.10 Foreground pixel gives full alpha.

pixel, and the alpha is set to completely opaque. The pixel colour is not modified. See Figure 4.10.

Note *You should note that the Screen Colour is a single colour. You are not picking lots of colours that are keyed out.*

Despilling

Although picking the screen colour will remove blue spill, it may not be enough. To despill more use the despillBias control. In the

rare case that the screen has been so badly shot that it's not really blue or green anymore, the alpha bias can be used to get a good key.

Despill Bias

Consider Figure 4.11 from the film Merlin.

Pulling a default key from this will give the extreme blue spill around the edges of her hair as shown in Figure 4.12. To reduce this select the despillBias colour and pick skin tones from the foreground image. This result is shown in Figure 4.13. The alpha channel is shown in

Figure 4.14.

Alpha Bias

You should not normally need to adjust the alpha bias but on those rare occasions when your green screen is more red than green it can be used to put things right. Consider the image shown in Figure 4.15.

This shot, from the film Executive Decision, is actually a green screen although it doesn't look it. The colour of the screen in the background is around 28% red, 25% green and 8% blue. This is in fact red, but only just. Note that the pilot in the cockpit is predominantly brown, at around 42% red, 25% green, 15% blue. So a default key from that screen colour would make the foreground transparent (Figure 4.16) as it is a more saturated red than the red of the screen colour.

In this situation, the **Alpha Bias** can help. Select the alphaBias and pick colours from his mask. This gives the result shown in Figure 4.17.

Clip Min and Max

The clip levels are adjusted using two parameters clipMin and clipMax. Any alpha value at or below clipMin will be set to zero and any alpha value at or above clipMax will be set to 1. Figure 4.18 shows the

original alpha of an image and Figure 4.19 on page 32 shows the result of clipping it. Notice how the grey areas in the black background have been reduced and that the grey edges have hardened up considerably. When compositing, the clipMin control can be used to improve the background image if parts of the foreground are showing through. The clipMax control on the other hand can be used to firm up the centre of the matte, making it less transparent to the background.

Original blue screen.

Figure 4.11 Original blue screen.

Figure 4.12 Default key.

Figure 4.13 Despill Bias.

Figure 4.14 Alpha.

Figure 4.15 Is this the worst green screen you've ever seen?

Figure 4.16 Default key showing the transparency of the foreground.

Figure 4.17 Alpha Bias.

Figure 4.18 clipMin = 0

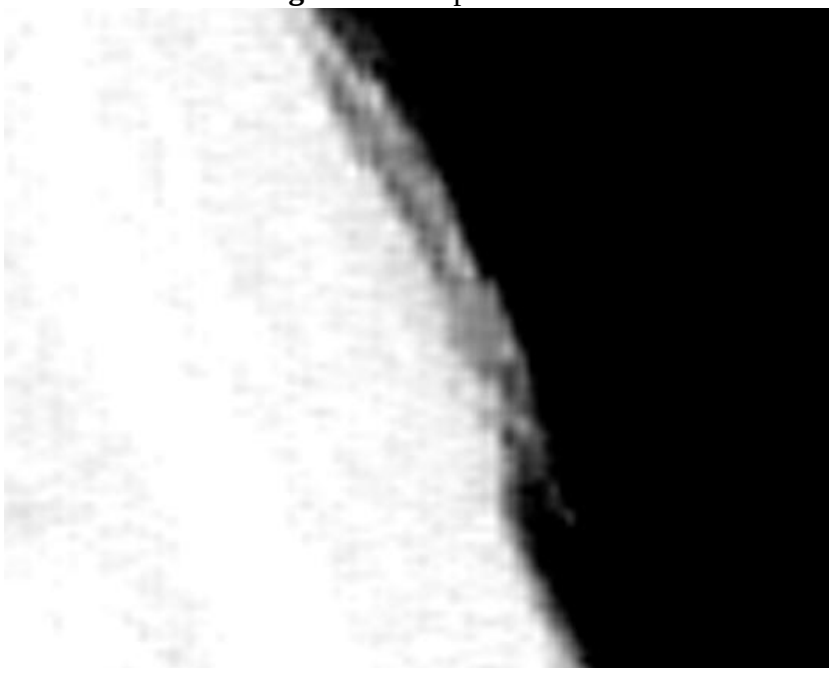

Figure 4.19 clipMin = 0.5

Note *You need to be really careful if you choose to use clipMin and clipMax that you don destroy the edges on your foreground. It is possible to use Clip Rollback to compensate for this.*

Screen Gain

The screen gain controls how much of the screen colour is removed to make the screen matte.

Increasing this value will key more. Figure 4.20 shows the Status after picking the screen colour. You can clearly see that parts of the background are grey where they should be black. When composited you may see faint pixels from the foreground where you should be seeing pure background. Increasing the screen gain will fix this, as shown in Figure 4.21, but increasing it too much will destroy your good work. Like many keying parameters it's a balance not too much, not too little. Increasing the screen gain too much will lead to the background showing through the foreground and edge detail will be destroyed.

Figure 4.23 on page 35 shows this quite well. Note the steering wheel is black when it should be white. If you look at the composite you will see the background showing through here. Also, some of the fine hair detail on the actor, visible in Figure 4.22, has been eroded in Figure 4.23.

Screen Balance

Saturation is measured by comparing the intensity of the primary component against a weighted average of the two other components. This is where the screenBalance control comes in. A balance of 1 means that the saturation will be measured against the smallest of the other two components in the screen colour.

A balance of 0 means that the saturation will be measured against the larger of the other two components. A balance of 0.5 will measure the saturation from the average of the other two components.

The appropriate balance point for each image sequence you key will be different depending on the colours in that image. Generally speaking, blue screens tend to work best with a balance of around 0.95 and green screens with a balance of around 0.5. The screen balance value is set automatically on pickinbg the screen colour but this can be adjusted. If the key is not working too well with these settings, try setting the balance to about 0.05, 0.5 and 0.95 and see what works best.

PreBlur

Some shots can be improved by softening the foreground image that is used to generate the key. The original image is then used

Figure 4.20 Status after picking the Screen Colour.

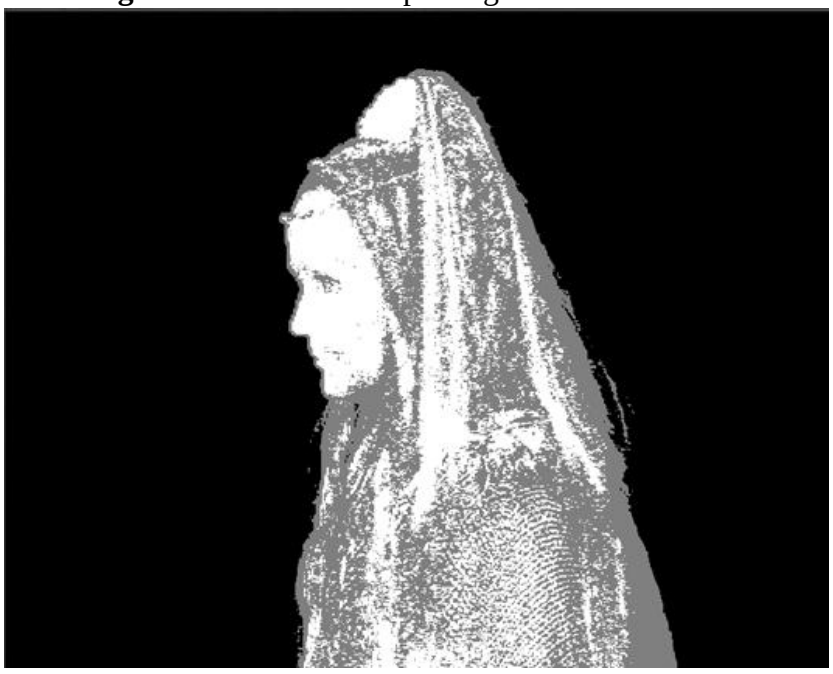

Figure 4.21 Status showing the increase in Screen Gain.

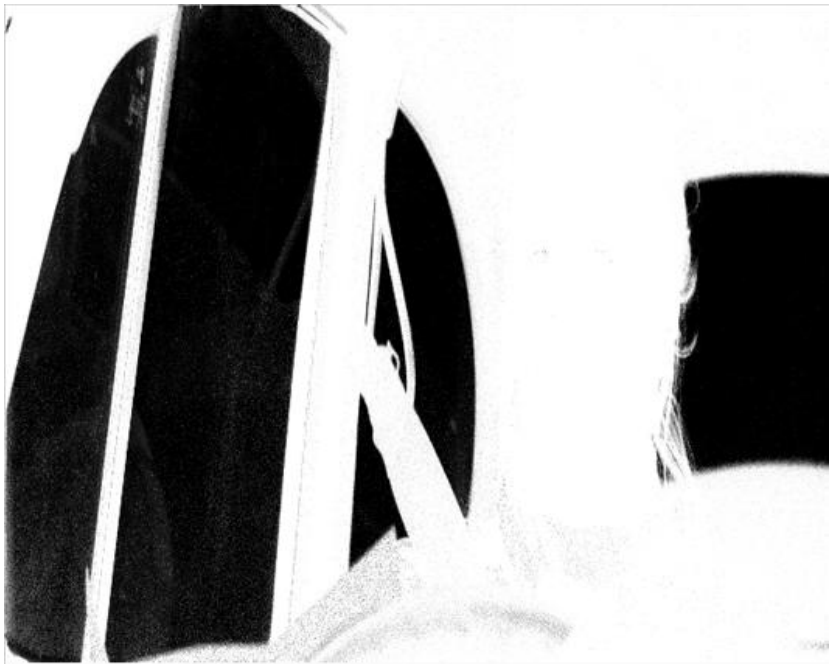

Figure 4.22 screenGain = 1.05 giving a good screen matte.

Figure 4.23 screenGain = 1.50 giving background show through and over eroded edges.

in the composite and colour corrections. The preBlur parameter is used to do this. DV footage or grainy shots may benefit from subtle use of this control.

Gains

Keylight creates the screen matte after the screen colour has been picked. You can make fine adjustments to this matte using the gain controls. Increasing the gain controls makes the screen matte more transparent by increasing the amount of screen colour showing through the matte. This tends to tint the edges the opposite of the screen colour (for blue screens, edges become yellow). Decreasing the gain makes the main matte more opaque by reducing the amount of screen colour showing through the matte.

The matte can be adjusted independently in the shadows, midtones and highlights giving more control than the clipping levels.

The level of the midtones can be adjusted too. For example, if you are working on a dark shot you may want to set the midtone level to a dark grey to make the gain controls differentiate between tones that would otherwise all be considered shadows.

Screen Processing

Once you have picked the screen colour and got the screen matte, you may wish to process this matte using the parameters in the screen processing group. The matte can be adjusted using clipping levels, it can be eroded or grown, despotted and softened.

For example, you could soften and erode the matte to produce a hold matte that could be connected to the inside input of another Keylight node further down the tree. This holdout matte could stop print through (foreground transparency).

Clip Rollback

Pulling a screen matte (Figure 4.24) will typically produce lots of transparency (grey) in the matte at the edges. This is good since this is what you need to key hair well. You may also get transparency in the foreground as shown in Figure 4.25. This is bad as your subject will appear slightly seethrough, and this should be corrected.

You can do this by connecting a matte into the third (inside) input, or you can use the clipMax parameter to turn these grey pixels white. This cleans up the foreground (Figure 4.26) but it will also destroy the edge detail you want to keep. This is where clipRollBack comes in. This is used to put back the edges to restore the detail that was lost. A rather exaggerated clip rollback is shown in Figure 4.27 to illustrate the point.

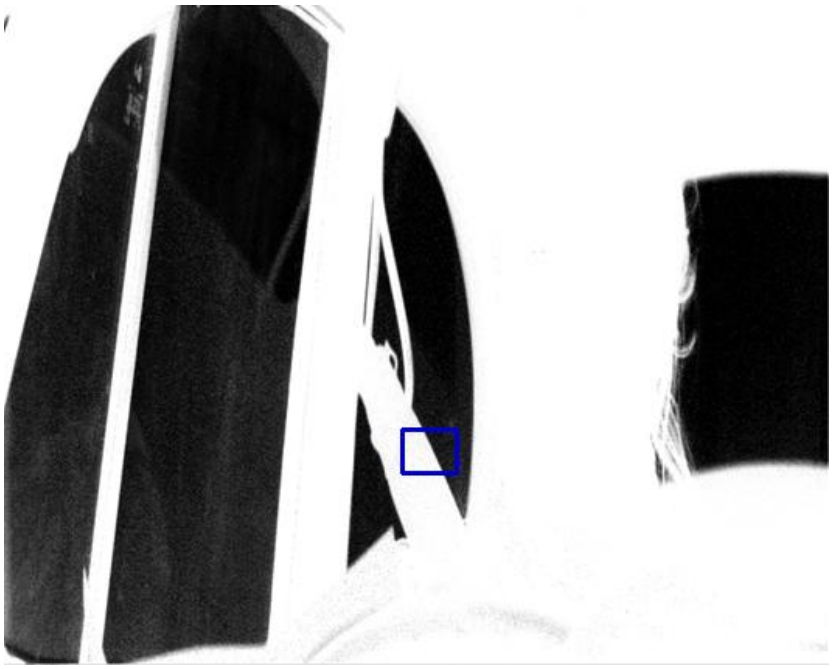

Figure 4.24 Screen matte highlighting the close up view as shown in Figure 4.25.

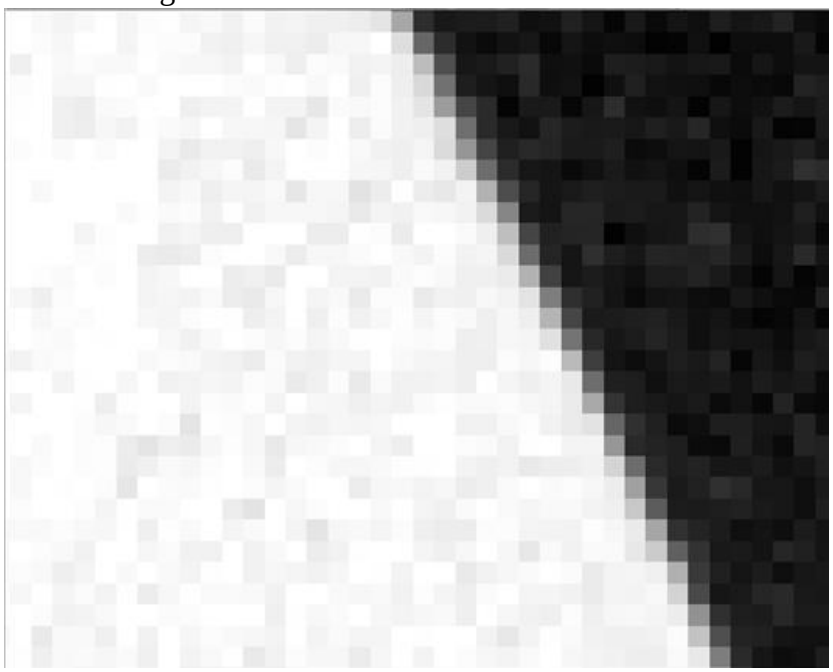

Figure 4.25 Close up screen matte showing unwanted (grey) transparency in the (white) foreground.

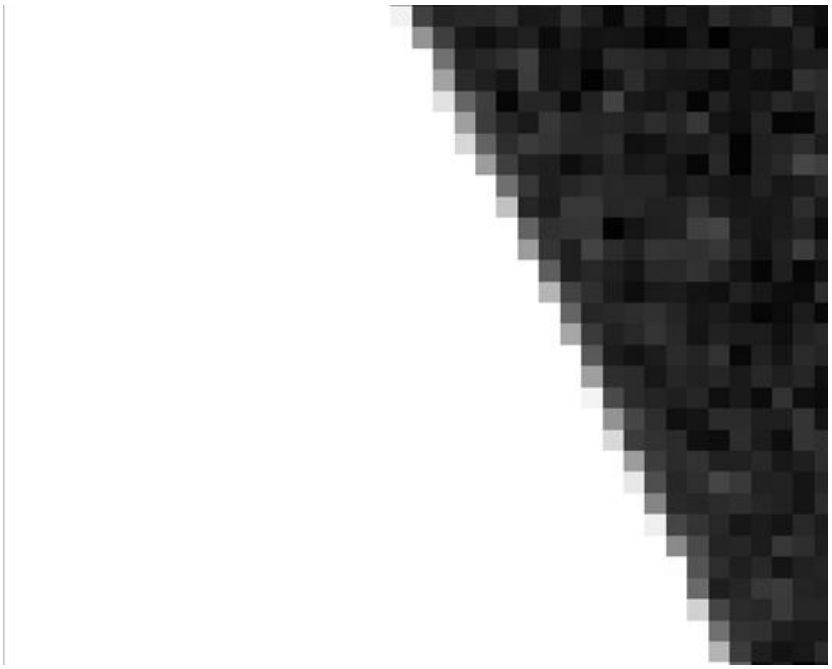

Figure 4.26 clipMax has been used to remove the unwanted grey pixels in the white matte.

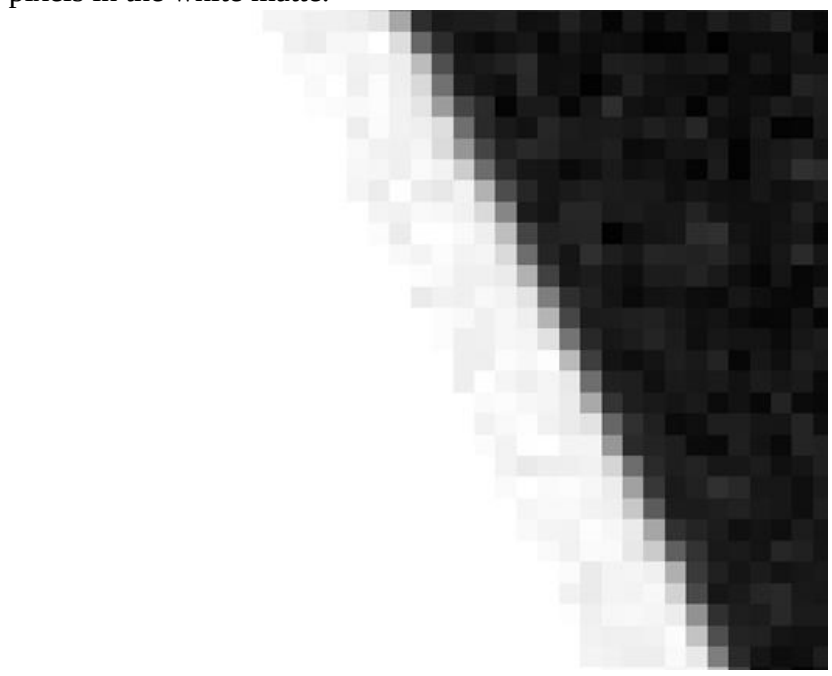

Figure 4.27 clipRollBack has been used to reduce the unwanted erosion of the edge.

Grow & Shrink

This control should not normally be used as eroding the edges can produce a very poor key. However, the growShrink parameter allows you to grow (if greater than zero) or shrink (if less than zero) the alpha in the Screen Matte. These controls are subpixel accurate.

Despot

This controls how much to simplify the matte. It coagulates similar regions so that, for example, black specks in the white matte can be absorbed by the surrounding white areas. Increasing the despotBlack will remove isolated spots of black in the white matte. Increasing despotWhite will remove isolated spots of white in the background up to that size.

Mattes

There are 4 mattes in Keylight.

1. Screen Matte
2. Inside Matte
3. Outside Matte
4. Alpha (Composite Alpha)

The screen matte is generated by the Keylight algorithm after the screen colour has been picked. It can be processed (clipped, eroded, etc) by the screen processing tools.

The inside matte is the hold out matte. It is used to confirm areas that are definately foreground. If your subject has blue eyes and is being shot in front of a blue screen, this matte can be used to key back the eyes. This matte is taken from the 3rd input to Keylight. The embedded alpha channel of the foreground input can be added to this matte using the foregroundAlpha parameter.

The outside matte is the garbage matte and is used to remove unwanted objects (lighting rigs, etc) from the foreground. The matte is taken from the 4th input to Keylight. The inverted embedded alpha channel of the foreground input can be added to this matte using the foregroundAlpha parameter.

The matte used to blend the foreground and background in the final composite is the alpha displayed in the alpha channel of the composite. This matte is the combination of the screen matte, inside and outside mattes.

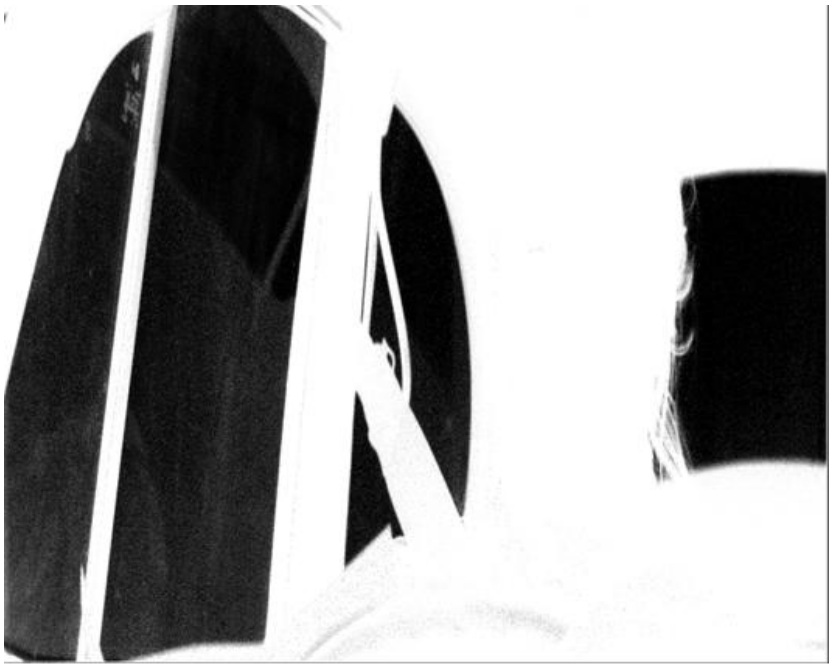

Figure 4.28 Screen Matte.Eroded Matte.

Figure 4.29

Figure 4.30 Eroded matte.

Figure 4.31 Despot.

Inside & Outside Mattes

If you can't adequately improve the screen matte using the clip levels, you can draw a QuickShape round the pixels you definitely want to be foreground or background and use this as a matte input. The inside matte makes the foreground less transparent and the outside matte is used to clean up the background that might have bits of the foreground showing through. It is sometimes referred to as the hold out matte.

The outside matte (garbage matte) is often used to clean up screens that are not a constant colour or have lighting rigs in shot (Figure 4.32) by forcing the alpha transparent.

Figure 4.32 Green screen with lighting rig visible.

The inside matte can be used to keep elements in the foreground that you don want to lose (an actor's blue eyes in front of a blue screen). These mattes should normally be softened externally to blend into the screen matte.

Figure 4.33 on the next page shows a QuickShape drawn around the lighting rig on the left side of the screen. Figure 4.34 shows the matte. Connect the matte to the fourth (outside) input of Keylight

and switch the parameter outsideChannel to alpha. The outside matte forces that part of the image to be in the background thus keying out the rig.

Figure 4.33 QuickShape drawn round the lighting rig.

Figure 4.34 QuickShape matte.

Foreground Alpha

This parameter determines what to do with any embedded alpha in the original foreground image.

- Ignore this will not add any embedded alpha to the screen matte.
- AddToInside the embedded alpha is added to the inside matte.
- AddNegativeToOutside the embedded alpha is inverted and added to the outside matte.

Colour Replacement

Remember that Keylight does two things it removes the screen colour to despill the image and generates an alpha (Screen Matte) to composite the foreground over the background layer.

If you then process the Screen Matte, for example, by eroding the alpha or changing the clip levels, Keylight will be removing the wrong amount of screen colour from the pixels whose transparency have now changed. The Replace Method instructs Keylight how to deal with such pixels. The Status will display which pixels use a replace method. Those pixels who use a replace method because the alpha processing tools modified the transparency will be green, whilst those pixels whose transparency was modified by the inside matte will be blue. See the Status View on page 22.

There are four options to the replace method, these are:

1. None the despilled image is left untouched if the alpha is modified.
2. Source the image will have a corresponding amount of the original pixel (screen colour and all) reintroduced/removed if the alpha is changed.
3. Hard Colour the despilled image has a corresponding amount of the replace colour added for any increase in alpha.
4. Soft Colour the despilled image has a corresponding amount of the replace colour added for any increase in alpha, however, it attempts to modulate the luminance of the resulting pixel so that it matches the original pixel. This will give a more subtle result than the Hard Colour option.

White Point

During its calculations, Keylight needs to know a nominal white point for the image it is keying. For 8 and 16 bit integer images,

this is simply the maximum pixel value of 255 and 65535 respectively.

However, with floating point images, pixels are not necessarily constrained between 0 and 1. Pixel values may be well above 1. When Keylight attempts to key such images it can make the alpha of the resulting key far too solid, as it is assuming the white point is at 1.0.

To get around this problem, Keylight 2.1 has a parameter, which you use to set the nominal white point of the image.

So if you are keying in float and your image has ordinary pixels that have values above 1.0 your alpha will be too hard.

To solve this, set the whitePoint slider to the maximum value of the pixels in your image that is being keyed. You don't have to be really accurate when doing this.

Example

Here's an example to illustrate the point.

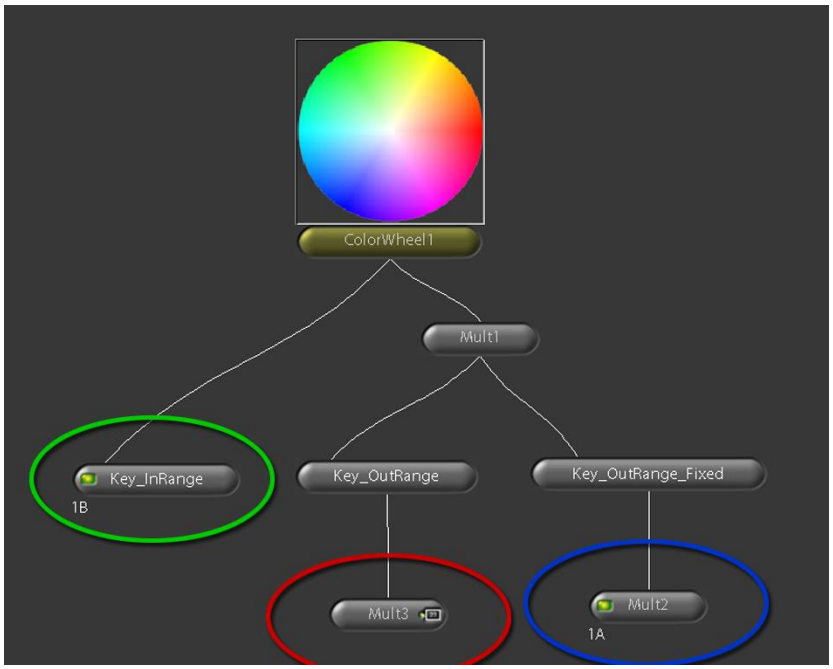

Figure 4.35 Example tree.

We have taken a standard Shake colour wheel and keyed it three different ways as shown in Figure 4.35.

1. Pixel range 0 to 1, white point = 1. This node is shown in green. This gives the correct key as shown in Figure 4.36 on page 47.

2. Pixel range 0 to 5, white point = 1. This node is shown in red. The colour wheel is multiplied by 5 then keyed and multiplied by 0.2. This gives an incorrect result as shown in Figure 4.37. Alpha is far too hard. It should match the alpha shown in Figure 4.36. RGBs are fine though.

3. Pixel range 0 to 5, white point = 5. This node is shown in blue. This gives a correct result. Alpha and RGB match example 1.

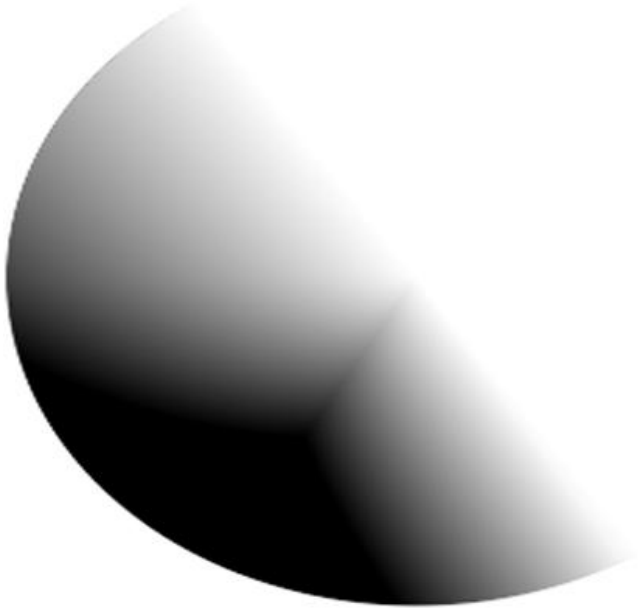

Figure 4.36 Normal range and white point = 1.

Figure 4.37 Abnormal range and white point = 1.

Tutorial

Introduction

We have included several tutorials with example images that you can use to practice Keylight.

- Tutorial 1: Simple Key
- Tutorial 2: Fine Tuning a Key
- Tutorial 3: Extreme Blue Spill
- Tutorial 4: A Red Green Screen
- Tutorial 5: Inside & Outside Masks

Example Images

The tutorial images referred to in this chapter can be downloaded from our web site www.thefoundry.co.uk

Tutorial 1: Simple Key

Using the blue screen clip from The Saint, you will composite the actor over the background. You will learn how to:

- Apply Keylight to a layer.
- Pick the Screen Colour.
- View the Final Result.

Download File

Saint.tar.gz

The clips you will need for this task are called SaintFG.tif and SaintBG.tif, pictures courtesy of CFC and Paramount British Pictures Ltd for the film The Saint. You should import them into After Effects and create a new composition containing the blue screen of Elizabeth Shue layered over the road.

Figure 5.1 is the blue screen foreground that should be composited over the background shown in Figure 5.2.

Step by Step

1. FileIn SaintFG.tif and SaintBG.tif. Load Keylight from the Key tab panel (Figure 1.3 on page 11). Connect SaintFG to the first (foreground) input and SaintBG to the second (background) input of the Keylight node.

Figure 5.1 Blue Screen SaintFG.tif

Background

Figure 5.2 SaintBG.tif

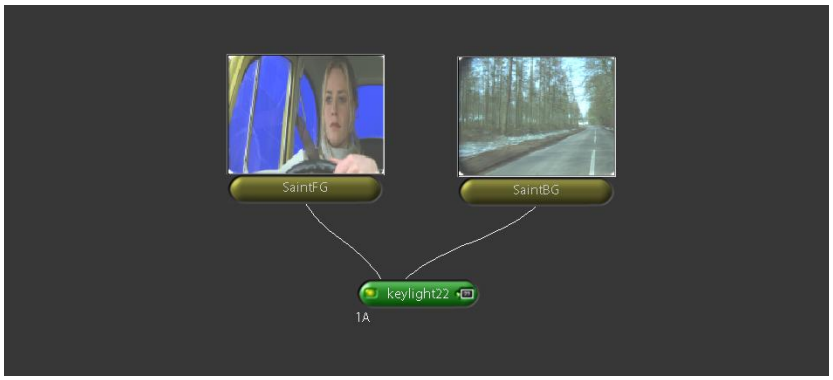

Figure 5.3 Node connections.

2. View and edit the Keylight node. Select screenColour and using the color picker tools select the blue screen colour from the Viewer (Figure 5.4). A good place to pick is

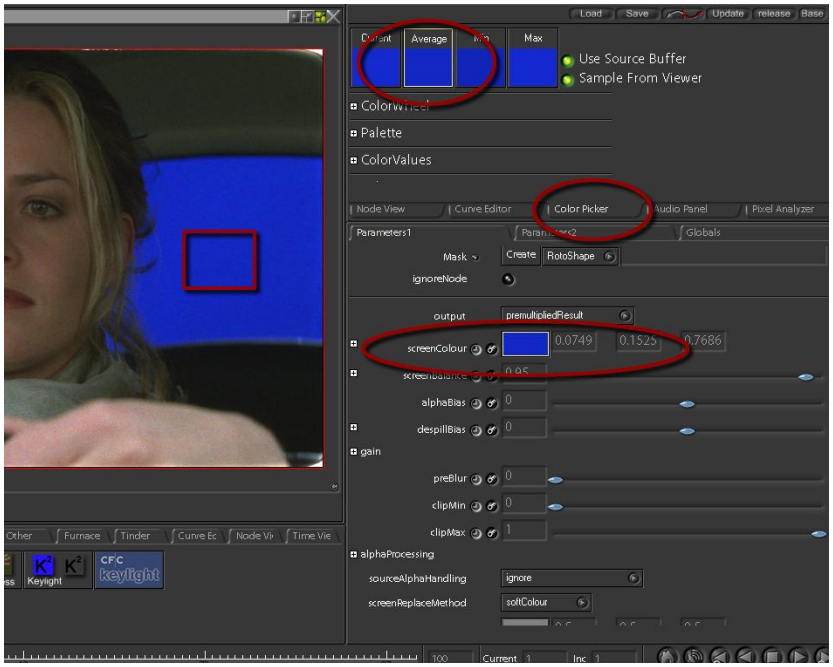

Figure 5.4 Picking the blue screen using Color Picker.

the blue from the back windscreen as this has no reflections. Picking this blue will key the back windscreen perfectly leaving reflections in the side window.

3. That's it. In many cases this is all you will need to do to perform a key, since selecting the screen colour creates a matte and despills the foreground. Switch output from premultipliedResult to composite to see the foreground keyed over the background. The final composite is shown in

Figure 5.5 Final composite.

Figure 5.5.

There are a couple of extra steps that can be taken to fine tune this key and these are discussed in Tutorial 2: Fine Tuning a Key on the next page.

Tutorial 2: Fine Tuning a Key

Using the images from the film The Saint, you will learn how to fine tune the key pulled in Tutorial 1. You will learn how to:

- Use Status to highlight any problems.
- Use the clipMin to improve the background.
- Use the Despill Bias to remove more blue spill.

Download File

Saint.tar.gz

Step by Step

1. FileIn SaintFG.tif and SaintBG.tif. Connect the blue screen foreground to the first input of Keylight and the road background to the second input. Select the screenColour and click and drag an area on the blue screen in the viewer window.

Note *These steps were covered in greater detail in the previous chapter.*

2. Before we do anything else we need to look at the quality of the key so far. On first inspection, the composite looks pretty good, but it's hard to judge. You should also look at the alpha channel to see how good the matte is in the foreground and backgrounbd areas. To see any potential problems more clearly, switch the output to Status as shown in Figure 5.6. Here we can see that the windscreens are a mix-

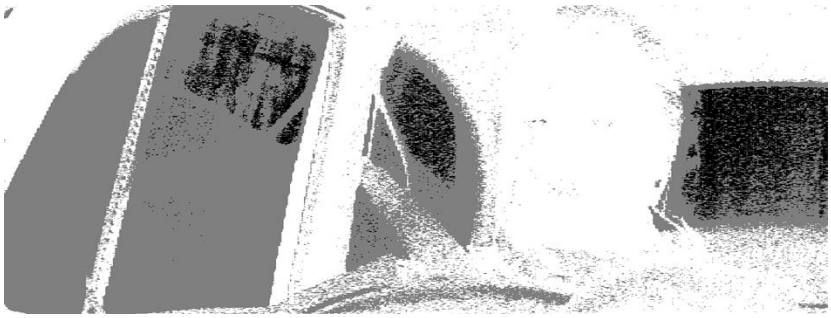

Figure 5.6 Status showing grey pixels in the background.

ture of black and grey pixels. The black pixels tell us that pure background will be seen here in the final composite. The grey pixels tell us that there will be a mix of foreground and background pixels. Now remember that the status view is an exagerrated version of the key to help us quickly see where there may be problems. You should always look at the screen matte alpha and the composite. But for now, let's

try and clean up the background showing through the windows, while leaving some reflections in the side window. In other words we need mostly black pixels with a few grey ones.

3. Figure 5.7 shows the Status view that we're aiming for. The background has been cleaned up and we still have some reflections in the side window. To get this you should increase the screenGain to 1.1.

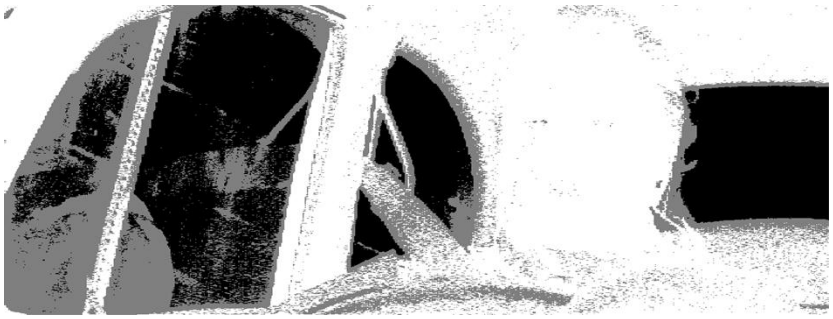

Figure 5.7 Improved screen matte with screenGain.

4. There is a small amount of print through in the foreground under the seat belt. You can see this as the grey pixels in Figure 5.7. This can be removed with a hold matte. We'll use Keylight to pull a screen matte then soften and erode the edges and feed it back into another Keylight node. The tree

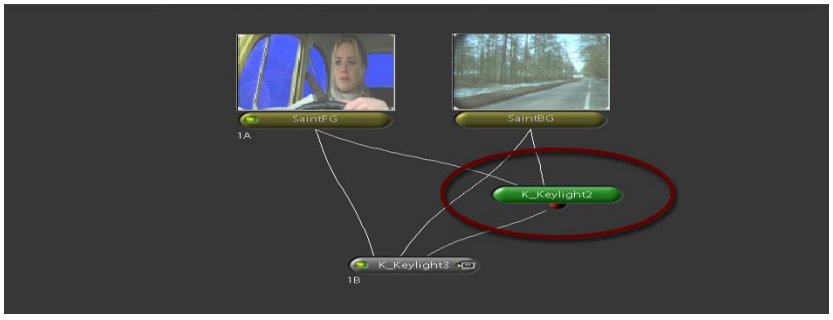

Figure 5.8 Keylight (highlighted in red) used to generate a hold matte for another downstream Keylight node.

is shown in Figure 5.8.

5. Finally, if you look closely at the composite you will see a tiny amout of blue spill on the woman's hand and in her hair. This was from reflected light from the blue screen. Select despillBias and pick some skin tones to remove it. Use the Comparison Buffer to see the difference.

Figure 5.9 Final Composite

Tutorial 3: Extreme Blue Spill

This is a really interesting clip from the film Merlin. The results with Keylight are certainly not perfect, indeed it is unlikely that you will ever end up with a truly realistic looking shot. However, there are some interesting things to observe. You will learn how to:

- Reduce the blue spill using Despill Bias.

Download File

MerlinBlue.tar.gz

Step by Step

1. FileIn the MerlinBlueFG.tif (Figure 5.10) and MerlinBlueBG.tif clips and connect to Keylight.
2. Pick the purest blue you can see for the screenColour.
3. Select despillBias and pick skin tones from the face and neck. View the composite as shown in Figure 5.12.

Tutorial 4: A Red Green Screen

Using the images from the film Executive Decision, you will learn how to pull a key from a poor green screen using the Despill Bias control. You will learn how to:

- Pick the Screen Colour.
- Drop the Despill Bias.
- Produce a final composite.

Download File

ExecutiveDecision.tar.gz

Figure 5.10 A tricky blue screen.

Figure 5.11 Serious blue spill.

Figure 5.12 Result.

Step by Step

1. FileIn ExecFG.tif (Figure 5.13) and ExecBG.tif. Connect to Keylight. The foreground image is actually a green screen

Figure 5.13 Poor Green Screen.

   shot, although it doesn't look it. If you analyse the pixels it's slightly more red than green. To key this we'll have to fool Keylight.

2. Pick the Screen Colour. You should go for the slightly darker green patch to the left of the pilot. Although feel free to experiment picking different parts of the green screen. The initial selection gives the composite shown in Figure 5.14.

3. To fix this select the alphaBias and pick colours from the mask. The result is shown in Figure 5.15.

4. If you look closely, the background and foreground needs

Figure 5.14 Default key.

Figure 5.15 Alpha Bias.

cleaning up. Figure 5.16 shows the Status View. We will

Figure 5.16 Status View.

use the alpha processing tools to make the cockpit windows black and the pilot white.

5. Increase clipMin to 0.2 to remove some of the foreground showing through the background. Decrease clipMax to 0.7 to improve the opacity of the foreground. Increase softness to 1, despotBlack to 2 and despotWhite to 2.

6. Use the screenReplaceMethod to pull some of the original source image through the composite. This improves the apparant graininess in the foreground. Figure 5.18 shows the differences.

Figure 5.17 Composite.

Figure 5.18 Close up view. The left image has Replace Method set to Source. The right image has Replace Method set to None.

Tutorial 5: Inside & Outside Masks

Using the 16 bit blue screen test card image you will learn how to:

- Use Inside and Outside mattes.
- Use the Replace Method to put back keyed out colours.
- Process the Screen Matte.

Download File

TestCard.tar.gz

Step by Step

1. FileIn TestCard1.cin (Figure 5.19) and ColourGrid1.cin and connect them to Keylight.
2. Pick the blue from the image. Keylight will create a matte and despill the foreground as shown in Figure 5.20.
3. To remove the garbage around the subject we will use an outside matte. Use a QuickShape to draw a spline around the person and test cards. This is shown as the yellow line in Figure 5.21. Connect the QuickShape node to the 4th input

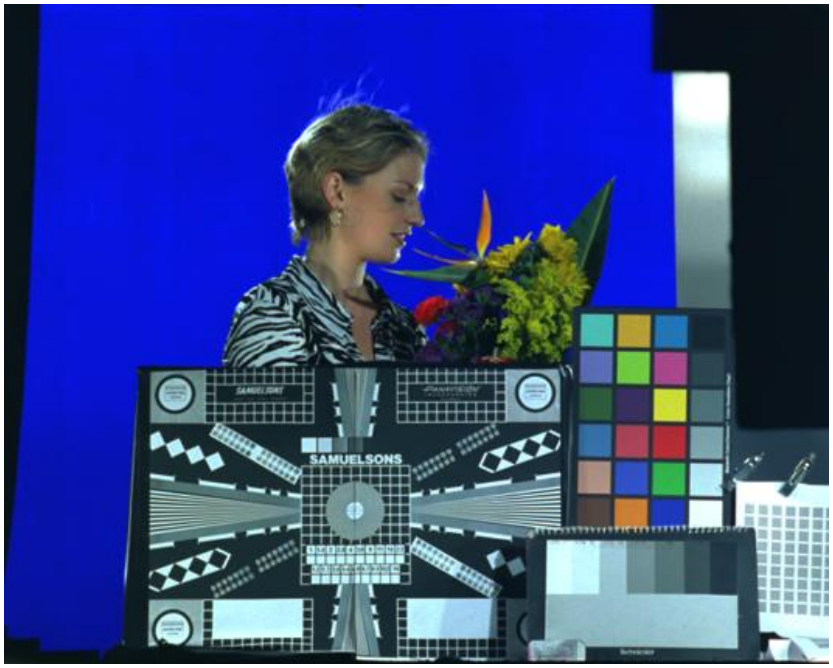

Figure 5.19 Test Card.

Figure 5.20 Basic Key.

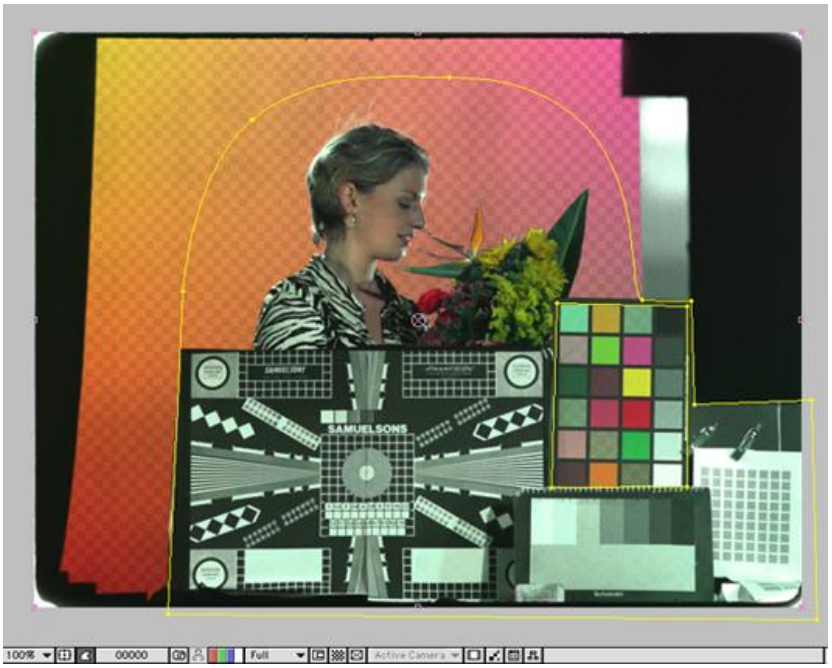

Figure 5.21 Outside matte.

of Keylight.

4. Add an Invert node after the QuickShape or you'll be removing the person rather than the unwanted pixels at the screen edges. You will also need to increase the QuickShape DOD by adding a SetDOD and setting the size to that of the TestCard (720x540).

5. You will have noticed in Figure 5.22 the "dirt" around the subject's head. Clealy we have to improve the key. You can also see the the faults in the matte if you view the Status as shown in Figure 5.23.

6. Increase the screenGain to 1.1. This cleans up some of the background as shown in Figure 5.24.

7. Sections 7 to 10 will cover the changes to the Screen Matte that will improve the key. Decrease the clipMax from 1 to 0.7. This will improve the foreground as shown in Figure 5.25. However, you will notice in the composite that the edges have become a little hard. We can fix this using the clipRollback and softness in the Screen Processing group.

8. Increase the clipRollback to 3 and the softness to 1. It's also worth trying to improve the key around the spikey flowers with a subpixel erode of the edge. Change Screen Grow/Shrink to 0.5.

9. To remove the foreground spots increase despotBlack to 1. The result is shown in Figure 5.26.

Figure 5.22

Figure 5.23

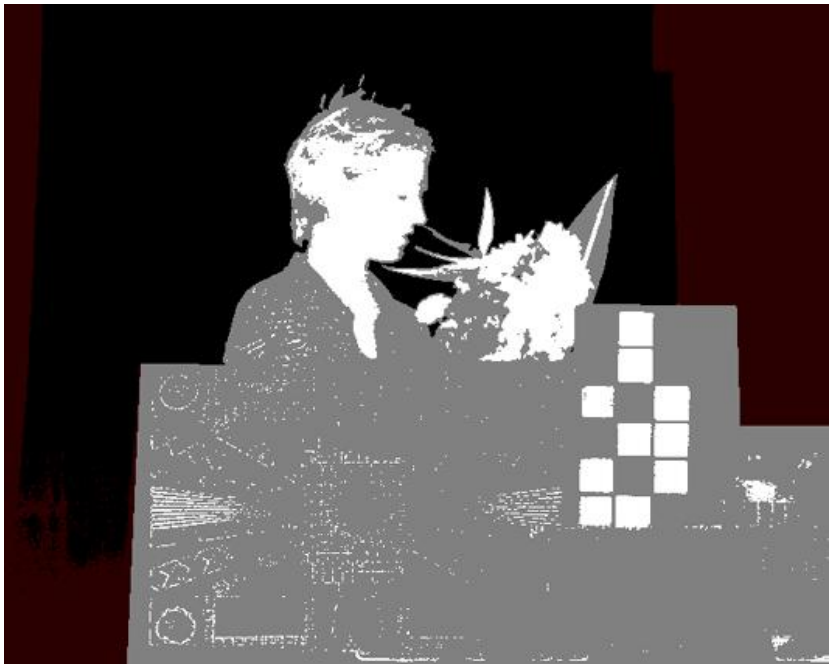

Figure 5.24 Increase clipMin.

Figure 5.25 Clip White.

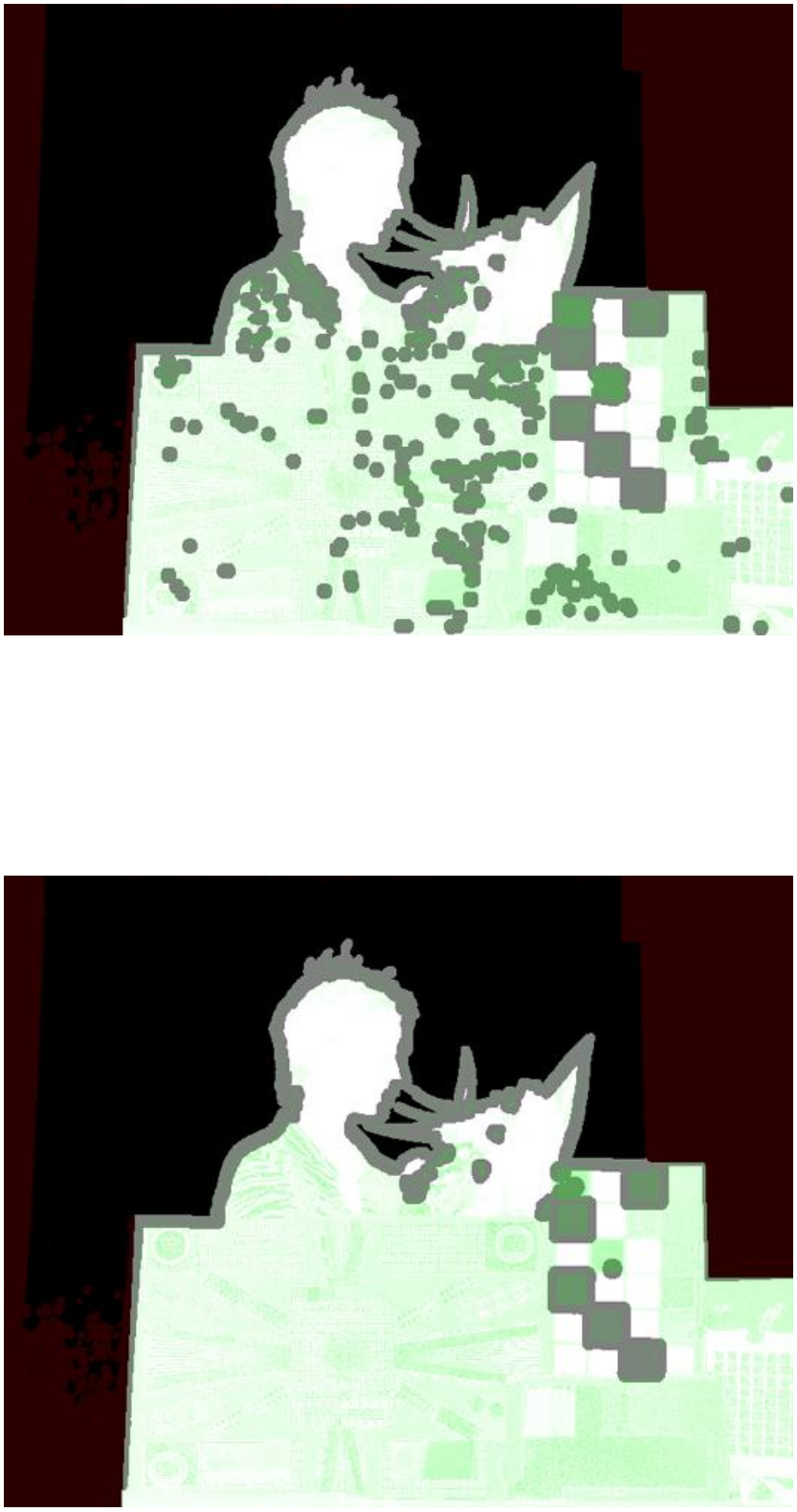

Figure 5.26 Despot Black.

10. The composite is shown in Figure 5.27. You will see that the colours of the colour swatches have been altered by the Keylight algorithm. This can be fixed with an inside matte.

Figure 5.27 Composite.

11. Draw a rough QuickShape around just the colour swatches . Set the insideReplaceMethod to Source to pull back the original colours.

12. That's it.

Figure 5.28 Inside matte and Replace Method Source.

Appendix A

Release Notes

This appendix describes the requirements, new features, improvements over previous versions, fixed bugs and known bugs and workarounds in Keylight.

Keylight 2.2v2

This is a maintenance release of Tinder for Shake to support Shake 4.10 and Intel Macs.

Release Date

1 June 2006.

Requirements

1. Shake 4.00/4.10 on Mac OS X (Intel and PPC) and Linux (RH9).
2. Foundry FLEXlm Tools (FFT) (4.0v1 or later) for floating license support.

New Features

There are no new features.

Improvements

There are no improvements.

Bug Fixes

There are no fixed bugs.

Known Bugs and Workarounds

1. DOD BUG ID 257 the softness parameter in screenProcessing pulls in black pixels from outside the frame. As a workaround increase the DOD of the plate before feeding it into Keylight.
2. DOD BUG ID 256 if you attached a quickshape into the third or fourth input then insert an invert node between the quickshape and Keylight, Keylight will use the inverse of the default dod of the shape rather than the image size, leading to an incorrect result. There is an easy workaround. Just insert a setDOD after the invert node and set this to the image size.

Keylight 2.2v1

This is a maintenence release of Keylight with FLEXlm licensing builtin.

Release Date

January 2006

Requirements

1. Shake 4.00 on Mac OS X (10.3.9+), Linux (RH9).
2. Foundry FLEXlm Tools (FFT) (4.0v1 or later) for floating license support.

New Features

There are no new features.

Improvements

FLEXlm the plug-ins are licensed with FLEXlm.

Bug Fixes

1. Various floating license bugs affecting sites running very large numbers of licenses have been fixed with the introduction of FLEXlm licensing.
2. Fixed handling of BWA (2 channel) images. Previously only the luminance (BW) channel was only applied to the green channel, leaving the red and the blue empty. In addition fixed a buffer allocation issue, where certain combinations of Foundry nodes could cause a crash. In particular, putting a ColourWheel into a LuminanceKey into some Tinder nodes would cause a crash. This has been fixed.

Known Bugs and Workarounds

1. DOD BUG ID 257 the softness parameter in screenProcessing pulls in black pixels from outside the frame. As a workaround increase the DOD of the plate before feeding it into Keylight.
2. DOD BUG ID 256 if you attached a quickshape into the third or fourth input then insert an invert node between the quickshape and Keylight, Keylight will use the inverse of the default dod of the shape rather than the image size, leading to an incorrect result. There is an easy workaround. Just insert a setDOD after the invert node and set this to the image size.

Keylight 2.1v3

This is a maintenence release of Keylight to support Shake 4.00.

Warning

If you upgrade to Keylight 2.1, old scripts built with Keylight 2.0 will not work and will need converting or redoing.

Release Date

July 2005

Requirements

1. Shake 4.00 on Mac OS X, Linux (RH9).
2. Foundry License Manager 3.1 or later. (Built with Licensing Library 3.0v7v4)

New Features

1. The Shake help button now launches a basic html help page.
2. The version number is now included in the parameter list of the plug-in.

Improvements

Improved timing of the memory allocation enabling more robust handling of large images.

Bug Fixes

There are no fixed bugs.

Known Bugs and Workarounds

1. DOD BUG ID 257 the softness parameter in screenProcessing pulls in black pixels from outside the frame. As a workaround increase the DOD of the plate before feeding it into Keylight.
2. DOD BUG ID 256 if you attached a quickshape into the third or fourth input then insert an invert node between the quickshape and Keylight, Keylight will use the inverse of the default dod of the shape rather than the image size, leading to an incorrect result. There is an easy workaround. Just insert a setDOD after the invert node and set this to the image size.

Keylight 2.1v2

This is a maintenence release of Keylight that fixes a bug.

Warning

If you upgrade to Keylight 2.1, old scripts built with Keylight 2.0 will not work and will need converting or redoing.

Release Date

17 February 2005

Requirements

1. Shake 3.01 or 3.50 on Mac OS X, Irix, Linux (RH9).
2. Foundry License Manager 3.1 or later. (Built with Licensing Library 3.0v7v3)

New Features

There are no new features.

Improvements

There are no improvements.

Fixed Bugs

1. Clipping on despill BUG ID 301 negative values in a floating point image that are despilled in Keylight would be clipped to zero. This has been fixed.
2. Multiple Ethernet cards on Mac OSX. Although multiple ethernet cards have been supported on OSX for a while, additional interfaces of a non ethernet type are being returned by OSX. These non ethernet interfaces have a null machine address and cause the plug-in to crash. The licensing has been updated in this release to safe guard against null machine addresses and machine addresses of insufficient length.

Known Bugs and Workarounds

1. DOD BUG ID 257 the softness parameter in screenProcessing pulls in black pixels from outside the frame. As a workaround increase the DOD of the plate before feeding it into Keylight.
2. DOD BUG ID 256 if you attached a quickshape into the third or fourth input then insert an invert node between the quickshape and Keylight, Keylight will use the inverse of the default dod of the shape rather than the image size, leading

to an incorrect result. There is an easy workaround. Just insert a setDOD after the invert node and set this to the image size.

Keylight 2.1v1

This is a new release of Keylight that changes the despill sliders and adds a white point parameter.

Warning

If you upgrade to Keylight 2.1, old scripts built with Keylight 2.0 will not work and will need converting or redoing.

Release Date

18 January 2005

Requirements

1. Shake 3.01 or 3.50 on Mac OS X, Irix, Linux (RH9).
2. Foundry License Manager 3.1 or later. (Built with Licensing Library 3.0v7)

New Features

1. A white point slider has been added. Keylight assumes a white point of 1 in its calculations. If you are using floating point images that routinely have pixels values greater than 1 (i.e. not just in very bright lights or specular highlights), you may see harsh edges in keys that you pull. The alpha will be incorrectly set due to the white point assumption. Increasing the white point value in this new slider will compensate for this. You will need to judge the key by eye in setting this value.

Improvements

1. The despillBias and alphaBias sliders have been changed to colour parameters. A very small subset of shots could not be despilled using the sliders in Keylight 2.0, and some customers preferred picking colours from the screen to despill (just like they used to with fgBias in Keylight 1). So we've put this back.
2. The screenStrength parameter has been remaned to screenGain.
3. The strengths group has been renamed to gain.

Bug Fixes

There are no fixed bugs.

Known Bugs and Workarounds

There are no known bugs.

Keylight 2.0v5

This is a maintenence release of Keylight to fix a bug.

Release Date

14 January 2005

Requirements

1. Shake 3.01 or 3.50 on Mac OS X, Irix, Linux (RH9).
2. Foundry License Manager 3.1 or later. (Built with Licensing Library 3.0v7)

New Features

There are no new features.

Improvements

There are no improvements to existing features.

Fixed Bugs

1. Blur BUG ID 214 alpha values of floating point images would fail to be clamped to 1 if either the preBlur or softness parameters were increased above their default values of 0. This has been fixed.
2. Version BUG ID 215 the version number of Keylight 2.0v4 for Shake 3.50 on Mac OS X was incorrectly called 2.0v43.01.
3. Screen Replace setting the Screen Replace Method to Soft gave overly harsh alphas. This has been fixed.

Known Bugs and Workarounds

There are no known bugs.

Keylight 2.0v4

This is a maintenence release of Keylight to fix a floating point bug.

Release Date

December 2004

Requirements

1. Shake 3.01 or 3.50 on Mac OS X, Irix, Linux (RH9).
2. Foundry License Manager 3.1 or later. (Built with Licensing Library 3.0v7)

New Features

There are no new features.

Improvements

1. There is a list of the parameter name mappings from Keylight 1 to Keylight 2 included in the Advanved Keying chapter. This should help compositors experienced with Keylight get used to the new version.

Fixed Bugs

1. Floating point BUG ID 174 if a matte was put into the third input (inside matte) of Keylight the pixels within the area defined by the matte would be clipped to 1.0. This has been fixed.

Known Bugs and Workarounds

There are no known bugs.

Keylight 2.0v3

This is a maintenence release of Keylight to fix a licensing bug.

Release Date

November 2004

Requirements

1. Shake 2.51 on Windows
2. Shake 3.01 or 3.50 on Mac OS X, Irix, Linux (RH9).
3. Foundry License Manager 3.1 or later. (Built with Licensing Library 3.0v7)

New Features

There are no new features.

Improvements

There are no improvements.

Bug Fixes

1. Licensing. If the plug-ins tried to get a license from a machine that wasn't running a foundry license server, the plug-in would crash. This has been fixed.

Known Bugs and Workarounds

There are no known bugs.

Keylight 2.0v2

This is a maintenence release of Keylight to fix a licensing bug.

Requirements

1. Shake 2.51 on Windows
2. Shake 3.01 or 3.50 on Mac OS X, Irix, Linux (RH9).
3. Foundry License Manager 3.1 or later. (Built with Licensing Library 3.0v6)

Release Date

September 2004

New Features

There are no new features.

Improvements

There are no improvements.

Bug Fixes

1. Licensing. Plug-ins would not release a floating license immediately after quitting Shake, but would wait until the server timeout. This has been fixed so that licenses are given back to the server on quitting Shake.

Known Bugs and Workarounds

There are no known bugs.

Keylight 2.0v1

Requirements

Shake 2.51 on Windows, Shake 3.01 or Shake 3.50 on Mac OS X, Irix or Linux.

Release Date

July 2004

New Features

1. Floating point support.
2. Builtin matte processing tools.
3. Improved Keylight algorithm.

Improvements

This section will describe improvements to existing features in later versions.

Bug Fixes

This section will describe fixed bugs in later versions.

Known Bugs and Workarounds

There are no known bugs.

Appendix B

End User License Agreement

The copyright in this Keylight software ("the Software") and its associated documentation is owned by The Foundry Visionmongers Limited ("the Owners").

This is a legally binding agreement between you and the Owners which will govern your use of the Software, you agree with the Owners to be bound by all of its terms and conditions. Please read this document carefully before opening this package and using the Software. If you do not accept these terms you may within 14 days of purchase return the Software, its packaging and documentation unused and intact to your supplier together with proof of purchase for a full refund of any license fee you may have paid for this software.

License

The Owners hereby grant to you a personal, non-transferable (except as expressly provided in clause 2 below) and non-exclusive right to use this software on a temporary trial basis if no license fee has been paid. If a license fee has been paid, you may continue to use this software under these same conditions. Your rights are to:

- load and install the Software by copying it onto the hard disk drive or into the CPU memory of a single computer which is under your control for use thereon; and
- make up to 3 full or partial copies of the Software, but only as necessary for backup or archival purposes, provided that:
  - your use and possession of such copies shall be solely under the terms and conditions of this agreement, and
  - you shall place the same proprietary and copyright notices on all such copies as are included by the Owners on the medium containing the authorised original copy of the Software.

You may not under any circumstances:

- store, use or allow the use of the Software in any manner on more than one computer at any time;
- save to the extent and in the circumstances expressly required to be permitted by law, copy, modify, adapt, merge,

translate, reverse engineer, de compile, disassemble or create derivative works, based on the whole or any part of, the Software;

- except as expressly provided in clause 2 below, distribute, lease, rent, sub-license, transfer, assign, loan or otherwise convey the Software or any portion thereof to anyone.

To the extent that local law gives you the right to de compile the Software in order to obtain information necessary to render the Software interoperable with other computer programs, the Owners hereby undertake to make that information readily available to you. The Owners shall have the right to impose reasonable conditions such as a reasonable fee for doing so. In order to ensure that you receive the appropriate information, you must first give the Owners sufficient details of your objectives and the other software concerned. Requests for the appropriate information should be made to the software development manager at The Foundry.

Transfers

If you transfer ownership or otherwise dispose of any property upon which the Software is copied, you may transfer the Software and all licenses and rights in the Software granted to you under this agreement to the transferee provided that:

- the Owners agree to such a transfer in writing; and
- such transferee agrees in writing to accept the terms and conditions of this agreement; and
- you contemporaneously transfer all copies of the Software you have made to such transferee or destroy all copies not transferred; and
- any applicable transfer fee is paid to The Owners.

If any transferee does not accept such terms, this agreement shall automatically terminate and you shall not retain any rights under this agreement in respect of the transferred Software.

Except as expressly provided in this clause, you may not sublicense, transfer or assign this agreement or any of your rights or obligations under this agreement, in whole or in part.

Ownership

You own only the media (or authorised replacement) on which the Software was recorded. You may retain the media on termination of this agreement provided you have ensured that the Software contained on or stored in it has been completely erased or otherwise destroyed.

You acknowledge that the Owners shall at all times retain ownership of the Software as recorded on the original media and all subsequent copies of it. The Owners transfer no ownership interest in the Software, in the intellectual property in any Software or in any copy of it, to you under this agreement or otherwise, and the Owners reserve all rights not expressly granted to you hereunder.

Termination

If you fail to comply with any of your obligations under this agreement, this agreement, and all rights and licenses granted hereunder shall automatically terminate without prejudice to any other rights which the Owners may possess in respect of your default. Upon termination, you will promptly, and in any event within 30 days of any such termination, destroy all copies of the Software and its documentation including any Software stored on any computer under your control.

Disclaimer and Limitation of Liability

The Owners provide the Software to you on an "as is" basis. The express terms of this agreement are in lieu of all warranties, conditions, undertakings, terms and obligations implied by statute, common law, trade usage, course of dealing or otherwise (including without any limitation any warranty of merchantability or fitness for a particular purpose), all of which are hereby excluded to the fullest extent permitted by law. No oral or written information or advice given by the Owners employees, representatives, agents or distributors will create a warranty for the Software and you may not rely on any such information or advice.

The Owner does not warranty that the Software will meet your requirements or that the operation of the Software will be uninterrupted or error-free or contain no defects. You shall load and use the Software at your own risk and in no event will the Owners be liable to you for any loss or damage of any kind (except personal injury or death resulting directly from the Owners' negligence) including lost profits or other indirect or consequential loss arising from your use or possession of or inability to use the Software or from errors or deficiencies in it, however arising and whether caused by negligence or otherwise except as expressly provided herein. In no event shall the Owners' total liability arising out of this agreement or your use or possession of the Software exceed the amount paid by you for the Software.

Law

This agreement shall be governed by and construed in accordance with English Law.