



©2010 The Foundry Visionmongers Ltd. All rights reserved.

FurnaceCore User Guide

This manual, as well as the software described in it, is furnished under license and may only be used or copied in accordance with the terms of such license. This manual is provided for informational use only and is subject to change without notice. The Foundry assumes no responsibility or liability for any errors or inaccuracies that may appear in this book.

No part of this manual may be reproduced, stored in a retrieval system, or transmitted in any form without the prior written permission of The Foundry.

The Foundry logo is a trademark of The Foundry Visionmongers Ltd. Nuke is a registered trademark of The Foundry Visionmongers Ltd. All other products or brands are trademarks or registered trademarks of their respective companies or organisations.

Software engineering: Simon Robinson, Ben Kent, Ralph McEntagart, Lucy Wilkes, Phil Parsonage, Andy Whitmore, Bruno Nicoletti, Mailys Levassort, and Jack Binks.

Algorithms: Dr. Bill Collis, Dr. Anil Kokaram of Trinity College Dublin, Prof. Paul White of the University of Southampton, Ben Kent, Phil Parsonage, and Dr. Francois Pitie.

Product testing: Jack Binks, Dan Allum and Mark Titchener.

Writing and layout design: Eija Narvanen using FrameMaker.

Proof reading Eija Narvanen.

Contents

Introduction	About this User Guide	6
	OFX Plug-ins	6
	What's New?	6
	Notation	6
	Installing FurnaceCore	6
	Licensing FurnaceCore	7
	Applying FurnaceCore Plug-ins	8
	On-Screen Tools	8
	Colour Space in FurnaceCore Plug-ins	9
	Customer Support	9
	Other Foundry Products	9
Overview	Grain Management	11
	Retiming	12
	Removing Flicker and In-Scene Objects	12
	Stabilisation and Alignment	13
	Grading	13
Align	Effect Type	14
	Introduction	14
	Quick Start	15
	Inputs	16
	Parameters	17
DeFlicker2	Effect Type	19
	Introduction	19
	Quick Start	20
	Inputs	20
	Parameters	20
DeGrain	Effect Type	21
	Introduction	21
	Colour Space	22
	Quick Start	22
	Inputs	23
	Parameters	23

DeNoise	Effect Type	26
	Introduction	26
	Colour Space.....	26
	Quick Start	27
	Inputs	27
	Parameters	27
Kronos	Effect Type	30
	Introduction	30
	Quick Start	31
	Inputs	33
	Parameters	34
MatchGrade	Effect Type	38
	Introduction	38
	Quick Start	38
	Inputs	39
	Parameters	40
MotionBlur	Effect Type	41
	Introduction	41
	Quick Start	41
	Inputs	43
	Parameters	43
ReGrain	Effect Type	45
	Introduction	45
	Colour Space.....	46
	Quick Start	46
	Response.....	48
	Checking the Result	48
	Proxy Resolutions.....	49
	Inputs	50
	Parameters	50
RigRemoval	Effect Type	53
	Introduction	53
	Quick Start	54
	Inputs	57

	Parameters	57
Steadiness	Effect Type	60
	Introduction	60
	Quick Start	61
	Inputs	62
	Parameters	62
Global Motion Estimation	Introduction	65
	What is Global Motion Estimation?	65
	The Analysing Global Motion Estimation Effects	66
	Controls	67
	Widgets	69
Local Motion Estimation	Introduction	70
	Background	70
	Parameters	70
Appendix A	Release Notes	73
Appendix B	Third Party Licenses	74
Appendix C	End User License Agreement	77
Index	A-Z	83

INTRODUCTION

Welcome to this User Guide for FurnaceCore on Avid DS.

About this User Guide

This User Guide will tell you how to install, license, and use the FurnaceCore plug-ins. Each plug-in is described in detail in later chapters.

Throughout this user guide, we assume you are familiar with the version of Avid DS you are running.

Note *For the most up-to-date information, please see the FurnaceCore product page and the latest FurnaceCore user guide on our web site at <http://www.thefoundry.co.uk>.*

OFX Plug-ins

These plug-ins have been written and compiled to the OFX plug-in standard. OFX is an open plug-in API for 2D visual effects. For a current list of supported host systems see <http://www.thefoundry.co.uk>.

What's New?

Have a look at the new features and improvements in Appendix A.

Notation

In this user guide, we will refer to machines running FurnaceCore and Avid DS as clients and machines that are running the Foundry FLEXIm Tools as servers.

Installing FurnaceCore

FurnaceCore is available as a download from our web site, <http://www.thefoundry.co.uk>. The downloads are in executable installer format for Windows machines. FurnaceCore should be installed on the client machines.

FurnaceCore on Windows

Follow these instructions to install FurnaceCore on a Windows machine running Avid DS.

1. Download one of the following files from our web site (www.thefoundry.co.uk):
Furnace_Core_4.3v1_Avid DS-win-x86-release-32.exe
Furnace_Core_4.3v1_Avid DS-win-x86-release-64.exe

Which file you should download depends on whether you are using a 32- or a 64-bit version of Avid DS.

2. Double-click on the exe file to launch the installer. Follow the on-screen instructions to install the plug-ins.
3. Proceed to "Licensing FurnaceCore" on page 7.

Default Install Directory

The FurnaceCore plug-ins are divided into three groups of Avid DS presets: Image Effects, Time Effects, and Tree Effects. By default, these presets are installed to one of the following directories:

```
C:\Program Files\Avid\DS_[version]\DSPresets\FurnaceCore or  
C:\Program Files (x86)\Avid\DS_[version]\DSPresets\FurnaceCore
```

The binaries are also installed to the OFX plug-ins directory. The default directories Avid DS searches for OFX plug-ins are as follows:

```
C:\Program Files\Common Files\OFX\ or  
C:\Program Files (x86)\Common Files\OFX\
```

Moving the OFX Plug-ins Directory

You can put the OFX plug-ins in any directory, as long as you set the environment variable OFX_PLUGIN_PATH to point to the directory they're in.

Licensing FurnaceCore

If you don't have a valid license key, a warning will be displayed and the finished render will have coloured dots scattered over it.

The license key is a sequence of numbers and letters, stored in a plain text file, that unlocks FurnaceCore. License keys can be created for a particular computer enabling those plug-ins to run only on that computer. These are called node locked licenses. We also supply floating licenses that will unlock FurnaceCore on any networked computer connected to a machine running the Foundry license server.

Tools to install license keys, manage floating licenses, and diagnose license problems can be downloaded from our web site, <http://www.thefoundry.co.uk/licensing>.

Applying FurnaceCore Plug-ins

To load FurnaceCore plug-ins:

1. Switch to the Compositing layout.
2. Right-click on the Effects Tree and select **Add Effect**.
3. Navigate to the **Image Effects** folder, select **OFX Host.Preset**, and click **OK**.
The OFX Host.preset gives you access to all the FurnaceCore plug-ins.
4. To select the plug-in you want to apply, open the preset's properties and use the **Effect** menu.

OR

1. Right-click on a clip in the timeline and select **Add Clip Effect**.
2. Navigate to the **Image Effects** folder, select **OFX Host.Preset**, and click **OK**.
The OFX Host.preset gives you access to all the FurnaceCore plug-ins.
3. To select the plug-in you want to apply, use the **Effect** menu in the preset's properties.

Note *F_Align and F_MatchGrade are multi-input effects and can only be applied when using the Compositing Layout.*

Note *F_Kronos should be applied as a Timewarp Container. For more information, see "Timewarp Container" on page 31.*

The FurnaceCore plug-ins are also divided into three groups of Avid DS presets: Image Effects, Time Effects, and Tree Effects. If you want to apply a plug-in using these presets (rather than the OFX Host.preset described above), do one of the following:

- To apply an Image Effect, use the Browser to navigate to **\DSPresets\FurnaceCore\Image Effects** and drag a preset onto a clip in the Timeline.
- To apply the only Time Effect, F_Kronos, see "Timewarp Container" on page 31.
- To apply a Tree Effect, switch to the Compositing Layout, use the Browser to navigate to **\DSPresets\FurnaceCore\Tree Effects**, and drag a preset onto the Effects Tree.

On-Screen Tools

Some plug-ins have their own on-screen tools for region selection or to facilitate the changing of various parameters.

All on-screen tools are transparent when not active. Parts of the widget that can be moved become solid when the mouse is moved near to them. When selected, they are extended out to the edges of the viewing area.

The rectangular tools used to select a region – for sampling or analysis, for example – sometimes apply to a single frame only. In this case, the on-

screen tool will have continuous lines on this frame, and dotted lines on all other frames. When a region is selected, the frame associated with it will be automatically updated to the current frame.

Colour Space in FurnaceCore Plug-ins

Some of the algorithms in the FurnaceCore tool set are sensitive to the colour space of the source footage. If the footage is not in the expected colour space, you may get poor results from some of the plug-ins.

FurnaceCore on Avid DS expects all footage to be in **sRGB** space.

Avid DS converts footage to sRGB, so your footage should be sRGB by the time it reaches the plug-in anyway.

Plug-ins most sensitive to colour space are:

- **F_DeGrain**
- **F_DeNoise**
- **F_MatchGrade**
- **F_ReGrain**

F_DeGrain, **F_ReGrain**, and **F_DeNoise** also have additional pop-up menus to allow you to adjust the algorithm to get the best results when using footage that was originally in another colour space. Please take a moment to read through the relevant chapters in this manual for these plug-ins, to make sure you are aware of how they behave.

Customer Support

Should questions arise that this user guide fails to address, you can contact Customer Support directly via e-mail at support@thefoundry.co.uk or via telephone to our London office on +44 (0)20 7434 0449 or to our Los Angeles office on (310) 399 4555 during office hours.

Other Foundry Products

The Foundry is a leading developer of visual effects and image processing technologies for film and video post production. Its products include Nuke, Mari, Furnace, FurnaceCore, Keylight, Ocula, and RollingShutter and run on a variety of compositing platforms including Adobe After Effects, Avid DS, and Apple Final Cut Pro. For the full list of products and supported platforms see our web site <http://www.thefoundry.co.uk>.

Nuke is an Academy Award® winning compositor. It has been used to create extraordinary images on scores of feature films including *Avatar*,

District 9, The Dark Knight, Iron Man, Quantum of Solace, The Curious Case of Benjamin Button, Transformers, and Pirates of the Caribbean: At World's End.

Mari is a creative texture-painting tool that can handle extremely complex or texture heavy projects. It was developed at Weta Digital and has been used on films such as District 9, The Day the Earth Stood Still, The Lovely Bones, and Avatar.

Keylight is an industry-proven 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.

Ocula is a collection of tools that solve common problems with stereoscopic imagery, improve productivity in post production, and ultimately help to deliver a more rewarding 3D-stereo viewing experience.

RollingShutter is a plug-in that tackles image-distortion problems often experienced by users of CMOS cameras. The plug-in will often vastly improve the look of distorted footage, by either minimising or eradicating image distortions. Unlike solutions tied to camera stabilisation, that stretch the image as a whole, the RollingShutter plug-in compensates for local skewing and distortion in the scene, by correcting each object individually.

Visit The Foundry's web site at <http://www.thefoundry.co.uk> for further details.

OVERVIEW

Great – so you’ve just downloaded, installed, and licensed up your copy of FurnaceCore, but where do you start? In this section, we’ll have a quick look at the plug-ins, broken down by what we would envision their application to be. Please bear in mind, many of the tools offer much greater flexibility than we can cover in this manual, so have a good play around with them.

Please select an overview from the list of sections below. When you see an individual plug-in that you’d like to find out a little more about, click on it and you will be taken to the plug-in reference section for that tool. [Grain Management](#) | [Retiming](#) | [Removing Flicker and In-Scene Objects](#) | [Stabilisation and Alignment](#) | [Grading](#).

Grain Management

The FurnaceCore grain management tools allow you to quickly remove and replicate grain patterns from both modern day fine-grained film stock and old, degraded, archive material. So where (and why) would you want to manage grain? In a whole host of different places; for example:

- When comping (or even editing) two shots together it is very obvious to the viewer when the grain characteristics vary; take, for example, comping a moving Computer Generated (CG) element against a still frame: the background still has static, unmoving grain, and the CG element will have no grain at all. In this circumstance, you’d replicate the grain from the still frame and generate further frames so that the grain appears to be in motion. This would then be applied back against both the original still and the CG element, helping create the illusion that it was all shot in camera.
- When colour grading you’ll find that as you begin to push the processing, the grain will be the first part of the image that clips and otherwise introduces undesirable artefacts. By removing it beforehand, then replicating it on the finished shot you’ll save yourself a great deal of trouble.

DeGrain is a fast (spatial) grain suppression tool, useful for quickly cutting levels of grain on modern film stocks. If you want to remove grain from a single frame, this is the tool you need.

DeNoise is a temporal grain and noise removal tool, good for everything from modern stock through to archival footage and video noise. If you want to remove grain from a sequence of frames, this should be the tool of your choice.

ReGrain allows you to reapply grain to a source clip. A number of preset stocks are included, or you can sample a plate of your choice and have ReGrain create a statistically identical moving grain sample.

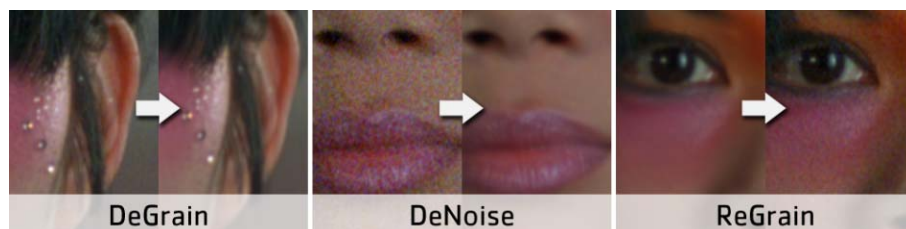


Figure 1. Grain management tools.

Retiming

Retiming is the process of altering the speed of a clip, to get either slow or fast motion. Historically, this was an ugly process which introduced juddery or artefact-ridden output. By making a best-guess about where objects move within a scene (motion estimation), we are able to create frames between existing ones. Once we know how these objects move, we are also able to add motion blur to these objects in motion as though it was shot in-camera.

Kronos is FurnaceCore's retimer. It gives you full control over the speed of playback of your clip and allows you to add motion blur.

MotionBlur allows you to add true-to-life motion blur to objects within a scene, without the added complexity of the clip retiming controls.



Figure 2. Retiming tools.

Removing Flicker and In-Scene Objects

DeFlicker2 deals with film flicker from a variety of sources. It works particularly well on in-scene flicker, poorly synchronised light rigs, and stray light.

RigRemoval Got an object moving in relation to the background (or vice

versa), that you want to get rid of? RigRemoval scans forwards and backwards within a sequence to find an area where the background in question was unobstructed by the object, and then copies that back into place. Great for unwanted traffic, people, and more.



Figure 3. .RigRemoval.

Stabilisation and Alignment

So you've got anything from a wobbly shot to a repair you want to stick over the defect? These tools are the answer.

Steadiness Suppress wobbles in handheld footage or lock the sequence position against a certain frame (great for when a small shake hits an otherwise locked off shot), Steadiness requires no tracking markers to be specified.

Align Lock one shot of a scene against another. Handy for anything from doubling up the crowd size by comping together two shots, to locking your freshly generated clean plate to the original.

Grading

Help speed up the grading process using this image-enhancing tool.

MatchGrade Got two shots from different times of day that need to be comped or edited together? This plug-in analyses the colour histogram of a reference image and automatically applies the result to your source sequence. This allows sequences that were shot with subtly different lighting to have the same 'look'.

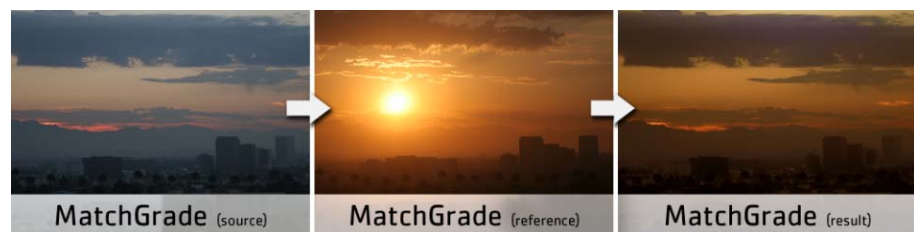


Figure 4. F_MatchGrade.

ALIGN

This chapter looks at how to use F_Align to register (line up) two shots that are of the same scene, but have slightly different camera motion and foreground objects. This can be useful, for example, for doubling up the crowd size by lining up and comping together two shots of the same scene, or locking your freshly generated clean plate to the original.

Effect Type

F_Align is a tree effect.

Introduction

F_Align takes two sequences that were shot of the same scene and lines them up spatially. It uses Global Motion Estimation (GME) to calculate a four-corner pin so that each frame in one shot (the source input) will be aligned with the corresponding frame in a second shot (the reference input). The result is the source image which has been repositioned to line up with the reference image.

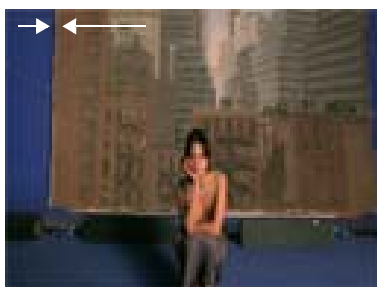


Figure 5. Source image. Notice the position of the background.

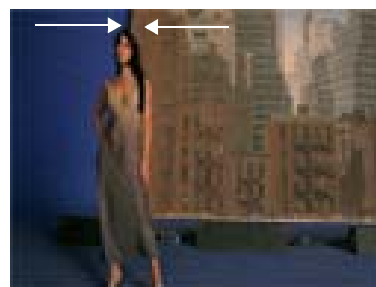


Figure 6. Reference image.

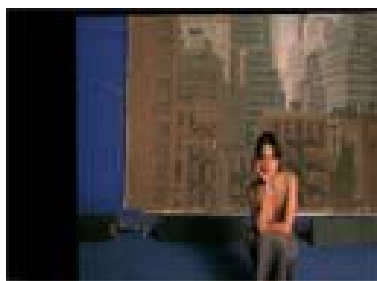


Figure 7. Output from F_Align. The source image has been repositioned so that the background lines up with the reference image.

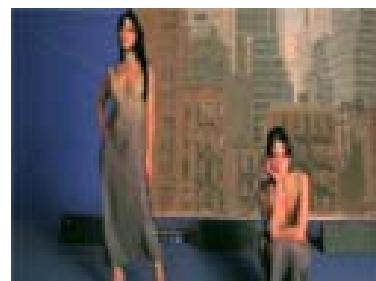


Figure 8. The output of F_align comped together with the reference image.

To be able to align the sequences, F_Align analyses them for global motion. This analysis can be triggered for the complete sequence, specified frame range, or a single frame when you press the **Analyse** button in the F_Align controls. Alternatively, it can be done on the fly for a single frame when you move to a new frame on the timeline. The advantage of pressing **Analyse** is that during the analysis, F_Align stores the calculated four-corner pin as key-framed parameters. When you then render the output of the plug-in later, F_Align can use these key frames without having to calculate them again.

If you analyse on the fly, you won't have direct access to the calculated corner pins. Any re-rendering will also be significantly slower, as the 'on the fly' calculations will have been lost and F_Align will have to analyse again.

If at any stage you modify the effect in such a way to invalidate the key-framed analysis (for example by changing the **Accuracy** parameter), a warning will be posted and the effect will analyse on-the-fly during render, ignoring the keyed analysis.

The on-screen widget and the **Analysis Region** parameters are used to control which section of the **Reference** frame is being matched to each **Source** frame. Typically, leaving the region at its default is good enough. However, a heavy mismatch in foreground detail may make it necessary to change the region to a section that is shared between shots.

For more of an overview of Global Motion Effects, and a description of the common way of working many of these effects have, please see the Global Motion Estimation chapter on page 65.

Quick Start

This section gives a very brief outline of how to use the plug-in. It covers both analysing using the Analyse button and analysing on the fly.

Analysing with the Analyse button

To align two shots and store the results as key-framed parameters, do the following:

1. Find two shots that are of the same scene, but have slightly different camera motion and foreground objects. Load both these shots.
2. Apply **F_Align**. See "Applying FurnaceCore Plug-ins" on page 8. Note that to apply a multi-input effect, such as F_Align, you first need to switch to the Compositing layout.
3. Connect the shot you want to reposition to the **Source** input of F_Align and the shot you want to match to the **Reference** input. View the output from F_Align.

The source will be immediately repositioned so that it aligns to the reference shot without any need for analysis.

You will see the following banner in the overlay:

Note: no valid key framed analysis found. Analysing during render.

4. Depending on the exact difference between the two shots, you may need to enable the **Scale** and/or the **Perspective** toggles to get a decent alignment.
5. You may also need to reposition the **Analysis Region** depending on the differences in foreground detail. However, leaving it at the default position works well for most shots.
6. Click on the **Analyse** button.

F_Align will now start analysing each frame in the shot, figuring out the four-corner pin and writing it as key frames to the corner pin parameters, **Bottom Left**, **Bottom Right**, **Top Left**, and **Top Right**. You will find these parameters under **Advanced > Four Corner Pin**.

If you interrupt the analysis, the pins it has keyed until that point will be retained.

7. Play or scrub through the aligned frames. The rendering will be faster as F_Align will no longer need to analyse on the fly. However, if you scrub to a frame where a corner pin has not been keyed, F_Align will re-analyse that frame on the fly.

Analysing On The Fly

To align two shots and calculate the alignment on the fly, do the following:

1. Find two shots that are of the same scene, but have slightly different camera motion and foreground objects. Load both these shots.
2. Apply **F_Align**. See "Applying FurnaceCore Plug-ins" on page 8. Note that to apply a multi-input effect, such as F_Align, you first need to switch to the Compositing layout.
3. Connect one of the shots to the **Source** input of F_Align and the other to the **Reference** input. View the output from F_Align.

The **Source** will be immediately repositioned so that it aligns to the **Reference** shot without any need for analysis.

You will see the following banner in the overlay:

Note: no valid key framed analysis found. Analysing during render.

4. Depending on the exact difference between the two shots, you may need to enable the **Scale** and/or the **Perspective** toggles to get a decent alignment.
5. You may also need to reposition the **Analysis Region** depending on the differences in foreground detail. However, leaving it at the default position works well for most shots.

Inputs

F_Align has two input clips: the **Source** input is moved so that each frame matches the **Reference** input frame.

Parameters

The parameters for this plug-in are described below.

Analyse - This is a push button which will trigger an analysis of the input clips and calculate a corner pin. Interrupting the analysis will not delete the corner pin keys that have already been calculated.

Clear Analysis - Pressing this push button will delete all key frames from the corner pin parameters, allowing you to force a re-analysis if you feel the need to.

Analysis Range - This controls the range of frames any analysis will be run over. It can be one of:

- **Specified Range** - which will look at the parameters **Analysis Start** and **Analysis Stop** for the range of frames to analyse,
- **Source Clip Range** - which will automatically determine the range of frames to analyse from the length of the input clip.
- **Current Frame** - the analysis occurs only on the current frame. This is useful for correcting any errors that may have occurred while analysing the entire clip.

Analysis Start - The first frame to analyse from if **Analysis Range** is set to **Specified Range**.

Analysis Stop - The last frame to analyse from if **Analysis Range** is set to **Specified Range**.

Scale - A toggle that indicates whether the calculated corner pin can include a scaling factor.

Rotate - A toggle that indicates whether the calculated corner pin can include rotations.

Translate - A toggle that indicates whether the calculated corner pin can include translations in x and y.

Perspective - A toggle that indicates whether the calculated corner pin can include perspective transforms.

Analysis Region - This is the region analysed to calculate the four-corner pin. This is especially useful when doing any form of frame locking, in which case, go to the lock frame, look at the reference clip and position the box over the area you want locked.

Analysis Region BL - controls the position of the bottom left corner of the analysis region.

Analysis Region TR - controls the position of the top right corner of the analysis region.

Advanced - The lesser used refinement controls.

Accuracy - This controls the time/accuracy trade off. The higher this is, the slower the analysis, but you have a better likelihood of a good result.

Filtering - Sets the filtering quality.

- **Low** - low quality but quick to render.
- **Medium** - uses a bilinear filter. This gives good results and is quicker to render than high filtering.
- **High** - uses a sinc filter to interpolate pixels giving a sharper repair. This gives the best results but takes longer to process.

Invert - if set, then the inverse of the calculated four-corner pin is used during render.

Four Corner Pin - The corner pins calculated during the analysis pass:

Bottom Left - the lower left corner pin.

Bottom Right - the lower right corner pin.

Top Left - the upper left corner pin.

Top Right - the upper right corner pin.

About - Click this to display the online help, FurnaceCore version number, and the location of the user guide.

DEFlicker2

When working in film, you sometimes have to deal with shots that have a luminance flicker. This chapter concentrates on removing flicker using F_DeFlicker2.

Effect Type

F_DeFlicker2 is an image effect.

Introduction

F_DeFlicker2 is used to remove flicker. It is particularly suitable for removing flicker that is localised and dependent on the geometry of the scene (that is, flicker that is not present across the whole of the image), such as that caused by an unsynchronised fluorescent light in a shot. It works by calculating the gain between the current frame and each frame in a small analysis range surrounding it. It then tries to adjust the gain so that it varies smoothly over this frame range. This means it is better at reducing fast flicker than flicker which varies slowly over the image sequence, as the latter will already appear smooth over the small frame range and F_DeFlicker2 will leave it largely untouched.

The algorithm used by F_DeFlicker2 can introduce blurring in areas where there is rapid motion. This problem could be alleviated by using local motion estimation before deflickering the frames. However, this process is complicated by the fact that the presence of flicker can adversely affect the results of the motion estimation. F_DeFlicker2 therefore adopts a two-stage approach to this problem. First, the normal deflickering process is performed. Then, the motion vectors for the sequence are calculated on the resulting deflickered frames, and applied to the original frames in order to align them. The deflicker calculation is then performed on the aligned frames to give the final result. To use this approach, turn on **Use Motion** in F_DeFlicker2.

Note that because F_DeFlicker2 looks at input frames outside the current frame when performing its calculation, it can be a computationally expensive plug-in. As such, using more than two instances of F_DeFlicker2 in a node tree will dramatically increase render times. It is strongly advised therefore, that you render each instance out separately.

Quick Start

To remove flicker from a sequence:

1. Load the sequence you want to remove flicker from.
2. Apply DeFlicker2. See “Applying FurnaceCore Plug-ins” on page 8.
3. View the output from DeFlicker2.
4. If you’re not happy with the results, adjust the DeFlicker2 parameters. The available parameters are described below.
5. Render the result.

Inputs

F_DeFlicker2 has a single input: the sequence to deflicker.

Parameters

The parameters for this plug-in are described below.

DeFlicker Amount - Use this to reduce flicker without removing it entirely; smaller values mean more will be left behind.

Block Size - To find where a certain pixel is located in the analysis range, the deflicker algorithm looks for a block of pixels centered around that pixel. Block size defines the width and height of these blocks (in pixels). On rare occasions, a large block size can produce data that’s lacking in detail. This is because a small feature can fit into a large block, causing the motion estimation to concentrate on the background motion and ignore the small feature. A small value, instead, can produce a noisy motion field, as there aren’t enough constraints in a small block to fit the motion accurately. In most cases, however, the default value is small enough so that details aren’t lost, and the smoothing step of the algorithm ensures the motion field isn’t too noisy. Therefore, this value very rarely needs editing.

Use Motion - Turn this on to do a second deflicker pass using motion-compensated frames. This can improve results in areas where there is fast motion, where the initial deflicker pass can introduce blurring.

Vector Detail - Determines the density of the motion vectors used when **Use Motion** is turned on. The maximum value of 1 will generate one vector per pixel. This will produce the most accurate vectors but will take longer to render. A value of 0.5 will generate a vector at every other pixel.

Analysis Range - The number of frames searched each side of the current frame when calculating the flicker. Higher values may give better results, but can also bring in erroneous information and take longer to process.

About - Click this to display the online help, FurnaceCore version number, and the location of the user guide.

DEGRAIN

This chapter looks at removing noise or grain using F_DeGrain. As a general guideline, if you have just *a single frame* to work on, we would recommend using F_DeGrain to achieve the best results. However, if you are looking to remove noise or grain from *a sequence of images*, better results are likely to be obtained by using the Furnace plug-in F_DeNoise, which is a fully motion-compensated noise reducer (described on page 26).

Effect Type

F_DeGrain is an image effect.

Introduction

F_DeGrain is used to remove grain from a frame. The aim is to remove as much grain as possible whilst doing as little damage to the image as possible.

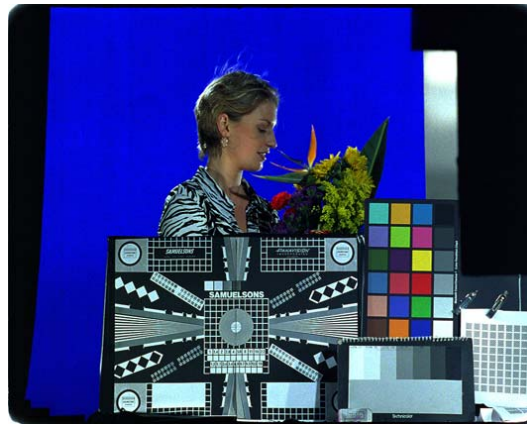


Figure 9. Test grain image.

F_DeGrain's spatial filtering involves averaging pixels within the same frame. This can lead to the blurring of the image, and so, to keep this to a minimum, a wavelet based technique is used. This decomposes the image into a number of different frequencies and scales before attempting to remove the grain. The high frequency spectrum can be isolated and processed using the **Process Small Frequencies** and **Small Amount** parameters. The low frequencies are processed using the **Process Large Frequencies** and **Large Amount** parameter. This allows you to select the frequencies that contain grain, process them by as much as you think is best, and leave details in the other frequencies untouched.

Colour Space

F_DeGrain works best when the colour space of the footage is the same as the 'colour space the grain exists in'. An example might make more sense here.

If you have a film scan, it's most likely to be in log (or Cineon) colour space, as it represents the print density of the film. When you import this into Avid DS, Avid DS converts it into sRGB space. Because the noise in the plate is from the film itself, this conversion exaggerates the noise in the highlights.

Because the F_DeGrain plug-in receives this sRGB data from Avid DS, you need to use the **Original Colour Space** menu in the F_DeGrain controls to tell the plug-in what colour space the image was originally in. In the case of our film scan, this would be **Cineon**. If you had footage from a digital video camera, this would most likely be **sRGB**.

If this isn't set correctly, you'll probably see that F_DeGrain doesn't remove grain very well in certain parts of the image as it's trying to remove the exaggerated grain.

Quick Start

To remove grain from an image, do the following:

1. Load the single-frame image you wish to remove the grain from.
2. Add an F_DeGrain effect after your image. See "Applying FurnaceCore Plug-ins" on page 8.
3. View the output from F_DeGrain.
4. Position the selection box over a plain area of the image (Figure 11).

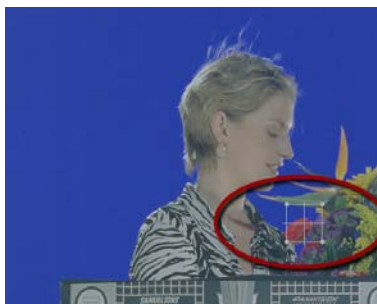


Figure 10. Bad sample position.

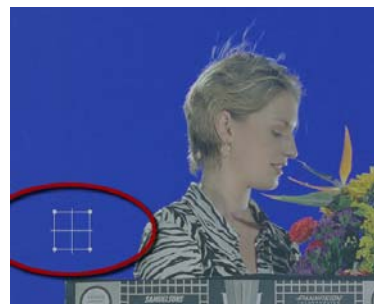


Figure 11. Good sample position.

Note *It is very important to position the selection box over a region with little image detail. Failure to do this will give poor results as the algorithm will think the image detail is grain and remove it.*

The sample selection automatically updates not only the **Analysis Region BL** and **Analysis Region TR** parameters in the **Analysis Region** group but the frame from

which the sample is taken. Whenever the sample rectangle is altered, the internal analysis of the grain in that region reoccurs. The sample analysis data is saved into hidden data for distribution across render farms.

Fine Tuning

If either not enough grain has been removed or the picture has been softened by removing too much grain, it will be necessary to fine tune the parameters:

1. Increase **DeGrain Amount** to remove more grain, reduce it to remove less. This is a fairly crude way of setting the parameters. Below, there are more advanced controls.
2. The easiest way to find the optimal setting for F_DeGrain is to look at what is removed from the image. To do this, change the **Output** parameter to **Grain**. This will display the grain that is being subtracted from the original image. If necessary, increase **Exaggerate Grain** to make it more obvious.
Only grain should be visible in this image. If you can see a lot of picture detail, it means the degrainer is working too hard and removing too much of the image, which will lead to a soft result.
3. The degrainer works by decomposing the image into four levels - **Small**, **Medium**, **Large** and **Huge**. To set the first level, make sure **Process Huge Frequencies** is the only **Process** control checked, and adjust **Huge Amount** until the image is only just visible. Then, repeat for **Large Amount**, **Medium Amount**, and **Small Amount**.
4. Often, the blue channel will contain more grain than the red and green. This can be checked by viewing the individual colour channels. If this is the case, increase **Blue Amount** until enough grain is being removed.
5. When you are happy with the settings, make sure you have checked **Process Small Frequencies**, **Process Medium Frequencies**, **Process Large Frequencies**, and **Process Huge Frequencies**.

Inputs

F_DeGrain has one input: the image to be degraded.

Parameters

The parameters for this plug-in are described below.

Original Colour Space - This tells F_DeGrain what colour space the footage was where the grain originated. It makes sure that the grain isn't exaggerated by any colour space conversions prior to processing.

DeGrain Amount - This is the coarse adjustment control. Increasing tune will remove more grain and decreasing it will leave more in.

Output - Whether to output the degrained image or the grain that was removed.

- **Result** - output the result of degrading the source.
- **Grain** - output the grain that was removed from the source.

Exaggerate Grain - When displaying the removed grain, increase this parameter to make it more visible.

Analyse - This is a push button which will trigger an analysis of the input clip. Press this button if the input clip from which the grain was analysed has changed but you do not want to move the analysis region to trigger re-analysis. Whenever the input clip changes, you will see the following warning in the Viewer:

The clip from which the grain was analysed has changed. Press Analyse or move the analysis region to re-analyse grain.

Analysis Region - The selection box that marks the region of the image used to analyse the grain and to set the internal degrading parameters automatically. It is important that this part of the frame contains no image detail, only grain.

Analysis Region BL - controls the position of the bottom left corner of the sample region.

Analysis Region TR - controls the position of the top right corner of the sample region.

Analysis Frame - sets the frame from which the sample rectangle should be taken.

Advanced - The lesser used refinement controls.

Frequencies - Sets which of the frequencies to process and the amount of grain removed. In other words, you can remove and process only the large grain leaving the others untouched. However, normally you would remove grain throughout the frequency spectrum.

Process Small Frequencies - toggle this on if you want the small scale grain removed.

Small Amount - increases or decreases the amount of small grain removed.

Process Medium Frequencies - toggle this on if you want the medium scale grain removed.

Medium Amount - increases or decreases the amount of medium grain removed.

Process Large Frequencies - toggle this on if you want the large scale grain removed.

Large Amount - increases or decreases the amount of large grain removed.

Process Huge Frequencies - toggle this on if you want the largest scale grain removed.

Huge Amount - increases or decreases the amount of huge grain removed.

Channels - Sets the amount of grain removed from the individual colour channels.

Red Amount - increases or decreases the amount of grain removed in the red channel.

Green Amount - increases or decreases the amount of grain removed in the green channel.

Blue Amount - increases or decreases the amount of grain removed in the blue channel.

About - Click this to display the online help, FurnaceCore version number, and the location of the user guide.

DENOISE

This chapter looks at removing noise or grain using F_DeNoise. As F_DeNoise is a fully motion compensated noise reducer, it is very good at removing noise or grain *from a sequence*. If you just have *a single frame*, you should use the FurnaceCore plug-in F_DeGrain (described on page 21). This is a powerful wavelet-based spatial noise reducer and is likely to give better results on single images.

Effect Type

F_DeNoise is both an image effect and a tree effect.

Introduction

F_DeNoise is designed to remove noise or grain from a clip. Assuming there is no motion in a sequence, the best way to reduce the noise is to take an average across a number of frames (temporal averaging). The noise which is different on each frame will be reduced and the picture which is the same will be reinforced. Temporal averaging is far superior to averaging pixels from within the same frame (spatial averaging) as it doesn't soften the image. Unfortunately, if there is motion in the sequence, the averaged image will be blurred as the image appears at different locations in each frame. However, by estimating the motion in the sequence using The Foundry's advanced motion estimation technology, it is possible to compensate for any motion and average frames temporally without introducing any blurring artefacts.

Colour Space

F_DeNoise works best when processing footage in the same colour space that the noise naturally exists in. Converting footage between colour spaces can distort the profile of the noise across the image, and make removing noise more difficult.

Because Avid DS coverts footage to sRGB internally, the F_DeNoise plug-in provides the **Original Colour Space** pop-up menu to allow you to tell the plug-in which colour space the noise originally exists in. This should generally be set to the original colour space of the footage. In some cases, it may be worth experimenting with this to see which gives you the best results.

Quick Start

To remove noise from a sequence, do the following:

1. Load the footage to be noise reduced.
2. Apply F_DeNoise. See “Applying FurnaceCore Plug-ins” on page 8.
3. View the output.
4. Select the **Plate Size** of your input frame. Note that this refers to the original size of the plate so even if you are working on a cropped part of a 2k plate, **Plate Size** should still be set to 2k.
5. F_DeNoise works by analysing the grain structure inside the sample region. To set the sample region, move the on-screen sample box over a plain area of the image. To get a good result, it is important that this area is free from image detail, so no textures or edges.
6. The output should now show the denoised frame. If you are not satisfied with the results, try moving the sample box to a different, flat area of the image, or analysing on a different frame.

F_DeNoise will reanalyse the grain structure every time the **Plate Size** parameter is changed or the **Analysis Centre** is repositioned.

7. To remove more noise, simply increase the **DeNoise Amount** parameter. You can also remove different amounts of noise from the red, green, and blue channels by altering the **Red Amount**, **Green Amount**, and **Blue Amount** parameters in the **Advanced** parameter group.
8. If ringing artefacts appear around edges, check **Supress Ringing**. This way, the plug-in will do two separate denoise passes over the image. The second pass is designed to give good results in the regions where spurious rings tend to appear in the first pass, so the two resulting images can then be recombined in such a way as to reduce these artefacts.
9. To view the removed noise, set the **Output** parameter to **Noise**. The output from this will be the noise that has been subtracted from the original image by F_DeNoise. If necessary, increase **Exaggerate Noise** to make the noise easier to see.

Inputs

F_DeNoise has two inputs:

- The **Source** clip from which to remove the noise.
- An optional **Noise** input. When a **Noise** clip is supplied, the noise will be analysed in this clip, rather than the **Source** clip. The **Noise** clip should have similar noise characteristics to the **Source** clip and should be used when your **Source** clip does not contain a suitable flat region on which to do the analysis.

Parameters

The parameters for this plug-in are described below.

Original Colour Space - This tells F_DeNoise what colour space the footage

was where the noise originated. It makes sure that the noise isn't exaggerated by any colour space conversions prior to processing.

DeNoise Amount - This adjusts the overall amount of noise or grain that is removed. Increase this value to remove more noise.

Plate Size - The algorithm automatically sets some parameters depending on the expected size of the noise and grain which can be related to the size of the image. As you may be processing a cropped region, we do not necessarily know this from the image size. Select PAL Or NTSC, 1K, 2K, or 4K depending on the *original* size of the scan.

Suppress Ringing - Switch this on to remove the ringing that can be introduced near to edges in the denoised image.

Output - Whether to output the denoised image or the noise that was removed.

- **Result** - output the denoised source image.
- **Noise** - output the noise that was removed from the source image.

Exaggerate Noise - If you have chosen to output the noise, increase this parameter to make it more obvious.

Analyse - This is a push button which will trigger an analysis of the input clip. Press this button if the input clip from which the noise was analysed has changed but you do not want to move the analysis region to trigger re-analysis. Whenever the input clip changes, you will see the following warning in the Viewer:

The clip from which the noise was analysed has changed. Press Analyse or move the analysis region to re-analyse noise.

Analysis Region - These parameters allow you to change the region used to analyse the grain, in order to improve the noise reduction.

Analysis Centre - The position of the centre of the analysis region.

Analysis Frame - The frame to analyse on.

Advanced - These parameters allow you to remove different amounts of noise in each of the colour channels.

Red Amount - increases or decreases the amount of noise removed in the red channel.

Green Amount - increases or decreases the amount of noise removed in the green channel.

Blue Amount - increases or decreases the amount of noise removed in the blue channel.

About - Click this to display the online help, FurnaceCore version number, and the location of the user guide.

KRONOS

This chapter looks at retiming sequences using F_Kronos.

Effect Type

F_Kronos is a time effect.

Introduction

F_Kronos is FurnaceCore's re-timer and is designed to slow down or speed up footage. It works by calculating the motion in the sequence in order to generate motion vectors. These motion vectors describe how each pixel moves from frame to frame. With accurate motion vectors, it is possible to generate an output image at any point in time throughout the sequence by interpolating along the direction of the motion.



Figure 12. Simple mix of two frames to achieve an inbetween frame.



Figure 13. Kronos vector interpolation of the same two frames.

By default, F_Kronos is set to perform a half speed slow down. This is achieved by generating a new frame at position 0.25 and 0.75 between the original frames at 0 and 1. Frames are created at a quarter and three quarters instead of zero (an original frame) and a half so as not to include any original frames in the re-timed sequence. This avoids the pulsing that would otherwise be seen on every other frame on a half speed slowdown.

When F_Kronos is used to speed up motion, motion blur will be seen.

F_Kronos contains a number of controls to allow you to trade off render time versus accuracy of vectors. For more information on vectors and local motion estimation in general, "Local Motion Estimation" on page 70.

Quick Start

Kronos can be applied as a Clip Effect but this is not recommended because trimming the start of the clip will not output the correct result.

The best way to use Kronos is as a Timewarp Container. We'll describe how to set this up and then how to use it to retime clips.

Timewarp Container

You can create a TimeWarp Container and apply Kronos in the following two ways.

Option 1

1. Select **Layout > Customize Toolbars**.
2. Select **Editing** on the left hand side and, scrolling down, select **Time Effect** on the right. (Do not select **Timeline Effect**).

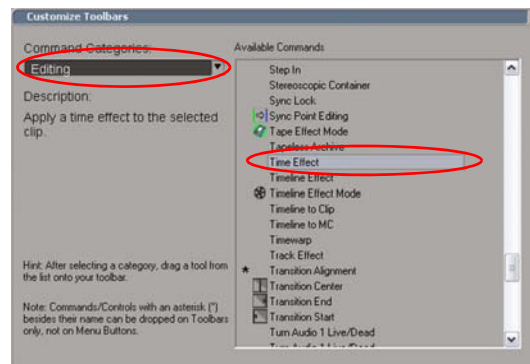


Figure 14. Customise Toolbar.

3. Drag **Time Effect** either onto a toolbar of your choice (the User toolbar is a good choice, Figure 15 red arrow) or onto the Time Effects button on the NLE Tools toolbar, Figure 15 green arrow.

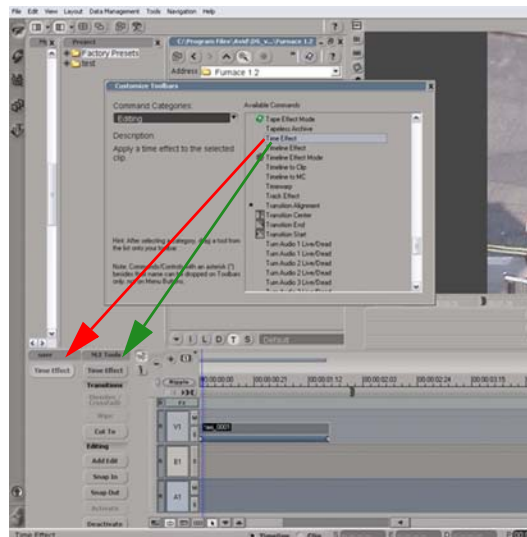


Figure 15. Create a Time Effect button.

4. Select your clip on the Timeline.
5. Select your new **Time Effect** button from the User toolbar (or Time Effects - Time Effect from the NLE toolbar).
DS will open a browser window in the Time Effects folder.
6. Navigate to the **DSPresets\Image Effects** directory and double click on **OFX Host.preset** to create a Timewarp container for the clip. Right-click on the F_KronosContainer on the Timeline and select **OFX Properties** from the menu to display the property pages. Set **Effect** to **F_Kronos**.

Option 2

1. Using the Browser, navigate to **\DSPresets\FurnaceCore\Time Effects**.
2. Drag **F_Kronos.preset** onto a toolbar of your choice (the User toolbar is a good location).
This creates an F_Kronos button.
3. Select your clip on the Timeline.
4. Click on the **Time Effects** button on the NLE toolbar and select **Timewarp**.
5. With your clip still selected, click on the F_Kronos button you created in step 2.

Applying Kronos

To retime a sequence, do the following:

1. Create a Timewarp Container and apply F_Kronos as described under "Timewarp Container" on page 31.
2. By default, the Speed control will be set to perform a half speed slow down. To adjust the slow down or to speed the sequence up, enter a new value for the **Speed** control. Values below 1 slow down the clip. Values above 1 speed up

movement. The default value, 0.5, created the half speed slow down. Quarter speed would be 0.25.

Alternatively, you can describe the retiming in terms of 'at frame 100 in the output clip, I want to see frame 50 of the source clip'. To do so, set **Timing** to **Source Frame**. Go to a frame in the timeline, and set **Frame** to the input frame you want to appear at that output position. You'll need to set at least two key frames for this to retime the clip. For example, to slow down a 50 frame clip by half, you can set **Frame** to 1 at frame 1, and to 50 at frame 100.

3. To create an arbitrarily changing speed for the sequence, see "Varying the Speed" below.
4. Render the retimed sequence using the default parameter settings. If you are not happy with the results, adjust the Vector Generation Parameters described under "Local Motion Estimation" on page 70.

Varying the Speed

To vary the speed in your sequence, do the following:

1. In the F_Kronos controls, choose **Source Frame** as the **Timing** method. This allows you to describe the retiming in terms of 'at frame 100 in the output clip, I want to see frame 50 of the source clip'.
2. Animate the **Frame** parameter. Select an output frame from the timeline and set **Frame** to the input frame you want to appear at that output position. Set a key frame. Move to another frame on the timeline and set **Frame** to the input frame you want to appear at that position. Set another key frame. For example, to do a four times slow down, move to frame 1 and set **Frame** to 1, set a key frame, move to frame 19, set **Frame** to 5, and set another key frame.

You should now have a linear time curve, which you can see in the Curve Editor. By using the Curve Editor to adjust this curve, you can create an arbitrarily changing speed for the sequence.

Instead of animating the **Frame** parameter, you can also switch **Timing** to **Speed**, and animate the **Speed** parameter.

Inputs

F_Kronos has the following inputs:

- **WarpSource** - The source image to retime.
- **Matte** - This sequence will be used as a foreground matte. This can improve the motion estimation by reducing the dragging of pixels that can occur between foreground and background objects.
- **Motion Source** - If supplied, motion vectors will be calculated from this sequence and applied to the input sequence. This can be useful if, for example, your input sequence is very noisy, as too much noise interferes with the motion estimation. In that case, you should supply a smoothed version of the sequence here.

Parameters

The parameters for this plug-in are described below.

Method - Sets the interpolation algorithm.

- **Nearest Frame** - the nearest original frame is displayed.
- **Frame Blending** - a mix between two frames is used for the inbetween frame. This is quick to render and useful when tweaking the timing on the curve before setting **Method** to **Motion Estimation**.
- **Motion Estimation** - vector interpolation is used to calculate the in-between frame.

Timing - Sets how to control the new timing of the clip.

- **Speed** - select this if you wish to describe the retiming in terms of overall duration, i.e. double speed will halve the duration of the clip or half speed will double the duration of the clip.
- **Source Frame** - select this if you wish to describe the retiming in terms of 'at frame 100 in the output clip, I want to see frame 50 of the source clip'. You'll need to set at least 2 key frames for this to retime the clip.

Speed - This parameter is only active if **Timing** is set to **Speed**. Values below 1 slow down the clip. Values above 1 speed up movement. For example, to slow down the clip by a factor of two (half speed), set this value to 0.5. Quarter speed would be 0.25.

Frame - This parameter is active only if **Timing** is set to **Source Frame**. Use this to specify the source frame at the current frame in the timeline. For example, to slow down a 50 frame clip by half set the **Source Frame** to 1 at frame 1 and the **Source Frame** to 50 at frame 100. The default expression will result in a half-speed retime.

Vector Detail - Adjust this to vary the density of the vector field. The larger vector detail is, the greater the processing time, but the more detailed the vectors should be. A value of 1.0 will generate a vector at each pixel. A value of 0.5 will generate a vector at every other pixel. For some sequences, a high vector detail near 1.0 generates too much unwanted local motion detail, and often a low value is more appropriate.

Smoothness - Vector fields usually have two important qualities: they should accurately match similar pixels in one image to another and they should be smooth rather than noisy. Often, it is necessary to trade one of these qualities off against the other. A high smoothness will miss lots of local detail, but is less likely to provide you with the odd spurious vector (think of it as if the vectors were combed until they are smooth with respect to their neighbours). A low smoothness will concentrate on detail matching, even if the resulting field is jagged. The default value of 0.5 should work

well for most sequences.

Shutter - Controls for adjusting the shutter time and the smoothness of the motion blur.

Shutter Time - Sets the equivalent shutter time of the retimed sequence. A shutter time of 1 is equivalent to averaging over plus and minus half an input frame which is equivalent to a shutter angle of 360 degrees. A shutter time of 0.5 is equivalent to a shutter angle of 180 degrees. Imagine a grey rectangle moving left to right horizontally across the screen. Figure 16 and Figure 17 show how **Shutter Time** affects the retimed rectangle.



Figure 16. Shutter Time 1.



Figure 17. Shutter Time 0.5.

Shutter Samples - Sets the number of in-between images used to create an output image during the shutter time. Increase this value for smoother motion blur.



Figure 18. Shutter Samples 2.



Figure 19. Shutter Samples 20.

Automatic Shutter Time - Automatically varies the **Shutter Time** throughout the sequence. This is on by default.

Output - Sets the final output display for the re-timed image. Selecting anything other than **Result** is only useful when a **Matte Component** is used.

- **Result** - displays the retimed Source image.
- **Matte** - displays the retimed Matte input.
- **Foreground** - displays the retimed foreground - the background regions outside the matte input may show garbage.

- **Background** - displays the retimed background - the foreground regions inside the matte input may show garbage.

Matte Component - The **Matte** input can be used to help the motion estimation algorithm understand what is foreground and background in the image so that the dragging of pixels between overlapping objects can be reduced. White areas of the matte are considered to be foreground, and black areas background. Grey areas are used to attenuate between foreground and background. When the matte input is filled with an appropriate clip, this parameter controls how the pixel values in the matte are used to do the masking.

- **None** - don't mask.
- **Src Alpha** - use the alpha of the source input.
- **Src Inverted Alpha** - use the inverted alpha of the source input.
- **Matte Luminance** - use the luminance of the matte input.
- **Matte Inverted Luminance** - use the inverted luminance of the matte input.
- **Matte Alpha** - use the alpha of the matte input.
- **Matte Inverted Alpha** - use the inverted alpha of the matte input.

Advanced - The lesser used refinement controls.

Block Size - To find where a certain pixel is located in the analysis range, the vector generation algorithm looks for a block of pixels centered around that pixel. Block size defines the width and height of these blocks (in pixels). On rare occasions, a large block size can produce data that's lacking in detail. This is because a small feature can fit into a large block, causing the motion estimation to concentrate on the background motion and ignore the small feature. A small value, instead, can produce a noisy motion field, as there aren't enough constraints in a small block to fit the motion accurately. In most cases, however, the default value is small enough so that details aren't lost, and the smoothing step of the algorithm ensures the motion field isn't too noisy. Therefore, this value very rarely needs editing.

Over smooth - This is a computationally intensive smoothing operation that performs a different vector-smoothing operation to normal. This generates highly smooth vector fields (as if the vectors were combed several times with a fine-toothed comb), but may also sacrifice a lot of required detail. In most cases, this level of smoothing isn't necessary. However, if you have a problem sequence that looks too 'broken up' in the retime, you may want to toggle **Over smooth** on.

Filtering - Sets the quality of filtering when producing in-between frames.

- **Extreme** - uses a sinc interpolation filter to give a sharper picture but takes a lot longer to render.
- **Normal** - use bilinear interpolation which gives good results and is a lot quicker than **Extreme**.

Warp Mode - Sets how to control the new timing of the clip.

- **Simple** - this is the quickest option, but may produce less than optimal results around moving objects and image edges.
- **Normal** - this is the standard option, with more optimal treatment of moving objects and image edges.
- **Occlusions** - this is an advanced option that can improve the results when not doing a separated picture build with multiple vector sequences and mattes. It attempts to reduce the level of background dragging that occurs between foreground and background objects.
- **Sharp Occlusions** - This option is similar to **Occlusions**, but produces fewer artifacts where the vector fields are generated from 3D CG sources.

Show Vectors - Switch this on to display the vectors on the screen.

Luminance Correct - LME is highly dependent upon the idea that the brightness of objects doesn't vary through a sequence. Where brightness varies rapidly - for example a highlight moving across the bodywork of a car - the motion calculation will perform poorly. The luminance of a shot can come from other sources too - such as an overall flicker problem. In these cases, where there is a global luminance shift, toggling this control on will allow the LME algorithm to take account of overall brightness changes between frames.

Tolerances - For efficiency, much of the LME is done on luminance only - that is, using monochrome images. The tolerances allow you to tune the weight of each colour channel when calculating the image luminance. These parameters rarely need tuning. However, you may, for example, wish to increase the red weighting **Weight Red** to allow the algorithm to concentrate on getting the motion of a primarily red object correct, at the cost of the rest of the items in a shot.

Weight Red

Weight Green

Weight Blue

About - Click this to display the online help, FurnaceCore version number, and the location of the user guide.

MATCHGRADE

This chapter looks at automatic colour matching using F_MatchGrade.

Effect Type

F_MatchGrade is a tree effect.

Introduction

It is often necessary to match the colours of one clip with those of another. When filming outside at different times of the day, you will inevitably get colour and luminance differences that will have to be corrected if the sequences are to be composited or edited together.

You can, of course, use colour correction tools and trial and error to try and match the clips. But this tends to be time-consuming and requires some considerable skill. F_MatchGrade does it all for you by automatically modifying the colour histogram of an image to match a reference image.



Figure 20. Source image.

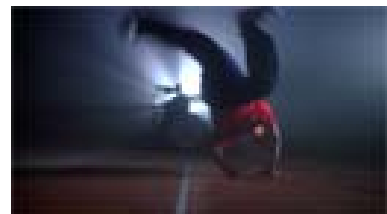


Figure 21. Reference image.

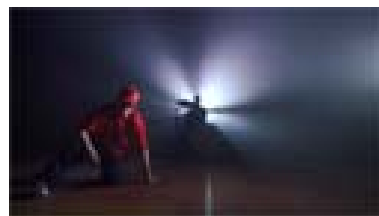


Figure 22. Output image.

This plug-in can also be used to boost desaturated images.

Quick Start

To match the colours of one clip with those of another, do the following:

1. Load the clip whose colours you want to change (source clip). Then, load the clip whose colours you want to match (reference clip).

2. Apply **F_MatchGrade**. See “Applying FurnaceCore Plug-ins” on page 8. Note that to apply a multi-input effect, such as **F_MatchGrade**, you first need to switch to the Compositing layout.
3. Connect the source clip to **F_MatchGrade**’s **Apply To** input, and the reference clip to the **TargetColour** input.
4. View the output which should now match the look of the **TargetColour** input.
5. If the match isn’t close enough, try increasing **Iterations**. This increases the number of refinement passes. More iterations should produce a better match but will take longer.

By default, **F_MatchGrade** calculates the transform needed to match the source image frame to the target image frame, and applies this transform to every frame of the source sequence. This way, the transformation is temporally uniform. However, if you want the colour transfer to update according to the current frame of the source and target clips, you can set the **Source Colour From** and **Target Colour From** parameters to **Current Frame**.

If you have already spent some time on a specific frame to make it look right and want to apply the same colour transform to the rest of the clip, do the following:

1. Connect the clip whose colours you want to adjust to **F_MatchGrade**’s **Apply To** input.
2. Connect the single original frame to the **SourceColour** input, and the modified frame to the **TargetColour** input.

F_MatchGrade will calculate the transform needed to match the **SourceColour** frame to the **TargetColour** frame, and apply this transform to every frame of the **Apply To** sequence.

Inputs

F_MatchGrade has three inputs:

- The first, **Apply To** input, is the sequence to which a colour transform will be applied.
- The second, **TargetColour** input, is the sequence whose colours **F_MatchGrade** will try to match. If only the first two inputs are supplied, this transform will make each frame of the **Apply To** sequence match the **TargetColour** input.
- If a third, **SourceColour** input, is supplied, **F_MatchGrade** will calculate the transform that matches the **SourceColour** input to the **TargetColour** input, and then apply that transform to each frame of the **Apply To** sequence.

Parameters

The parameters for this plug-in are described below.

Iterations - The number of refinement passes. More iterations should produce a better match but will take longer.

Advanced - the lesser used refinement controls.

Source Colour From - sets how the source clip is used to calculate the colour transformation. This parameter is only available if a **Source** input is connected.

- **Specified Frame** - the colour transformation is calculated using a single frame from the source clip. You can specify this frame using the **Source Frame** parameter below. This is the default.
- **Current Frame** - the colour transformation is calculated so that it updates according to the frames in the source clip.

Source Frame - sets the source clip frame that is used to calculate the colour transformation. By default, this is set to frame 0, which means frame 1 in Avid DS. This parameter is only available if a **Source** input is connected.

Target Colour From - sets how the target clip is used to calculate the colour transformation.

- **Specified Frame** - the colour transformation is calculated using a single frame from the target clip. You can specify this frame using the **Target Frame** parameter below. This is the default.
- **Current Frame** - the colour transformation is calculated so that it updates according to the frames in the target clip.

Target Frame - sets the target clip frame that is used to calculate the colour transformation. By default, this is set to frame 0, which means frame 1 in Avid DS.

About - Click this to display the online help, FurnaceCore version number, and the location of the user guide.

MOTIONBLUR

This chapter looks at adding motion blur using F_MotionBlur.

Effect Type

F_MotionBlur is both an image effect and a tree effect.

Introduction

F_MotionBlur uses the Foundry's advanced motion estimation technology to add realistic motion blur to a sequence.



Figure 23. Before F_MotionBlur.



Figure 24. After F_MotionBlur.

F_MotionBlur uses the same techniques and technology as the motion blur found in F_Kronos (see "Kronos" on page 30), but presents the controls in a less complex, more user friendly way. However, if you need precise control over the motion vectors used for adding blur, or a large temporal range (that is, a very high shutter time), you should use F_Kronos.

Quick Start

To add motion blur to a sequence, do the following:

1. Load the clip you want to add motion blur to.
2. Apply F_MotionBlur. See "Applying FurnaceCore Plug-ins" on page 8.
3. View the output.
4. Select a suitable **Shutter Time**, depending on the amount of blur you wish to add.



Figure 25. A low Shutter Time.



Figure 26. A high Shutter Time value.

5. Render the sequence to see the motion blurred result.
6. If you can see that the motion blur has been created from a few discrete images, try increasing **Shutter Samples**. This will result in more in-between images being used to generate the motion blur and so result in a smoother blur.

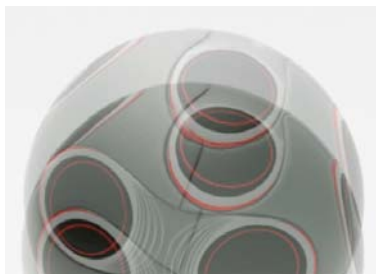


Figure 27. Shutter Samples = 2.

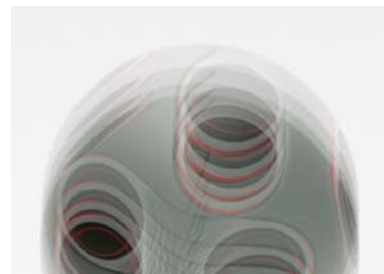


Figure 28. Shutter Samples = 4.

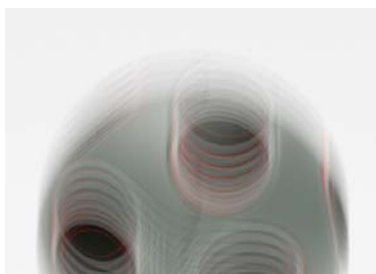


Figure 29. Shutter Samples = 8.



Figure 30. Shutter Samples = 30.

If your sequence is composed of a foreground object moving over a background, the motion estimation is likely to get confused at the edge between the two. To fix this, you can do the following:

1. Add a matte of the foreground region to the **Matte** input.
2. Use **Matte Component** to select which component of the matte to use.

This causes the motion of the foreground to be calculated separately to the motion of the background and so should produce less artefacts in the motion blur.

Inputs

F_MotionBlur has two inputs:

- **Warp Source** is the sequence to add motion blur to.
- **Matte** is an optional matte of the foreground to help improve the motion blur at foreground/background boundaries.

Parameters

The parameters for this plug-in are described below.

Shutter Time - Sets the equivalent shutter time of the re-timed sequence. A shutter time of 1 is equivalent to averaging over plus and minus half an input frame which is equivalent to a shutter angle of 360 degrees. A shutter time of 0.5 is equivalent to a shutter angle of 180 degrees. Imagine a grey rectangle moving left to right horizontally across the screen. Figure 31 and Figure 32 show how **Shutter Time** affects the retimed rectangle.



Figure 31. Shutter Time 1.



Figure 32. Shutter Time 0.5.

Shutter Samples - Sets the number of in-between images used to create an output image during the shutter time. Increase this value for smoother motion blur.



Figure 33. Shutter Samples 2.



Figure 34. Shutter Samples 20.

Vector Detail - Adjust this to vary the density of the vector field. The larger vector detail is, the greater the processing time, but the more detailed the vectors should be. A value of 1.0 will generate a vector at each pixel. A value of 0.5 will generate a vector at every other pixel. For some sequences, a high vector detail near 1.0 generates too much unwanted local motion detail, and often a low value is more appropriate.

Matte Component - What to use as the (optional) foreground matte for the motion estimation.

- **None** - don't use a matte.
- **Src Alpha** - use the alpha of the source input.
- **Src Inverted Alpha** - use the inverted alpha of the source input.
- **Matte Luminance** - use the luminance of the **Matte** input.
- **Matte Inverted Luminance** - use the inverted luminance of the **Matte** input.
- **Matte Alpha** - use the alpha of the **Matte** input.
- **Matte Inverted Alpha** - use the inverted alpha of the **Matte** input.

About - Click this to display the online help, FurnaceCore version number, and the location of the user guide.

REGRAIN

This chapter looks at adding grain to sequences using F_ReGrain.

Effect Type

F_ReGrain is both an image effect and a tree effect.

Introduction

F_ReGrain is used to add grain to a sequence. It has been designed to sample an area of grain from one image and then to generate unlimited amounts of this grain with exactly the same statistics as the original. This new grain can then be applied to another image.

Figure 35 shows an enlarged and exaggerated sample of grain from Kodak 320 film stock. F_ReGrain was used to sample the original Kodak 320 stock and synthesize a plate of grain. The result is shown in Figure 36. Note that the grain characteristics closely match the original.

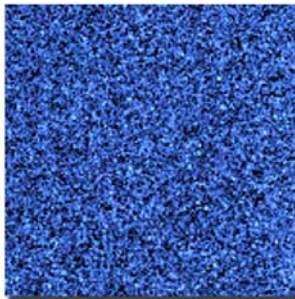


Figure 35. Kodak 320.

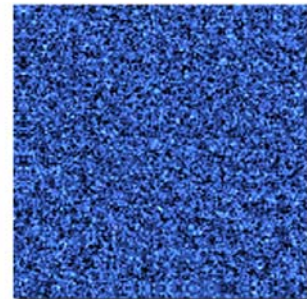


Figure 36. F_ReGrain.

Similarly, Figure 37 is a sample from Kodak 500 film stock and Figure 38 shows this replicated using F_ReGrain.

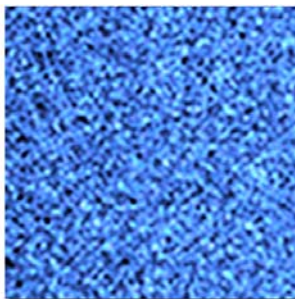


Figure 37. Kodak 500.

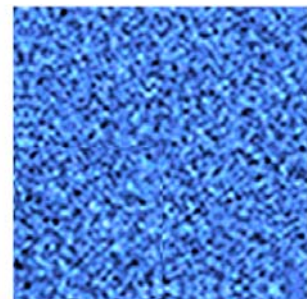


Figure 38. F_ReGrain.

Colour Space

Because colour space transformations can distort the look of grain, and by default, Avid DS converts your footage to sRGB, if you are sampling grain from your own plate, you need to make sure you tell the plug-in what space the plate was in originally, so the sample isn't distorted.

This is accomplished by setting the **Grain Colour Space** menu in the F_ReGrain controls to the right space. For example, if you were sampling grain from a film scan, you would want to set this to **Cineon**. If you had footage from a digital video camera, this would most likely be **sRGB**. F_ReGrain will automatically set the right colour space when using one of the pre-sampled grain clips, which are in **sRGB**.

F_ReGrain also works best when applying grain in the same colour space that the sampled grain originally existed. There is a menu in the **Response** section of the F_ReGrain controls which allows you to match or override this. It defaults to applying the grain in the same space as the sample.

Quick Start

Adding Sampled Grain

To add grain to a sequence, do the following:

1. Load the sequence you want to add grain to. Then, load the image you want to sample grain from.
2. Make sure you are working at full resolution and not proxy resolution. F_ReGrain will not work at proxy resolution. (See "Proxy Resolutions" on page 49.)
3. Apply F_ReGrain. See "Applying FurnaceCore Plug-ins" on page 8.
4. Connect the sequence that you want to have grain to F_ReGrain's **Source** input. Then, connect the sequence you want to sample grain from to the **Grain** input. View the output from F_ReGrain.
5. Set **Grain Type** to **From Grain Clip**.
6. Position the on-screen sample region over an area of the **Grain** sequence just containing grain and no picture detail. See Figure 39. It is important to get your selection right. You should avoid any image detail or even a plain area that has luminance variations underneath the grain. The better this initial selection, the better the result will be. If you can't find a decent sample area on the current frame, then try other frames from the same film stock. The default size of the sample area should be enough to gather information about the grain characteristics of your image. However, you may need to change its size and shape to fit over a plain area free of image detail.

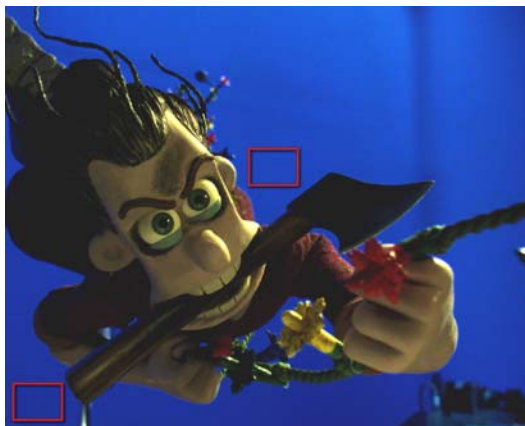


Figure 39. This shows two possible selection regions that contain no edge detail and little luminance variation.

Warning! *There is a minimum size of this sample area below which the statistical analysis of the grain will be unreliable. If the sample area you select is too small, you will see a warning message which prompts you to select a larger region. (See "Proxy Resolutions" on page 49.)*

7. View the output of F_ReGrain to judge the results. The output will now contain the **Source** image with grain from the **Grain** image applied. Both the size and the luminance of the new grain can be manually tweaked using **Grain Size** and **Grain Amount** respectively. It helps to view the **Grain** input while editing the parameters of F_ReGrain.

The grain is sampled on a single frame which is set when you adjust the sample area (or by manual adjustment of the **Analysis Frame** parameter). Although it is sampled on only one frame, the algorithmically created grain will change from frame to frame but mirror the characteristics of the sample grain.

Using Pre-Sampled, Standard Grain Types

If you don't have an image to sample grain from, you can also select from a variety of pre-sampled, standard grain types. Do the following:

1. Load the sequence you want to add grain to.
2. Make sure you are working at full resolution and not proxy resolution. F_ReGrain will not work at proxy resolution. (See "Proxy Resolutions" on page 49.)
3. Apply F_ReGrain.
4. Connect the sequence that you want to have grain to F_ReGrain's **Source** input.
5. Set **Grain Type** to **Preset Stock**.
6. Try the different grain types using the **Preset Stock** menu. 2K, 4K, aperture corrected, and non aperture corrected stocks are included. Individual colour channels can be selected and adjusted using the **Advanced** parameters.

Response

In its default setting, F_ReGrain adds the same amount of grain over the whole image. However, the amount of grain on an image is normally a function of luminance. Various parameters in the **Grain Response** group allow you to adjust how the amount of grain added varies with luminance:

- Pressing **Sample Grain Response** will cause the variation of the amount of grain with luminance to be calculated from the **Grain** input, and switching on **Use Sampled Response** will apply these curves to the grain added to the **Src** sequence.
- To view the sampled response curves, switch on **Draw Response**; an example is shown in Figure 40.
- The amount of grain added to the lowlights, midtones and highlights of the image can be adjusted using the **Low Gain**, **Mid Gain** and **High Gain** parameters. The effect of adjusting these can also be seen on the response curves.

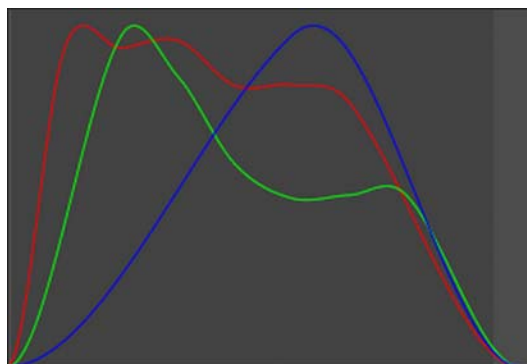


Figure 40. This shows an example of the grain response with luminance. The x axis represents luminance and the y axis the amount of grain.

Checking the Result

To test that the new grain is the same as the old grain, set **Output** to **Grain Plate**.

This generates a sheet of grain with the same luminance level as the mean of the sample region. The sample region with the original grain is also displayed. It should be impossible to differentiate between the two regions. Figure 41 shows a good selection area giving a good test plate of grain in Figure 42.



Figure 41. Good selection area...



Figure 42. ...producing a good test plate of grain, free of artefacts.

Figure 43 shows a poor selection area since it contains image detail. Figure 44 shows the resulting test plate which clearly highlights the problem.

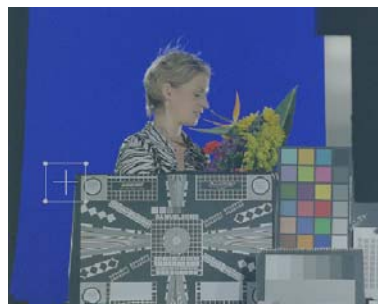


Figure 43. Bad selection area...

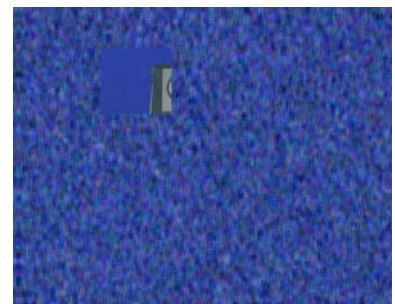


Figure 44. ...producing a poor result.

Proxy Resolutions

Grain manipulation at proxy resolution should be avoided as the results are unreliable. The grain selection area may be too small at proxy resolution to give a good result, and making this area larger may drag in unwanted detail from the image. If you try to use F_ReGrain at proxy resolution, we simply pass the image through untouched and issue the following warning:

Cannot work at proxy scale.

We decided that this was preferable behaviour to doing poor grain replication at proxy resolution. You can, of course, crop the input clip and work with that rather than the proxy. There is a minimum size for the selection box, which is about 37x37 at base resolution. If the box you select is smaller, you will get this warning along the top of the viewer:

Sample box is too small - please select a larger sample of the grain.

Inputs

F_ReGrain has two inputs:

- The **Source** is the image to which the grain will be added.
- The **Grain** is the image from which grain will be sampled. When a **Grain** input is supplied, the plug-in will automatically switch to using grain sampled from this input. However, the supplied grain stocks are still available.

Parameters

The parameters for this plug-in are described below:

Grain Type - Selects whether the grain is sampled from the **Grain** image (**From Grain Clip**) or from a set of standard stocks.

- **Preset Stock** - grain characteristics are sampled from a supplied film stock. 2K, 4K, aperture corrected and non aperture corrected stocks are supplied. Although standard stocks are included, it is recommended where possible that you sample from the film stock you are trying to match.
- **From Grain Clip** - samples and reconstructs the grain characteristics from the **Grain** input.

Preset Stock - The film stock the grain characteristics are sampled from when **Grain Type** has been set to **Preset Stock**. Common Fuji and Kodak stocks are supplied. The exposure can be under, over, or, if left blank, non aperture corrected. The size is either 2K or 4K pixels. For example, **FUJIF500 2K** refers to the grain characteristics sampled from a 2K plate of Fuji Film 500 film stock non aperture corrected.

Grain Amount - Adjusts the brightness of the grain. Setting this to 0 means no grain is added.

Grain Size - Adjusts the size of the grain granules. The larger the value, the bigger and softer the granules.

Output - Sets whether to render the result or a test image.

- **Result** - shows the **Source** image with the grain applied.
- **Grain Plate** - shows a test image with the grain applied. This test image is composed from a section of the input image surrounded by a uniform solid colour sampled from the image with the grain applied (Figure 44 on page 49). If the inner area is indistinguishable from the outer area, then you have a good grain sample (Figure 42 on page 49).

Analyse - This is a push button which will trigger an analysis of the input clip. Press this button if the input clip from which the grain was analysed

has changed but you do not want to move the analysis region to trigger re-analysis. Whenever the input clip changes, you will see the following warning in the Viewer:

The clip from which the grain was analysed has changed. Press Analyse or move the analysis region to re-analyse grain.

Analysis Region - A selection box that marks the region of image used to analyse the grain when **Grain Type** is set to **From Grain Clip**. This part of the frame must contain no image detail, only grain. (Figure 41 on page 49.)

Analysis Region BL - controls the position of the bottom left corner of the analysis region.

Analysis Region TR - controls the position of the top right corner of the analysis region.

Analysis Frame - sets the frame to sample the grain from.

Grain Colour Space - This tells F_ReGrain what colour space the grain sample clip was in when the grain originated. Setting this correctly ensures that the grain is not exaggerated by any colour space conversions prior to sampling.

- Cineon
- sRGB
- Linear

Advanced - The parameters under **Advanced** allow detailed adjustment of the grain.

Process Red - Switch this on to process the red channel.

Red Amount - sets the brightness of the grain in the red channel.

Red Size - adjusts the size of the grain granules in the red channel.

Process Green - Switch this on to process the green channel.

Green Size - adjusts the size of the grain granules in the green channel.

Green Amount - sets the brightness of the grain in the green channel.

Process Blue - Switch this on to process the blue channel.

Blue Amount - sets the brightness of the grain in the blue channel.

Blue Size - adjusts the size of the grain granules in the blue channel.

Grain Response - The parameters under **Grain Response** allow the amount of grain added to be varied as a function of the image luminance.

Apply Grain In - This controls what colour space the grain sample is re-applied to the image.

Generally, this should be set to **Grain Colour Space** to ensure the most accurate recreation.

You may want to override this though for some looks or special cases.

- **Grain Colour Space** - The Grain sample will be applied in the colour space set in the **Grain Colour Space** menu, in the **Grain Sample** section.
- **Linear / sRGB / Cineon** - The grain sample will be applied in the specified space.

Low Gain - adjusts the gain of the grain in the lowlights.

Mid Gain - adjusts the gain of the grain in the midtones.

High Gain - adjusts the gain of the grain in the highlights.

Use Sampled Response - switch this on to scale the brightness of the grain as a function of the luminance of the Grain image.

Sampled Response Mix - this control is usually set to 1. Decreasing it reduces the effect of the response curves until, at 0, they have no effect on the output. This parameter is only available if **Use Sampled Response** is on.

Sample Grain Response - press this to update the response curves from the current frame. Multiple presses accumulate the grain response rather than resetting every time. This parameter is only available if **Use Sampled Response** is on.

Reset Grain Response - press this to reset the grain curves to their default (flat) response. This parameter is only available if **Use Sampled Response** is on.

Draw Response - overlays the response curves on the bottom left corner of the viewer. This parameter is only available if **Use Sampled Response** is on.

About - Click this to display the online help, FurnaceCore version number, and the location of the user guide.

RIGREMOVAL

This chapter looks at the removal of unwanted objects (rigs) from image sequences without accurate rotoscoping or keying to produce a clean plate.

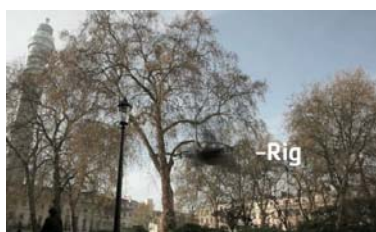


Figure 45. Before with bird.



Figure 46. After applying F_RigRemoval.

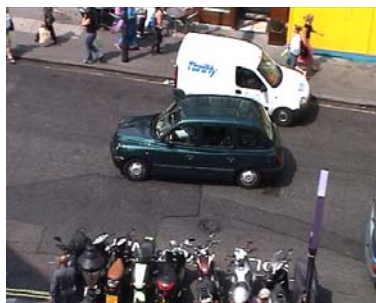


Figure 47. Before with taxi.



Figure 48. After applying F_RigRemoval.

Effect Type

F_RigRemoval is both an image effect and a tree effect.

Introduction

In this context, we define a rig as a foreground element in a sequence that moves over a background element. The plug-in will only work satisfactorily if it is possible to model the background motion by a global 3D transform. For example, if the background contains multiple objects moving in different directions, the results will be poor. Typically, good results will only be achieved in situations where a skilled artist could generate, and track in, a clean plate in order to repair the sequence. However, this plug-in should make the process quicker and easier. The rig removal algorithm works by estimating the background motion between successive frames, ignoring the foreground object, and then using the motion information to look forward and backward in the sequence in order to find the correct piece of background to fill in the missing region. The size of the missing region and

the speed of the background dictate how far away from the current frame it is necessary to search to find the correct information.

Quick Start

To remove an unwanted object from an image sequence, do the following:

1. Load the sequence with an unwanted object.
2. Apply F_RigRemoval. See “Applying FurnaceCore Plug-ins” on page 8.
3. Connect your image sequence to F_RigRemoval’s **Source** input. View the output of F_RigRemoval.
4. On each frame, define the area that will be repaired. You can do this in the following three ways:
 - If the image sequence has an embedded alpha channel, then you can use that alpha to define the area to be repaired. To do so, set **Rig Region** to **Src Alpha**.
 - Using F_RigRemoval’s **Rig Mask** input, you can feed in a matte sequence to define the area to be repaired. To use this matte, set **Rig Region** to **Rig Mask Alpha** (or one of the other **Rig Mask** options).
 - You can use the on-screen rectangle to define the area to be repaired. To do so, set **Rig Region** to **Box**. Position the on-screen rectangle on top of the unwanted object. To key-frame the position to follow the object, set keys on **Rig Region BL** and **Rig Region TR** in the **Rig Region Box** parameter group. Move to a new frame and reposition the on-screen rectangle. Set another key frame. Repeat as necessary until the rectangle covers the object on every frame you want to remove the object from.

Whichever method you choose, the region does not need to be the exact foreground region, but just a rough outline. However, you should avoid making it unnecessarily large as this will increase rendering time.

5. Having defined the region to repair throughout the clip, set **Frame Range** to the number of frames that the plug-in needs to analyse forwards and backwards to find enough data to repair the sequence. On the first frame, this will be quite time consuming as the algorithm needs to estimate the motion between each pair of frames. Subsequent frames will be much quicker.
6. If it has not been possible to replace all the foreground pixels, either because **Frame Range** was set too low or the background information does not exist anywhere within the sequence, the pixels will be displayed in red. Try to adjust **Frame Range** until no red pixels are visible and then render the sequence.

In Figure 49, we are using a box to define the pixels to replace, and **Frame Range** is set to zero. Increasing this value, as shown in Figure 50, gathers pixels from other frames and improves the result. To completely remove the red pixels, you’d need a **Frame Range** value of 5.



Figure 49. Frame Range = 0.



Figure 50. Frame Range = 3.

7. View the results.



Figure 51. Original image.



Figure 52. The output of F_RigRemoval.

Tip

F_RigRemoval is fairly slow to process, which can make the initial keyframing of the rectangular region frustrating.

Sometimes, the easiest way to adjust the region is to load up F_RigRemoval, and to view the source so the effect is not processed, but the parameters are visible. Then, animate the rectangular region over the foreground object you're trying to remove throughout the sequence. When you're happy with the region position, click back onto F_RigRemoval's output and wait for it to update. Slowly increase the **Frame Range** parameter until the whole region is repaired, then check the output on other frames.

Note

F_RigRemoval uses frames from a wide range around the current frame. Feeding the output of one F_RigRemoval node into another will greatly increase the memory overhead, as the second F_RigRemoval node will require the first F_RigRemoval node to calculate all its frames before passing the result on. Therefore, it is strongly recommended that you

process out the first result and then use the processed result with the second F_RigRemoval node.

Occlusions

The algorithm used in F_RigRemoval is unable to differentiate between multiple foreground objects. If there is another foreground object in the sequence that moves through the background region that is being used in the repair, this second foreground object will also be cut up and used, resulting in an incorrect repair. To try and assist in these situations, it is possible to mark regions of the image as not to be used for repair by setting their alpha value to mid grey. This will ensure that spurious bits of other foreground objects do not appear in the repair.

In Figure 53, we are trying to remove the woman in the centre of the screen as she walks from left to right down the street. At this frame, a man walks in the opposite direction and her feet and his head overlap.



Figure 53. Original shot.

Figure 54 shows the normal matte for the woman, and Figure 55 shows the result of using this in F_RigRemoval. Note that the man's head interferes with the repair and the reconstruction of the pavement is particularly bad, probably due to the man becoming the dominant motion source.



Figure 54. The normal matte.



Figure 55. The output.

To fix this, we can adapt the matte to include a mid grey area over the man. This tells the rig removal algorithm to ignore that area in the repair. This matte is shown in Figure 56, and the result is shown in Figure 57. Note that the repair on the pavement is improved, and the man is simply clipped rather than being used in the repair.

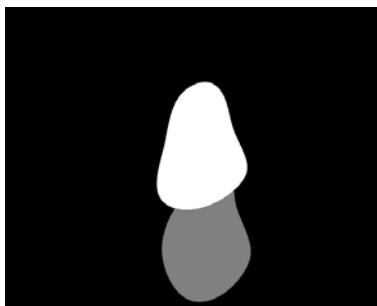


Figure 56. The matte with a mid-grey area.



Figure 57. The output.

Inputs

F_RigRemoval has two inputs:

- The **Source** image with the unwanted object (rig). The source may contain an alpha channel to define the rig area.
- The optional **Rig Mask** to designate the rig area to be removed.

Parameters

The parameters for this plug-in are described below.

Rig Region - Defines the area to repair.

- **Box** - repair the area inside a rectangular box, controlled by the box parameters below or the on-screen box.
- **Src Alpha** - repair the region defined by the alpha of the source input.
- **Src Inverted Alpha** - repair the region defined by the inverted alpha of the source input.
- **RigMask Luminance** - repair the region defined by the luminance of the **Rig Mask** input.
- **RigMask Inverted Luminance** - repair the region defined by the inverted luminance of the **Rig Mask** input.
- **RigMask Alpha** - repair the region defined by the alpha of the **Rig Mask** input.
- **RigMask Inverted Alpha** - repair the region defined by the inverted alpha of the **Rig Mask** input.

Frames Searched - Sets whether to search forwards, backwards, or in both directions to find missing data.

- **Forward and Backward** - searches before and after the current frame.
- **Forward** - searches frames after the current frame.
- **Backward** - searches frames before the current frame.

Frame Range - Sets the number of frames the algorithm should look forwards and backwards in the sequence to find the missing data. If you are getting red pixels, then increase this value. See Figure 50 on page 55.

Frames Used in Range - If **Frame Range** has to be set to a large number to make an effective repair, the rendering time can be prohibitive. **Frames Used in Range** can speed up the repair by not using every frame to fill the foreground region, effectively skipping frames. However, this may reduce the quality of the result.

- **All Frames** - use every frame in the searched frame range to construct the repair.
- **Half of Frames** - use every other frame in the searched frame range to construct the repair.
- **Quarter of Frames** - use every fourth frame in the searched frame range to construct the repair.
- **10% of Frames** - use every tenth frame in the searched frame range to construct the repair.
- **Max 25 Frames** - use no more than 25 frames from the searched frame range to construct the repair. This option can be useful if **Frame Range** has been set to a very large number.

Max Rig Movement - To avoid perspective changes, F_RigRemoval searches for the missing data inside an area immediately around the rig region. **Max Rig Movement** defines the width of this area (in pixels). Fast movement in the **Src** footage requires a higher value than slow movement. However, because the area used for the repair may be from a different part of the image, high values can introduce perspective problems.

Rig Region Box - The rectangular area used to define the repair when **Rig Region** is set to **Box**.

Rig Region BL - controls the position of the bottom left corner of the rig region.

Rig Region TR - controls the position of the top right corner of the rig region.

Advanced - The lesser used refinement controls.

Filtering - Sets the filtering quality.

- **Low** - low quality but quick to render.
- **Medium** - uses a bilinear filter. This gives good results and is quicker to render than high filtering.
- **High** - uses a sinc filter to interpolate pixels giving a sharper repair. This gives the best results but takes longer to process.

Luminance Correct - Switch this on to correct for luminance changes from information taken from other frames. This is particularly important if the lighting changes throughout the sequence.

Perspective Correct - Switch this on to correct for minor perspective changes.

Overlap Correct - The repair is built up using slices of information from other frames in the sequence. These slices can be overlapped and blended to give a more natural looking repair. This parameter controls how much the regions overlap. Increasing this parameter too much will degrade image sharpness.

Repair Fail Marker Opacity - Sets the level of transparency of the red pixels used to show where the repair has failed.

Preserve Alpha - Switch this on to preserve the original alpha channel. By default, this is switched off, and the alpha channel is set to white where the repair has failed and black everywhere else.

About - Click this to display the online help, FurnaceCore version number, and the location of the user guide.

STEADINESS

This chapter looks at how to stabilise a shot using F_Steadiness.

Effect Type

F_Steadiness is an image effect.

Introduction

F_Steadiness uses Global Motion Estimation (GME) to calculate a four-corner pin, so that camera motion within a single shot can be smoothed out over a range of frames or removed by locking to a specific frame.

F_Steadiness needs to analyse the input clip before it can render useful output. This analysis is done when you press **Analyse** in the F_Steadiness controls. During the analysis, F_Steadiness key-frames a four-corner pin which will stabilise the clip in subsequent renders. Without having performed an analysis pass, F_Steadiness will not do anything useful on render.

F_Steadiness can work in two ways. These are:

1. **Smooth** - A range of frames around each frame is analysed for motion and an average of that motion used to calculate the corner pin. Use this to keep the overall camera motion, but to smooth out sharp bumps and kicks.
2. **Lock** - A lock frame is specified and steadiness attempts to register each individual frame to that lock frame. Use this to completely remove camera motion from the sequence.

In lock mode, each frame in the clip must share a substantial amount of the scene with the lock frame, so you can't lock each frame in a 360 degree pan to the first one. However, in smooth mode, because F_Steadiness is only working on a small window of frames around the current frame, you can use it in shots which change completely over time.

The analysis region is used to control which section of the reference frame is being matched to each source frame. In lock mode, the reference is the lock frame, so you should position the analysis region when looking at the lock frame. In smooth mode, it looks at the incremental differences between frames, in which case you should place the analysis region in the area you want to appear with smooth motion.

For an overview of Global Motion Estimation, and a description of the common way Global Motion Effects work, please see "Global Motion Estimation" on page 65.

Quick Start

This section gives a very brief outline of how to use the plug-in.

Smoothing Out Camera Motion

To keep the overall camera motion but to smooth out sharp bumps and kicks, do the following:

1. Find a shot that has some camera shake in it and load it.
2. Apply F_Steadiness. See "Applying FurnaceCore Plug-ins" on page 8.
3. View the output.

You will see the following banner in the overlay:

No valid key framed analysis found, please press Analyse.

4. In the **Advanced** F_Steadiness controls, make sure **Mode** has been set to **Smooth**.
5. Click on the **Analyse** button.

F_Steadiness will now start analysing each frame in the shot, figuring out the smoothing corner pin and writing it as key frames to the corner pin parameters, **Bottom Left**, **Bottom Right**, **Top Left**, and **Top Right**.

After a brief pause (while F_Steadiness calculates the transforms in the frame range), F_Steadiness will update the timeline and you will see the steadied image render in the viewer.

If at any point you interrupt the analysis, the pins it has calculated until that point will be retained.

6. Play or scrub through the stabilised frames.
7. If you want to make the result smoother, increase the **Smoothing** parameter in the **Advanced** parameter group and the corner pin will be recalculated immediately to give a smoother shot. You don't need to re-analyse the sequence if you do this; as F_Steadiness has kept the raw inter-frame transforms cached away, all it needs to do is re-write the keys for the average smoothing pin.

Locking To A Frame

To completely remove camera motion from a sequence, do the following:

1. Find a shot that has camera shake but where all frames share scene information and load it.
2. Apply F_Steadiness. See "Applying FurnaceCore Plug-ins" on page 8.
3. View the output.

You will see the following banner in the overlay:

No valid key framed analysis found, please press Analyse.

4. In the **Advanced** F_Steadiness controls, set **Mode** to **Incremental Lock**.
5. Choose a frame somewhere in the middle of the sequence that you want to lock to, and set the **Lock Frame** parameter to that frame.
6. Scrub back and forth through the shot and look for a region of the shot that is shared by all frames and doesn't change much (for example, avoid a region with people walking in front of it).

7. Whilst looking at the lock frame, position the on-screen widget for the **Analysis Region** over that region.
8. Hit **Analyse**.
The effect will start to analyse, working first forward from the lock frame, then backwards from it, until all frames to be analysed are done.
The timeline will immediately update to give you feedback in the viewer.

Inputs

F_Steadiness has a single input: the shot to stabilise.

Parameters

The parameters for this plug-in are described below.

Analyse - This is a push button which will trigger an analysis of the input clip and calculate a corner pin. Interrupting the analysis will not delete the corner pin keys that have already been calculated.

Clear Analysis - Pressing this push button will delete all key frames from the corner pin parameters, allowing you to force a re-analysis if you feel the need to.

Analysis Range - This controls the range of frames any analysis will be run over. It can be one of:

- **Specified Range** - which will look at the parameters **Analysis Start** and **Analysis Stop** for the range of frames to analyse,
- **Source Clip Range** - which will automatically determine the range of frames to analyse from the length of the input clip.

Analysis Start - The first frame to analyse from if **Analysis Range** is set to **Specified Range**.

Analysis Stop - The last frame to analyse from if **Analysis Range** is set to **Specified Range**.

Scale - A toggle that indicates whether the calculated corner pin can include a scaling factor.

Rotate - A toggle that indicates whether the calculated corner pin can include rotations.

Translate - A toggle that indicates whether the calculated corner pin can include translations in x and y.

Perspective - A toggle that indicates whether the calculated corner pin can include perspective transforms.

Analysis Region - This is the region analysed to calculate the four-corner pin. This is especially useful when doing any form of frame locking, in which case, go to the lock frame, look at the reference clip and position the box over the area you want locked.

Analysis Region BL - controls the position of the bottom left corner of the analysis region.

Analysis Region TR - controls the position of the top right corner of the analysis region.

Advanced - the lesser used refinement controls.

Mode - This parameter controls whether F_Steadiness is smoothing the shot while keeping the overall camera motion, or locking the shot to a single frame to completely remove camera motion. It can be set to:

- **Incremental Lock** - in which case, it will calculate the pin that takes each frame to the lock frame. This calculates the pin by working from the lock frame out to each frame, calculating the GME between each frame incrementally and accumulating it to create the corner pin.
- **Absolute Lock** - this also calculates a pin that takes each frame to the lock frame. However, it does so by doing GME directly from the frame in question directly to the lock frame.
- **Smooth** - in which case, we are smoothing the shot for a range of frames described by the **Smoothing** parameter. This is the default value. It can be used to keep the overall camera motion, but to smooth out sharp bumps and kicks.

Incremental locks work better in some situations, whilst absolute locks work better in others. However, absolute locks, when they work, are often more accurate. If, for example, you want to get a lock between frame 0 and frame 100, try using absolute lock first. If absolute lock doesn't work, it can be because frame 0 and frame 100 are too different. In this case, incremental lock can help if there is a gradual spatial progression from frame 0 to frame 100.

Smoothing - This controls the range of frames to average motion over when **Mode** is set to **Smooth**. The default value is 10.

Lock Frame - This controls the frame which will be locked to when **Mode** is set to either of the lock modes. By default, this is set to 0. Note that what is frame 0 for F_Steadiness is frame 1 in Avid DS. Therefore, if you are looking at frame 3 in the Viewer and want to use that frame as the lock

frame, you need to enter 2 as the **Lock Frame** value.

Accuracy - This controls the time/accuracy trade off. The higher this is, the slower the analysis, but you have a better likelihood of a good result.

Filtering - Sets the filtering quality.

- **Low** - low quality but quick to render.
- **Medium** - uses a bilinear filter. This gives good results and is quicker to render than high filtering.
- **High** - uses a sinc filter to interpolate pixels giving a sharper repair. This gives the best results but takes longer to process.

Invert - If set, then the inverse of the calculated four-corner pin is used during render. This works best with the lock modes, and can be used to track static locked-off plates back into a shot.

Auto Scale - To smooth out or remove camera motion, F_Steadiness translates and rotates the frames in the source clip. This leaves black pixels around the image. The Auto Scale parameter lets you fill in the black gaps at the edges by scaling the output image up. A value of 1 (the default) scales the image up until no black is visible, whereas a value of 0 disables scaling and leaves the black edges untouched. Auto Scale uses the minimum scale necessary to remove the black gaps at the edges to preserve as much detail as possible.

Four Corner Pin - The corner pins calculated during the analysis pass:

Bottom Left - the lower left corner pin.

Bottom Right - the lower right corner pin.

Top Left - the upper left corner pin.

Top Right - the upper right corner pin.

About - Click this to display the online help, FurnaceCore version number, and the location of the user guide.

GLOBAL MOTION ESTIMATION

Introduction

FurnaceCore has three effects based on global motion estimation (GME). They all calculate a four-corner pin, which finds the best fit of one image onto another, and then apply that pin. These effects differ in how that corner pin is calculated and to which image that pin is applied. This chapter describes the general idea behind these plug-ins; for more detailed information on each plug-in, please read their individual chapters.

These effects are:

- **F_Align** - which lines up two shots of the same scene, by finding a corner pin from each source clip frame to the corresponding reference clip frame. For example, you can use this to align two separate but similar steady cam shots of the same scene.
- **F_RigRemoval** - which removes unwanted objects (rigs) from image sequences without accurate rotoscoping or keying to produce a clean plate.
- **F_Steadiness** - which removes motion from a single clip, by calculating a pin that either locks all frames in the clip to a single reference frame, or smooths that motion out over a window of frames. For example, you can use this to remove camera shake.

What is Global Motion Estimation?

Global motion estimation is a technique that attempts to map one image onto another with a simple four-corner pin. This differs from local motion estimation (LME), which attempts to find where each individual pixel in the image is in the other image. (For more information about local motion estimation, please see "Local Motion Estimation" on page 70). GME is much cheaper to compute than LME, but gives you less information about the image. Nevertheless, it is still very powerful for a variety of applications.

Using the plug-ins' parameters, you can tell the GME engine what type of motion to expect. This can be a combination of any of:

1. **translation** - which allows the four corners to translate by the same amount,
2. **rotation** - which allows the corners to rotate about their centre,
3. **scale** - which allows the size of the area defined by the corners to change,
4. **perspective** - which allows the angles at the corners to change, so that the area defined by them is no longer a rectangle.

The more types of motion you allow, the more expensive the motion

estimation becomes. For many scenes, rotation and translation are sufficient.

The GME effects have an accuracy control, which controls the amount of work The Foundry's GME engine does to calculate the global motion. Typically, the higher this is, the better the estimation, but the more expensive it is.

Limitations of GME

As stated above, global motion estimation simply calculates a four-corner pin to transform one image onto another. This means that GME can't be used to match two images where there is heavy parallax, very complicated foreground motion, changing objects, and so on.

The best way to think of what GME can do is that if you can do it with a four-corner pin, it can; if you can't, it can't. However, GME will take the pain out of hand matching pins frame by frame.

The Analysing Global Motion Estimation Effects

F_Align and F_Steadiness work in a similar way, which is distinct from the way F_RigRemoval works. These two effects calculate a four-corner pin for each frame and save it into the corner pin parameters. These pins are then used during the render to move the source image.

Using Them

These effects analyse images over a range of frames to figure out their four-corner pins. This is done in response to the user pressing the **Analyse** button in the effects control panel. During analysis, the effect will run through a range of frames adding keys to the corner pin parameters. These corner pins are then applied to the source clip to render a frame.

F_Steadiness has a separate analysis pass that happens interactively; it then uses the previously computed and key-framed corner pins during render. This speeds up its operation, as the analysis step only has to be done once, and once it has been done this plug-in is very quick to render. However, F_Align, which only ever needs the two current frames from each clip, can compute the corner pin on the fly (but not keyframe it!). This leads to a slightly different mode of operation for the following effects:

- F_Steadiness
 - needs to have an analysis run before it renders useful output,
 - will always use the value in the corner pin parameters when rendering the output image.

- **F_Align**
 - no need to have the analysis run to render output, but doing so will give you a key-framed corner pin,
 - during render, it will use the value of the corner pin parameters only if there is a keyframe there, otherwise, it will analyse on the fly during render. This means that analysis will speed up later renders, as just rendering the corner pin is much cheaper than calculating it.

Some parameters to the effect control how the effect performs GME analysis, and some only affect the rendering. If you ever modify one of these parameters, then any analysis you may have performed will be out of date. To let you know, an overlay warning will be posted whenever this happens. You don't have to re-analyse and your renders will still look at the keyed corner pins.

If you have not modified a parameter that affects analysis (the warning overlay will let you know), pressing **Analyse** will only re-analyse a frame if there is no key on the corner pin at that time. This avoids redundant re-analysis if you have interrupted the analysis or extended the analysis range. However, if you want to force re-analysis, press **Clear Analysis** and all keys will be deleted.

F_Steadiness and F_Align have an analysis region rectangle parameter which is used to specify which area of the reference image should be analysed during GME. So, for example, with F_Steadiness set to **Lock Mode**, this is the area inside the lock frame that a match will be sought for. The documentation for each plug-in describes exactly how to use the analysis region.

Controls

The controls common to all GME plug-ins are described below. They are grouped into two sections: ones that determine how analysis is carried out, and ones that control the rendering of the output.

Controls That Affect Analysis

The following parameters and controls affect the analysis of the four-corner pin.

Analyse - This is a push button which will trigger an analysis of the input clips and calculate a corner pin. Interrupting the analysis will not delete the corner pin keys that have already been calculated.

Clear Analysis - Pressing this push button will delete all keyframes from the corner pin parameters, allowing you to force a re-analysis if you feel the need to.

Analysis Range - This controls the range of frames any analysis will be run over. It can be one of:

- **Specified Range** - which will look at the parameters **Analysis Start** and **Analysis Stop** for the range of frames to analyse,
- **Source Clip Range** - which will automatically determine the range of frames to analyse from the length of the input clip.

Analysis Start - The first frame to analyse from if **Analysis Range** is set to **Specified Range**.

Analysis Stop - The last frame to analyse from if **Analysis Range** is set to **Specified Range**.

Scale - A toggle that indicates whether the calculated corner pin can include a scaling factor.

Rotate - A toggle that indicates whether the calculated corner pin can include rotations.

Translate - A toggle that indicates whether the calculated corner pin can include translations in x and y.

Perspective - A toggle that indicates whether the calculated corner pin can include perspective transforms.

Analysis Region - This is the region analysed to calculate the four-corner pin. This is especially useful when doing any form of frame locking, in which case, go to the lock frame, look at the reference clip and position the box over the area you want locked.

Analysis Region BL - controls the position of the bottom left corner of the analysis region.

Analysis Region TR - controls the position of the top right corner of the analysis region.

Accuracy - This controls the time/accuracy trade off in the GME engine. The higher this is, the slower it goes, but you have a better likelihood of a good result.

Parameters That Affect Rendering

These following parameters control how a GME effect renders the four-corner pin. Some of them are set during the analysis pass.

Filtering - This controls the quality of the rendering.

- **Low** - uses nearest neighbour filtering. This gives low quality but is quick to render.
- **Medium** - uses a bilinear filter. This gives good results and is quicker to render than high filtering.
- **High** - uses a sinc filter to interpolate pixels giving a sharper repair. This gives the best results but takes longer to process.

Invert - If set, then the inverse of the calculated four-corner pin is used during render.

Four Corner Pin - The corner pins calculated during the analysis pass:

- **Bottom Left** - the lower left corner pin.
- **Bottom Right** - the lower right corner pin.
- **Top Left** - the upper left corner pin.
- **Top Right** - the upper right corner pin.

Widgets

All the Analysing GME effects have two on-screen widgets: one to provide feedback and one to set up the analysis region.

Analysis Region Widget - This is a rectangle widget which you use to set the analysis region over the reference image.

Four Corner Widget - This is a widget that shows the state of the four-corner pin that has been calculated. You can change it by grabbing any of the corners and tweaking the shape of the pin. To give you more feedback as to what frames have been analysed, it will be drawn solid if there is a key in the corner pin at the frame being displayed; otherwise, it will be drawn dashed.

LOCAL MOTION ESTIMATION

Introduction

Several tools in FurnaceCore make use of Motion Estimation technology. Motion Estimation tends to fall into two areas: Global Motion Estimation and Local Motion Estimation. In this chapter, we look at the parameters for Local Motion Estimation (or LME), which is the per-pixel motion analysis used in tools such as F_Kronos.

Background

The easiest way to understand LME is to think in terms of vector fields. A vector field for an image in a sequence has the same dimensions as the image, but contains an (x,y) offset per pixel. These offsets show how to warp a neighbouring image onto the current image. Clearly, as most of the images in a sequence have two neighbours, each can have two vector fields. These are called the 'forward motion vectors' where they represent the warp of the image in front of the current one, and 'backward motion vectors' where they represent the warp of the image behind the current one.

This is an approximation to what is really going on in an image sequence. A single vector field can be thought of as representing a warp or a morph - sometimes referred to as a 'rubber sheet' warp and cannot truly represent multiple overlapping motions. This effect can be seen where moving foreground objects appear to drag the background. To help cope with this restriction, the **Occlusions** parameter attempts to improve the rubber-sheet effect.

Parameters

Not all tools using LME have all of these parameters. They can be categorised into two groups: one for the generation of the vectors, and one for their use in picture warping.

Vector Generation Parameters

Vector Detail - Adjust this to vary the resolution of the vector field. The larger vector detail is, the greater the processing time, but the more detailed the vectors should be. A value of 1.0 will generate a vector at each pixel. A value of 0.5 will generate a vector at every other pixel. For some sequences, a high vector detail near 1.0 generates too much unwanted local motion detail, and often a low value is more appropriate.

Smoothness - Vector fields usually have two important qualities: they

should accurately match similar pixels in one image to another and they should be smooth rather than noisy. Often, it is necessary to trade one of these qualities off against the other. A high smoothness will miss lots of local detail, but is less likely to provide you with the odd spurious vector (think of it as if the vectors were combed until they are smooth with respect to their neighbours). A low smoothness will concentrate on detail matching, even if the resulting field is jagged. The default value of 0.5 should work well for most sequences.

Block Size - To find where a certain pixel is located in the analysis range, the vector generation algorithm looks for a block of pixels centered around that pixel. Block size defines the width and height of these blocks (in pixels). On rare occasions, a large block size can produce data that's lacking in detail. This is because a small feature can fit into a large block, causing the motion estimation to concentrate on the background motion and ignore the small feature. A small value, instead, can produce a noisy motion field, as there aren't enough constraints in a small block to fit the motion accurately. In most cases, however, the default value is small enough so that details aren't lost, and the smoothing step of the algorithm ensures the motion field isn't too noisy. Therefore, this value very rarely needs editing.

Over Smooth - This is a computationally intensive smoothing operation that performs a different vector-smoothing operation to normal. This generates highly smooth vector fields (as if the vectors were combed several times with a fine-toothed comb), but may also sacrifice a lot of required detail. In most cases, this level of smoothing isn't necessary.

Luminance Correct - LME is highly dependent upon the idea that the brightness of objects doesn't vary through a sequence. Where brightness varies rapidly - for example a highlight moving across the bodywork of a car - the motion calculation will perform poorly. The luminance of a shot can come from other sources too - such as an overall flicker problem. In these cases where there is a global luminance shift, toggling this control on will allow the LME algorithm to take account of overall brightness changes between frames.

Tolerances - For efficiency, much of the LME is done on luminance only - i.e., using monochrome images. The tolerances allow you to tune the weight of each colour channel when calculating the image luminance. These parameters rarely need tuning. However, you may, for example, wish to increase the red weighting **Weight Red** to allow the algorithm to concentrate on getting the motion of a primarily red object correct, at the cost of the rest of the items in a shot.

Weight Red

Weight Green

Weight Blue

Picture Warping Parameters

Filtering - Sets the quality of filtering when producing in-between (F_Kronos) frames.

- **Normal** - use bilinear interpolation which gives good results and is a lot quicker than **Extreme**.
- **Extreme** - uses a sinc interpolation filter to give a sharper picture but takes a lot longer to render.

Warp Mode - Sets how to control the new timing of the clip.

- **Simple** - this is the quickest option, but may produce less than optimal results around moving objects and image edges.
- **Normal** - this is the standard option, with more optimal treatment of moving objects and image edges.
- **Occlusions** - this is an advanced option that can improve the results when not doing a separated picture build with multiple vector fields and mattes. It attempts to reduce the level of background dragging that occurs between foreground and background objects.

Show Vectors - Switch this on to display the vectors on the screen.

APPENDIX A

Release Notes

This section describes the requirements, new features, improvements, fixed bugs and known bugs & workarounds for each release of FurnaceCore.

FurnaceCore 4.3v1 This is the first release of FurnaceCore for Avid DS.

Release Date

July 2010

Requirements

- Windows XP SP2 or XP64.
- Avid DS 10.2.x or later.
- Foundry FLEXIm Tools (FFT) 5.0v1 or later for floating license support.

New Features

There are ten plug-ins in this release.

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

- BUG ID 7365 - Installer not taking into account localised languages.
The installer does not check for local language versions. It is necessary to redirect the installer to install to the correct location in these instances.
- BUG ID 9448 - 4K/2K plates on Windows 32-bit.
When running on Windows 32-bit, FurnaceCore often fails with "There is insufficient memory to complete the operation". This is due to lack of memory. There is currently not workaround to this issue.

APPENDIX B

Third Party Licenses

This appendix lists third party libraries used in FurnaceCore, along with their licenses.

Library	Description	License
Boost	Source code function / template library	<p>Boost Software License - Version 1.0 - August 17th, 2003</p> <p>Permission is hereby granted, free of charge, to any person or organization obtaining a copy of the software and accompanying documentation covered by this license (the "Software") to use, reproduce, display, distribute, execute, and transmit the Software, and to prepare derivative works of the Software, and to permit third-parties to whom the Software is furnished to do so, all subject to the following:</p> <p>The copyright notices in the Software and this entire statement, including the above license grant, this restriction and the following disclaimer, must be included in all copies of the Software, in whole or in part, and all derivative works of the Software, unless such copies or derivative works are solely in the form of machine-executable object code generated by a source language processor.</p> <p>THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, TITLE AND NON-INFRINGEMENT. IN NO EVENT SHALL THE COPYRIGHT HOLDERS OR ANYONE DISTRIBUTING THE SOFTWARE BE LIABLE FOR ANY DAMAGES OR OTHER LIABILITY, WHETHER IN CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.</p>
Expat	XML parser	<p>Copyright © 1998, 1999, 2000 Thai Open Source Software Center Ltd and Clark Cooper Copyright © 2001, 2002, 2003, 2004, 2005, 2006 Expat maintainers.</p> <p>Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:</p> <p>The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.</p> <p>THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NON-INFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.</p>

Library	Description	License
FFTW	Fast Fourier transform library	Copyright © 2003 Matteo Frigo Copyright © 2003 Massachusetts Institute of Technology All rights reserved. Used under terms of a commercial license. http://www.fftw.org .
FreeType	Font support	Portions of this software are copyright © 2008 The FreeType Project (www.freetype.org). All rights reserved.
FTGL	OpenGL support	FTGL - OpenGL font library Copyright © 2001-2004 Henry Maddocks ftgl@opengl.geek.nz Copyright © 2008 Sam Hocevar sam@zoy.org Copyright © 2008 Sean Morrison learner@brcad.org Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software. THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Library	Description	License
OFX	Plug-in API	<p>Copyright (c) 2003-2009, The Open Effects Association Ltd. All rights reserved.</p> <p>Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:</p> <ul style="list-style-type: none"> • Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer. • Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution. • Neither the name The Open Effects Association Ltd, nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission. <p>THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.</p>

APPENDIX C

End User License Agreement

IMPORTANT: BY INSTALLING THIS SOFTWARE YOU ACKNOWLEDGE THAT YOU HAVE READ THIS AGREEMENT, UNDERSTAND IT AND AGREE TO BE BOUND BY ITS TERMS AND CONDITIONS. IF YOU DO NOT AGREE TO THE TERMS OF THIS AGREEMENT DO NOT INSTALL, COPY OR USE THE SOFTWARE.

This END USER SOFTWARE LICENSE AGREEMENT (this "Agreement") is made by and between The Foundry Visionmongers Ltd., a company registered in England and Wales, ("The Foundry"), and you, as either an individual or a single entity ("Licensee").

In consideration of the mutual covenants contained herein and for other good and valuable consideration (the receipt and sufficiency of which is acknowledged by each party hereto) the parties agree as follows:

Section 1. Grant of License.

Subject to the limitations of Section 2, The Foundry hereby grants to Licensee a limited, non-transferable and non-exclusive license to install and use a machine readable, object code version of this software program (the "Software") and the accompanying user guide and other documentation (collectively, the "Documentation") solely for Licensee's own internal business purposes (collectively, the "License"); provided, however, Licensee's right to install and use the Software and the Documentation is limited to those rights expressly set out in this Agreement.

Section 2. Restrictions on Use.

Licensee is authorized to use the Software in machine readable, object code form only, and Licensee shall not: (a) assign, sublicense, transfer, pledge, lease, rent, share or export the Software, the Documentation or Licensee's rights hereunder; (b) alter or circumvent the copy protection mechanisms in the Software or reverse engineer, decompile, disassemble or otherwise attempt to discover the source code of the Software; (c) modify, adapt, translate or create derivative works based on the Software or Documentation; (d) use, or allow the use of, the Software or Documentation on any project other than a project produced by Licensee (an "Authorized Project"); (e) allow or permit anyone (other than Licensee and Licensee's authorized employees to the extent they are working on an Authorized Project) to use or have access to the Software or Documentation; (f) copy or install the Software or Documentation other than as expressly provided for herein; or (g) take any action, or fail to take action, that could adversely affect the trademarks, service marks, patents, trade secrets, copyrights or other intellectual property rights of The Foundry or any third party with intellectual property rights in the Software (each, a "Third Party Licensor"). Furthermore, for purposes of this Section 2, the term "Software" shall include any derivatives of the Software.

Licensee shall install and use only a single copy of the Software on one computer, unless the Software is installed in a "floating license" environment, in which case Licensee may install the Software on more than one computer; provided, however, Licensee shall not at any one time use more copies of the Software than the total number of valid Software licenses purchased by Licensee.

Furthermore, the Software can be licensed on an "interactive" or "non-interactive" basis. Licensee shall be authorized to use a non-interactive version of the Software for rendering purposes only (i.e., on a CPU, without a user, in a non-interactive capacity) and shall not use such Software on workstations or otherwise in a user-interactive capacity. Licensee shall be authorized to use an interactive version of the Software for both interactive and non-interactive rendering purposes, if available.

Finally, if the Software is an "Educational Version," Licensee may use it only for the purpose of training and instruction, and for no other purpose. Educational Versions of the Software may not be used for commercial, professional or for-profit purposes.

Section 3. Back-Up Copy.

Notwithstanding Section 2, Licensee may store one copy of the Software and Documentation off-line and off-site in a secured location owned or leased by Licensee in order to provide a back-up in the event of destruction by fire, flood, acts of war, acts of nature, vandalism or other incident. In no event may Licensee use the back-up copy of the Software or Documentation to circumvent the usage or other limitations set forth in this Agreement.

Section 4. Ownership.

Licensee acknowledges that the Software and Documentation and all intellectual property rights relating thereto are and shall remain the sole property of The Foundry and the Third Party Licensors. Licensee shall not remove, or allow the removal of, any copyright or other proprietary rights notice included in and on the Software or Documentation or take any other action that could adversely affect the property rights of The Foundry or any Third Party Licensor. To the extent that Licensee is authorized to make copies of the Software or Documentation under this Agreement, Licensee shall reproduce in and on all such copies any copyright and/or other proprietary rights notices provided in and on the materials supplied by The Foundry hereunder. Nothing in this Agreement shall be deemed to give Licensee any rights in the trademarks, service marks, patents, trade secrets, copyrights or other intellectual property rights of The Foundry or any Third Party Licensor, and Licensee shall be strictly prohibited from using the name, trademarks or service marks of The Foundry or any Third Party Licensor in Licensee's promotion or publicity without The Foundry's express written approval.

Section 5. License Fee.

Licensee understands that the benefits granted to Licensee hereunder are contingent upon Licensee's payment in full of the license fee payable in connection herewith (the "License Fee").

Section 6. Taxes and Duties.

Licensee agrees to pay, and indemnify The Foundry from claims for, any local, state or national tax (exclusive of taxes based on net income), duty, tariff or other impost related to or arising from the transaction contemplated by this Agreement.

Section 7. Limited Warranty.

The Foundry warrants that, for a period of ninety (90) days after delivery of the Software: (a) the machine readable electronic files constituting the Software and Documentation shall be free from errors that may arise from the electronic file transfer from The Foundry and/or its authorized reseller to Licensee; and (b) to the best of The Foundry's knowledge, Licensee's use of the Software in accordance with the Documentation will not, in and of itself, infringe any third party's copyright, patent or other intellectual property rights. Except as warranted, the Software and Documentation is being provided "as is." THE FOREGOING LIMITED WARRANTY IS IN LIEU OF ALL OTHER WARRANTIES OR CONDITIONS, EXPRESS OR IMPLIED, AND The Foundry DISCLAIMS ANY AND ALL IMPLIED WARRANTIES OR CONDITIONS, INCLUDING, WITHOUT LIMITATION, ANY IMPLIED WARRANTY OF TITLE, NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE, REGARDLESS OF WHETHER The Foundry KNOWS OR HAS REASON TO KNOW OF LICENSEE'S PARTICULAR NEEDS. The Foundry does not warrant that the Software or Documentation will meet Licensee's requirements or that Licensee's use of the Software will be uninterrupted or error free. No employee or agent of The Foundry is authorized to modify this limited warranty, nor to make additional warranties. No action for any breach of the above limited warranty may be commenced more than one (1) year after Licensee's initial receipt of the Software. To the extent any implied warranties may not be disclaimed under applicable law, then ANY IMPLIED WARRANTIES ARE LIMITED IN DURATION TO NINETY (90) DAYS AFTER DELIVERY OF THE SOFTWARE TO LICENSEE.

Section 8. Limited Remedy.

The exclusive remedy available to the Licensee in the event of a breach of the foregoing limited warranty, TO THE EXCLUSION OF ALL OTHER REMEDIES, is for Licensee to destroy all copies of the Software, send The Foundry a written certification of such destruction and, upon The Foundry's receipt of such certification, The Foundry will make a replacement copy of the Software available to Licensee.

Section 9. Indemnification.

Licensee agrees to indemnify, hold harmless and defend The Foundry and The Foundry's affiliates, officers, directors, shareholders, employees, authorized resellers, agents and other representatives (collectively, the "Released Parties") from all claims, defense costs (including, but not limited to, attorneys' fees), judgments, settlements and other expenses arising from or connected with the operation of Licensee's business or Licensee's possession or use of the Software or Documentation.

Section 10. Limited Liability.

In no event shall the Released Parties' cumulative liability to Licensee or any other party for any loss or damages resulting from any claims, demands or actions arising out of or relating to this Agreement (or the Software or Documentation contemplated herein) exceed the License Fee paid to The Foundry or its authorized reseller for use of the Software. Furthermore, IN NO EVENT SHALL THE RELEASED PARTIES BE LIABLE TO LICENSEE UNDER ANY THEORY FOR ANY INDIRECT, SPECIAL, INCIDENTAL, PUNITIVE, EXEMPLARY OR CONSEQUENTIAL DAMAGES (INCLUDING DAMAGES FOR LOSS OF BUSINESS OR LOSS OF PROFITS) OR THE COST OF PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES, REGARDLESS OF WHETHER THE RELEASED PARTIES KNOW OR HAVE REASON TO KNOW OF THE POSSIBILITY OF SUCH DAMAGES AND REGARDLESS OF WHETHER ANY REMEDY SET FORTH HEREIN FAILS OF ITS ESSENTIAL PURPOSE. No action arising out of or related to this Agreement, regardless of form, may be brought by Licensee more than one (1) year after Licensee's initial receipt of the Software; provided, however, to the extent such one (1) year limit may not be valid under applicable law, then such period shall be limited to the shortest period allowed by law.

Section 11. Term; Termination.

This Agreement is effective upon Licensee's acceptance of the terms hereof (by clicking on the "Accept" button) and Licensee's payment of the License Fee, and the Agreement will remain in effect until termination. If Licensee breaches this Agreement, The Foundry may terminate the License granted hereunder by notice to Licensee. In the event the License is terminated, Licensee will either return to The Foundry all copies of the Software and Documentation in Licensee's possession or, if The Foundry directs in writing, destroy all such copies. In the latter case, if requested by The Foundry, Licensee shall provide The Foundry with a certificate signed by an officer of Licensee confirming that the foregoing destruction has been completed.

Section 12. Confidentiality.

Licensee agrees that the Software and Documentation are proprietary and confidential information of The Foundry and that all such information and any communications relating thereto (collectively, "Confidential Information") are confidential and a fundamental and important trade secret of The Foundry. Licensee shall disclose Confidential Information only to Licensee's employees who are working on an Authorized Project and have a "need-to-know" such Confidential Information, and shall advise any recipients of Confidential Information that it is to be used only as authorized in this Agreement. Licensee shall not disclose Confidential Information or otherwise make any Confidential Information available to any other of Licensee's employees or to any third parties without the express written consent of The Foundry. Licensee agrees to segregate, to the extent it can be reasonably done, the Confidential Information from the confidential information and materials of others in order to prevent commingling. Licensee shall take reasonable security measures, which such measures shall be at least as great as the measures Licensee uses to keep Licensee's own confidential information secure (but in any case using no less than a reasonable degree of care), to hold the Software, Documentation and any other Confidential Information in strict confidence and safe custody. The Foundry may request, in which case Licensee agrees to comply with, certain reasonable security measures as part of the use of the Software and Documentation.

Licensee acknowledges that monetary damages may not be a sufficient remedy for unauthorized disclosure of Confidential Information, and that The Foundry shall be entitled, without waiving any other rights or remedies, to such injunctive or equitable relief as may be deemed proper by a court of competent jurisdiction.

Section 13. Inspection.

Licensee shall advise The Foundry on demand of all locations where the Software or Documentation is used or stored. Licensee shall permit The Foundry or its authorized agents to inspect all such locations during normal business hours and on reasonable advance notice.

Section 14. Nonsolicitation.

Licensee agrees not to solicit for employment or retention, and not to employ or retain, any of The Foundry's current or future employees who were or are involved in the development and/or creation of the Software.

Section 15. U.S. Government License Rights.

The Software, Documentation and/or data delivered hereunder are subject to the terms of this Agreement and in no event shall the U.S. Government acquire greater than RESTRICTED/LIMITED RIGHTS. At a minimum, use, duplication or disclosure by the U.S. Government is subject to the applicable restrictions of: (i) FAR §52.227-14 ALTS I, II and III (June 1987); (ii) FAR §52.227-19 (June 1987); (iii) FAR §12.211 and 12.212; and/or (iv) DFARS §227.7202-1(a) and DFARS §227.7202-3.

The Software is the subject of the following notices:

* Copyright (c) 2010 The Foundry Visionmongers, Ltd. All Rights Reserved.

* Unpublished-rights reserved under the Copyright Laws of the United Kingdom.

Section 16. Survival.

Sections 2, 4, 5, 6, 7, 8, 9, 10, 12, 13, 14, 15, 16, 17 and 18 shall survive any termination or expiration of this Agreement.

Section 17. Import/Export Controls.

To the extent that any Software made available hereunder is subject to restrictions upon export and/or reexport from the United States, Licensee agrees to comply with, and not act or fail to act in any way that would violate, the applicable international, national, state, regional and local laws and regulations,

including, without limitation, the United States Foreign Corrupt Practices Act, the Export Administration Act and the Export Administration Regulations, as amended or otherwise modified from time to time, and neither The Foundry nor Licensee shall be required under this Agreement to act or fail to act in any way which it believes in good faith will violate any such laws or regulations.

SECTION 18. MISCELLANEOUS.

This Agreement is the exclusive agreement between the parties concerning the subject matter hereof and supersedes any and all prior oral or written agreements, negotiations, or other dealings between the parties concerning such subject. This Agreement may be modified only by a written instrument signed by both parties. If any action is brought by either party to this Agreement against the other party regarding the subject matter hereof, the prevailing party shall be entitled to recover, in addition to any other relief granted, reasonable attorneys' fees and expenses of litigation. Should any term of this Agreement be declared void or unenforceable by any court of competent jurisdiction, such declaration shall have no effect on the remaining terms of this Agreement. The failure of either party to enforce any rights granted hereunder or to take action against the other party in the event of any breach hereunder shall not be deemed a waiver by that party as to subsequent enforcement of rights or subsequent actions in the event of future breaches. This Agreement shall be governed by, and construed in accordance with English Law.

Copyright (c) 2010 The Foundry Visionmongers Ltd. All Rights Reserved. Do not duplicate.

INDEX

A-Z

A

- adding
 - grain 45
 - motion blur 41
- Align 14
- aligning
 - clips spatially 14
- alignment tools 13
- analysing global motion estimation effects 66

B

- backward vector fields 70
- bugs
 - fixed 73
 - known 73

C

- camera motion
 - removing 60
 - stabilising 60
- colour matching 38
- contacting The Foundry 9
- contexts 9
- customer support 9

D

- DeFlicker2 19
- DeGrain 21
- DeNoise 26

E

- end user license agreement 77

F

- film grain
 - adding 45
 - removing 21
 - stocks 47
- flicker
 - reducing 19
- forward vector fields 70

G

- global motion effects 65
- global motion estimation 65
 - limitations 66
- GME 65
- grading tools 13
- grain
 - adding 45
 - removing 21, 26
 - stocks 47
- grain management tools 11

I

- improvements 73
- install directory 7
- installation 6
- introduction 6

K

- known bugs 73
- Kronos 30

L

- licensing 7
- limitations
 - of global motion estimation 66
- LME 70
- local motion estimation 70

M

- MatchGrade 38
- motion
 - stabilising 60
- motion blur
 - adding 41
- motion estimation
 - global 65
 - local 70
- MotionBlur 41

N

- new features 73
- noise
 - removing 26

O

- occlusions in rig removal 56

- OFX plug-ins 6
- on-screen tools 8
- overview
 - of the plug-ins 11

P

- plug-in contexts 9
- proxy resolution warnings 49

R

- ReGrain 45
- release dates 73
- release notes 73
- removing
 - camera motion 60
 - flicker 19
 - grain 21, 26
 - noise 26
 - rigs 53, 65
 - unwanted objects 12, 53, 65
- retiming 30
- retiming tools 12
- rig removal 53, 65
- RigRemoval 53

S

- slowing down footage 30
- spatial alignment of clips 14
- speeding up footage 30
- stabilisation tools 13
- stabilising footage 60
- Steadiness 60
- system requirements 73

T

- The Foundry
 - contacting 9
- The Foundry products 9

U

- unwanted objects
 - removing 53, 65

V

- vector fields 70
 - backward 70
 - forward 70

W

web site 9

www.thefoundry.co.uk 9, 10