

# OLM Open Tools OLM Keylight User Manual

OLM Digital, Inc.



**Table of Contents**

<b>Introduction</b>	<b>4</b>
Supported Environment	4
Install	4
<b>Using OLM Keylight</b>	<b>4</b>
<b>OLM Keylight parameters</b>	<b>5</b>
<b>Other</b>	<b>6</b>

## **Copyright, End User License Agreement**

OLM Open Tools are licensed under the Apache License, Version 2.0.

## **Change Logs**

2020/11/04     First Version

Figure 1

*Tears of Steel* (CC) Blender Foundation | [mango.blender.org](http://mango.blender.org)

## 1. Introduction

OLM Keylight is a light reimplementation of The Foundry® Nuke® plug-in Keylight for color keying.

### 1.1. Supported Environment

Please refer to OLM Open Tools [website](#) for a complete list of supported environments.

### 1.2. Install

The zip archive you can download from OLM OpenTools web site contains the files listed in Table 1. To install OLM Keylight copy the plug-in file (for windows: OLMKeylight.dll, for Linux:OLMKeylight.so) in a directory registered in the environment variable NUKE\_PATH (example: for windows: C:\Users\USER\_NAME\.nuke, for Linux: ~/.nuke)

Folder	File Name	Explanation
Plugins\[Linux/Windows] \[Version]	OLMKeylight.so OLMKeylight.dll	The plug-in files.
doc	OLMKeylight UserManual.pdf	The user manual for OLM Keylight

**Table 1 : Content of the zip archive**

## 2. Using OLM Keylight

To create an OLMKeylight node, use the menu: (Others)→(All plugins)→(Update). Then you can use (Others)→(All plugins)→(OLM) →(OLMKeylight) to create the node, or press TAB key and type OLMKeylight. It is up to you to create a menu.py and/or init.py file to set up the plugin automatically.

Once the node created you can setup a simple node graph with some input file, OLM Keylight node and a viewer like in Figure 1



**Figure 1 OLM Keylight node network**

Be aware that to work properly OLM Keylight requires inputs in linear color space.

### 3. OLM Keylight parameters

OLM Keylight parameters are as follows (Figure 2)

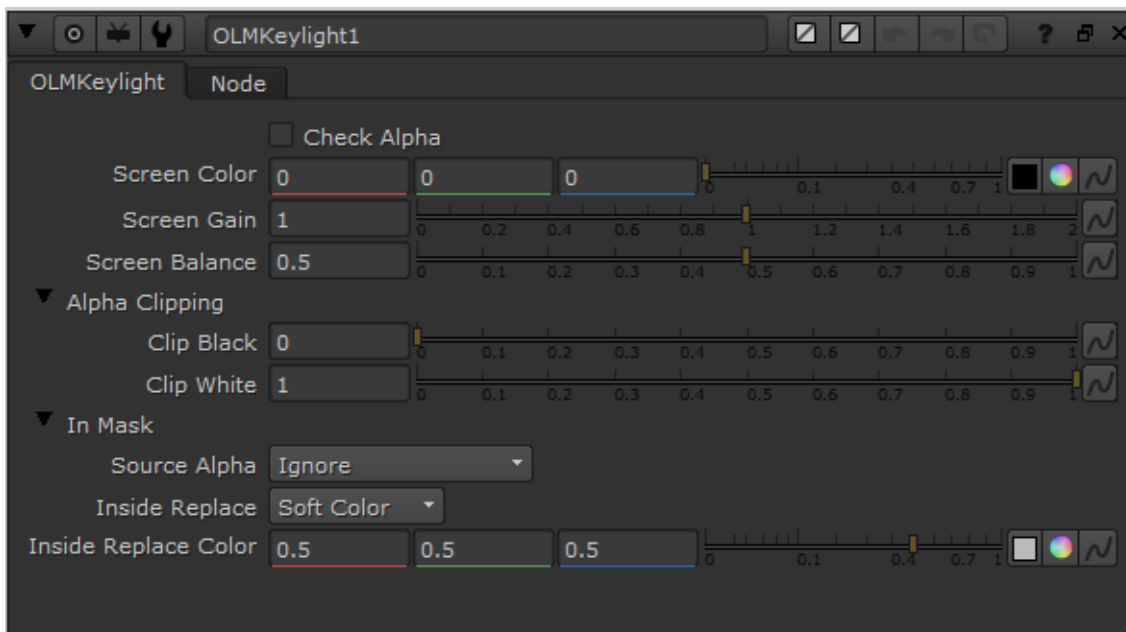


Figure 2 OLM Keylight parameters

- **Check Alpha** — When this parameter is set to true, the output color components (R, G, B) are all set to the computed alpha value.
- **Screen Color** — This parameter is the color to be keyed, usually green for a green screen background or blue for a blue screen background.
- **Screen Gain** — This value represents how much of the screen color is removed. The higher it is, the more the screen color is removed..
- **Screen Balance** — This value controls the weight of the two minor color components when computing the saturation of the major component. A good value for a green screen is 0.5.
- **Alpha clipping**
  - **Clip Black** — Below this threshold, the alpha value is set to 0.
  - **Clip White** — Above this threshold, the alpha value is set to 1.
- **Source Alpha** — This parameter controls how the alpha value of the input is used in the keying process.
  - **Ignore** means input alpha is not used. Alpha is 1.
  - **Add to Inside Mask** or **Normal** means input alpha is used as it.
- **Inside Replace** — only valid if **Source Alpha** is **Add to Inside Mask** or **Normal**. When the computed alpha value is smaller than the input alpha, then the output color is modified accordingly to this parameter value.
  - **None** means no modifications.
  - **Source** means that the color of the input is blended with the color of the output.
  - **Hard Color** means that **Inside Replace Color** is added to the output color.
  - **Soft Color** means that **Inside Replace Color** is added to the output color, but with some adjustments to match the luminance of the original pixel color.
- **Inside Replace Color** — This parameter is the color that is used in the **Hard Color**

and **Soft Color** modes of the **Inside Replace** parameter.

#### 4. Other

If you have problem using the plug-in in any environment, if you find a bug, have a feature request, or for any kind of feedback feel free to contact us at the following mail address:  
[opentools@olm.co.jp](mailto:opentools@olm.co.jp)