

# The Theory of OpenEXR Deep Samples

Peter Hillman  
Weta Digital Ltd

November 10, 2013

## Abstract

This document derives the techniques for splitting and combining two non-solid samples of equal depth and thickness. These should be used by deep image ‘flattening’ algorithms to compute the combined colour of two samples. The formulas are defined in the document “Interpreting OpenEXR Deep Pixels” by Florian Kainz. This document derives those formulas, and is for information only.

## Contents

<b>1</b>	<b>Sample model</b>	<b>2</b>
1.1	Alpha vs Transparency . . . . .	2
1.2	Computing $t$ . . . . .	3
1.3	Sample properties . . . . .	3
<b>2</b>	<b>Sample splitting</b>	<b>3</b>
2.1	Alpha of a subsample . . . . .	3
2.2	Colour of a subsample . . . . .	4
2.3	Splitting transparent samples . . . . .	5
<b>3</b>	<b>Sample merging</b>	<b>6</b>
3.1	Merging when one sample is transparent . . . . .	7
3.2	Merging two transparent samples . . . . .	7
3.3	Merging solid samples . . . . .	7
<b>4</b>	<b>Summary</b>	<b>8</b>

## Definitions

$r$  ratio of length of original sample to a subsample:  $0 \leq r \leq 1$

$C_i$  Colour of sample  $i$

$C'_i$  Colour of a subsample of  $i$

$\alpha_i$  alpha value of sample  $i$

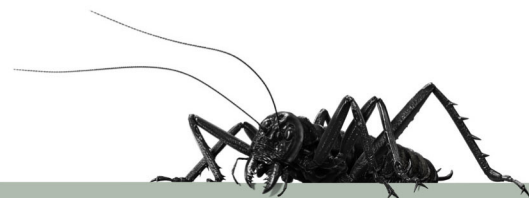
$\alpha'_i$  alpha value of a subsample of  $i$

$T_i$  transparency/transmission of sample  $i$ ;  $T_a = 1 - \alpha_a$

$c_i$  colour per unit length (‘instantaneous colour’) of  $i$

$t_i$  transparency per unit length (‘intense transparency’) of  $i$

Subscripts  $a$  and  $b$  refer to the input samples; subscript  $c$  refers to the combined output sample



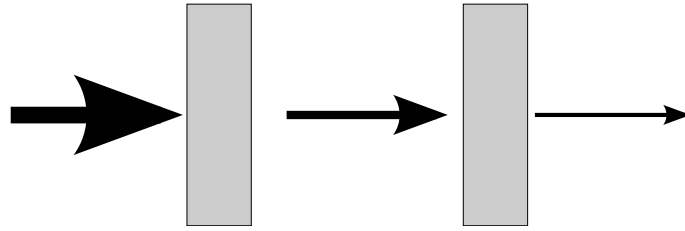


Figure 1: Light passing through multiple objects from left to right. Each object attenuates a fraction  $t$  of the light

## 1 Sample model

Many deep compositing operations require subdividing a sample into two subsamples, or else clipping a sample at a given point. When this happens, the alpha value of the subsample(s) must be recomputed, and the colour will also change. To derive an equation for computing the new alpha, the amount of light attenuated by each subsample must be considered.

Light passing through a light absorbing object will be attenuated according to the *transparency*  $t$  of the object. Thus, if  $i$  is the amount of light entering the object, the light  $o$  leaving the object is given by

$$o = t \cdot i \quad (1)$$

Where light passes through two objects, as in Figure 1, the light is attenuated twice (the output from the first is fed into second)

$$o = t_2 \cdot (t_1 \cdot i) \quad (2)$$

where  $t_2$  and  $t_1$  are the transmission of the individual samples. In general, if the transmission  $t$  of all objects are the same, the light emerging from  $n$  adjacent samples is given by the *Beer-Lambert* law:

$$o = i \cdot t^n \quad (3)$$

$$T = \frac{o}{i} = t^n \quad (4)$$

where  $T$  is the total transmission. If  $t$  is considered the *transmission per unit length* and  $n$  the length, the equation is continuous, applying also for non-integer values of  $n$ <sup>1</sup>.

An OpenEXR deep sample can then be modelled as an object of a specified length  $n$  which absorbs light according to equation (4).

### 1.1 Alpha vs Transparency

OpenEXR images use alpha  $\alpha$  instead of transparency.  $T = 1 - \alpha$ : if  $\alpha = 1$  (so  $T = 0$ ) then all light is absorbed; if  $\alpha = 0$  (so  $T = 1$ ), then all light is transmitted and the material is transparent. Substituting  $T$  into (1) and (2) gives the *screen* compositing operation:

$$\alpha_c = 1 - (1 - \alpha_1)(1 - \alpha_2) \quad (5)$$

where  $\alpha_c$  is the combined alpha value, and  $\alpha_1$  and  $\alpha_2$  are the alpha values of the two samples.

Throughout this document,  $T$  is preferred to  $\alpha$  where it gives simpler equations.

<sup>1</sup>Various forms of the Beer-Lambert law exist, using different constants. For example,  $o = i \cdot 10^{-\epsilon l c}$ , where  $\epsilon$  and  $c$  are the molar absorptivity and concentration respectively. This is equivalent to (4) if  $t = 10^{-\epsilon c}$  and  $l = n$



## 1.2 Computing $t$

OpenEXR images samples store the  $\alpha$  of the entire sample. This gives us the total transmission  $T$  of the sample, not the transmission per unit length  $t$ . Rearranging (4) gives

$$t = T^{-n} \quad (6)$$

## 1.3 Sample properties

The following are assumptions made by the formulas derived here:

- Samples have **constant optical density and colour**: if a sample has length  $n$  and a subsample of length  $n'$  is extracted from it, the RGBA colour of the subsample will be the same regardless from where in the sample it is extracted. In particular, if a sample is split into  $k$  subsamples of equal length, each subsample will have the same RGBA colour.
- Sample attenuation is **non-scattering** and **pixel independent**: light travelling through the sample is either absorbed or transmitted; it is not reflected back down the sample or scattered into neighbouring samples. Scattering causes point lights to appear blurred when passing through fog, and also tends to make fog look more optically dense than it really is, since detail is lost very quickly, even though light is being transmitted. This effect is not modelled with OpenEXR volumetric samples, and must be approximated by including the scattered light within the volume or applied as a post-process. Since the light attenuation profile throughout a pixel will not follow the Beer-Lambert equation (4), extra samples must be used to model the pixel.
- Sample behaviour is **unit** and **scale independent**: if a sample is divided into  $k$  subsamples of equal length, then scaling the depth channels  $z_{\text{front}}$  and  $z_{\text{back}}$  of the deep image will not change the colour of the samples. When merging two deep images, prescaling the depth of each image by the same amount then merging the images is identical to merging the original images then scaling the depth of the result. This scaling property allows any unit to be used to store depth, and the unit to be changed by scaling the depth channels without modification of the RGBA channels.
- Sample behaviour is **position independent**: moving a sample in depth will not change the RGBA values obtained by subdividing it. If a sample is 10 units long and a subsample of 2 units extracted from it, the subsample will have the same RGBA values regardless of the position in depth of the sample. Shifting two images by adding a constant  $c$  to the depth channels of each image then merging them is identical to merging the original images then shifting the result.
- Sample splitting is **non-destructive**: using the equations derived here, the samples of a pixel can be arbitrarily subdivided. Recombining the subsamples with the *over* compositing operation will yield the RGBA colour of the original sample (assuming sufficient numerical precision).

## 2 Sample splitting

### 2.1 Alpha of a subsample

When a subsample of length  $n'$  is extracted from an original sample of length  $n$ , the alpha value  $\alpha'$  for subsample must be computed.

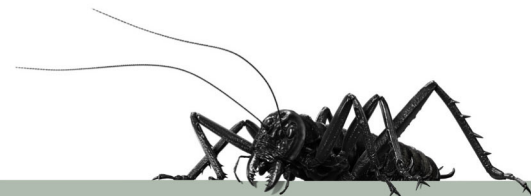




Figure 2: Model for a sample with colour and alpha. The grey box is light absorbent material with transparency  $t$ ; the red dots are light sources each of which has colour  $c$ .

The total transmission  $T' = t^{n'}$ . Substituting for  $t$  computed for the entire sample length using (6) gives:

$$\begin{aligned} T' &= (T^{-n})^{n'} \\ T' &= T^{\frac{n'}{n}} \end{aligned} \quad (7)$$

Here,  $n$  refers to the *length* of the original sample, but samples are specified in OpenEXR with a front and back depth. Thus, if a sample  $T$  with front and back depths  $z_{\text{front}}$  and  $z_{\text{back}}$  is split at point  $z$  in space, the transmission  $T'$  of the front subsample is given by

$$T' = T^{\frac{z - z_{\text{front}}}{z_{\text{back}} - z_{\text{front}}}} \quad (8)$$

$$\alpha' = 1 - (1 - \alpha)^{\frac{z - z_{\text{front}}}{z_{\text{back}} - z_{\text{front}}}} \quad (9)$$

Since  $(z - z_{\text{front}})/(z_{\text{back}} - z_{\text{front}})$  is the *ratio of the original sample length to the extracted sample length*, we can substitute this ratio for  $r$  for the remainder. This is equivalent to considering subdivision of a sample of unit length.

## 2.2 Colour of a subsample

A sample with RGB information is modelled as a cylinder. The alpha channel(s) of the sample are represented by some non-scattering, absorbent material within it, which attenuates the light passing through the sample. The colour channel(s) of the sample is represented by infinitely many light sources, which emit coloured light (but don't absorb anything), with identical intensities  $c$ , evenly spaced through the sample, as shown in Figure 2.

A light source at distance  $x$  into the sample will be attenuated by the amount of absorber between it and the front of the sample. According to Beer-Lambert, the total transmission of that part of the sample  $t$  is given by

$$t' = T^x \quad (10)$$

$$\alpha' = 1 - (1 - \alpha)^x \quad (11)$$

The light which reaches the front of the sample from the individual light source at distance  $x$  is  $cT^x$ . If we had a finite number  $N$  sources, the total light reaching the front of the sample would be:

$$C = \sum_{k=0}^N cT^{k/N} \quad (12)$$



where  $k/N$  is the position of light  $k$ , equal to  $x$ . As  $N$  tends to infinity, this becomes

$$\mathbf{C} = \int_0^1 \mathbf{c} T^x \, dx \quad (13)$$

$$\begin{aligned} &= \left. \frac{\mathbf{c} T^x}{\log(T)} \right|_{x=0}^{x=1} \\ &= \frac{\mathbf{c}(T - 1)}{\log(T)} \end{aligned} \quad (14)$$

The value  $\mathbf{C}$  is the sample's RGB value, as stored in the OpenEXR image. Rearranging (14) gives us the colour of each light (or perhaps the 'instantaneous colour' of the sample)

$$\mathbf{c} = \frac{\mathbf{C} \log(T)}{T - 1} \quad (15)$$

If the entire sample is reduced to a subsample length  $r$ , only a subsection of lights are included, and we must compute the new RGB colour  $\mathbf{C}'$  for the subsample. Hence:

$$\begin{aligned} \mathbf{C}' &= \int_0^r \mathbf{c} T^x \, dx \\ &= \frac{\mathbf{c}(T^r - 1)}{\log(T)} \end{aligned}$$

Substituting for  $\mathbf{c}$  from (15) gives

$$\begin{aligned} \mathbf{C}' &= \frac{\mathbf{C} \log(T)}{T - 1} \frac{(T^r - 1)}{\log(T)} \\ &= \mathbf{C} \frac{(T^r - 1)}{T - 1} \end{aligned}$$

Noting that  $T - 1 = -\alpha$  and  $T^r - 1 = T' - 1 = -\alpha'$  and multiplying top and bottom by  $-1$  gives:

$$\mathbf{C}' = \mathbf{C} \frac{\alpha'}{\alpha} \quad (16)$$

This is exactly equivalent to **unpremultiplying** the colour by the original alpha value, computing the new alpha value, and **premultiplying** by the new value.

## 2.3 Splitting transparent samples

When  $\alpha = 0$  (or alternatively,  $T = 1$ ), equation (16) requires a division by zero. A special case is derived for  $\alpha = 0$ . The sample model of Figure 2 with the colour provided by many light sources still applies, but there is no attenuation. Therefore, each light contributes equally to the observed colour  $\mathbf{C}$ . The 'instantaneous' colour equals the final colour, (assuming sample length is normalised to 1)

$$\mathbf{C} = \int_0^1 \mathbf{c} \, dx \quad (17)$$

$$\mathbf{C} = \mathbf{c} \quad (18)$$



A subsection  $C'$  of length  $r$  is given by

$$\begin{aligned} C' &= \int_0^r c \, dx \\ C' &= rC \end{aligned} \tag{19}$$

That is, if  $\alpha = 0$ , then scaling a sample's length by  $r$  scales the colour by the same amount.

### 3 Sample merging

Now, consider combining two samples  $a$  and  $b$  together. This operation is required when “tidying” a deep sample list, which is essential before flattening a deep image into a regular one. We assume that  $z_{\text{front}_a} = z_{\text{front}_b}$  and  $z_{\text{back}_a} = z_{\text{back}_b}$  so the lengths of each sample are the same, and they fully overlap in space. If this is not the case, the samples should be subdivided and those subsamples merged separately as described in the document “Interpreting OpenEXR Deep Samples”

Again, the sample model from Figure 2 is used. However, at each location, there are now two light sources,  $c_a$  and  $c_b$ , and the light is attenuated by both  $t_a$  and  $t_b$ . We can treat this as being attenuated first by  $t_b$ , then by  $t_a$ , so the total light reaching the end of the sample is:

$$c_c = (c_a + c_b)(T_a T_b)^x \tag{20}$$

By substitution into (14) and then substituting for  $c_a$  and  $c_b$  from (15) gives

$$\begin{aligned} C_c &= \frac{(c_a + c_b)(T_a T_b - 1)}{\log(T_a T_b)} \\ C_c &= \frac{\left( \frac{C_a \log(T_a)}{T_a - 1} + \frac{C_b \log(T_b)}{T_b - 1} \right) (T_a T_b - 1)}{\log(T_a T_b)} \\ C_c &= \frac{\left( \frac{C_a \log(1 - \alpha_a)}{-\alpha_a} + \frac{C_b \log(1 - \alpha_b)}{-\alpha_b} \right) ((1 - \alpha_a)(1 - \alpha_b) - 1)}{\log((1 - \alpha_a)(1 - \alpha_b))} \end{aligned}$$

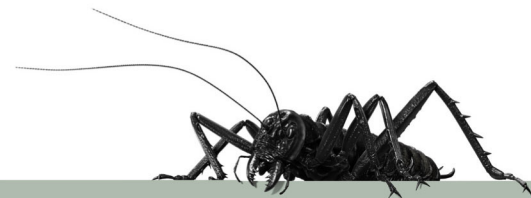
Multiplying both parts of the top line by -1 gives

$$C_c = \frac{\left( \frac{C_a \log(1 - \alpha_a)}{\alpha_a} + \frac{C_b \log(1 - \alpha_b)}{\alpha_b} \right) (1 - (1 - \alpha_a)(1 - \alpha_b))}{\log((1 - \alpha_a)(1 - \alpha_b))} \tag{21}$$

Note that the combined transmission is given by  $T_c = T_a T_b$ , implying that the combined alpha follows the screen equation (5). Substituting into both top and bottom of (21) gives the formula for final colour:

$$C_c = \frac{\left( \frac{C_a \log(1 - \alpha_a)}{\alpha_a} + \frac{C_b \log(1 - \alpha_b)}{\alpha_b} \right) \alpha_c}{\log(1 - \alpha_c)} \tag{22}$$

This is the **premultiplied** combined colour. For the unpremultiplied colour, the  $\alpha_c$  term can be omitted. This gives the transmission weighted average of the unpremultiplied input colours.



### 3.1 Merging when one sample is transparent

Now suppose one of the samples (say,  $b$ ) has no alpha, so  $\alpha_b = 0$  and  $T_b = 1$ . From (18),  $\mathbf{c}_b = \mathbf{C}_b$ . The combined colour  $\mathbf{C}_c$  is attenuated according to  $T_a$ , the transmission of  $a$  alone, and the combined colour is given by:

$$\mathbf{c}_c = (\mathbf{c}_a + \mathbf{C}_b)(T_a)^x$$

and substituting into (14) gives:

$$\begin{aligned} \mathbf{C}_c &= \frac{(\mathbf{c}_a + \mathbf{C}_b)(T_a - 1)}{\log(T_a)} \\ &= \frac{\left(\frac{\mathbf{C}_a \log(1 - \alpha_a)}{-\alpha_a} + \mathbf{C}_b\right) ((1 - \alpha_a) - 1)}{\log(1 - \alpha_a)} \\ \mathbf{C}_c &= \frac{\left(\frac{\mathbf{C}_a \log(1 - \alpha_a)}{\alpha_a} - \mathbf{C}_b\right) \alpha_a}{\log(1 - \alpha_a)} \end{aligned} \quad (23)$$

### 3.2 Merging two transparent samples

Where both samples are transparent, the colours simply add together:

$$\mathbf{C}_c = \mathbf{C}_a + \mathbf{C}_b \quad (24)$$

### 3.3 Merging solid samples

For numerical stability, we must give a sensible value for  $\mathbf{C}_c$  when  $\alpha_b = 1$ . Equation (22) is undefined, since  $\log(1 - \alpha_b) = \log(0) = -\infty$ . We follow the initial model of light sources in an absorbing material, but now the absorbing material absorbs all light. Thus, we must be observing only its closest light source (that at  $x = 0$ ) in the solid sample: the sample will absorb light from its other sources. Thus, we can treat  $b$  as an infinitely thin ‘discrete’ sample at the front of the sample, which will block all light behind it; we can simply composite  $b$  over  $a$ . Therefore, if  $\alpha_b = 1$  and  $\alpha_a < 1$ , the total observed colour will be  $\mathbf{C}_b$ . Transposing  $a$  and  $b$  gives  $\mathbf{C}_c = \mathbf{C}_a$  if  $\alpha_a = 1$ .

Following the definitions above, when  $\alpha_a = 1$  and  $\alpha_b = 1$ , then  $\mathbf{C}_c = \mathbf{C}_a + \mathbf{C}_b$ . However, for stability, it makes sense to define

$$\mathbf{C}_c = \frac{\mathbf{C}_a + \mathbf{C}_b}{2} \quad (25)$$

This approach gives more stable results when there is random sampling error in the depth channel. Assume that the front depth of sample  $A$  differs by some small amount from that of  $B$ , so that  $z_a = \delta + z_b$ . Only the front sample will be visible: if  $\delta < 0$ , then  $\mathbf{C}_c = \mathbf{C}_a$ ; if  $\delta > 0$ , then  $\mathbf{C}_c = \mathbf{C}_b$ . Setting  $\mathbf{C}_c = \mathbf{C}_a + \mathbf{C}_b$  when  $\delta = 0$  gives an image which is (potentially) twice as bright. If  $\delta$  is random noise due to sampling, we will get a random pattern of bright pixels whenever  $\delta = 0$ . Using the mean colour seems a safer option. However, this is *not* associative: when combining three solid samples  $d, e$  and  $f$  does not give the same result as combining first  $d$  and  $e$ , then combining the result with  $f$ .



## 4 Summary

To **clip a sample** covering range  $z_{\text{front}} - z_{\text{back}}$  into a subsample  $z_{\text{front}} - z$ , or **extract a subsample** of length  $l'$  from a sample of length  $l$ :

$$r = \frac{z - z_{\text{front}}}{z_{\text{back}} - z_{\text{front}}} = \frac{l'}{l} \quad (26)$$

$$\alpha' = 1 - (1 - \alpha)^r \quad (27)$$

$$\mathbf{C}' = \begin{cases} r\mathbf{C}, & \text{if } \alpha = 0 \\ \mathbf{C} \frac{\alpha'}{\alpha}, & \text{otherwise} \end{cases} \quad (28)$$

where  $\alpha$  and  $\alpha'$  are the alpha values of the original sample and the subsample respectively, and  $\mathbf{C}$  and  $\mathbf{C}'$  are the colours of the original sample and the subsample respectively

To **merge two samples**  $a$  and  $b$  into a combined sample  $c$ :

$$\alpha_c = 1 - (1 - \alpha_a)(1 - \alpha_b) \quad (29)$$

$$\mathbf{C}_c = \begin{cases} \mathbf{C}_a + \mathbf{C}_b, & \text{if } \alpha_a = 0 \text{ and } \alpha_b = 0 \\ \mathbf{C}_a, & \text{if } \alpha_a = 1 \text{ and } \alpha_b < 1 \\ \mathbf{C}_b, & \text{if } \alpha_b = 1 \text{ and } \alpha_a < 1 \\ (\mathbf{C}_a + \mathbf{C}_b)/2, & \text{if } \alpha_a = 1 \text{ and } \alpha_b = 1 \\ \alpha_a \left( \frac{\mathbf{C}_a \log(1 - \alpha_a)}{\alpha_a} - \mathbf{C}_b \right) / \log(1 - \alpha_a), & \text{if } 0 < \alpha_a < 1 \text{ and } \alpha_b = 0 \\ \alpha_b \left( \frac{\mathbf{C}_b \log(1 - \alpha_b)}{\alpha_b} - \mathbf{C}_a \right) / \log(1 - \alpha_b), & \text{if } 0 < \alpha_b < 1 \text{ and } \alpha_a = 0 \\ \alpha_c \left( \frac{\mathbf{C}_a \log(1 - \alpha_a)}{\alpha_a} + \frac{\mathbf{C}_b \log(1 - \alpha_b)}{\alpha_b} \right) / \log(1 - \alpha_c), & \text{otherwise} \end{cases} \quad (30)$$

