

## Abstract

NeuS 和后面的 VolSDF 是同一时期 (Nips 2021) 的工作，也是使用可微分渲染来解决之前多视角重建需要 mask 的问题，两份工作也同样采用 SDF 作为几何表征。他们同样设法利用一个转换函数将 SDF (  $f(\mathbf{x}), \mathbf{x} = \mathbf{o} + \mathbf{d}t$  ) 转换成了体渲染中的权值 (  $w(t)$  ), 以实现：

$$C(\mathbf{o}, \mathbf{v}) = \int_0^{+\infty} w(t) c(\mathbf{p}(t), \mathbf{v}) dt$$

只是两者的实现不太一样。NeuS 是通过一个可学习的标准差 (  $s$  ) 来控制转换函数的带宽。一开始的带宽非常宽，利于学习到整体形状（低频信号），也便于优化。随着学习过程的深入，带宽逐渐变窄，越来越关注细节的优化（高频信号），Volume Rendering 越来越接近 Surface Rendering 的效果。

## Volume Rendering vs Surface Rendering

NeuS 里面说明了为什么基于 Volume Rendering (NeuS, VolSDF) 的多视角神经隐式场重建相比基于 Surface Rendering (IDR) 的多视角神经隐式场重建要好的一个重要原因是：基于 Surface Rendering 的方法仅关注其与表面的交点部分，而基于 Volume Rendering 的方法的样本是光线上的很多采样点，所以后者能够更合理和全面地对隐式场进行监督。NeuS 给下图所示的例子，以此来说明基于 Volume Rendering 的方法对这种突变 (abrupt) 深度能实现更好的拟合。

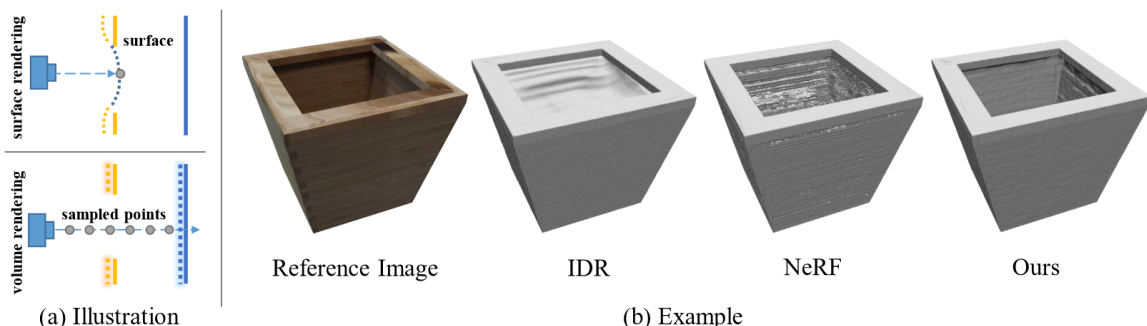


Figure 1: (a) Illustration of the surface rendering and volume rendering. (b) A toy example of bamboo planter, where there are occlusions on the top of the planter. Compared to the state-of-the-art methods, our approach can handle the occlusions and achieve better reconstruction quality.

这里要补充的一点是当前的绝大多数神经隐式 SDF 的表面重建方法会先利用 Sign agnostic learning of shapes from raw data 提出的几何初始化使得初始的 SDF 表征一个类球形表面，也就是提供一个先验，降低拟合的难度。优化的过程可视作是对这个类球形表面施加变形，使之接近目标物体。而 NeuS 想表达的意思，个人认为是基于 Volume Rendering 能够是这个变形更“深入”，因为它能够在当前表面的远处也产生监督，而 Surface Rendering 与之相比则极易容易陷入到当前表面附近的局部最优解。

## SDF（符号距离函数）

符号距离函数就是欧几里得上最近的物体距离，负数为在物体里面，0为在物体表面上，正数为在物体外面。

理论上，SDF的导数模长应该为1。为什么呢？

是因为 SDF 的定义基于空间中任意一点到物体表面的最近距离，换句话说，SDF 的值随着空间中实际距离的变化而变化。SDF 的梯度  $\nabla f(\mathbf{p})$  指向点  $\mathbf{p}$  到物体表面最近点的方向。更确切地说，SDF 的梯度是垂直于物体表面的，并且其方向指向物体的外部。

如果在 SDF 的梯度方向上移动一小段距离 $\Delta d$ ，那么 SDF 的值应该恰好变化 $\Delta f = \Delta d$ 。这种情况下，我们可以写出梯度的定义：

$$\Delta f \approx \nabla f(\mathbf{p}) \cdot \Delta \mathbf{p}$$

如果沿着梯度的方向 $\Delta \mathbf{p}$ 移动（即 $\Delta \mathbf{p}$ 和 $\nabla f(\mathbf{p})$ 是同方向的），则我们期望 SDF 的值  $f(\mathbf{p})$  的变化量和物理距离 $\Delta d$ 完全一致，即：

$$\Delta f = \Delta d$$

为了满足这个条件，SDF 的梯度 $\nabla f(\mathbf{p})$ 的模长必须为 1，即：

$$|\nabla f(\mathbf{p})| = 1$$

这意味着在空间中，SDF 的梯度不仅指向最短距离的方向，其大小也表示该方向上每单位距离的变化率。

从几何角度来看，SDF 的梯度在物体表面上的值是物体表面法线的方向，且法线的模长为 1。如果梯度的模长不为 1，表示 SDF 的值随物理距离的变化不准确，这会导致几何表面重建过程中产生误差。

尽管理论上 SDF 的梯度模长为 1，但在实际计算过程中（如数值方法、离散采样等），由于误差或近似，可能会出现梯度模长偏离 1 的情况。这种偏差在很多 SDF 相关的算法中通过正则化项（如 Eikonal 方程中的正则项）加以修正，以保持梯度模长尽可能接近 1。

例如，Eikonal 方程的正则项用于确保 SDF 的梯度保持模长为 1：

$$\mathcal{L}_{\text{reg}} = (\|\nabla f(\mathbf{p})\|_2 - 1)^2$$

这项损失函数用于约束模型输出的 SDF 使其梯度的模长接近 1。

所以SDF作为一个度量函数，为了保证其变化的均匀性，论文里令其梯度：

$$\|\nabla \phi(x)\| = 1$$

## logistic density function

---

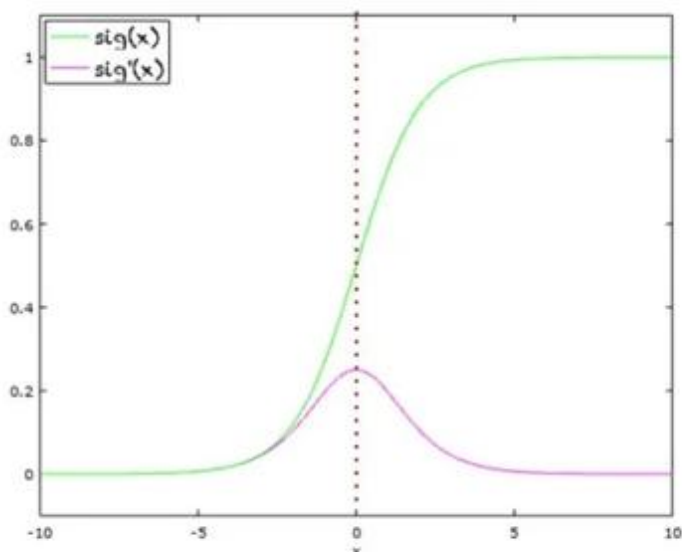
其实只要找一个在0处导数最大、并且满足原函数取值在0-1的函数就行

### sigmoid函数

$$\Phi_s(x) = \frac{1}{1+e^{-sx}}$$

### logistic density function

$$\phi(x) = \Phi'_s(x) = \frac{se^{-sx}}{(1+e^{-sx})^2}$$



Plot of  $\sigma(x)$  and its derivate  $\sigma'(x)$

Domain:  $(-\infty, +\infty)$

Range:  $(0, +1)$

$\sigma(0) = 0.5$

Other properties

$$\sigma(x) = 1 - \sigma(-x)$$

$$\sigma(x) = \frac{1}{1 + e^{-x}} = \frac{e^x}{e^x + 1}$$

$$\sigma'(x) = \sigma(x)(1 - \sigma(x))$$

知乎 @唐三三三

从上图可知， $\phi(x)$  为一个在0点对称的概率密度函数，在0点达到最大值，数据从0点开始衰减。

## Motivation

### 原始Nerf的密度（形状）估计

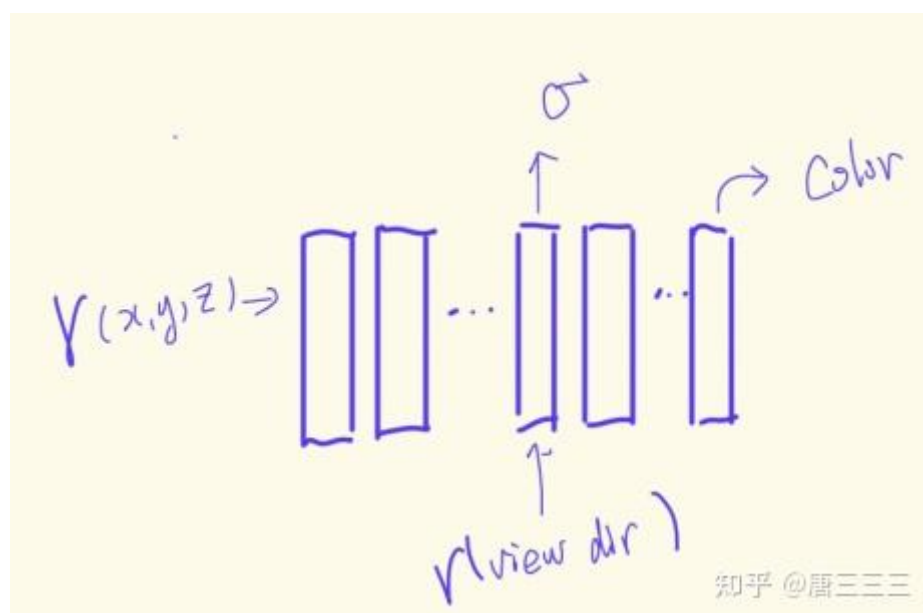
因为原Nerf网络是对密度和颜色的联合优化，虽然多视角生成任务效果还不错，但是其重建的外观通常不平滑和不准确，容易生成floaters现象，以后表面坑坑洼洼的，本论文就旨在解决这个问题。

他的想法和volsdf其实差不多，但具体实现方法有些区别，就是在密度估计方面加入一个先验信息，让网络知道这个“密度” $\sigma$  是跟物体外观有关的一个东西。

### 原始的Nerf

$$(x, y, z) \rightarrow \sigma$$

$$(x, y, z, \text{viewdir}) \rightarrow \text{color}$$



知乎 @唐三三三

# Neus (Naive Idea)

## SDF guided Volume Rendering

回到如何将 SDF 和体渲染结合起来。最直接的方式就是将 SDF 值  $f(x)$  过一个转换函数得到  $\sigma$  然后参与积分，因为我们直观上认为  $\sigma$  应当在表面即 SDF=0 的地方有较高的值，所以我们只要用一个尖峰偶函数来作为转换函数即可。而 NeuS 中用到的转换函数则是：

$$\phi_s(x) = \frac{d\Phi_s(x)}{dx} = \frac{s \exp(-sx)}{(1 + \exp(-sx))^2}, \quad \Phi_s(x) = (1 + \exp(-sx))^{-1}$$

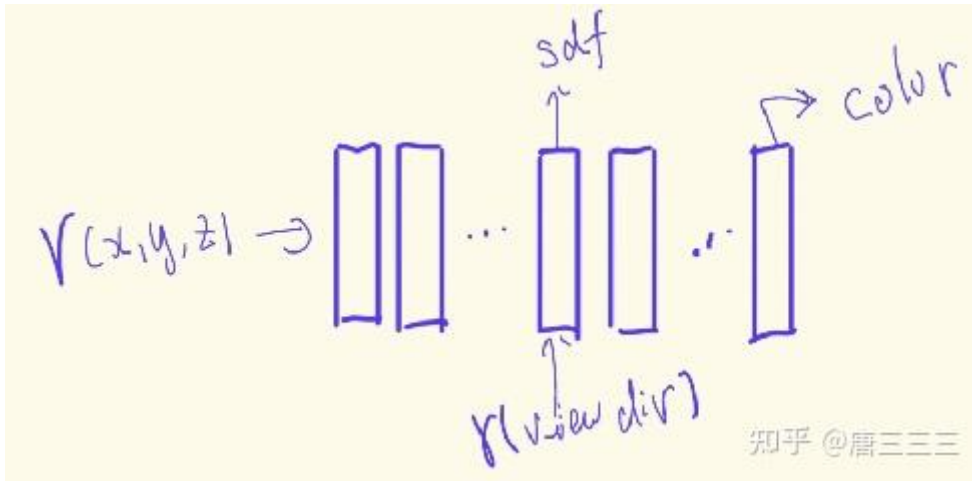
可以看到，这里的转换函数  $\phi_s(x)$  就是 Sigmoid 激活函数  $\Phi_s(x)$  的通用形式的导数，是一个偶函数，在负半轴单调递增，正半轴单调递减。

其中  $s$  是个学习参数，用来控制带宽， $s$  越大，也就是  $1/s$  越小 ( $1/s$  即  $\phi_s(x)$  其标准差)，带宽越窄； $s$  越接近零，则密度分布越接近冲激函数。

## 网络框架

$(x, y, z) \rightarrow sdf$

$(x, y, z, viewdir) \rightarrow color$



### 密度的计算

把预测出来的sdf值用  $f(p(t))$  来表示，其中  $p(t)$  表示采样到的点的  $(x, y, z)$  坐标。密度的表示可用 logistic density function来表示，即：

$$\sigma(t) = \phi(f(p(t)))$$

体渲染公式为：

$$C = \int T(t)c(t)\sigma(t)dx$$

其中：

$$T(t) = e^{-\int_0^t \sigma(s)dt}$$

但是，论文里提出这种Naive idea是有一些问题的。

我们设：

$$w(t) = T(t)\sigma(t)$$

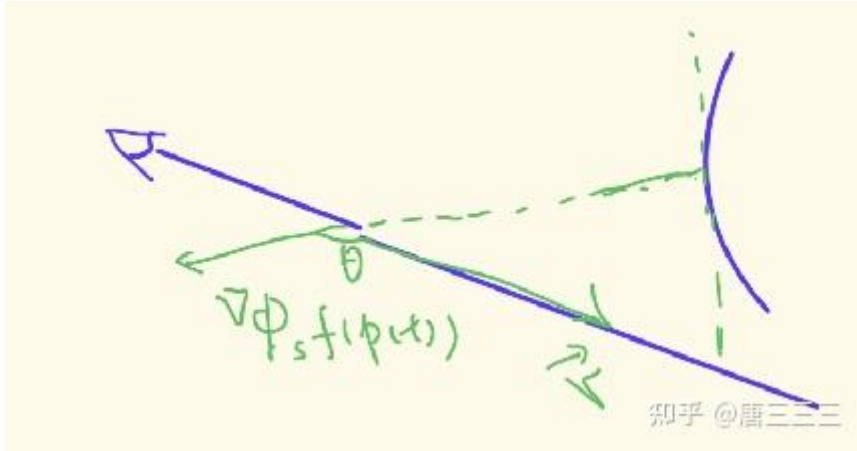
也就是这里的  $w(t)$  表示，空间中不同粒子点对最后像素点颜色的贡献值，一条射线出来，贡献点的最高点应该在与曲面交界上那一点的  $t^*$ ，也就是说：

$$\frac{dw}{dt} = 0 \text{ when } t = t^*$$

下面来算一下：

$$\begin{aligned}\frac{dw}{dt} &= \frac{dT(t)}{dt}\sigma(t) + T(t)\frac{d\sigma(t)}{dt} \\ &= -e^{-\int_0^t \sigma(s)dt}\sigma(t)\sigma(t) + T(t)\frac{d\sigma(t)}{dt} \\ &= T(t)\left(\frac{d\sigma(t)}{dt} - \sigma^2(t)\right)\end{aligned}$$

把符号距离函数的公式代入：



$$\begin{aligned}\frac{dw(t)}{dt} &= T(t^*)\left(\frac{d\phi_s(f(p(t)))}{dt} - \phi_s^2(f(p(t)))\right) \\ &= T(t^*)(\phi'_s(f(p(t)))(\nabla f(p(t)) \cdot \mathbf{v}) - \phi_s^2(f(p(t)))) \\ &= T(t^*)(\phi'_s(f(p(t)))\cos\theta - \phi_s^2(f(p(t))))\end{aligned}$$

假设  $t^*$  为这一射线与曲面第一次相交的点。

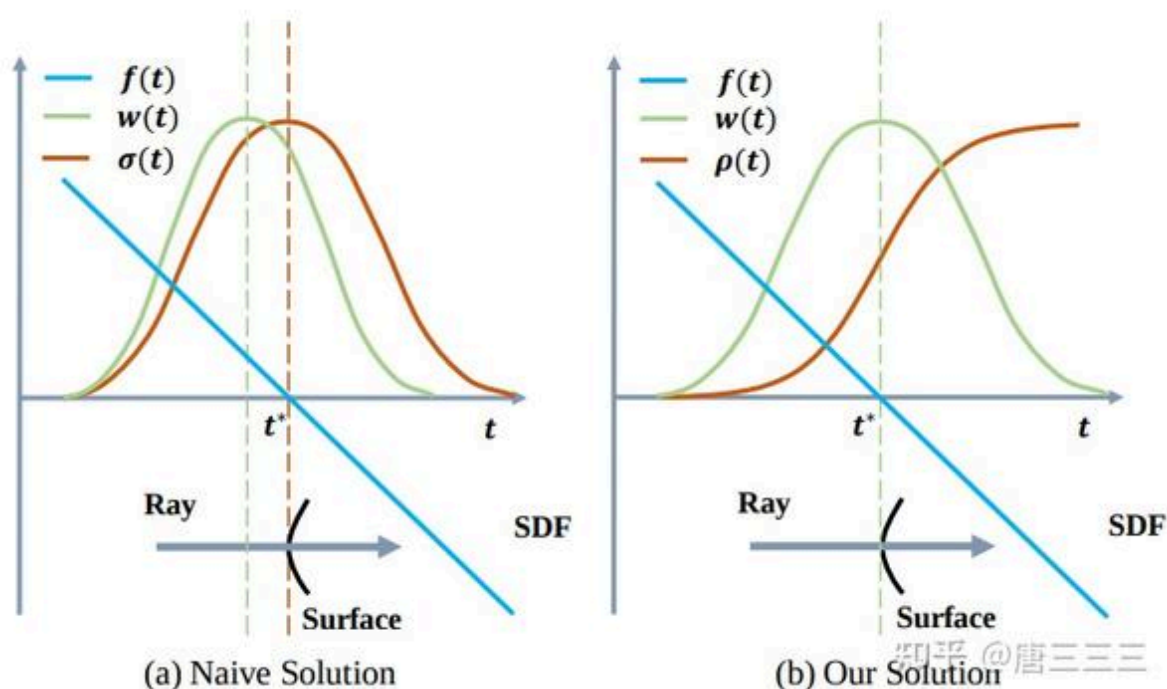
$$\begin{aligned}f(p(t^*)) &= 0 \\ \phi'_s(f(p(t^*))) &= 0\end{aligned}$$

所以：

$$\frac{dw(t^*)}{dt^*} = -T(t^*)\phi_s^2(0) < 0$$

从上面的推导可知，在曲面上的一个点，梯度不为0。因此原naive idea是有问题的。

其实 NeRF 积分公式中的透明度项  $T(t)$  就使得权重函数满足遮挡关系，但这会导致权重函数有偏（超前：峰值点落在交点前，附录也给出了有偏的证明和推导）。对此，NeuS 对体渲染公式设计了一个关于 SDF 一阶无偏且满足遮挡关系的权重函数。



## Neus (proposed)

请注意，从这里开始 NeuS 仅利用体渲染公式的积分（累加）性质，而不再考虑其最终提出的权重建模的物理意义是否与实际物理意义有着严格的对应关系。因此我觉得下面的  $\rho$  应当仅当成独属于 NeuS 的建模对待，不应当再与原生的体渲染公式涉及到的物理原理有过多的联系，而作者的最终目的也仅仅是为了得到  $w(t)$  与 SDF（即  $f(p(t))$ ）的转换关系，只是作者后面在将  $\rho(t)$  带入体渲染方程并做离散化的过程中，作者发现了能够绕过  $\rho(t)$  直接构建  $\alpha_i$ ，所以在实际的操作（代码）中，NeuS 将 SDF 值转成  $\alpha_i$  再进行体渲染的积分（累加）计算。

$$C(\mathbf{o}, \mathbf{v}) = \int_0^{+\infty} w(t) c(p(t), \mathbf{v}) dt$$

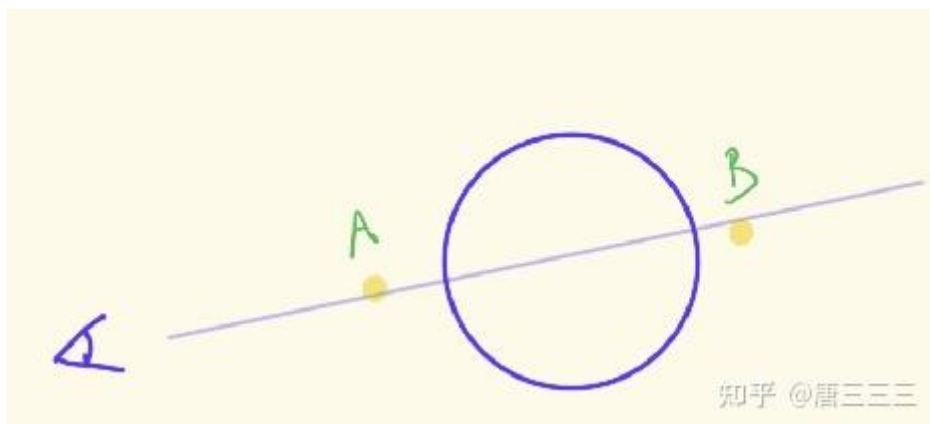
这里  $w(t)$  应该怎么计算呢？我们设计的原则应该遵守两个：

### Unbiased(无偏)

如果  $t^*$  是射线与曲面第一次相交的点，那  $w(t)$  在这一点上应该取最大值，也就是这一点对像素颜色贡献值应该最大。 $w(t)$  峰值应落在零值面  $f(\mathbf{p}(t^*)) = 0$ ，即光线与物体表面的交点。

### Occlusion Aware

如果有两个点  $f(t_0) = f(t_1)$ ，也就是两个点离曲面最近点的距离一样，且有  $w(t_0) > 0, w(t_1) > 0, t_0 < t_1$ ，也就是当两个点 SDF 一样的情况，在射线前面部分的点对颜色的贡献度更大。 $w(t)$  呈双峰分布，NeuS 做的就是第二个峰抑制掉，保证  $w(t)$  只有一个峰值且落在第一个交点处。



Motivation



一个简单的关于  $w(t)$  的计算：

$$w(t) = \frac{\phi_s(f(p(t)))}{\int_0^{+\infty} \phi_s(f(p(t))) dt}$$

(实际上就是多加了个简单的归一化操作)但是这个公式只适应于每个射线与曲线只有一个交点的情况下，当光线仅进入表面时(即仅发生一次相交时)，该权重函数是满足上面提到的两个性质的。但当光线出表面或再次进入另一表面时，则会出现第二个甚至第三个峰值，这就不满足遮挡关系。而 NeuS 的操作简单来说就是**仅保留第一个峰值，想办法抑制后面的峰值**。NeuS 的推导可以分步走：

1. 在仅有一次相交的情况下，已知无偏的权重函数  $w(t)$ ，利用一阶近似求解得到  $\rho(t)$
2. 得到  $\rho(t)$  后再考虑实际情况(入射后会出射)，想办法抑制第二个峰值

所以论文里觉得还是应该沿用之前体渲染计算方法：

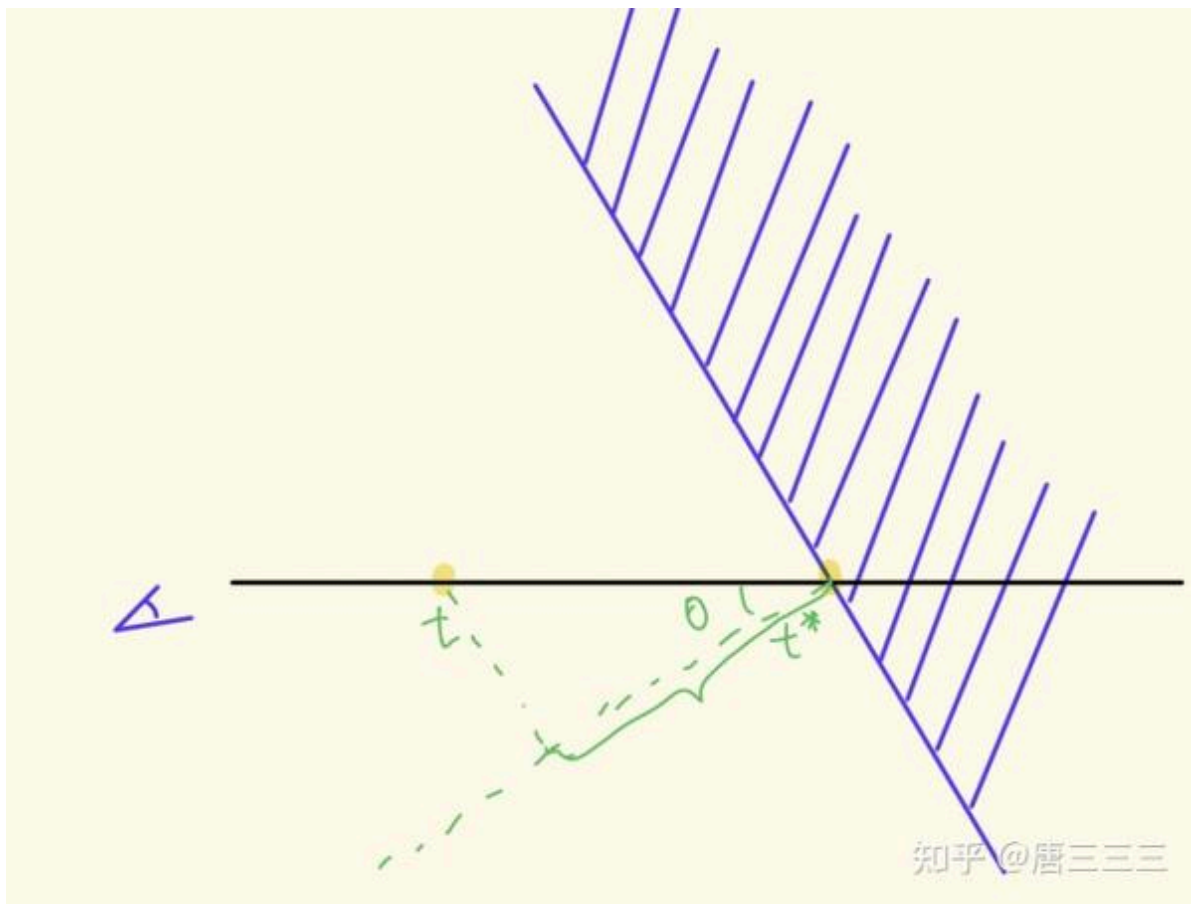
$$w(t) = T(t)\rho(t)$$

$$T(t) = e^{-\int_0^t \rho(s) ds}$$

只是这里的  $\rho(t) \neq \phi_s(f(p(t)))$ 。

### 推导

我们还是从比较简单的情况开始，假定空间里面只有一个平面，平面在相机的无穷远处，则每条射线只与一个点相交，且  $t^*$  为射线与曲面上相交点，在这种情况下有：



$$f(p(t)) = -|\cos\theta|(t - t^*)$$

其中  $f(p(t^*)) = 0$  且  $\theta$  是视角向量  $\mathbf{v}$  与表面法  $\mathbf{n}$  的夹角

这里  $\cos\theta$  就是一个常量。

$$w(t) = \lim_{t^* \rightarrow +\infty} \frac{\phi_s(f(p(t)))}{\int_0^{+\infty} \phi_s(f(p(t))) dt}$$

$$= \lim_{t^* \rightarrow +\infty} \frac{\phi_s(f(p(t)))}{\int_0^{+\infty} \phi_s(-|\cos\theta|(u - t^*)) du}$$

$$\text{令 } u^* = u - t^*$$

$$= \lim_{t^* \rightarrow +\infty} \frac{\phi_s(f(p(t)))}{\int_{-t^*}^{+\infty} \phi_s(-|\cos\theta|u^*) du^*}$$

$$\text{令 } \hat{u} = -|\cos\theta|u^*$$

$$= \lim_{t^* \rightarrow +\infty} \frac{\phi_s(f(p(t)))}{|\cos\theta|^{-1} \int_{-t^*}^{+\infty} \phi_s(\hat{u}) d\hat{u}}$$

$$= \lim_{t^* \rightarrow +\infty} \frac{|\cos\theta| \phi_s(f(p(t)))}{\int_{-t^*}^{+\infty} \phi_s(\hat{u}) d\hat{u}}$$

$$= |\cos\theta| \phi_s(f(p(t)))$$

or

$$\int_{-\infty}^{\infty} \phi(x) = \int_{-\infty}^{\infty} \frac{e^{-x}}{(1 + e^{-x})^2} = \frac{1}{1 + e^{-x}} \Big|_{x=-\infty}^{+\infty} = 1 - 0 = 1$$

$$\int_{-\infty}^{\infty} \phi_s(x) = \int_{-\infty}^{\infty} \frac{se^{-sx}}{(1 + e^{-sx})^2} = \frac{1}{(1 + e^{-sx})} \Big|_{x=-\infty}^{+\infty} = 1 - 0 = 1$$

$$\int_{-\infty}^{\infty} \phi_s(|\cos(\theta)|x) = \int_{-\infty}^{\infty} \frac{se^{-s|\cos(\theta)|x}}{(1 + e^{-s|\cos(\theta)|x})^2}$$

$$= \frac{1}{|\cos(\theta)|(1 + e^{-s|\cos(\theta)|x})} \Big|_{x=-\infty}^{+\infty}$$

$$= \frac{1 - 0}{|\cos(\theta)|} = \frac{1}{|\cos(\theta)|}$$

(实际上，能够直接推积分结果出来，中途把  $|\cos(\theta)|$  抽出来反而很奇怪)

也就是：

$$T(t)\rho(t) = |\cos\theta|\phi_s(f(p(t)))$$

由于

$$T(t)\rho(t) = -\frac{dT(t)}{dt}$$

这也能对应上后面给的

$$|\cos\theta|\phi_s(f(p(t))) = -\frac{\Phi_s(f(p(t)))}{dt}$$

所以：

$$-\frac{dT(t)}{dt} = -\frac{\Phi_s(f(p(t)))}{dt}$$

也就有：  $C(\mathbf{o}, \mathbf{v}) = \int_0^{+\infty} w(t)c(p(t), \mathbf{v})dt$

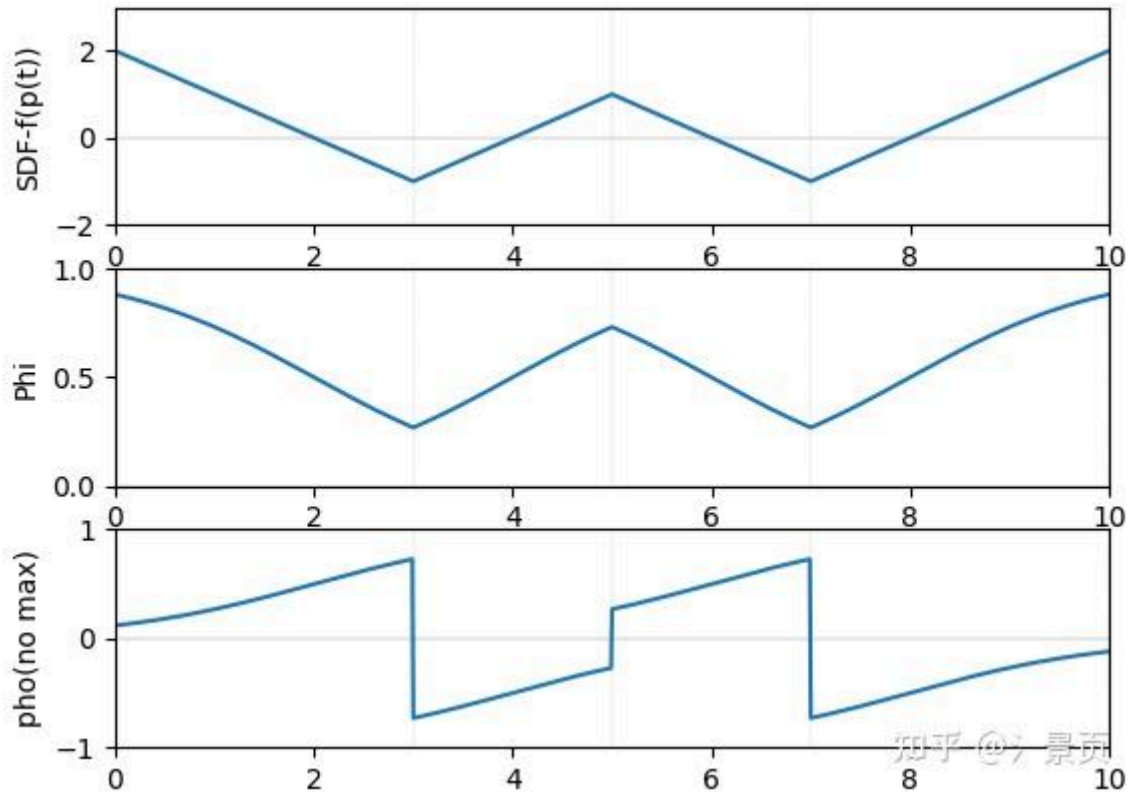
$$T(t) = \Phi_s(f(p(t)))$$

$$\int_0^t \rho(s)ds = -\ln\Phi_s(f(p(t)))$$

$$\rho(t) = -\frac{\frac{d\Phi_s(f(p(t)))}{dt}}{\Phi_s(f(p(t)))}$$

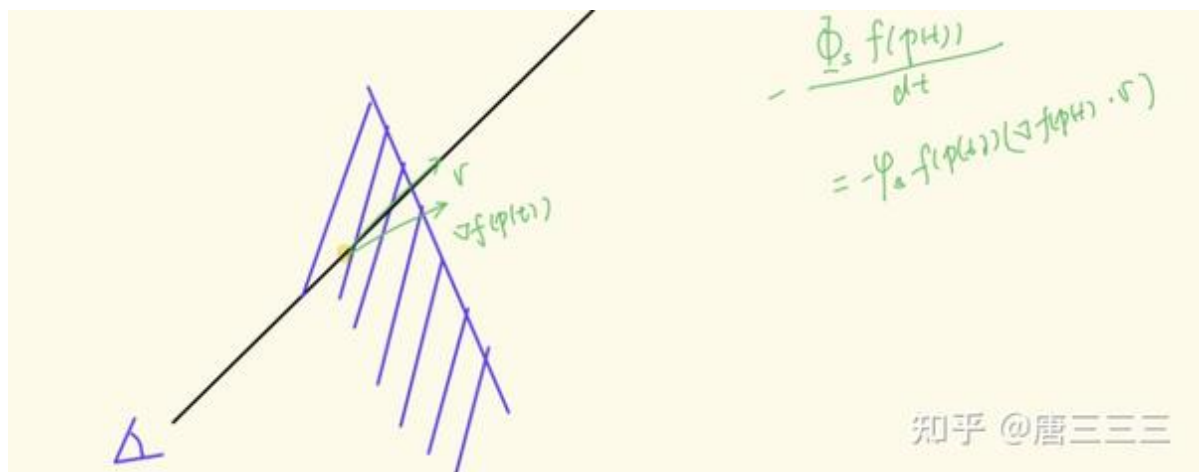


此时就满足了 $w(t)$ 的一阶无偏性。作者在文章中说，当光线出射时，SDF值会增大， $\frac{-\Phi_s(f(p(t)))}{\Phi_s(f(p(t)))}$  会小于零，为了更直观的展示，可以参考下面第三行，第二行为  $\Phi(f(p(t)))$ ，其中横轴均为  $t$ 。



之前的推导是一个特定的假设情况下，下面看下能不能把这个结论进行扩展。

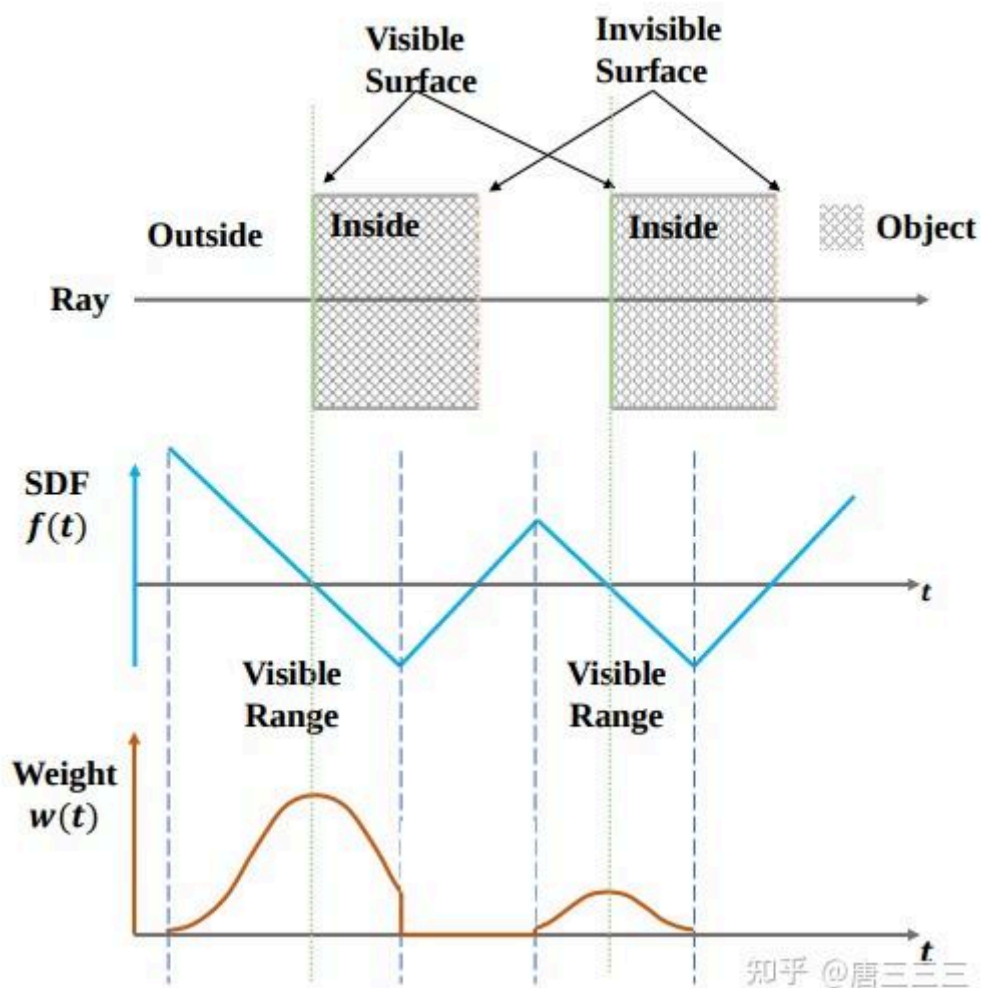
首先，如果一般射线穿过两个及以上的面， $-\frac{d\Phi_s(f(p(t)))}{dt}$  是有可能为负的，所以为了防止它为负，我们加一个约束：



$$\rho(t) = \max\left(-\frac{d\Phi_s(f(p(t)))}{dt}, 0\right)$$

同时，我们使用新的：

$$C(\mathbf{o}, \mathbf{v}) = \int_0^{+\infty} w(t) c(p(t), \mathbf{v}) dt$$



算出来呢，可以看出，相交的第一个面的贡献度是最大的，后面进行了衰减，从图中也可以看出，他是服从我们之前的设计原则（Unbiased and Occlusion Aware）。

### 离散化处理

跟原始nerf一样，我们推出来的这个连续积分的形式，计算是无法处理的，所以我们需要对这个公式进行离散化处理，又方便我们编程，其推导过程跟原始的Nerf是一样的。

原始Nerf的体渲染公式：

$$C = \int c(t)\sigma(t)T(t)dt$$

在离散情况下，我们先把射线分为一段一段，假定每一段内颜色  $c(t)$  和密度  $\sigma(t)$  是不变的，先算出其每段的不透明度：

$$\alpha_i = 1 - e^{-\int_{t_i}^{t_{i+1}} \sigma(t)dt}$$

$$\alpha_i = 1 - e^{-\delta_i \sigma_i}$$

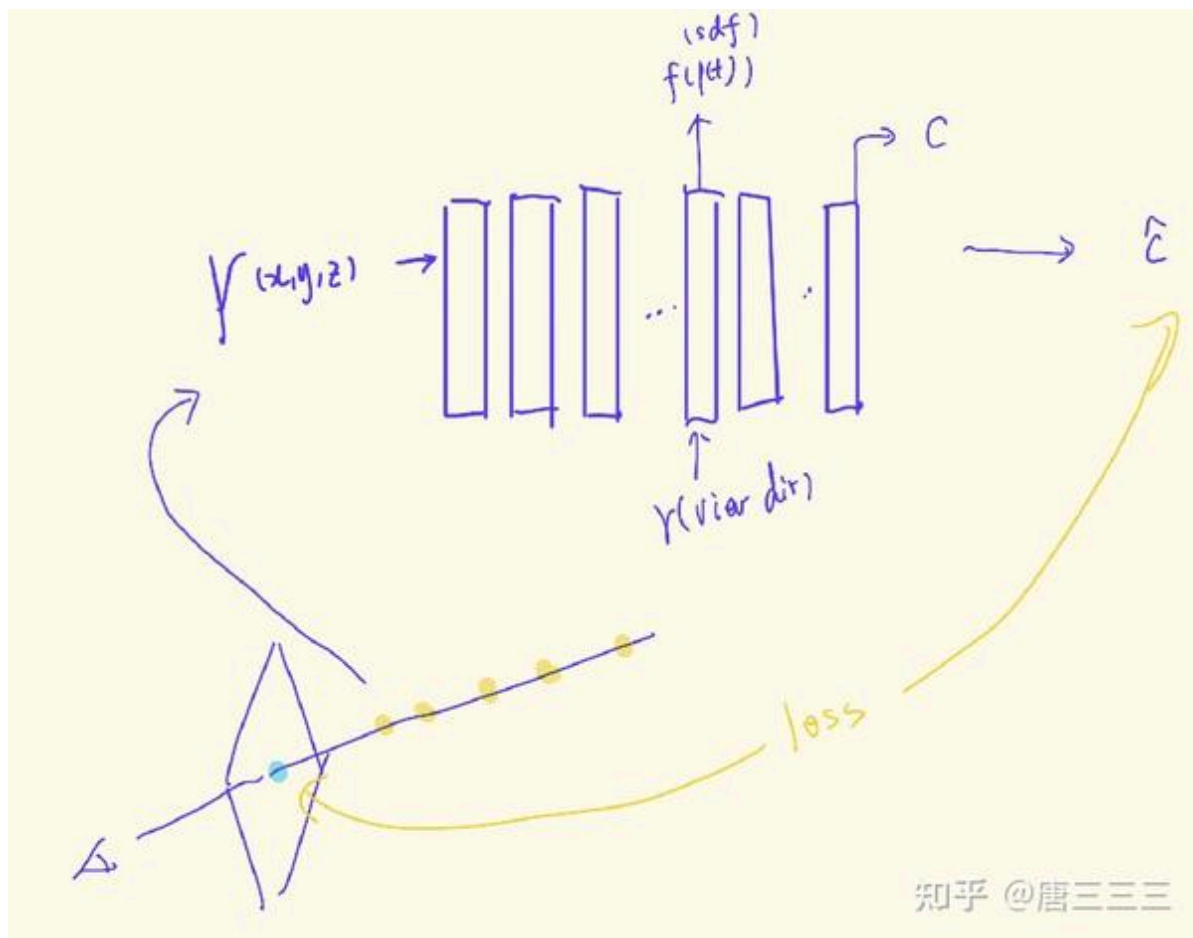
最后的颜色预测为：

$$C = c_0\alpha_0 + c_1\alpha_1(1 - \alpha_0) + \dots + c_N\alpha_N(1 - \alpha_{N-1}) \dots (1 - \alpha_0)$$

Neus这个方法下也是一样。

$$\begin{aligned}
\alpha_i &= 1 - e^{\int_{t_i}^{t_{i+1}} \rho(t) dt} \\
&= 1 - e^{\int_{t_i}^{t_{i+1}} -\frac{d\Phi_s(f(p(t)))}{\Phi_s(f(p(t)))} dt} \\
&= 1 - e^{\ln\Phi_s(f(p(t))) + C|_{t_i}^{t_{i+1}}} \\
&= 1 - e^{\ln\Phi_s(f(p(t_{i+1}))) - \ln\Phi_s(f(p(t_i)))} \\
&= 1 - \frac{\Phi_s(f(p(t_{i+1})))}{\Phi_s(f(p(t_i)))} \\
&= \frac{\Phi_s(f(p(t_i))) - \Phi_s(f(p(t_{i+1})))}{\Phi_s(f(p(t_i)))}
\end{aligned}$$

## 具体实现



$$L = L_{color} + \lambda L_{reg} + \beta L_{mask}$$

$$L_{color} = \frac{1}{m} \sum_k (C_k, \hat{C}_k)$$

$$L_{reg} = \frac{1}{mn} \sum_{k,i} (\|\nabla f(\hat{p}_{k,i})\|_2 - 1)$$

## 总结

整个过程可以理解为作者从体渲染公式自顶  $w(t)$  向下  $\rho(t)$  出发进行推导，然后将这个过程离散化的过程中，作者发现中间  $\alpha_i$  是最合适的切换点。

虽然  $w(t), \rho(t), \alpha_i$  的关系在上一段话应该已经交代清楚了（我猜）。但最后还有一个奇怪的点，那就是为什么从  $w(t)$  推导得到  $\rho(t)$ 。个人认为的看法是，作者希望在保持无偏的情况下再实现遮挡关系的满足，此时只能从  $w(t)$  往回推，试图找到能够引入遮挡关系的点，推到这里发现能通过一个 max 就能简单的解决这个问题；另一原因可能是作者还是想和体渲染中的物理概念扯上关系，因此塑造了这个  $\rho(t)$  让他具有和  $\sigma(t)$  一样的地位。但我个人觉得，按照这个思路作者好像也能够直接推到  $\alpha(t)$  并直接给出离散表示  $\alpha_i$ ，只是本人还未证明过。

关于物理关系，我个人认为得当 NeuS 从利用归一化操作得到无偏的权重函数  $w(t) = \frac{\phi_s(f(\mathbf{p}(t)))}{\int_0^{+\infty} \phi_s(f(\mathbf{p}(u)))du}$  开始推导时，物理关系就被破坏了（想象一下我仅改变光线后方样本的密度（请理解我用密度这个词），前方样本的密度不变，但却使前方的贡献变小了）。此时 NeuS 仅借鉴了体渲染的积分（累加）形式罢了。

回过头来看，个人觉得从无偏的权重函数  $w(t) = \frac{\phi_s(f(\mathbf{p}(t)))}{\int_0^{+\infty} \phi_s(f(\mathbf{p}(u)))du}$  开始推导颇有点机械降神的意味。或许对于搞机器学习理论的人来说，VolSDF 推导上下界的才更为正常。