

技术文档

前言

背景介绍

漫反射光照重建是计算机图形学中的一个重要问题，其目的是从一组光照样本中重建出一个光照向量，以便于后续的渲染。在实际应用中，由于光照样本的噪声，重建出的光照向量往往会受到噪声的影响，因此需要对光照向量进行降噪处理。

目的

本文档旨在介绍漫反射光照重建的算法，包括光照向量的编码、降噪、重建等过程，以及降噪后的光照向量的方差估计。

注：

这个算法是我受到一阶球谐函数光照近似启发而来，不清楚是否有类似的算法，可能是原创。如果非原创欢迎在 issue 中指出。

漫反射光照降噪与重建算法

模型

定义

假设给一组 $N + 1$ 维向量 V_i (向量组自身记为 V ，第 $N + 1$ 维非负) 和一个能量度量函数 $w(v) : \mathbb{R}^{N+1} \rightarrow \mathbb{R}$ ， w 定义为

$$w(v) := \sqrt{\sum_{i=1}^N v_i^2 + v_{N+1}}$$

定义映射

$$T : \text{List}(\mathbb{R}^{N+1}) \rightarrow \mathbb{R}^{N+1}$$

其中 List 是一个有序集合， $T(V)$ 是 V 中所有向量的到 \mathbb{R}^{N+1} 的映射， T 满足：

1. 能量守恒

对于任意 $V \in \text{List}(\mathbb{R}^{N+1})$ 有

$$w(T(V)) = \sum_{i=1}^n w(V_i)$$

2. 结合不变性

对于任意 $x \in \mathbb{R}^{N+1}$ 和 $V \in \text{List}(\mathbb{R}^{N+1})$ 有

$$T(\{x, V\}) = T(\{x, T(V)\})$$

3. 线性性

对于任意 $x \in \mathbb{R}^{N+1}$ ， $n \in \mathbb{N}_+$ 有

$$T(\{x, x, \dots, x\}) = nx$$

其中 $\{x, x, \dots, x\}$ 是 n 个 x 的有序集合

记

$$V_i = (\vec{v}_i, I_i)$$

$$\Lambda(V_i) := v_i, I(V_i) := I_i$$

其中

$$\vec{v}_i \in \mathbb{R}^N, I_i \in \mathbb{R}_+$$

则上述映射 T 存在，其形式为：

$$T(V) = \left(\sum_{i=1}^n v_i, \sum_{i=1}^n \omega(V_i) - \left| \sum_{i=1}^n v_i \right| \right)$$

其中

$$\omega(V_i) := |v_i| + I_i$$

物理意义

选定 $N = 3$ ，取 V 为光照样本集合，每个样本 V_i 为一个光照向量， V_i 的前三维为方向光照，第四维为环境光照， $w(V_i)$ 为光照的能量， T 为光照的合成算子， $T(V)$ 为合成光照， T 满足能量守恒，结合不变性和线性性。

算法

取算法的输入为一组输入进 BRDF 模型的光照向量 V ，输出为一个光照向量 $T(V)$ ，算法的目标是降低光照向量的噪声，重建光照向量。

算法流程

1. 编码 V_i 为 $V_i = (I * \vec{d}, 0)$ ，其中 I 为采样的光照强度， \vec{d} 为采样的入射光照方向， 0 为环境光照强度
2. 在时间域内对 V_i 进行滤波，得到 $V_{\text{out}} = \frac{1}{n} T(V)$ ，其中 n 为采样的光照样本数
3. 对 V_{out} 运用 SVGF 算法进行降噪得到 V'_{out}
4. 将 V'_{out} 输入进 BRDF 模型得到降噪后的光照

算法实现

1. 实际上编码 V_i 为 $V_i = (I * \vec{d}, I)$ ，其中 I 为采样的光照强度， \vec{d} 为采样的入射光照方向
2. 在加权混合时直接混合 V_i
3. 重建时由 T 算子的表达式可知，操作 2 中的混合操作等价于对 V_i 的前三维进行加权求和，对第四维进行加权求和，并且能正确构建出结果
4. 只有在解码时运用 T 算子，其余情况下只需要对 V_i 进行类似正常流程里对普通光照的处理即可

关键代码如下：

```
struct T
{
    mediump vec4 v_I; // (I * arrow(d), I)
    mediump vec2 CoCg; // (Co, Cg)
};

vec3 project_T_irradiance(T L, vec3 N)
{
    float Y = L.v_I.w;
    float T = Y - L.CoCg.y * 0.5;
```

```

    float G = L.CoCg.y + T;
    float B = T - L.CoCg.x * 0.5;
    float R = B + L.CoCg.x;

    vec3 irradiance = vec3(R,G,B) * (max(dot(L.v_I.xyz, N),0) + (Y -
length(L.v_I.xyz))) / (Y+1e-3);
    return max(irradiance, vec3(0.0));
}

T irradiance_to_T(vec3 irradiance, vec3 dir)
{
    T result;
    float Co = irradiance.r - irradiance.b;
    float t = irradiance.b + Co * 0.5;
    float Cg = irradiance.g - t;
    float Y = max(t + Cg * 0.5, 0.0);

    result.CoCg = vec2(Co, Cg);
    result.v_I = vec4(dir * Y,Y);
}

T mix_T(T a, T b, float s)
{
    T result;
    result.v_I = mix(a.v_I, b.v_I, s);
    result.CoCg = mix(a.CoCg, b.CoCg, s);
    return result;
}

T init_T()
{
    T result;
    result.v_I = vec4(0);
    result.CoCg = vec2(0);
    return result;
}

T scaleT(T A, float x) {
    T tmp;
    tmp.CoCg = A.CoCg * x;
    tmp.v_I = A.v_I * x;
    return tmp;
}

void accumulate_T(inout T accum, T b, float scale)
{
    accum.v_I += b.v_I * scale;
    accum.CoCg += b.CoCg * scale;
}

```

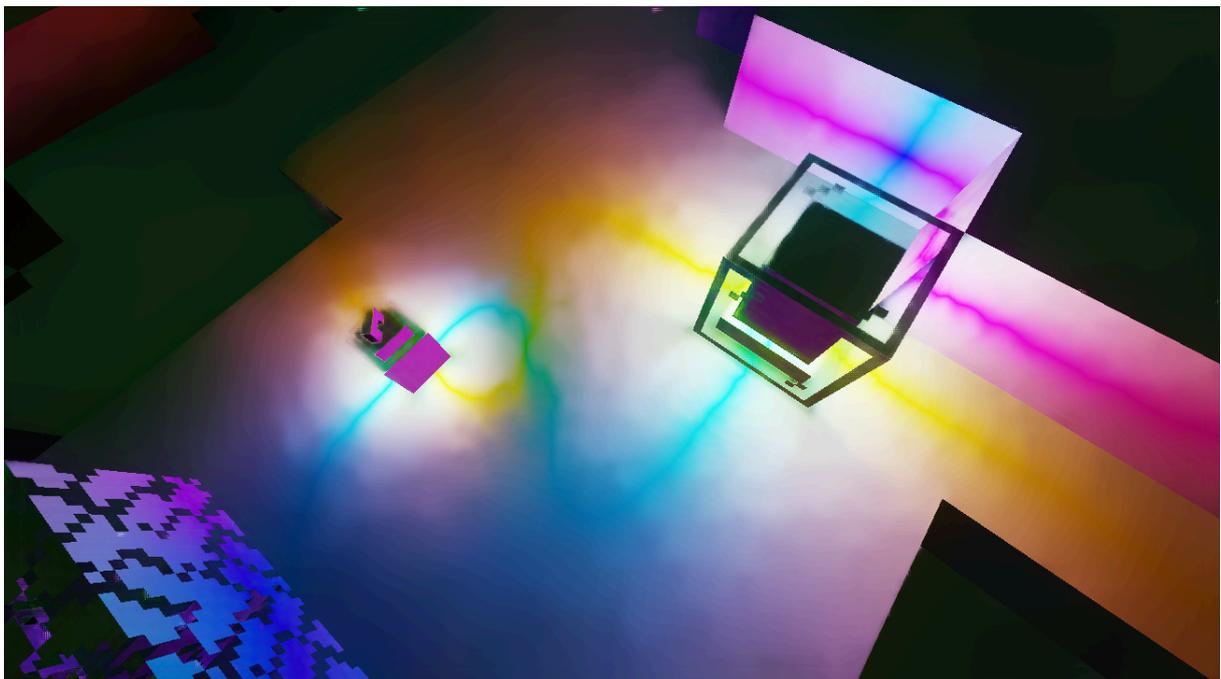
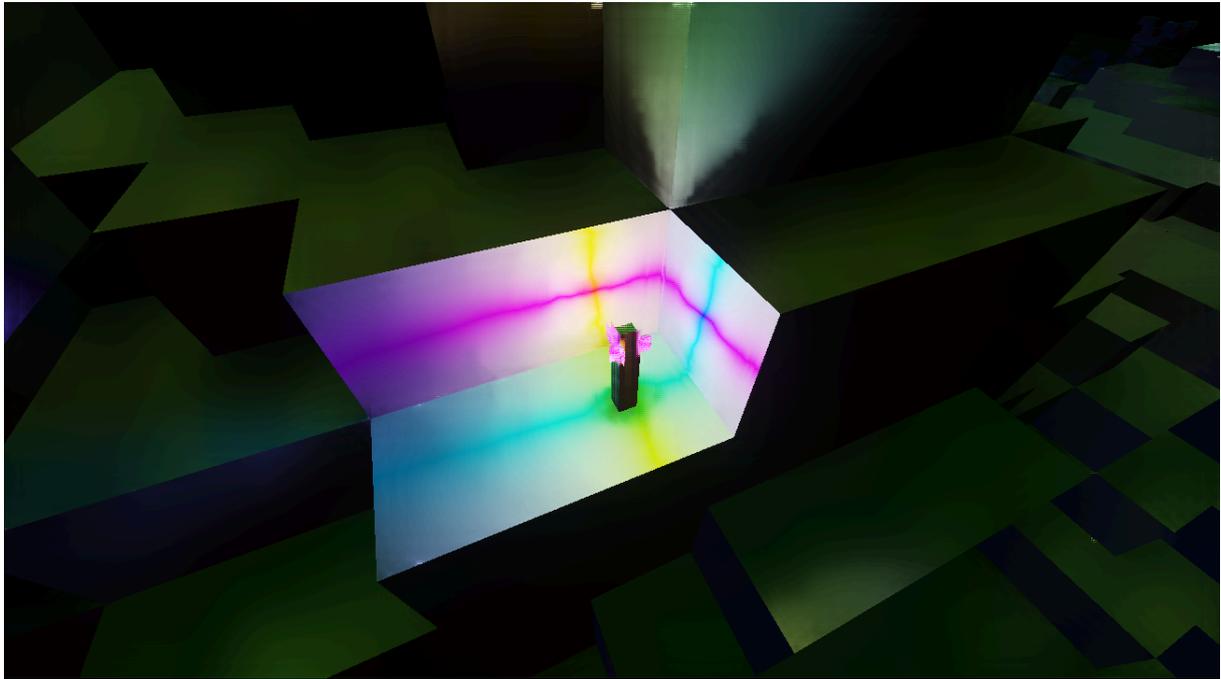
光照重建

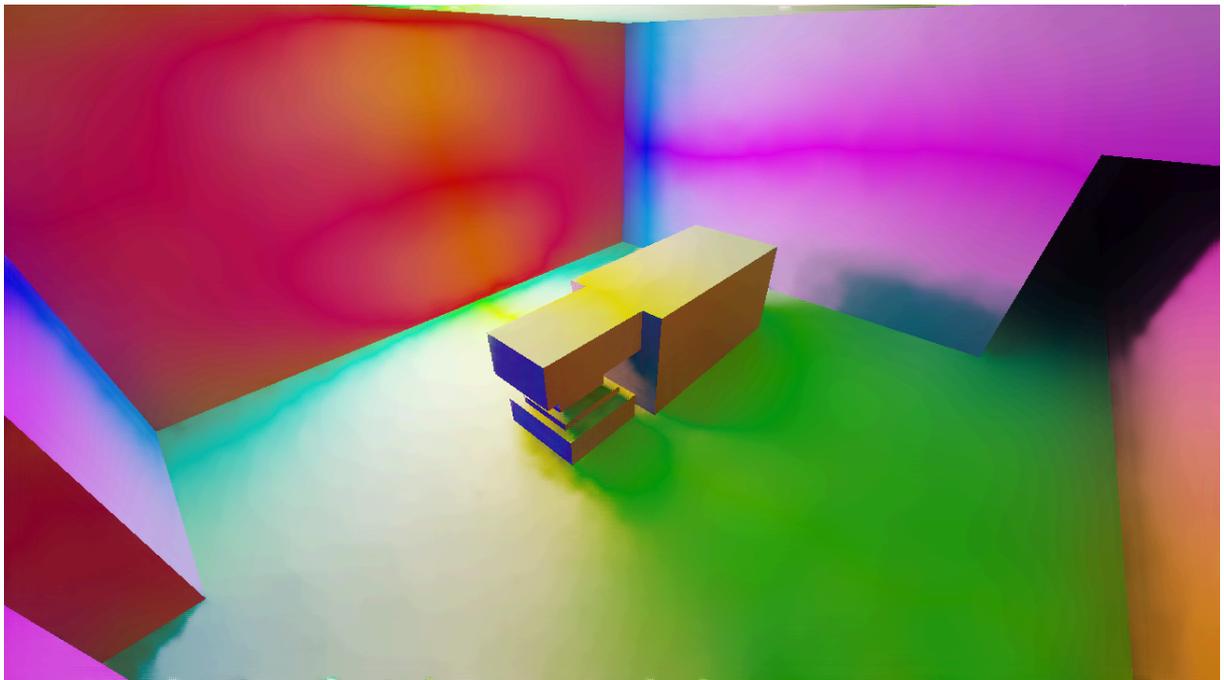
假设已经得到了降噪后的光照 $V'_{\text{out}} = (\vec{D}, E)$ 其中 \vec{D} 是方向光向量, E 是环境光分量, 对于一个给定的 BRDF 模型, 可以通过以下方法得到重建后的光照:

$$I_o = f_{\text{brdf}}(\vec{e}, \vec{D}, \vec{N}) + \int_{\omega} f_{\text{brdf}}(\vec{e}, E\vec{\omega}, \vec{N}) d\omega$$

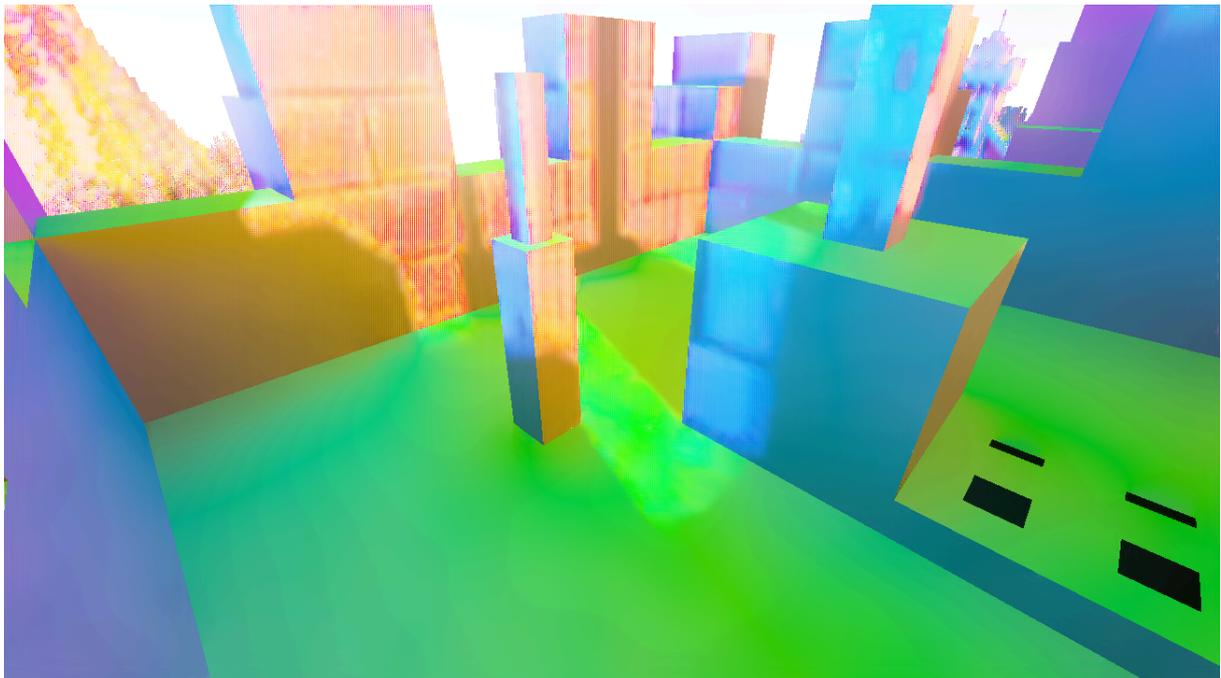
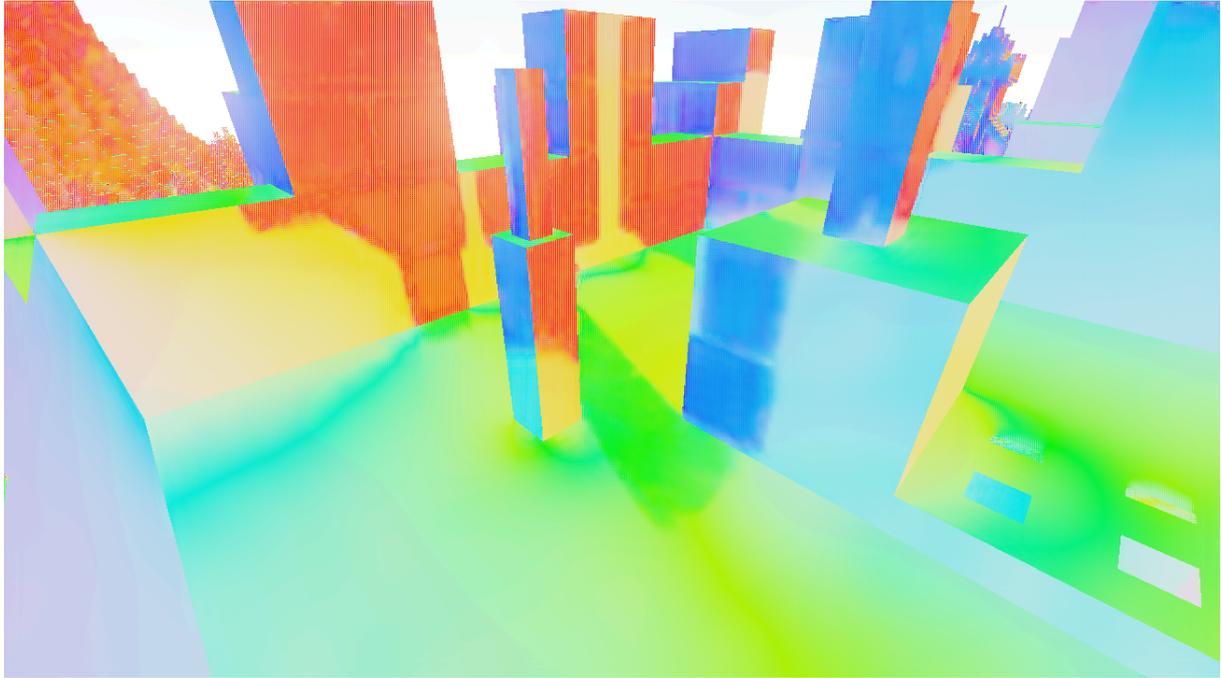
其中 I_o 为重建后的光照强度, f_{brdf} 为 BRDF 函数, \vec{e} 为观察方向, \vec{N} 为法向量, $\vec{\omega}$ 为入射光照方向, $d\omega$ 为立体角微分

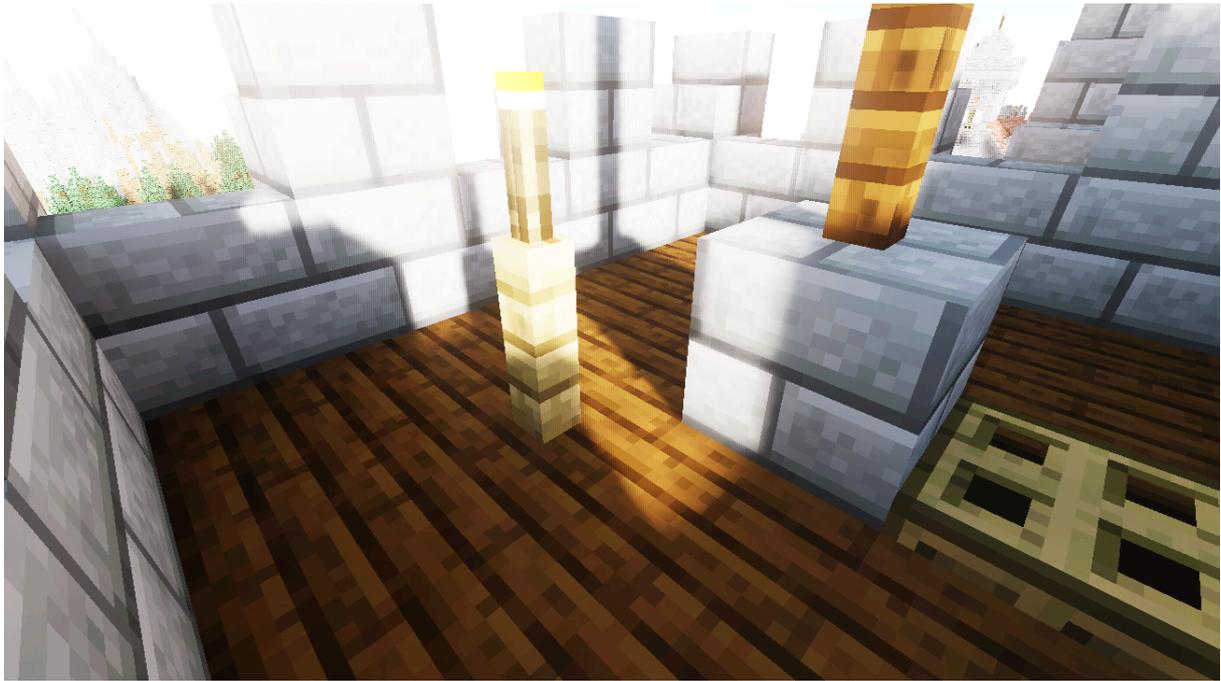
下面的图片显示了方向分量





下面的图片显示了方向分量以及其漫反射光照重建结果





显然我们可以看出来降噪后的光照的方向分量大致指向光照贡献较大的方向。那么我们就可以复用这个方向分量，将其输入到采样器中进行重要性采样，得到更为准确的光照重建结果。

预计未来的路径追踪器会通过复用上一帧的光照方向分量，进行重要性采样，以得到更为准确的光照重建结果，而不是单纯使用 BRDF 的 pdf 进行无目的的采样。

方差估计 (可能有用?)

考虑如下的最优化问题 (暂时忽略 $N + 1$ 维分量):

$$E(\vec{V}^2) = \min \frac{1}{n} \sum_{i=1}^n |\vec{V}_i|^2$$

$$s.t. \frac{1}{n} \sum_{i=1}^n \vec{V}_i = \vec{\mu}$$

$$s.t. \frac{1}{n} \sum_{i=1}^n |\vec{V}_i| = I$$

其中 $\vec{\mu}$ 为 \vec{V} 的期望, I 为 $|\vec{V}|$ 的期望

我们要求解这个问题, 得到 $E(\vec{V}^2)$

如果考虑对称性, 不妨设 $\vec{V}_i = \frac{1}{n}\vec{\mu} + \vec{e}_i$, 满足 $\vec{\mu} \cdot \vec{e}_i = 0$ 并且 $\sum \vec{e}_j = 0$, $|\vec{e}_i| = |\vec{e}_j|$

显然第一个约束条件成立, 第二个约束条件变为

$$\frac{1}{n} \sum_{i=1}^n \left| \frac{1}{n}\vec{\mu} + \vec{e}_i \right| = I$$

即

$$\frac{1}{n} \sum_{i=1}^n \sqrt{\left(\frac{1}{n}\vec{\mu} + \vec{e}_i \right)^2} = I$$

即

$$\frac{1}{n} \sum_{i=1}^n \sqrt{\frac{1}{n^2} |\vec{\mu}|^2 + \frac{2}{n} \vec{\mu} \cdot \vec{e}_i + |\vec{e}_i|^2} = I$$

考虑 $\vec{\mu} \cdot \vec{e}_i = 0$, 则

$$\frac{1}{n} \sum_{i=1}^n \sqrt{\frac{1}{n^2} |\vec{\mu}|^2 + |\vec{e}_i|^2} = I$$

由于 $|\vec{e}_i| = |\vec{e}_j|$, 则

$$\frac{1}{n} \sum_{i=1}^n \sqrt{\frac{1}{n^2} |\vec{\mu}|^2 + |\vec{e}_i|^2} = \sqrt{\frac{1}{n^2} |\vec{\mu}|^2 + |\vec{e}_i|^2}$$

于是

$$\vec{V}_i^2 = I^2$$

即

$$E(\vec{V}^2) = I^2$$

考虑原始问题, 我们有

$$E(V^2) = \frac{1}{n} \omega(T(V))^2$$

则方差估计为

$$D(T(V)) = \frac{1}{n} \omega(T(V))^2 - \frac{1}{n^2} \Lambda(T(V))^2$$

简记为 $D(T(V)) = \sigma^2(V)$

方差的性质

$$\sigma^2(\lambda V) = \lambda^2 \sigma^2(V)$$