

TESI DI LAUREA MAGISTRALE IN  
INFORMATICA

# **Simulazione della traslucenza in tempo reale mediante campionamento dinamico basato sul contrasto locale**

CANDIDATO

Dragan Vuletic

RELATORE

Prof. Andrea Fusiello

**CONTATTI DELL'ISTITUTO**

Dipartimento di Scienze Matematiche, Informatiche e Fisiche

Università degli Studi di Udine

Via delle Scienze, 206

33100 Udine — Italia

+39 0432 558400

<https://www.dmif.uniud.it/>

Ai miei genitori, per il supporto costante e i sacrifici fatti per  
permettermi di raggiungere questo traguardo.

A mia sorella, per aver sempre creduto in me, anche  
nei momenti più difficili.

Non avrei potuto farcela senza di voi.

Grazie di cuore.



# Sommario

La tesi affronta l'importanza del subsurface scattering (SSS) nel rendering grafico, specialmente per materiali come la pelle umana. Viene descritta la teoria alla base del SSS, inclusi i modelli matematici e fisici che lo descrivono, e sono presentate diverse tecniche in tempo reale basate su tali modelli che lavorano in screen space. Un aspetto originale che viene introdotto in questa tesi è il campionamento a 2 livelli basato sul contrasto. È un meccanismo che permette di ottimizzare la distribuzione dei campioni utilizzati per calcolare il contributo del subsurface scattering, compromettendo al minimo la qualità finale. L'analisi percettiva dei render, incluso quello con l'estensione proposta, viene effettuata tramite questionari e fornisce una valutazione dell'efficacia delle tecniche di rendering sviluppate.



# Indice

<b>1</b>	<b>Introduzione</b>	<b>1</b>
<b>2</b>	<b>Fondamenti Teorici</b>	<b>5</b>
2.1	Unità Radiometriche . . . . .	5
2.1.1	Flusso Radiante . . . . .	5
2.1.2	Irradianza . . . . .	5
2.1.3	Radianza . . . . .	6
2.2	Teoria Rendering . . . . .	8
2.2.1	Rendering Equation . . . . .	8
2.2.2	Volume Rendering Equation . . . . .	9
2.3	Teoria del Subsurface Scattering . . . . .	13
2.3.1	Modello Dipole . . . . .	13
	Approssimazione con dipolo . . . . .	22
2.3.2	Modello Multipole . . . . .	24
	Riflettanza singola slab . . . . .	25
	Combinazione multipli layer . . . . .	27
2.3.3	Modello Burley . . . . .	29
	Configurazione Searchlight . . . . .	30
	Configurazione Diffuse Surface Transmission . . . . .	32
	Considerazioni . . . . .	33
<b>3</b>	<b>Implementazione Tecniche SSS</b>	<b>35</b>
3.1	Approcci Real-Time . . . . .	35
3.1.1	Texture Space . . . . .	36
3.1.2	Screen Space . . . . .	37
3.2	Jimenez (2009) . . . . .	38
3.3	Jimenez (2015) . . . . .	40
3.3.1	Guided Optimization . . . . .	41
3.3.2	Analytic Pre-Integration . . . . .	42
3.3.3	Artist Friendly Model . . . . .	42
3.3.4	Artefatti . . . . .	43
3.4	Golubev (2018) . . . . .	44
3.4.1	Importance Sampling . . . . .	47
3.5	Jimenez Traslucenza (2010) . . . . .	48
3.6	Texturing . . . . .	50
<b>4</b>	<b>Campionamento a 2 Livelli Basato sul Contrasto</b>	<b>53</b>
4.1	Razionale delle Assunzioni . . . . .	53
4.2	L'algoritmo . . . . .	54
4.3	Determinazione Valori Ottimali dei Parametri . . . . .	58
4.4	Analisi performance . . . . .	61

<b>5 Analisi Percettiva Mediante Questionario</b>	<b>63</b>
5.1 Struttura Questionario . . . . .	63
5.2 Analisi Risultati . . . . .	65
5.2.1 Primissimo Piano . . . . .	67
5.2.2 Primo Piano . . . . .	68
5.2.3 Mezzo Primo Piano . . . . .	69
5.3 Ranking tramite processo di gerarchia analitica . . . . .	71
<b>6 Considerazioni Finali</b>	<b>75</b>

# 1

## Introduzione

Nel vasto panorama del rendering grafico, uno dei fenomeni ottici cruciali per rendere un’immagine foto-realistica è il *subsurface scattering*, comunemente abbreviato con SSS. Quest’ultimo si verifica nel momento in cui la luce interagisce con materiali traslucidi, penetrando nello strato superficiale e diffondendosi al loro interno prima di emergere dalla superficie in un punto diverso da quello di entrata con un aspetto diffuso e morbido. Il subsurface scattering è una manifestazione della complessa interazione della luce con la materia, che coinvolge processi di assorbimento, diffusione e trasmissione all’interno del materiale stesso. La comprensione del subsurface scattering è fondamentale per il rendering grafico, poiché questo fenomeno contribuisce in modo significativo alla percezione di realismo e naturalezza delle immagini generate al computer. Oltre ad influenzare l’aspetto visivo dei materiali traslucidi, il subsurface scattering è responsabile della trasmissione della luce attraverso la materia, influenzando la distribuzione spettrale della luce emergente e la percezione del colore e della luminosità. Per comprendere appieno il subsurface scattering, è necessario considerare una serie di fattori chiave che influenzano il suo comportamento. Questi includono le proprietà ottiche del materiale, come l’indice di rifrazione e il coefficiente di assorbimento, che determinano l’efficienza con cui la luce viene assorbita e diffusa all’interno del materiale. Inoltre, l’angolo di incidenza della luce e la profondità a cui la luce penetra nel materiale influenzano la quantità e la direzione del subsurface scattering, creando effetti visivi distinti in base alla geometria e alla composizione del materiale. Il subsurface scattering è un fenomeno onnipresente in natura e si osserva in una vasta gamma di materiali, tra cui la pelle umana, il marmo, il latte e molti altri (Fig. 1.1).

Nonostante l’importanza del subsurface scattering nel rendering grafico, la sua simulazione accurata è una sfida tecnica significativa. Questo fenomeno ottico coinvolge processi complessi di interazione della luce con la materia, che richiedono l’implementazione di algoritmi sofisticati e la considerazione di molteplici fattori. Tra questi, come abbiamo visto prima, le proprietà ottiche del materiale, quali l’indice di rifrazione, il coefficiente di assorbimento e il coefficiente di scattering. Inoltre, la simulazione del subsurface scattering è di sua natura computazionalmente costosa, dove tale complessità computazionale deriva dalla necessità di calcolare la distribuzione spaziale della luce all’interno del materiale, considerando le interazioni multiple della luce con le particelle del materiale stesso. Questo richiede l’implementazione di algoritmi efficienti e l’ottimizzazione delle risorse computazionali per garantire prestazioni accettabili senza compromettere la qualità visiva dell’output renderizzato. In particolare nei contesti di rendering



(a) Mani illuminate da dietro



(b) Statua di giada illuminata uniformemente



(c) Foglie illuminate da dietro



(d) Mandarino illuminato da dietro

Figura 1.1: Immagini di diversi materiali che, essendo traslucidi, esibiscono caratteristiche di subsurface scattering. Notare, in particolare, la tinta rossastra nelle mani, dettata dall’interazione della luce con uno strato interno della pelle particolarmente vascolarizzato, chiamato derma.

in tempo reale, dove è fondamentale rientrare nel budget di rendering (in genere 16ms o 33ms per frame), per poter garantire all’utente un’esperienza fluida. Nonostante le sfide tecniche, l’adozione del subsurface scattering ha portato a significativi progressi nel campo del rendering grafico. Questo fenomeno ottico è ampiamente utilizzato in settori come l’industria cinematografica, l’industria dei videogiochi e la visualizzazione architettonica per creare effetti visivi realistici e coinvolgenti. L’implementazione di algoritmi e tecniche avanzate per la simulazione del subsurface scattering ha permesso agli artisti e ai designer di creare materiali traslucidi con un elevato grado di realismo e dettaglio, migliorando notevolmente la qualità visiva delle produzioni digitali e aumentando l’impatto emotivo e visivo delle immagini e delle animazioni generate al computer. L’importanza di tale effetto è pure riconosciuta dal fatto che, nel 2004, Henrik Wann Jensen ed i suoi collaboratori ricevono un premio per l’innovazione tecnica da parte degli Academy Awards per il loro ruolo pionieristico nell’ambito [1]. Nel 2001, infatti, presentano il loro lavoro nel paper ”A Practical Model for Subsurface Light Transport” [19] che rivoluzionerà la computer grafica in ambito cinematografico, permettendo l’utilizzo del subsurface scattering in ambienti di produzione. In particolare, il suo primo utilizzo in un film sarà realizzato dallo studio Weta FX per “Il Signore degli Anelli - Le due Torri”, principalmente per la creazione in CG del personaggio chiamato “Gollum” (Fig. 1.2). Da questo momento in poi, tale tecnica, così come tutti gli altri metodi sviluppati in futuro che cercano di simulare la traslucenza, riceveranno una sempre maggiore adozione dall’industria, vedendo successivamente il loro utilizzo arrivare anche in ambienti interattivi.

In questo lavoro di tesi, andremo per prima cosa ad esplorare quali siano le basi fondanti del sub-



Figura 1.2: Il personaggio chiamato “Gollum” realizzato completamente in CG nel film “Il Signore degli Anelli - Le Due Torri” da parte di Weta FX. È il primo esempio di uso del subsurface scattering nel cinema.

surface scattering nell’ambito del rendering offline. A partire da queste fondamenta esploreremo gli approcci e tecniche per il rendering del subsurface scattering nel contesto del rendering in tempo reale sviluppate basandosi sui metodi offline. Attraverso un’analisi dettagliata delle teorie, delle metodologie e delle applicazioni pratiche esistenti, miriamo a identificare le sfide e le opportunità nel rendering del subsurface scattering e a proporre una nuova soluzione, dove quest’ultima vada ad affrontare uno dei limiti delle tecniche già presenti. La qualità di tale proposta verrà quindi verificata mediante la somministrazione di un questionario ad un gruppo di utenti, con l’obbiettivo di stabilire le differenze percettive rilevate dai partecipanti nelle varie configurazioni. I risultati ottenuti verranno, quindi, analizzati per estrapolare informazioni significative in merito alla qualità delle varie tecniche prese in considerazione e come queste si comparino tra loro.



# 2

## Fondamenti Teorici

### 2.1 Unità Radiometriche

Quando si parla di computer grafica uno degli elementi di maggior rilevanza è il comportamento della luce e della sua interazione con la materia. Il risultato di questa interazione determina come noi percepiamo ciò che vediamo. Per poter capire tutto ciò, bisogna, per prima cosa, comprendere come descriviamo la luce. Comunemente nell'ambito della computer grafica si utilizza *l'ottica geometrica* come modello con cui descrivere il comportamento della luce. In questo modello, la luce è rappresentata tramite flussi di particelle, che chiamiamo fotoni. Quest'ultimi si muovono lungo linee rette chiamate raggi, aventi origine da sorgenti luminose, e trasportano con sé un po' di energia. Tali particelle non interferiscono tra di loro e ciascuna di esse ha una certa direzione e lunghezza d'onda. Andiamo quindi a introdurre alcune delle unità radiometriche più importanti, tenendo in considerazione che all'interno di questa tesi si parlerà esclusivamente dello spettro visibile. Non si terrà in conto, quindi, del comportamento su tutto lo spettro.

#### 2.1.1 Flusso Radiante

Comunemente indicato con  $\Phi$ , il *flusso radiante* è la potenza emessa da una fonte di luce (Fig. 2.1). Essendo una potenza è definita come energia radiante emessa nell'unità di tempo, dove l'energia radiante è l'energia delle radiazioni elettromagnetiche emesse nell'ambiente circostante. L'unità di misura nel SI per l'energia radiante è il Joule e di conseguenza il flusso radiante si misura in Watt ( $W$ ). Questa è una delle misure radiometriche fondamentali, in quanto tutte le altre che andremo a utilizzare sono definite in funzione del flusso radiante. Di seguito la sua formulazione matematica, dove il termine  $Q$  rappresenta l'energia, mentre  $t$  è il tempo.

$$\Phi = \frac{\partial Q}{\partial t} \quad (2.1)$$

#### 2.1.2 Irradianza

Una delle misure derivate dal flusso radiante è l'*irradianza* (chiamata anche flux density), che viene descritta come la potenza di radiazione elettromagnetica incidente su una certa superficie per area unitaria, risultando quindi definita come  $Wm^{-2}$ . Tale superficie di incidenza può essere reale o anche

immaginaria, dove, quest'ultima, viene comunemente usata in modelli teorici per studiare la distribuzione della radianza all'interno di un'ipotetica scena. Quando si parla di irradianza, in realtà, si può fare riferimento a diverse tipologie, che vengono qui di seguito presentate.

1. Quando si utilizza semplicemente il termine irradianza, in genere ci si riferisce alla sua definizione classica, ovvero il flusso radiante incidente su un solo lato della superficie di interesse. (Fig. 2.3). Comunemente indicato con  $E$ , la seguente è la sua formulazione matematica, dove  $\Phi_i$  è il flusso radiante ricevuto e  $A$  è l'area di ricezione.

$$E = \frac{\partial \Phi_i}{\partial A} \quad (2.2)$$

2. **Net Irradiance:** in questo caso si considera il flusso radiante incidente su entrambi i lati della superficie, considerando però quello incidente sul lato opposto come un contributo negativo (Eq. 2.3). Comunemente indicato con  $\bar{E}$ , la seguente è la sua formulazione matematica.

$$\bar{E} = E(x, -n) - E(x, n) \quad (2.3)$$

3. **Vector Irradiance:** è una misura vettoriale, a differenza di quelle viste fino ad ora che erano scalari. Viene rappresentata con un vettore di 3 componenti, dove ciascuno di questi componenti è relativo ad uno degli assi delle coordinate che definiscono lo spazio tridimensionale. Quindi, ad ogni componente è associata la relativa net irradiance, calcolata considerando il vettore rappresentante l'asse come la normale utilizzata per determinare se il contributo sia positivo o negativo. Comunemente indicata con  $\vec{E}$ , la seguente è la sua formulazione matematica.

$$\vec{E} = \begin{bmatrix} \bar{E}(x, e_1) \\ \bar{E}(x, e_2) \\ \bar{E}(x, e_3) \end{bmatrix} \quad (2.4)$$

4. **Scalar Irradiance:** comunemente chiamata anche *fluence* (fluence) e rappresentata con  $\phi$ . Considera, invece di superficie, una sfera attorno al punto  $x$ , e considera i contributi di tutti le particelle passanti nel vicinato di  $x$  definito dalla sfera, indipendentemente dalla direzione da cui provengono. Questa misura è comunemente utilizzata quando si parla di rendering volumetrico, o più in generale del comportamento dei fotoni all'interno di un medium. La sua formulazione matematica è la seguente

Mantenendo la potenza costante, l'irradianza diminuisce all'aumentare della distanza dalla sorgente in maniera quadratica. Equivalentemente aumenta al diminuire della distanza dalla sorgente.

### 2.1.3 Radianza

Una delle unità radiometriche più importanti è la *radianza*, definita come flusso radiante emesso o ricevuto, incidente su una certa area  $A$  attorno ad un punto  $x$ , perpendicolare alla direzione  $\omega$  con un certo *angolo solido* definito in steradiani (Fig. 3). Sostanzialmente misura la densità di fotoni che

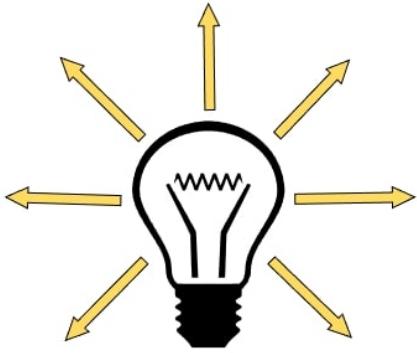


Figura 2.1: Flusso radiante, emesso in tutte le direzioni

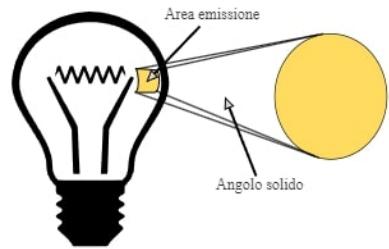


Figura 2.2: Radianza emessa da un area in un angolo solido concentrato nella direzione  $\vec{\omega}$

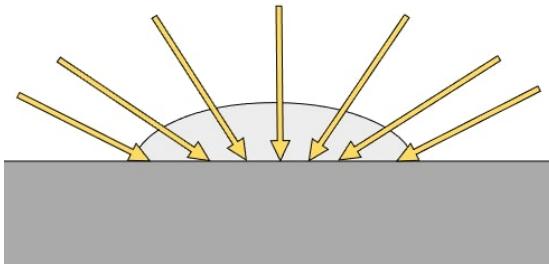


Figura 2.3: Irradianza, considera il flusso radiante proveniente da un emisfero di direzioni incidente su un area differenziale della superficie

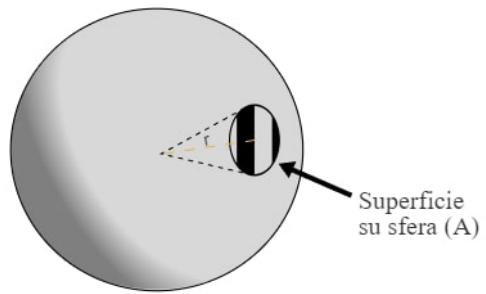


Figura 2.4: Angolo solido dove gli steradiani corrispondono alla superficie sulla sfera nel caso in cui il raggio sia unitario

passano vicino a  $x$  e viaggiano in direzione vicina a  $\omega$ , più precisamente all'interno dell'angolo solido definito. Dettagli di cosa voglia dire “vicino” sono fondamentali per la definizione. Comunemente indicata con  $L$ , la sua formulazione matematica è la seguente, dove  $A \cdot \cos \theta$  è l'area proiettata sulla superficie in base alla direzione di provenienza  $\omega$ ,  $\Omega$  è l'angolo solido e  $\Phi$  è il flusso radiante.

$$L = \frac{\partial^2 \Phi}{\partial \Omega \partial (A \cdot \cos \theta)} \quad (2.5)$$

Gli steradiani, generalmente indicati con  $sr$ , sono l'unità di misura per gli angoli solidi, ovvero l'equivalente di quello che sono i radianti in uno spazio bidimensionale, estesi però nello spazio tridimensionale. Non si parla quindi di angoli nel piano, ma di angoli solidi nello spazio, intuitivamente visualizzabili come coni con la punta nell'origine di una sfera. Così come i radianti sono definiti sulla circonferenza di un cerchio nel piano, gli steradiani sono definiti come proiezioni sulla superficie di una sfera con un certo raggio  $r$  (Fig. 2.4). In particolare, l'equazione per calcolare gli steradiani è data dal rapporto fra due superfici (Eq. 2.6), risultando quindi essere un numero puro, ovvero adimensionale. La prima superficie ( $A$ ) è data dall'area sulla superficie della sfera in considerazione, circoscritta dalle semirette partenti dall'origine che descrivono il cono avente un certo angolo solido. La seconda “superficie” è semplicemente il quadrato del raggio della sfera. Comunemente quest'ultima ha raggio unitario,

con la conseguenza che gli steradiani sono semplicemente l'area  $A$ .

$$\Omega = A/r^2 \quad (2.6)$$

L'unità di misura per la radianza, quindi, risulta essere definita come  $Wm^{-2}sr^{-1}$ . E' evidente quindi che per aumentare la radianza di una fonte di luce ci sono una serie di opzioni:

1. La più elementare consiste nel semplicemente aumentare la potenza emessa dalla fonte di luce
2. Un'altra possibilità è diminuire l'area da cui viene emessa tale luce
3. L'ultima opzione è quella di ridurre l'angolo di emissione, andando sostanzialmente a concentrare la stessa potenza su un'area più piccola.

Ovviamente per, invece, ridurre la radianza, il processo è equivalente ed opposto.

Una proprietà importante della radianza è che, in assenza di alcuna forma di attenuazione, risulta essere costante lungo il raggio. Questo è evidente dalla relazione fra l'intensità della sorgente  $I$  e l'angolo solido  $\Omega$  che descrivono la radianza. L'intensità segue la legge del inverso del quadrato, ovvero diminuisce proporzionalmente al quadrato della distanza dalla sorgente.

$$I \propto \frac{1}{r^2} \quad (2.7)$$

L'angolo solido segue la stessa proporzionalità.

$$\Omega \propto \frac{1}{r^2} \quad (2.8)$$

Ciò che ne risulta è che, essendo la radianza definita dall'intensità per angolo solido, il raggio al quadrato di intensità ed angolo solido si cancellano a vicenda, rendendo la radianza indipendente dalla distanza.

$$L \propto \frac{I}{\Omega} \quad (2.9)$$

## 2.2 Teoria Rendering

Quando si parla di rendering si fa riferimento al processo di sintetizzazione di un'immagine, foto-realistica o meno, a partire da modelli 2D o 3D. Nella maggior parte dei casi le scene renderizzate mostrano l'interazione di un medium con una o più fonti di luce. Questo fenomeno è uno degli aspetti principali, se non quello principale, quando si vogliono renderizzare immagini verosimili. Considerando la complessità necessaria per descrivere tale tipo di fenomeno fisico, si è storicamente usato realizzare una moltitudine di modelli che rappresentassero solo una parte degli aspetti inerenti l'interazione della luce con un medium. Un cambiamento in questo senso avviene nel 1986, durante il quale Kajiya presenta il suo paper seminale sulla *Rendering Equation*.

### 2.2.1 Rendering Equation

La Rendering Equation [23], chiamata anche Light Transport Equation, è una specializzazione della *Radiative Transfer Equation* in assenza di media partecipanti e vuole andare a sostituire tutta una

serie di algoritmi di rendering precedenti e fornire un framework unificato con cui guardare a quello che sostanzialmente è un unico problema. Lo fa definendo un'equazione integrale sotto la quale la radianza uscente da un punto è definita dalla somma di un termine relativo alla radianza emessa e di un termine che descrive la radianza riflessa (Eq. 2.10). Questa valutazione avviene nel contesto in cui seguiamo l'approssimazione ottica geometrica. Ciò vuol dire che approssimiamo la luce come raggi su cui si spostano in serie delle particelle che trasportano energia, piuttosto che associare il comportamento della luce a delle onde.

$$L_o(x, \vec{\omega}_o) = L_e(x, \vec{\omega}_o) + \int_{\Omega} f_r(x, \vec{\omega}_i, \vec{\omega}_o) L_i(x, \vec{\omega}_i) (\vec{\omega}_i \bullet \vec{n}) d\vec{\omega}_i \quad (2.10)$$

Nell'equazione precedente  $L_o$  e  $L_i$  rappresentano rispettivamente la radianza uscente e quella entrante. Anche  $L_e$  indica la radianza uscente, ma, più in particolare, la radianza che viene emessa da quel punto. Comunemente, quando si considera una superficie "normale", il valore di  $L_e$  è nullo, in quanto la superficie non emette energia propria. Questo non è il caso per superfici che sono anche fonti di luce (e.g. lampadine). In ogni caso, la radianza viene valutata in un punto specifico della superficie, corrispondente ad  $x$ . I due omega,  $\omega_o$  ed  $\omega_i$ , invece, indicano rispettivamente le direzioni di uscita della radianza dal punto  $x$  e le direzioni della radianza incidente sul punto  $x$ . Per quanto riguarda la radianza entrante, non si considera una sola direzione, ma tutte le direzioni rappresentate dall'emisfero indicato con  $\Omega$  ed è proprio per questo che viene applicato l'integrale. La normale della superficie è rappresentata da  $\vec{n}$  ed il suo prodotto scalare con la direzione di incidenza rappresenta un fattore di riduzione dell'irradianza. Infine abbiamo la funzione  $f_r$  che, dato un punto  $x$  sulla superficie, indica quanta della luce proveniente dalla direzione  $\omega_i$  viene riflessa nella direzione  $\omega_o$ . Questa funzione viene comunemente chiamata *Bi-directional Reflectance Distribution Function* (BRDF) e sostanzialmente descrive le caratteristiche del materiale con cui la luce interagisce. Esistono una moltitudine di modelli utilizzati come BRDF, come per esempio, nel caso più semplice, il modello lambertiano, che sostanzialmente descrive un comportamento perfettamente diffusivo. Sono presenti anche altri modelli più avanzati che descrivono in maniera più accurata tale comportamento, in quanto, nonostante il modello lambertiano sia physically-based, la maggior parte dei diffusori non presenta una diffusione perfetta [5]. In particolare, molti esibiscono cali o picchi nella retroriflessione ad angoli radenti [5]. In maniera equivalente ci sono anche modelli che descrivono il termine speculare, come per esempio il modello Cook-Torrance. Tutti questi modelli, però, sono limitati al fatto che rappresentano solo interazioni su una superficie in un unico punto. Questo, in realtà, è più un limite della Rendering Equation, che nonostante la sua generalità, non modella tutte le possibili situazioni. In particolare è di nostro interesse il fatto che non gestisca la trasmissione ed il subsurface scattering, ovvero fenomeni in cui il punto di entrata ed uscita della radianza non è necessariamente lo stesso, dato che avviene lo scattering non soltanto sulla superficie, ma anche dentro al volume. Proprio per gestire questo tipo di casi, esiste una generalizzazione della Rendering Equation, chiamata *Volume Rendering Equation*.

## 2.2.2 Volume Rendering Equation

La Volume Rendering Equation (VRE), conosciuta anche come *Radiative Transfer Equation* (RTE), è un fondamentale concetto nell'ambito della computer grafica, particolarmente rilevante nel rendering

di materiali che coinvolgono l’interazione della luce con medium traslucidi, come quelli che presentano fenomeni di subsurface scattering, sia nel caso di medium omogeneo che eterogeneo (per medium eterogeneo si intende che le sue proprietà variano nello spazio, a differenze delle proprietà che rimangono costanti nel caso di un medium omogeneo). A differenza della Rendering Equation classica, la Volume Rendering Equation si concentra sulla modellazione dell’interazione della luce all’interno di un volume tridimensionale anziché su superfici bidimensionali. Quindi, mentre la Rendering Equation classica si concentra sul calcolo della luce che colpisce e viene riflesso da superfici esterne, la Volume Rendering Equation considera l’interazione della luce all’interno del volume, consentendo di rappresentare in modo accurato materiali con comportamenti ottici complessi. La RTE ha una formulazione matematica che descrive come la luce interagisce con le particelle all’interno di uno spazio volumetrico. La forma generale della RTE viene espressa come un’equazione integro-differenziale 2.11, che descrive la variazione della radianza lungo una direzione  $\omega$ , causata da fenomeni di *assorbimento* e di *scattering* (sia interno che esterno) della radianza all’interno del volume, oltre che dal *emissione*. Questa equazione è fondamentale per descrivere l’interazione della luce con materiali che presentano fenomeni di translucenza, come la pelle, il latte, la cera o il marmo, dove la luce penetra nel materiale e interagisce con esso prima di emergere dalla superficie lontano dal punto di incidenza. Oltre, però a modellare il classico subsurface scattering per materiali opachi, viene anche utilizzata per gestire fenomeni quali fumo, nuvole, nebbia e molti altri.

$$(\vec{\omega}_o \bullet \nabla) L(x, \vec{\omega}_o) = -\sigma_t L(x, \vec{\omega}_o) + \sigma_s L_s(x, \vec{\omega}_o) \quad (2.11)$$

È evidente da questa equazione come vari la radianza. Ovvvero, c’è una perdita di radianza causata da assorbimento ed *out-scattering* indicata dal primo termine ed un guadagno causato da *in-scattering* ed emissione indicato dal secondo termine. Andiamo quindi ad analizzare singolarmente questi fenomeni:

### 1. Assorbimento

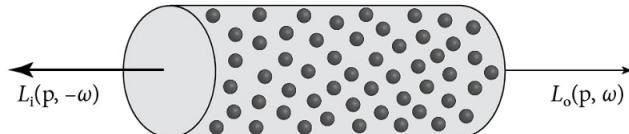


Figura 2.5: Assorbimento all’interno di un cilindro differenziale [27]

Fenomeno per cui la radianza viene ridotta a causa della trasformazione della luce in un’altra forma di energia. Ciò viene causato dall’interazione di luce con le particelle del medium. Viene descritto mediante il coefficiente di assorbimento  $\sigma_a$  che indica la probabilità che la luce venga assorbita per unità di distanza. Notare che non è un numero puro, ma è descritto in termini del reciproco della distanza ( $m^{-1}$ ) e di conseguenza non è neanche limitato all’intervallo  $[0, 1]$ . È matematicamente descritta come la variazione di radianza lungo una distanza differentiale  $dr$  lungo il raggio o più generalmente la differenza fra la radianza uscente e quella entrante.

$$dL(x, \vec{\omega}_o) = -\sigma_a L(x, \vec{\omega}_i) dr \quad (2.12)$$

### 2. Out-Scattering

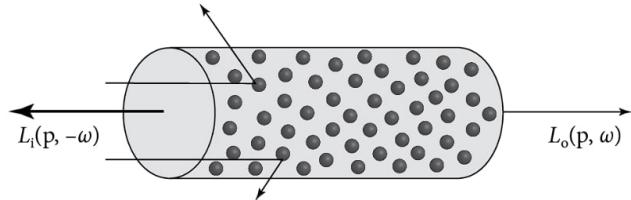


Figura 2.6: Out-scattering all'interno di un cilindro differenziale [27]

Fenomeno per cui la radianza viene ridotta a causa di eventi di scattering che portano i raggi di luce che viaggiavano in direzione  $\vec{\omega}_o$  ad essere rifiutati in un'altra direzione. Viene descritto sfruttando il coefficiente di scattering  $\sigma_s$ , che indica la probabilità che la luce subisca un evento di scattering (indipendentemente che sia in o out) per unità di distanza. Così come il coefficiente di assorbimento, anche il coefficiente di scattering è definito in termini del reciproco della distanza. La formulazione matematica dell'out-scattering è la seguente.

$$dL(x, \vec{\omega}_o) = -\sigma_s L(x, \vec{\omega}_i) dr \quad (2.13)$$

L'assorbimento insieme all'out-scattering formano il primo termine della RTE, ovvero quello riguardante la riduzione di radianza.

### 3. In-Scattering

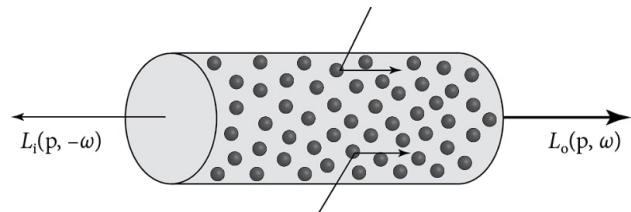


Figura 2.7: In-scattering all'interno di un cilindro differenziale [27]

Fenomeno per cui vi è un guadagno di radianza a causa di eventi di scattering che portano i raggi di luce che provenivano dall'esterno ad essere rifiutati all'interno del volume nella direzione  $\vec{\omega}_o$ . In questo caso è fondamentale la funzione di fase, che approfondiremo successivamente, ma, in breve, descrive la probabilità che data una direzione di arrivo  $\vec{\omega}_i$ , il raggio venga rifiutato in direzione  $\vec{\omega}_o$ . Possiamo quindi fare l'integrale in tutte le direzioni entranti, modulato dalla funzione di fase, per stabilire quali di questi raggi venga rifiutato in direzione  $\vec{\omega}_o$ . Applichiamo poi il coefficiente di scattering  $\sigma_s$  per "filtrare" la frazione di raggi che effettivamente hanno subito un evento di scattering. Di seguito la formulazione matematica del processo appena descritto, dove  $\Omega_{4\pi}$  rappresenta entrambi gli emisferi.

$$dL(x, \vec{\omega}_o) = \sigma_s \int_{\Omega_{4\pi}} p(x, \vec{\omega}_o, \vec{\omega}_i) L(x, \vec{\omega}_i) d\vec{\omega}_i dr \quad (2.14)$$

### 4. Emissione

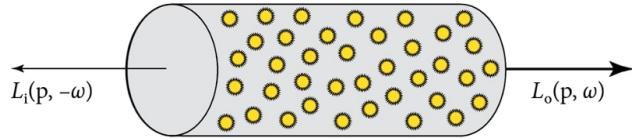


Figura 2.8: Emissione all'interno di un cilindro differenziale [27]

Fenomeno per cui vi è un guadagno di radianza a causa di un qualche fenomeno chimico all'interno del volume, che porta alla trasformazione di energia in luce (e.g. esplosione), con conseguente emissione nell'ambiente circostante. Di seguito la sua formulazione matematica, in cui  $L_e(x, \vec{\omega}_o)$  indica la radianza emessa nella direzione  $\vec{\omega}_o$ .

$$dL(x, \vec{\omega}_o) = \sigma_a L_e(x, \vec{\omega}_o) dr \quad (2.15)$$

L'emissione insieme all'in-scattering formano il secondo termine della RTE, ovvero quello riguardante il guadagno di radianza.

Il termine  $\sigma_t$  all'interno della RTE (Eq. 2.11) è semplicemente la somma del coefficiente di assorbimento e del coefficiente di scattering, ovvero il *coefficiente di attenuazione totale* (Eq. 2.16). Esso descrive la probabilità che un fotone subisca un qualsiasi forma di interazione con il medium, che sia scattering o assorbimento.

$$\sigma_t = \sigma_a + \sigma_s \quad (2.16)$$

Da questo coefficiente possiamo anche stabilire un nuovo valore che ci sarà utile in futuro, ovvero il *mean free path*, definito come il reciproco del coefficiente di attenuazione totale (Eq. 2.17). Considerando da cosa è derivato, è facile vedere che corrisponde alla distanza media che un fotone percorre prima di subire un evento di scattering o assorbimento.

$$\ell = \frac{1}{\sigma_t} \quad (2.17)$$

Un ulteriore valore che possiamo introdurre è l'albedo  $\alpha$ , più precisamente chiamato *single-scattering albedo* (Eq. 2.18). Nonostante il suo valore abbia implicazioni sul colore di un medium, non corrisponde al significato comunemente associato alla parola albedo nel rendering, ovvero al colore diffusivo, per esempio, di una texture. Invece, è definito come la probabilità (in questo caso effettivamente nell'intervallo  $[0, 1]$ ) che l'evento di interazione di un fotone con una particella di un medium sia un evento di scattering, piuttosto che di assorbimento.

$$\alpha = \frac{\sigma_s}{\sigma_t} \quad (2.18)$$

All'interno della RTE, la funzione di fase è un concetto cruciale che descrive la distribuzione angolare della luce all'interno di un mezzo partecipante. Questa funzione determina come la luce viene riflessa all'interno del volume e influenza quindi l'aspetto finale del materiale rendendolo più o meno traslucido, opaco o diffusivo. La funzione di fase si suddivide sostanzialmente in due tipologie, ovvero:

- Isotropa** : la probabilità che venga scelta una certa direzione è equivalente per tutte le direzioni di entrambi gli emisferi.

**2. Anisotropia :** certe direzioni hanno più probabilità di essere scelte rispetto ad altre. Il modello di Henyey-Greenstein è una funzione di fase approssimata, controllata da un *parametro di anisotropia*  $g$ . Quando questo parametro si avvicina a  $-1$  la distribuzione angolare tende verso la direzione di provenienza del raggio e si parla quindi di *back scattering*. Nel caso, invece,  $g$  tenda verso  $1$ , la distribuzione angolare tenderà di più nella stessa direzione di quella di arrivo e si parla quindi di *forward scattering*.

La funzione di fase deve anche soddisfare ad una condizione, ovvero che:

$$\int_{\Omega_{4\pi}} p(x, \vec{\omega}_o, \vec{\omega}_i) d\vec{\omega}_i = 1 \quad (2.19)$$

Ciò significa che può essere considerata una funzione di densità di probabilità. Inoltre, la funzione di fase può variare a seconda delle proprietà del materiale, come la composizione chimica, la struttura microscopica e la densità del volume. Nei materiali omogenei, può essere approssimata in modo semplice, ad esempio tramite modelli come quello di Henyey-Greenstein. Tuttavia, nei materiali più complessi o con strutture microscopiche irregolari, la funzione di fase può essere più difficile da modellare e quindi richiedere approcci più sofisticati, come la simulazione di scattering multipli o l'uso di dati sperimentali per determinare la distribuzione angolare della luce.

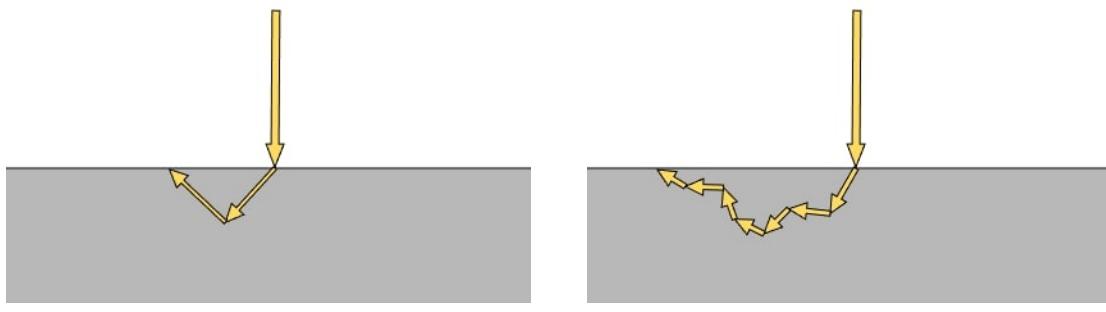
## 2.3 Teoria del Subsurface Scattering

La ricerca nella modellazione del fenomeno di subsurface scattering è un ambito molto ampio e negli anni sono state proposte una moltitudine di soluzioni. In questo capitolo cerchiamo di presentare alcune delle proposte più significative. Non sarà quindi assolutamente una lista esaustiva, ma i modelli che vedremo sono alcuni di quelli che hanno avuto una grossa importanza storica e, soprattutto, sono i modelli che sono stati usati per derivare le varie tecniche utilizzate per implementare il subsurface scattering in tempo reale.

Due dei modelli che presenteremo di seguito, ovvero il modello dipolo ed il modello multipolo, risultano essere particolarmente complessi a livello teorico e le sole pubblicazioni su cui sono stati presentati non sono sufficienti per una comprensione totale. Fortunatamente Mikkelsen, [24] durante una sua analisi sullo stato dell'arte riguardante il rendering della pelle, presenta una disamina completa ed approfondita di tutti gli aspetti dei due modelli. Quindi, vorrei far notare che nelle prossime due sezioni gran parte dei concetti e delle informazioni sono tratte da tale lavoro [24].

### 2.3.1 Modello Dipole

Nel 2001 Jensen [19] presenta il suo lavoro seminale nella definizione di una BSSRDF per simulare in maniera efficiente gli effetti che un modello BRDF non può catturare, come il bleeding del colore all'interno dei materiali e diffusione della luce lungo i bordi delle ombre, ovvero il fenomeno di subsurface scattering. Il nuovo metodo modella separatamente l'effetto di *single scattering* e di *multiple scattering* (Fig. 2.9). Il fenomeno di single scattering viene implementato con una soluzione esatta, mentre quello di multiple scattering viene approssimato mediante un modello di diffusione basato su un dipolo.



(a) single scattering, dove il raggio subisce un unico fenomeno di scattering prima di uscire dalla superficie

(b) multiple scattering, dove il raggio subisce multipli fenomeni di scattering prima di uscire dalla superficie

Figura 2.9: Fenomeno di (a) single e (b) multiple scattering

Precedentemente a questo lavoro la ricerca si è concentrata sullo sviluppo di modelli per la Bidirectional Reflectance Distribution Function (BRDF). Questo tipo di modelli fa un'assunzione particolarmente significativa, ovvero che la luce che entra in un certo punto, venga riflessa sempre dallo stesso punto. In sostanza non considera alcun tipo di subsurface scattering globale, con la conseguenza che indipendentemente dalla qualità dell'algoritmo di light transport utilizzato non sarà possibile produrre risultati convincenti per una vasta gamma di materiali con un modello così semplificato. Mentre questa semplificazione può funzionare per i metalli, in quanto essi non presentano alcuna translucenza, non vale lo stesso per i materiali dielettrici, in particolare per tessuti organici come la pelle. In questi casi il subsurface scattering viene approssimato da una semplice componente lambertiana, che risulta ovviamente insufficiente in quanto considera un raggio estremamente piccolo per il subsurface scattering. Più precisamente, il raggio è talmente piccolo che ricade dentro un singolo pixel. Il subsurface transport può essere simulato in maniera accurata andando ad utilizzare metodi che risolvono l'intera Radiative Transfer Equation, come per esempio mediante simulazioni di Monte Carlo. Benché questo tipo di metodi simuli correttamente tutti gli aspetti del subsurface scattering e produca risultati estremamente accurati, è al contempo computazionalmente estremamente costoso rispetto al rendering di materiali opachi che non presentano effetti di translucenza, in particolare quando il medium ha un alto coefficiente di scattering (equivalentemente si può dire medium ad *alto albedo*). Un medium ad alto albedo implica che poca energia venga assorbita ad ogni evento di scattering, necessitando conseguentemente la simulazione di molti di questi ultimi. Per esempio, il latte al cioccolato ha un valore di single scattering albedo pari a 0.99883 nella banda delle frequenze rappresentanti il colore rosso. Ciò implica che l'assorbimento ha un impatto estremamente piccolo sull'estinzione dell'energia ad ogni evento di scattering. Infatti possiamo facilmente calcolare e vedere come:

1. Dopo 100 eventi di scattering, l'88.95% dell'energia originale è ancora presente
2. Dopo 1000 eventi di scattering, il 31% dell'energia originale è ancora presente
3. Dopo 3000 eventi di scattering, il 2.98% dell'energia originale è ancora presente

In contrapposizione, un medium come il succo di mela ha nella stessa banda di frequenze un single scattering albedo pari a 0.1654. In questo caso, dopo soli 5 eventi di scattering, la percentuale di energia originale preservata è al 0.01%.

Questi approcci in cui si simula il processo completo di volume scattering tramite metodi di Monte Carlo quali il Photon Mapping, non sono gli unici per renderizzare materiali traslucidi. È possibile farlo, anche, mediante l'impiego di una BSSRDF che approssima tale processo, andando a considerare solo il punto di entrata e di uscita, oltre che alle direzioni di entrata ed uscita sul boundary del medium. Il problema è che la BSSRDF è comunque particolarmente complessa, essendo una funzione 8D (2.20). Questa dimensionalità è data dal fatto che, oltre all'azimut ed al zenith dei due vettori rappresentanti le direzioni di entrata ed uscita della luce, è necessario considerare anche le coordinate dei due punti di entrata ed uscita, a differenza di una BRDF dove la luce viene riflessa nello stesso punto:

$$S(x_i, \vec{\omega}_i, x_o, \vec{\omega}_o) = S(x_i, y_i, \theta_i, \phi_i, x_o, y_o, \theta_o, \phi_o) \quad (2.20)$$

Il problema dell'utilizzo della BSSRDF in questa forma completa e la conseguenza di dover campionare uno spazio 8-dimensionale, che risulta ancora poco praticabile a livello produttivo.

Inoltre, determinare una funzione per  $S$  non è semplice. Generalmente per le BRDF viene fatta una separazione in 2 termini (2.21), uno rappresentante la componente speculare ed un altro la componente diffusiva.

$$f \simeq f_r + f_d \quad (2.21)$$

Questa semplificazione usata per approssimare una BRDF può sembrare non adatta per una BSSRDF, ma in realtà per medium anisotropi con un alto coefficiente di scattering, quale la pelle, è abbastanza naturale, in quanto, come vedremo in maniera più approfondita successivamente, dopo un alto numero di eventi di scattering la luce perde completamente di direzionalità, assumendo un comportamento isotropo. Ciò che possiamo fare, quindi, è controllare la frazione di luce che viene riflessa nel punto  $x_o$  con un termine speculare simile a quello utilizzato in una normale BRDF. Mentre la frazione di luce incidente in  $x_i$ , che viene trasmessa dentro il medium e subisce una serie di eventi di scattering prima di uscire dal punto  $x_o$  in direzione  $\vec{\omega}_o$ , la possiamo approssimare con una componente diffusiva  $S_d$  (2.22).

$$S(x_i, \vec{\omega}_i, x_o, \vec{\omega}_o) \simeq S_d(x_i, \vec{\omega}_i, x_o, \vec{\omega}_o) + S_r(x_o, \vec{\omega}_i, \vec{\omega}_o) \quad (2.22)$$

Il termine  $S_d$  rimane ancora un termine con tutti i parametri iniziali. Da notare però, che in questo caso la dipendenza da  $\vec{\omega}_i$  e  $\vec{\omega}_o$  è dettata solo dalla differenza degli indici di rifrazione dei due media che formano l'interfaccia. Questa differenza determina la frazione di luce che viene riflessa ( $\rho_r(x, \vec{\omega}_i)$ ) e la frazione che viene trasmessa ( $\rho_t(x, \vec{\omega}_i)$ ). Generalmente la quantità riflessa è determinata mediante equazioni di Fresnel o, ancora più semplicemente, mediante l'approssimazione fornita da Schlick. Possiamo usare Fresnel in quanto accettiamo di soddisfare approssimativamente i suoi vincoli, ovvero che l'interfaccia, in questo caso aria/pelle, sia planare e che i medium siano omogenei. Avendo la frazione di luce riflessa, possiamo ottenere la frazione trasmessa banalmente sottraendo ad 1 la quantità riflessa, ovvero  $\rho_t = 1 - \rho_r$ . Ciò significa che all'interno della funzione  $S_d$  i valori di  $\vec{\omega}_i$  e  $\vec{\omega}_o$  verranno usati esclusivamente per calcolare la frazione di luce riflessa e trasmessa, mediante le equazioni di Fresnel (o l'approssimazione di Schlick). È possibile semplificarla ulteriormente, in quanto, come vedremo successivamente, i profili diffusivi che ottiene Jensen sono radialmente simmetrici. Ciò ci permette di non dover più considerare

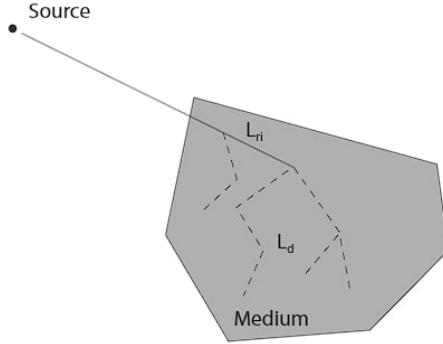


Figura 2.10: La radianza ridotta  $L_{ri}$  è indicata con la linea netta, mentre la radianza diffusiva  $L_d$  è indicata con la linea tratteggiata [24]

esplicitamente i due punti  $x_i$  e  $x_o$ , ma solo la distanza fra loro, dove quest'ultima rappresenta il "raggio" nel profilo diffusivo, ovvero la distanza dal punto di entrata.

È necessario quindi fornire una formulazione per il termine  $S_d$ . Come vedremo è possibile fare ciò analizzando le fluttuazioni di densità della luce entrante nel medium e fortunatamente questo processo è implicitamente descritto dalla *Diffusion Approximation*. Quest'ultima è formata da due parti:

1. **Diffusion Equation:** è un'equazione differenziale usata comunemente nell'ambito della fisica per descrivere a livello macroscopico il comportamento di particelle in moto Browniano, causato dal movimento e dalle collisioni casuali delle particelle. Qui possiamo intravedere l'utilità di questa equazione nel nostro caso, in quanto, come abbiamo visto in precedenza, un medium con un alto coefficiente di scattering porta ad uno scattering che risulta essenzialmente casuale, ossia un comportamento simile a quello descritto dalla Diffusion Equation.
2. **Boundary Condition :** è un vincolo o requisito che applichiamo all'equazione differenziale che stiamo prendendo in considerazione, dove tale vincolo descrive specificatamente le condizioni in cui stiamo cercando di risolvere il problema fisico.

Per quanto riguarda la Diffusion Equation, la otteniamo a partire dalla Radiative Transfer Equation (RTE) di cui abbiamo già discusso in precedenza. Per fare ciò, partiamo dal fatto che comunemente per studiare il trasporto, all'interno di un medium omogeneo, della radianza, quest'ultima viene separata in due termini:

- **reduced radiance ( $L_{ri}$ ):** la parte della radianza entrante che non è ancora stata completamente assorbita e che non ha ancora subito alcun evento di scattering (linea continua nella figura 2.10).
- **diffuse radiance ( $L_d$ ):** la parte della radianza che viene creata all'interno del medium, attraverso fenomeni di emissione o in-scattering da parte della radianza ridotta/diffusiva (linea tratteggiata nella figura 2.10).

Con questa formulazione possiamo visualizzare la radianza guadagnata dalla radianza diffusiva mediante in-scattering della radianza ridotta come una forma di emissione.

$$\varepsilon(x, \vec{\omega}_o) = \sigma_s \int_{\Omega_{4\pi}} p(x, \vec{\omega}_o, \vec{\omega}_i) L_{ri}(x, \vec{\omega}_i) d\vec{\omega}_i \quad (2.23)$$

La Radiative Transfer Equation della radianza diffusiva  $L_d$  può essere quindi riscritta includendo la nuova componente emissiva  $\varepsilon$ .

$$(\vec{\omega} \bullet \nabla) L_d(x, \vec{\omega}) = -\sigma_t L_d(x, \vec{\omega}) + \sigma_s \int_{\Omega_{4\pi}} p(x, \vec{\omega}', \vec{\omega}) L_d(x, \vec{\omega}') d\vec{\omega}' + \varepsilon(x, \vec{\omega}) \quad (2.24)$$

Questa nuova RTE per la radianza diffusiva può essere approssimata con l'utilizzo di 2 termini, ovvero l'irradianza scalare ( $\phi$ ), chiamata anche fluenza, e l'irradianza vettoriale ( $\vec{E}$ ), dai quali andremo poi ad ottenere quella che viene chiamata diffusion equation. Tale approssimazione viene ottenuta mediante la proiezione della radianza diffusiva  $L_d$  sulle prime 2 bande delle armoniche sferiche. Per poter fare la proiezione su una base ortonormale, come quella delle armoniche sferiche, è necessario che la funzione proiettata sia almeno continua a tratti. In questo caso stiamo prendendo in considerazione la radianza, che viene concettualmente trattata come una funzione continua, quando in realtà può presentare delle discontinuità causate da bruschi cambiamenti nei valori di riflettanza e trasmittanza. Nonostante la possibilità di tale discontinuità, assumiamo che la funzione sia continua a tratti. Ciò ci permette di fare la proiezione, utilizzando  $l = 2$ , dove  $l \in \mathbb{N}$  ed  $l$  rappresenta il numero di bande su cui fare la proiezione, ottenendo la seguente equazione.

$$L_d(\vec{\omega}_i) \simeq \frac{1}{4\pi} \phi(x) + \frac{3}{4\pi} \vec{E}(x) \bullet \vec{\omega}_i \quad (2.25)$$

A questo punto abbiamo tutti gli strumenti necessari per derivare la diffusion equation. Per prima cosa definiamo l'irradianza scalare  $\phi$  e l'irradianza vettoriale  $\vec{E}$  in termini della componente emissiva  $\varepsilon$  precedentemente introdotta, che chiameremo rispettivamente  $Q_0$  e  $\vec{Q}_1$ .

$$Q_0(x) = \int_{\Omega_{4\pi}} \varepsilon(x, \vec{\omega}_o) d\vec{\omega}_o \quad (2.26)$$

$$\vec{Q}_1(x) = \int_{\Omega_{4\pi}} \varepsilon(x, \vec{\omega}_o) \vec{\omega}_o d\vec{\omega}_o \quad (2.27)$$

Integriamo quindi entrambi i lati della RTE della radianza diffusiva (2.24) ottenendo quindi l'equazione 2.30, la quale descrive la divergenza dell'irradianza vettoriale. Integriamo per prima cosa la parte sinistra.

$$\begin{aligned} \int_{\Omega_{4\pi}} (\vec{\omega}_o \bullet \nabla) L_d(x, \vec{\omega}_o) d\vec{\omega}_o &= \nabla \bullet \int_{\Omega_{4\pi}} L_d(x, \vec{\omega}_o) \vec{\omega}_o d\vec{\omega}_o \\ &= \nabla \bullet \vec{E}(x) \end{aligned} \quad (2.28)$$

Proseguendo poi con l'integrazione della parte destra.

$$\begin{aligned} \int_{\Omega_{4\pi}} -\sigma_t L_d(x, \vec{\omega}) + \sigma_s \int_{\Omega_{4\pi}} p(x, \vec{\omega}', \vec{\omega}) L_d(x, \vec{\omega}') d\vec{\omega}' + \varepsilon(x, \vec{\omega}) d\vec{\omega} &= \int_{\Omega_{4\pi}} -\sigma_t L_d(x, \vec{\omega}) \\ &\quad + \sigma_s \phi(x) + \varepsilon(x, \vec{\omega}) d\vec{\omega} \\ &= -\sigma_a \phi(x) - \sigma_s \phi(x) + \sigma_s \phi(x) + Q_0(x) \\ &= -\sigma_a \phi(x) + Q_0(x) \end{aligned} \quad (2.29)$$

Dall'integrazione abbiamo quindi ottenuto la seguente equazione, che descrive la divergenza dell'irradianza vettoriale, ovvero quanto flusso tende ad entrare o ad uscire da un volume infinitesimale nel punto  $x$ . È evidente come la perdita sia dettata dall'assorbimento, mentre il guadagno sia dato da una qualsiasi sorgente che inietti radianza ridotta all'interno del medium.

$$\nabla \bullet \vec{E}(x) = -\sigma_a \phi(x) + Q_0(x) \quad (2.30)$$

Per procedere possiamo sostituire la radianza diffusiva approssimata  $s_2(L_d)$  all'interno della RTE per la radianza diffusiva (2.24). Per fare ciò, necessario che consideriamo una serie di identità e definizioni utili, quali:

- Coordinate sferiche:** il coseno di un angolo  $\delta$  può essere descritto in termini di coordinate sferiche. Utilizziamo la convenzione matematica per rappresentare le coordinate, ovvero con  $\theta$  indichiamo l'azimut, mentre con  $\varphi$  indichiamo il zenith.

$$\cos \delta = \cos \varphi_1 \cos \varphi_2 + \sin \varphi_1 \sin \varphi_2 \cos (\theta_1 - \theta_2) \quad (2.31)$$

- Integrale coseno:** l'integrale del coseno nell'intervallo  $[0; 2\pi[$  è nullo.

$$\int_0^{2\pi} \cos \theta d\theta = 0 \quad (2.32)$$

- Coseno medio:**  $g$  è il coseno medio della funzione di fase, equivalente al parametro di anisotropia utilizzato per la funzione di fase definita da Henyey-Greenstein.

$$g = \int_{\Omega_{4\pi}} p(\vec{\omega}_i \bullet \vec{\omega}_o)(\vec{\omega}_i \bullet \vec{\omega}_o) d\vec{\omega}_i \quad (2.33)$$

- Coefficienti ridotti:** il nuovo *coefficiente di scattering ridotto*  $\sigma'_s$  scala il coefficiente di scattering originale con il parametro di anisotropia  $g$  della funzione di fase. Intuitivamente, quando la luce diventa isotropa, solo il backward scattering influisce sul flusso radiante netto, mentre il forward scattering è essenzialmente equivalente all'assenza di scattering [19]. Ovviamente introduciamo di conseguenza anche il *coefficiente di attenuazione totale ridotto*  $\sigma'_t$  per considerare il nuovo coefficiente  $\sigma'_s$ .

$$\sigma'_s = \sigma_s(1 - g) \quad (2.34)$$

$$\sigma'_t = \sigma'_s + \sigma_a \quad (2.35)$$

- Identità vettore:** è possibile definire delle identità per un vettore  $\vec{v}$  qualsiasi (queste identità sono fornite da [18]).

$$\int_{\Omega_{4\pi}} \vec{\omega}(\vec{\omega} \bullet \vec{v}) d\vec{\omega} = \frac{4\pi}{3} \vec{v} \quad (2.36)$$

$$\int_{\Omega_{4\pi}} \vec{\omega}(\vec{\omega} \bullet \nabla(\vec{v} \bullet \vec{\omega})) d\vec{\omega} = \vec{0} \quad (2.37)$$

A questo punto possiamo procedere, sostituendo inizialmente la radianza diffusiva approssimata  $s_2(L_d)$  solo all'interno dell'integrale della RTE della radianza diffusiva, per rendere i passaggi più chiari.

$$\begin{aligned}
\int_{\Omega_{4\pi}} p(\vec{\omega}_i \bullet \vec{\omega}_o) s_2(L_d)(\vec{\omega}_i) d\vec{\omega}_i &= \int_{\Omega_{4\pi}} p(\vec{\omega}_i \bullet \vec{\omega}_o) \left( \frac{1}{4\pi} \phi(x) + \frac{3}{4\pi} \vec{E}(x) \bullet \vec{\omega}_i \right) d\vec{\omega}_i \\
&= \int_{\Omega_{4\pi}} \frac{1}{4\pi} \phi(x) p(\vec{\omega}_i \bullet \vec{\omega}_o) d\vec{\omega}_i + \int_{\Omega_{4\pi}} \frac{3}{4\pi} (\vec{E}(x) \bullet \vec{\omega}_i) p(\vec{\omega}_i \bullet \vec{\omega}_o) d\vec{\omega}_i \\
&= \frac{1}{4\pi} \phi(x) + \frac{3}{4\pi} \int_{\Omega_{4\pi}} p(\vec{\omega}_i \bullet \vec{\omega}_o) \|\vec{E}(x)\| \|\vec{\omega}_i\| \cos \delta d\vec{\omega}_i \\
&= \frac{1}{4\pi} \phi(x) + \frac{3}{4\pi} \int_{\Omega_{4\pi}} p(\vec{\omega}_i \bullet \vec{\omega}_o) (\|\vec{E}(x)\| \cos \varphi_1 \cos \varphi_2 \\
&\quad + \|\vec{E}(x)\| \sin \varphi_1 \sin \varphi_2 \cos (\theta_1 - \theta_2)) d\vec{\omega}_i \\
&= \frac{1}{4\pi} \phi(x) + \frac{3}{4\pi} \int_{\Omega_{4\pi}} p(\vec{\omega}_i \bullet \vec{\omega}_o) (\vec{E}(x) \bullet \vec{\omega}_o) (\vec{\omega}_o \bullet \vec{\omega}_i) d\vec{\omega}_i \\
&\quad + \frac{3}{4\pi} \int_{\Omega_{4\pi}} p(\vec{\omega}_i \bullet \vec{\omega}_o) \|\vec{E}(x)\| \sin \varphi_1 \sin \varphi_2 \cos (\theta_1 - \theta_2) d\vec{\omega}_i \\
&= \frac{1}{4\pi} \phi(x) + \frac{3}{4\pi} (\vec{E}(x) \bullet \vec{\omega}_o) \int_{\Omega_{4\pi}} p(\vec{\omega}_i \bullet \vec{\omega}_o) (\vec{\omega}_o \bullet \vec{\omega}_i) d\vec{\omega}_i + 0 \\
&= \frac{1}{4\pi} \phi(x) + \frac{3}{4\pi} (\vec{E}(x) \bullet \vec{\omega}_o) g \\
&= \frac{\phi(x) + 3g\vec{E}(x) \bullet \vec{\omega}_o}{4\pi}
\end{aligned} \tag{2.38}$$

Possiamo quindi procedere estendendo la sostituzione ad entrambe le parti della Radiative Transfer Equation della radianza diffusiva, opportunamente includendo i nuovi coefficienti ridotti definiti in precedenza. A partire dal risultato di tale sostituzione andiamo inoltre a scalare l'equazione con  $\vec{\omega}$  e ad integrare su tutta la sfera, ovvero da tutte le direzioni possibili.

$$\frac{\nabla \phi(x)}{3} = -\sigma'_t \vec{E}(x) + \int_{\Omega_{4\pi}} \varepsilon(x, \vec{\omega}_i) \vec{\omega}_i d\vec{\omega}_i \tag{2.39}$$

Continuiamo isolando il termine  $\vec{E}(x)$  all'interno di tale equazione, osservando come il termine di emissione che sta venendo integrato corrisponda al precedentemente definito valore  $\vec{Q}_1(x)$ . Sia, inoltre, la costante diffusiva  $D$  associata al termine  $\frac{1}{3\sigma'_t}$ .

$$\begin{aligned}
\vec{E}(x) &= -\frac{\nabla \phi(x)}{3\sigma'_t} + \frac{1}{\sigma'_t} \vec{Q}_1(x) \\
&= -D \nabla \phi(x) + 3D Q_1(x)
\end{aligned} \tag{2.40}$$

Recuperando l'equazione che descrive la divergenza dell'irradianza vettoriale (Eq. 2.30), vi sostituiamo il valore  $\vec{E}(x)$  appena trovato, ottenendo quella che viene chiamata diffusion equation, ovvero un'equazione differenziale parziale che descrive le fluttuazioni di densità.

$$D \nabla^2 \phi(x) = \sigma_a \phi(x) - Q_0(x) + 3D \nabla \bullet \vec{Q}_1(x) \tag{2.41}$$

Per determinare la fluenza  $\phi(x)$ , questa deve essere una soluzione della diffusion equation. Consi-

derando la situazione di cui abbiamo discusso nella sezione precedente, ovvero un medium omogeneo finito, non esiste una soluzione esplicita per ottenere la fluenza. Esiste, però, una soluzione per il caso triviale in cui abbiamo un medium omogeneo infinito con una sorgente puntiforme isotropa. Di seguito l'equazione che descrive l'emissione di radianza ridotta da parte della sorgente, con  $P_0$  che indica la potenza e  $\delta$  la funzione delta di Dirac, che garantisce l'emissione avvenga solo in  $x_0$ .

$$\varepsilon_{x_0}(x) = \frac{P_0}{4\pi} \cdot \delta(\|x - x_0\|) m^{-3} sr^{-1} \quad (2.42)$$

Tenendo in considerazione tale equazione, andiamo a trovare la soluzione alla diffusion equation, per prima cosa dividendo da entrambe le parti per  $D = \frac{1}{3\sigma'_t}$ . Teniamo a mente anche la definizione di  $\vec{Q}_1(x)$  e il fatto che  $\varepsilon_{x_0}(x)$  sia indipendente dalla direzione  $\vec{\omega}$ .

$$\begin{aligned} \nabla^2 \phi(x) &= \frac{\sigma_a \phi(x)}{D} - \frac{Q_0(x)}{D} + 3\nabla \bullet \vec{Q}_1(x) \\ \nabla^2 \phi(x) &= 3\sigma'_t \sigma_a \phi(x) - 3\sigma'_t Q_0(x) + 3\nabla \bullet \int_{\Omega_{4\pi}} \varepsilon_{x_0}(x) \vec{\omega} d\vec{\omega} \\ \nabla^2 \phi(x) - 3\sigma'_t \sigma_a \phi(x) &= -3\sigma'_t Q_0(x) + 3\nabla \bullet \varepsilon_{x_0}(x) \int_{\Omega_{4\pi}} \vec{\omega} d\vec{\omega} \end{aligned} \quad (2.43)$$

Introduciamo un nuovo coefficiente  $\sigma_{tr} = \sqrt{3\sigma_a \sigma'_t}$  chiamato *coefficiente di effective transport*. Osserviamo, inoltre, come l'integrale di un vettore su entrambi gli emisferi sia nullo.

$$\begin{aligned} \nabla^2 \phi(x) - \sigma_{tr}^2 \phi(x) &= -3\sigma'_t Q_0(x) + 3\nabla \bullet \varepsilon_{x_0}(x) \cdot 0 \\ \nabla^2 \phi(x) - \sigma_{tr}^2 \phi(x) &= -3\sigma'_t \int_{\Omega_{4\pi}} \varepsilon_{x_0}(x) d\vec{\omega} \\ \nabla^2 \phi(x) - \sigma_{tr}^2 \phi(x) &= -3\sigma'_t \varepsilon_{x_0}(x) \int_{\Omega_{4\pi}} 1 d\vec{\omega} \\ \nabla^2 \phi(x) - \sigma_{tr}^2 \phi(x) &= -\frac{1}{D} \varepsilon_{x_0}(x) \cdot 4\pi \end{aligned} \quad (2.44)$$

Adesso è sufficiente sostituire  $\varepsilon_{x_0}(x)$  all'interno della precedente equazione.

$$\begin{aligned} \nabla^2 \phi(x) - \sigma_{tr}^2 \phi(x) &= -\frac{1}{D} \frac{P_0}{4\pi} \cdot \delta(\|x - x_0\|) \cdot 4\pi \\ \nabla^2 \phi(x) - \sigma_{tr}^2 \phi(x) &= -\frac{P_0}{D} \cdot \delta(\|x - x_0\|) \end{aligned} \quad (2.45)$$

Quella che abbiamo trovato è una equazione differenziale di secondo ordine, che una volta risolta fornisce la seguente soluzione alla diffusion equation con un medium omogeneo infinito e una sorgente puntiforme isotropa.

$$\phi(r) = \frac{P_0}{D} \frac{e^{-\sigma_{tr} r}}{4\pi r} \quad (2.46)$$

Trovare la fluenza  $\phi(x)$  non sempre è così semplice. Prendiamo in considerazione una situazione in cui abbiamo un medium omogeneo finito, all'interno di uno spazio vuoto. Un primo vincolo per poter trovare la fluenza è che questa sia una soluzione della diffusion equation. Un altro vincolo è quello dettato da una boundary condition, introdotta a causa della formulazione che abbiamo dato al problema. La radianza diffusiva può uscire dal medium, ma non vi tornerà più, in quanto il medium

esterno è il vuoto, che non presenta quindi alcun fenomeno di scattering. Abbiamo quindi, per un punto sul boundary, due emisferi, uno negativo  $\Omega_{2\pi-}$  ed uno positivo  $\Omega_{2\pi+}$ . Per quest ultimo, la radianza diffusiva nel punto  $x$  equivale a 0, in quanto non ci sono contributi dall'esterno. Proprio per soddisfare tale proprietà, viene introdotta la prima citata boundary condition, che però non può essere soddisfatta nella sua forma esatta, in quanto abbiamo ottenuto la diffusion equation tramite l'approssimazione della radianza diffusiva  $L_d$ . È necessario quindi utilizzare una boundary condition approssimata in cui è definito che il totale flusso radiante ricevuto dall'esterno per un punto  $x$  sia 0. Tale condizione viene definita dalla seguente equazione.

$$\int_{\Omega_{4\pi}} L_d(x, \vec{\omega}_i) (\vec{n} \bullet \vec{\omega}_i) d\vec{\omega}_i = 0 \quad (2.47)$$

Se, invece, consideriamo una situazione in cui il nostro medium sia contenuto in un altro medium avente un differente indice di rifrazione, sull'interfaccia fra i due medium si presenteranno anche fenomeni di riflessione, invece che solo di trasmissione. Ovvero, una parte della radianza diffusiva uscente verrà riflessa all'interno del medium, con il risultato che il totale flusso radiante ricevuto dall'emisfero positivo  $\Omega_{2\pi+}$  non è più nullo, ma è descritto dalla seguente equazione, che tiene in considerazione la frazione di energia riflessa. Quest'ultima è determinata dal termine  $F_{dr}$ , ovvero la riflettanza diffusiva, computato sulla base della funzione di Fresnel parametrizzata con il valore  $\mu$ , ovvero l'indice di rifrazione fra i due medium che costituiscono l'interfaccia.

$$\int_{\Omega_{2\pi+}} L_d(x, \vec{\omega}_i) (\vec{n} \bullet \vec{\omega}_i) d\vec{\omega}_i = F_{dr} \int_{\Omega_{2\pi-}} L_d(x, \vec{\omega}_i) (-\vec{n} \bullet \vec{\omega}_i) d\vec{\omega}_i \quad (2.48)$$

$$F_{dr} = \frac{1}{\pi} \int_{\Omega_{2\pi}} F_r(\mu, \vec{n} \bullet \vec{\omega}_i) (\vec{n} \bullet \vec{\omega}_i) d\vec{\omega}_i \quad (2.49)$$

Se assumiamo che la reduced radiance  $\varepsilon(x, \vec{\omega})$  in ogni punto  $x$  sia isotropa, è possibile esprimere la boundary condition solo in termini di fluenza  $\phi$ , in quanto  $\vec{Q}_1(x) = 0$ . Questo è dato dalla definizione di  $\vec{Q}_1(x)$  perché, se la radianza ridotta è isotropa, vorrà dire che per ogni contributo positivo da una certa direzione, vi sarà un equivalente contributo negativo dalla direzione opposta. Tale approssimazione ci permette quindi di ottenere la seguente equazione.

$$\vec{E}(x) \simeq -D \nabla \phi(x) \quad (2.50)$$

Possiamo procedere sostituendo la precedente equazione nella parte sinistra dell'equazione relativa al boundary condition di un medium finito all'interno di un altro medium (Eq. 2.48).

$$\begin{aligned} \int_{\Omega_{2\pi+}} \left( \frac{1}{4\pi} \phi(x) + \frac{3}{4\pi} \vec{E}(x) \bullet \vec{\omega} \right) (\vec{n} \bullet \vec{\omega}) d\vec{\omega} &= \frac{\phi(x)}{4} + \frac{3}{4\pi} \int_{\Omega_{2\pi+}} (\vec{n} \bullet \vec{\omega}) (\vec{E}(x) \bullet \vec{\omega}) d\vec{\omega} \\ &= \frac{\phi(x)}{4} + \frac{3}{4\pi} \vec{n} \bullet \vec{E}(x) \int_{\Omega_{2\pi+}} (\vec{n} \bullet \vec{\omega})^2 d\vec{\omega} \\ &= \frac{\phi(x)}{4} + \frac{3}{4\pi} \vec{n} \bullet \vec{E}(x) \frac{2\pi}{3} \\ &= \frac{\phi(x)}{4} + \frac{2}{4} \vec{n} \bullet \vec{E}(x) \end{aligned} \quad (2.51)$$

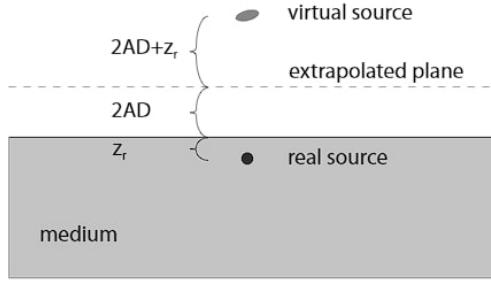


Figura 2.11: Configurazione dipolo, con sorgente reale e sorgente virtuale. Sono posizionate alla stessa distanza rispetto al piano estrapolato [24]

Osserviamo come l'integrale da entrambi i lati dell'equazione 2.48 sia sostanzialmente uguale, se non per la normale invertita. Possiamo quindi riutilizzare il risultato dell'equazione precedente per semplificare direttamente una sezione della parte destra, ottenendo quindi la seguente equazione facendo le opportune sostituzioni.

$$\begin{aligned}
 \frac{\phi(x)}{4} + \frac{2}{4} \vec{n} \bullet \vec{E}(x) &= F_{dr} \cdot \left( \frac{\phi(x)}{4} + \frac{2}{4} (-\vec{n} \bullet \vec{E}(x)) \right) \\
 \phi(x) + 2\vec{n} \bullet \vec{E}(x) &= F_{dr} \cdot (\phi(x) + 2(-\vec{n} \bullet \vec{E}(x))) \\
 \phi(x) + 2(\vec{n} \bullet (-D\nabla\phi))(x) &= F_{dr} \cdot (\phi(x) + 2(-\vec{n} \bullet \nabla(-D\nabla\phi))(x)) \\
 \phi(x) - 2D(\vec{n} \bullet \nabla\phi)(x) &= F_{dr} \cdot (\phi(x) - 2D(-\vec{n} \bullet \nabla\phi)(x))
 \end{aligned} \tag{2.52}$$

Riordinando, infine, gli opportuni termini, otteniamo la versione finale del boundary condition approssimato per un medium finito all'interno di un altro medium, dove  $A = \frac{1+F_{dr}}{1-F_{dr}}$ .

$$\phi(x) - 2DA(\vec{n} \bullet \nabla\phi)(x) = 0 \tag{2.53}$$

### Approssimazione con dipolo

Come abbiamo visto in precedenza, un medium omogeneo finito non ha una soluzione analitica per ottenere  $\phi(x)$ . Fortunatamente, però, il fenomeno di subsurface reflectance può essere modellato tramite un piano che separi in  $\mathbb{R}^3$  due medium i quali quindi risultano semi-infiniti. Tali media sono omogenei, ma con proprietà diverse, come per esempio pelle e aria. Ulteriormente, data l'assunzione che abbiamo fatto di avere un medium con un alto valore di single-scattering albedo (alto scattering), possiamo considerare la radianza incidente indipendentemente dalla direzione di arrivo, in quanto la radianza trasmessa diventa rapidamente isotropa. Possiamo quindi assumere di usare un'approssimazione in cui la totalità della radianza incidente nel punto  $x_i$  sul medium provenga da un beam di luce perpendicolare al piano. Con tale approssimazione dobbiamo quindi stabilire la fluenza  $\phi(x)$  dato un immaginario beam di luce sopra ad  $x_i$ . Nonostante anche questa configurazione non fornisca una semplice soluzione analitica, è possibile risolverla utilizzando un modello basato su un dipolo, che approssima la distribuzione all'interno del medium conseguente all'utilizzo di un beam di luce cilindrico. In tale modello due luci puntiformi vengono posizionate rispettivamente dentro al medium di interesse e sopra ad esso (Fig. 2.11). Nel caso della luce dentro al medium, chiamata *sorgente reale*, questa emette radianza positiva in tutte le direzioni

e viene posizionata ad una distanza di un mean free path sotto al punto di incidenza  $x_i$ . Più precisamente tale distanza la indicheremo con  $z_r = \frac{1}{\sigma_t}$ . Scegliamo questa distanza in quanto è la distanza media, in un medium con tali coefficienti, che un fotone percorre prima di subire un evento di scattering. Inoltre, una parte della radianza trasmessa verso la sorgente reale viene assorbita, portando alla necessità di scalare la potenza della sorgente reale con il termine di reduced albedo  $\alpha' = \frac{\sigma_s'}{\sigma_t}$ . In sostanza la sorgente reale simula il comportamento della luce emessa da un beam cilindrico dopo il primo evento di scattering all'interno del medium. Per quanto riguarda, invece, la luce fuori dal medium di interesse, chiamata sorgente virtuale, questa emette radianza negativa in tutte le direzioni ed è posizionata ad una distanza  $z_v = z_r + 4AD$ . Questa distanza, che risulta maggiore di quella della sorgente reale, è data dal fatto che il modello dipolo non soddisfa la condizione di boundary (Eq. 2.53), soddisfando però una condizione approssimata chiamata extrapolated boundary. Tale boundary è posizionata ad una distanza  $2AD$  sopra al boundary fisico, con la conseguenza che entrambe le sorgenti si trovano ad una distanza  $2AD + z_r$  rispetto al extrapolated boundary. Per ogni punto  $x$  possiamo definire le funzioni che descrivono la distanza rispetto alle sorgenti reali e verticali nel seguente modo.

$$\begin{aligned} d_r(x) &= \|(x_i - z_r \cdot \vec{n}) - x\| \\ d_v(x) &= \|(x_i + z_v \cdot \vec{n}) - x\| \end{aligned} \quad (2.54)$$

A questo punto possiamo determinare l'approssimazione della fluenza di un beam cilindrico incidente perpendicolarmente utilizzando l'equazione 2.46 precedentemente definita e considerando che la fluenza, dato un dipolo, è data dalla differenza fra la fluenza della sorgente reale e la fluenza della sorgente virtuale.

$$\phi(x) = \phi(x)_r - \phi(x)_v = \frac{\alpha' P_0}{4\pi D} \left( \frac{e^{-\sigma_{tr}d_r}}{d_r} - \frac{e^{-\sigma_{tr}d_v}}{d_v} \right) \quad (2.55)$$

Abbiamo tutti gli strumenti necessari per determinare il profilo diffusivo  $R(x_i, x_o)$ , ovvero quanta della radianza uscente da  $x_o$  sia conseguenza dell'irradianza incidente in  $x_i$ . Tale profilo è ottenuto con la seguente formula approssimata dove  $B(x_o)$  è la densità del flusso radiante uscente.

$$\begin{aligned} R(x_o, x_i) &= \frac{dB(x_o)}{d\Phi(x_i)} \\ &= \frac{D(\vec{n} \bullet \nabla \phi)(x_o)}{d\Phi(x_i)} \end{aligned} \quad (2.56)$$

Per semplificare i passaggi, iniziamo applicando  $\vec{n} \bullet \nabla$  alla sorgente reale, tenendo in considerazione che  $\nabla d_r(x) = \frac{(x_i - z_r \cdot \vec{n}) - x}{d_r(x)}$  e  $\nabla \frac{1}{d_r(x)} = -\frac{(x_i - z_r \cdot \vec{n}) - x}{d_r^3(x)}$ . In questo caso, inoltre, visto che  $x$  ed  $x_i$  risiedono sullo stesso piano ed il vettore che li unisce è perpendicolare con la normale  $\vec{n}$ , vale anche l'identità  $\vec{n} \bullet (x_i - x) = 0$ . Applichiamo anche, sia la regola del prodotto del operatore di divergenza per campi

scalari, che dice  $\nabla(\alpha\beta) = \alpha\nabla\beta + \beta\nabla\alpha$ , sia la regola di associatività del operatore di divergenza.

$$\begin{aligned}
 (\vec{n} \bullet \nabla) \frac{e^{-\sigma_{tr}d_r(x)}}{d_r(x)} &= \vec{n} \bullet \left( \frac{1}{d_r(x)} \nabla e^{-\sigma_{tr}d_r(x)} + e^{-\sigma_{tr}d_r(x)} \nabla \frac{1}{d_r(x)} \right) \\
 &= \vec{n} \bullet \left( -\sigma_{tr} \frac{(x_i - z_r \cdot \vec{n}) - x}{d_r^2(x)} e^{-\sigma_{tr}d_r(x)} - e^{-\sigma_{tr}d_r(x)} \frac{(x_i - z_r \cdot \vec{n}) - x}{d_r^3(x)} \right) \\
 &= -e^{-\sigma_{tr}d_r(x)} \frac{d_r(x)\sigma_{tr} + 1}{d_r^3(x)} \vec{n} \bullet ((x_i - x) - z_r \cdot \vec{n}) \\
 &= -e^{-\sigma_{tr}d_r(x)} \frac{d_r(x)\sigma_{tr} + 1}{d_r^3(x)} (0 - z_r \cdot \vec{n} \bullet \vec{n}) \\
 &= e^{-\sigma_{tr}d_r(x)} \frac{d_r(x)\sigma_{tr} + 1}{d_r^3(x)} z_r \\
 &= (d_r(x)\sigma_{tr} + 1) \frac{z_r e^{-\sigma_{tr}d_r(x)}}{d_r^3(x)}
 \end{aligned} \tag{2.57}$$

Possiamo riformulare le funzioni che descrivono la distanza di un punto da una sorgente reale o virtuale, introducendo la variabile  $r = \|x - x_i\|$  ed utilizzando il teorema di Pitagora.

$$\begin{aligned}
 d_r(r) &= \sqrt{r^2 + z_r^2} \\
 d_v(r) &= \sqrt{r^2 + z_v^2}
 \end{aligned} \tag{2.58}$$

Infine possiamo finalmente ottenere il profilo diffusivo della riflettanza  $R(x, x_i)$ , facendo le opportune sostituzioni ed assumendo che la potenza iniettata in  $x_i$  dal beam cilindrico di luce sia  $d\Phi(x_i) = P_0$ .

$$\begin{aligned}
 R(x, x_i) &\simeq \frac{D(\vec{n} \bullet \nabla\phi)(x)}{d\Phi(x_i)} \\
 &= \frac{\alpha' P_0}{4\pi D} \frac{D}{P_0} \left( (d_r\sigma_{tr} + 1) \frac{z_r e^{-\sigma_{tr}dr}}{d_r^3} + (d_v\sigma_{tr} + 1) \frac{z_v e^{-\sigma_{tr}dv}}{d_v^3} \right) \\
 &= \frac{\alpha'}{4\pi} \left( (d_r\sigma_{tr} + 1) \frac{z_r e^{-\sigma_{tr}dr}}{d_r^3} + (d_v\sigma_{tr} + 1) \frac{z_v e^{-\sigma_{tr}dv}}{d_v^3} \right)
 \end{aligned} \tag{2.59}$$

### 2.3.2 Modello Multipole

Donner [11] osserva come il modello proposto da Jensen produca un effetto troppo ceroso, che può essere adatto per un certo tipo di materiali, ma non lo è, per esempio, per la pelle. Suggerisce quindi un approssimazione simile, ma basata su multipli layer. Infatti, per modellare un materiale come la pelle è necessaria considerare i parametri che assume il medium a diverse profondità. Questa configurazione può essere ottenuta mediante l'utilizzo di multiple slab (una slab è un medium di spessore finito, con un interfaccia planare nella parte superiore ed inferiore), una sopra l'altra, con un beam di luce incidente. In quanto una slab, a differenza di un medium semi-infinito, ha due boundary (uno superiore ed uno inferiore Fig. 2.12), non è più sufficiente l'utilizzo di un singolo dipolo. Si rende necessario l'utilizzo del *modello Multipole* (chiamato anche Multi-Dipole) per soddisfare la condizione di multipli boundary. Come vedremo successivamente, mediante questo modello è possibile produrre i profili di riflettanza diffusiva (o trasmissione) per ciascuna slab. Per combinare tutti questi profili verrà utilizzata una particolare serie di convoluzioni.

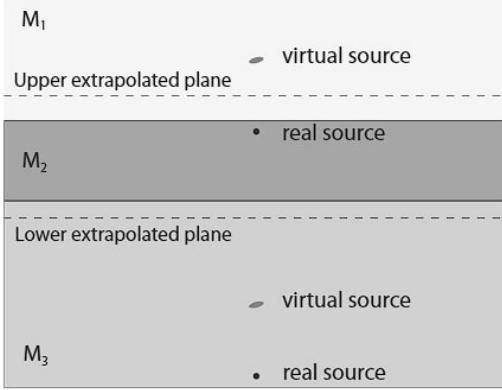


Figura 2.12: Configurazione multipole, ovvero con multipli dipoli.  $M_2$  è una slab con interfaccia sia superiore che inferiore. I dipoli, anche dopo essere stati specchiati, mantengono la stessa polarizzazione. [24]

### Riflettanza singola slab

Iniziamo determinando il profilo diffusivo di una singola slab, andando poi in un passo successivo a combinarlo con quelli delle altre slab. Partiamo dal presupposto che la nostra slab si trovi in un sandwich di due media semi-infiniti con diversi indici di rifrazione ( $\mu_1$  per il media superiore e  $\mu_3$  per quello inferiore, con la slab avente indice di rifrazione  $\mu_2$ ). Sia  $A_0$  a rappresentare  $A(\frac{\mu_2}{\mu_1})$  per la parte superiore e  $A_s$  a rappresentare  $A(\frac{\mu_2}{\mu_1})$  per la parte inferiore. Anche in questo caso, come per il modello dipole, per approssimare la redistribuzione del flusso iniettato tramite un beam di luce, utilizziamo una sorgente puntiforme posizionata ad una distanza di un mean free path ( $\frac{1}{\sigma_t}$ ) sotto al boundary. Quindi per soddisfare la condizione dell'extrapolated boundary aggiungiamo anche una sorgente puntiforme virtuale, posizionata ad una distanza di  $l + 4A_0D$  sopra la superficie. Osservandola, questa è la stessa configurazione basata sul modello dipole, con però una differenza, ovvero la presenza di anche un boundary inferiore, con il conseguente vincolo. Ovvero, il flusso radiante deve essere nullo ad una distanza  $2A_sD$  sotto la slab. Per fare ciò, è sufficiente specchiare la posizione del primo dipolo attorno al extrapolated boundary inferiore, mantenendo però la polarità del dipolo. L'introduzione di questo secondo dipolo ha risolto un problema, ma ne ha introdotto un altro, ovvero ha rotto il vincolo precedentemente stabilito per il boundary superiore. Andiamo quindi a ripetere il processo di rispecchiamento, utilizzando però il secondo dipolo e l'extrapolated boundary superiore e così via, andando a formare una somma infinita di dipoli (Fig. 2.12). Possiamo determinare le coordinate dei dipoli con le seguenti semplici formule ricorsive, dove  $j \in \mathbb{N}$  e  $j$  rappresenta l'indice del dipolo ( $j = 0$  è il primo dipolo). I pedici  $r$  e  $v$  rappresentano rispettivamente le sorgenti reali e le sorgenti virtuali, mentre  $k_u$  e  $k_l$  rappresentano rispettivamente le coordinate del boundary superiore (upper) e inferiore (lower).

$$\begin{aligned}
 z_{r,j} &= -z_{v,-j} + 2k_u \\
 z_{v,j} &= -z_{r,-j} + 2k_u \\
 z_{r,-j} &= -z_{v,j-1} + 2k_l \\
 z_{v,-j} &= -z_{r,j-1} + 2k_l
 \end{aligned} \tag{2.60}$$

Da queste equazioni, andando a sostituire la quarta nella prima e la seconda nella terza, otteniamo delle equazioni che sia a sinistra che a destra fanno riferimento allo stesso tipo di sorgente(reale o virtuale), ed assumono la forma  $x_{j+1} = x_j + k$ .

$$\begin{aligned} z_{r,j} &= z_{r,j-1} + 2(k_u - k_l) \\ z_{r,-j} &= z_{r,-(j-1)} - 2(k_u - k_l) \end{aligned} \quad (2.61)$$

Ripetendo lo stesso processo del passo precedente possiamo ottenere delle nuove forme ricorsive anche per le sorgenti virtuali, permettendoci di definire le seguenti formule dirette per ottenere le coordinate di sorgenti virtuali e reali. Le coordinate  $z_{r,0}$  e  $z_{v,0}$  corrispondono a quelle che abbiamo definito in precedenza per il primo dipolo.

$$\begin{aligned} z_{r,i} &= z_{r,0} + i2(k_u - k_l) \\ z_{v,i} &= z_{v,0} + i2(k_u - k_l) \end{aligned} \quad (2.62)$$

Sostituendo in maniera opportuna i valori, otteniamo la seguente formulazione finale, che indica lo spostamento rispetto al dipolo iniziale.

$$\begin{aligned} z_{r,i} &= -l + i * 2(2A_0D + s + 2A_sD) \\ z_{v,i} &= l + 4A_0D + i * 2(2A_0D + s + 2A_sD) \end{aligned} \quad (2.63)$$

Per un punto  $x$  qualsiasi sulla superficie, possiamo esprimere la sua distanza da una sorgente reale o da una sorgente virtuale con le seguenti equazioni.

$$\begin{aligned} d_{r,i}(x) &= \|(x_i + z_{r,i} \cdot \vec{n}) - x\| \\ d_{v,i}(x) &= \|(x_i + z_{v,i} \cdot \vec{n}) - x\| \end{aligned} \quad (2.64)$$

Abbiamo visto in precedenza nella configurazione basata su un singolo dipolo come ottenere la riflettanza  $R$ . Con una configurazione basata su multipoli, la riflettanza è data dalla somma dei contributi dei  $2n+1$  dipoli. L'equazione ottenuta approssima la porzione di luce del beam incidente, che attraversa il medium per poi raggiungere il punto  $x$  sulla superficie.

$$R(\|x - x_i\|) \simeq \sum_{i=-n}^n \frac{\sigma'}{4\pi} \left( (d_{v,i}\sigma_{tr} + 1) \frac{z_{v,i} e^{-\sigma_{tr} d_{v,i}}}{d_{v,i}^3} - (d_{r,i}\sigma_{tr} + 1) \frac{z_{r,i} e^{-\sigma_{tr} d_{r,i}}}{d_{r,i}^3} \right) \quad (2.65)$$

Tale equazione descrive la situazione in cui il lato da cui entra la luce è lo stesso lato da cui esce. Similmente possiamo definire un equazione  $T$  che descriva il fenomeno di trasmissione, ovvero quant'è la porzione di luce che raggiunge il fondo in un punto  $x'$ , distante  $s$  unità dalla superficie superiore. Possiamo quindi ridefinire le funzioni per ottenere la distanza di un punto da una sorgente e ripetere il processo di prima per ottenere la trasmittanza  $T$ .

$$\begin{aligned} d'_{r,i}(x') &= \|(x'_i + (x_{r,i} + s) \cdot \vec{n}) - x'\| \\ d'_{v,i}(x') &= \|(x'_i + (x_{v,i} + s) \cdot \vec{n}) - x'\| \end{aligned} \quad (2.66)$$

$$T(\|x' - x'_i\|) \simeq \sum_{i=-n}^n \frac{\sigma'}{4\pi} ((d'_{r,i}\sigma_{tr} + 1) \frac{z_{r,i}e^{-\sigma_{tr}d'_{r,i}}}{d'^3_{r,i}} - (d'_{v,i}\sigma_{tr} + 1) \frac{z_{v,i}e^{-\sigma_{tr}d'_{v,i}}}{d'^3_{v,i}}) \quad (2.67)$$

Le formule per le distanze possono, in realtà, essere ulteriormente semplificate semplicemente sfruttando, come abbiamo fatto nel modello precedente, il teorema di Pitagora. Sappiamo che  $\|x - x_i\| = \|x' - x'_i\|$ , in quanto le varianti con  $'$  rappresentano la stessa posizione sulla superficie, ma proiettata ortogonalmente sulla superficie inferiore di una slab. Definendo quindi la distanza fra  $x$  e  $x'$  con  $r$ , è possibile riformulare la funzione della distanza nel seguente modo.

$$\begin{aligned} d_{r,i}(r) &= \sqrt{r^2 + z_{r,i}^2} \\ d_{v,i}(r) &= \sqrt{r^2 + z_{v,i}^2} \end{aligned} \quad (2.68)$$

$$\begin{aligned} d'_{r,i}(r) &= \sqrt{r^2 + (z_{r,i} + s)^2} \\ d'_{v,i}(r) &= \sqrt{r^2 + (z_{v,i} + s)^2} \end{aligned} \quad (2.69)$$

### Combinazione multipli layer

Nella sezione precedente abbiamo visto come stabilire le funzioni di riflettanza  $R(r)$  e trasmittanza  $T(r)$  per una slab, quando questa si trova fra due medium semi-infiniti. Fortunatamente è possibile seguire le stesso processo per stabilire tali funzioni anche quando le slab fanno parte di uno stack.

Sia  $\tilde{T}(x) = T(\|x\|)$  per una singola slab, la seguente equazione stabilisce il flusso radiante uscente trasmesso dalla parte inferiore di una slab, in particolare nel punto  $x'$ . Il punto di incidenza è conosciuto ed è  $x_i$ .

$$\begin{aligned} B(x') &= \int_{\mathbb{R}^2} \tilde{T}(x - x_i) I(x_i) dx_i \\ &= (\tilde{T} * I)(x) \end{aligned} \quad (2.70)$$

Nell'equazione precedente il passaggio in cui passiamo all'utilizzo dell'operazione di convoluzione è dato dalla definizione stessa di convoluzione. Ovvero, date due funzioni  $f$  e  $g$ , la loro convoluzione è data dall'integrale del prodotto delle due funzioni dopo che una è stata riflessa rispetto all'asse  $y$  e traslata. In particolare, la seguente è la sua formulazione matematica e risulta evidente il passaggio precedente.

$$(f * g)(t) := \int_{-\infty}^{\infty} f(\tau)g(t - \tau)d\tau \quad (2.71)$$

Per descrivere il flusso radiante trasmesso uscente dalla parte inferiore di uno stack di due slab, aventi i profili trasmittivi  $T_1$  e  $T_2$ , possiamo utilizzare  $B(x'') = (\tilde{T}_2 * (\tilde{T}_1 * I))(x)$  dove  $x''$  è il punto di uscita dalla seconda slab. Questa formulazione è applicabile, in quanto, assumiamo che ciascuna slab sia spessa almeno un paio di mean free path, il che implica che le interazioni fra slab adiacenti siano tutte dettate da multiple scattering. Inoltre, data l'associatività della convoluzione, possiamo convolare in anticipo i profili trasmittivi indipendentemente dal fatto se conosciamo il flusso radiante incidente.

$$\tilde{T}_2 * (\tilde{T}_1 * I) = (\tilde{T}_2 * \tilde{T}_1) * I \quad (2.72)$$

Questa formulazione della trasmittanza, ovvero  $\tilde{T}_{12} = \tilde{T}_2 * \tilde{T}_1$ , è però incompleta. Questo perché considera la radianza diffusiva trasmessa solo tramite un passaggio diretto di trasmissione per ciascuna slab. Non considera, quindi, la situazione in cui, per esempio, la radianza trasmessa nel secondo slab, dopo una serie di scattering ritorni sulla superficie della seconda slab. Questo comportamento viene descritto da un profilo di riflettanza, in questo caso relativo al secondo slab, ovvero  $R_2$ . Da lì può subire ulteriori scattering nella prima slab, per poter poi tornare nella parte inferiore di quest'ultima, ed in questo caso si utilizza il profilo di riflettanza relativo alla parte inferiore della prima slab. Questo ping-pong della radianza diffusiva fra le due slab può ripetersi un numero arbitrario di volte. È evidente, inoltre, la necessità di considerare separatamente i profili di riflettanza per la parte superiore ed inferiore di una slab, per i quali utilizzeremo rispettivamente la seguente nomenclatura, ovvero  $R^+(r)$  e  $R^-(r)$ , ed equivalentemente per la trasmittanza. A partire da queste informazioni, possiamo quindi definire la trasmittanza combinata nel seguente modo, dove utilizziamo il termine  $\delta$  avente per le convoluzioni la proprietà  $f * \delta = f$ .

$$\tilde{T}_{12}^+ = \tilde{T}_2^+ * (\delta + \tilde{R}_1^- * \tilde{R}_2^+ + \tilde{R}_1^- * \tilde{R}_2^+ * \tilde{R}_1^- * \tilde{R}_2^+ + \dots) * \tilde{T}_1^+ \quad (2.73)$$

$$\tilde{T}_{12}^- = \tilde{T}_1^- * (\delta + \tilde{R}_2^+ * \tilde{R}_1^- + \tilde{R}_2^+ * \tilde{R}_1^- * \tilde{R}_2^+ * \tilde{R}_1^- + \dots) * \tilde{T}_2^- \quad (2.74)$$

Similmente a come abbiamo definito i profili di trasmittanza è possibile definire i profili di riflettanza.

$$\tilde{R}_{12}^+ = \tilde{T}_1^- * (\delta + \tilde{R}_2^+ * \tilde{R}_1^- + \tilde{R}_2^+ * \tilde{R}_1^- * \tilde{R}_2^+ * \tilde{R}_1^- + \dots) * \tilde{R}_2^+ * \tilde{T}_1^+ + \tilde{R}_1^+ \quad (2.75)$$

$$\tilde{R}_{12}^- = \tilde{T}_2^+ * (\delta + \tilde{R}_1^- * \tilde{R}_2^+ + \tilde{R}_1^- * \tilde{R}_2^+ * \tilde{R}_1^- * \tilde{R}_2^+ + \dots) * \tilde{R}_1^- * \tilde{T}_2^- + \tilde{R}_2^- \quad (2.76)$$

Nel caso usassimo più di due slab, questo processo di combinazione di profili può essere ripetuto iterativamente. Sfortunatamente questo processo non produce, come nel caso di un modello dipole con un medium semi-infinito, una semplice funzione analitica. Esiste però un modo per risolvere tale problema, osservando come i profili siano sequenza di convoluzioni. Questo ci permette di ripostulare il problema allo spazio delle frequenze utilizzando il *teorema di convoluzione*, il quale dice che la trasformata di Fourier della convoluzione di due funzioni  $f$  e  $g$ , sia equivalente al prodotto fra la trasformata di  $f$  e la trasformata di  $g$ .

$$\mathcal{F}\{f * g\} = k \cdot \mathcal{F}\{f\} \cdot \mathcal{F}\{g\} \quad (2.77)$$

A partire dalla definizione di trasformata di Fourier possiamo quindi ridefinire sia il profilo di riflettanza che di trasmittanza nel seguente modo. Da notare che generalmente  $d = \infty$ , ma in questo caso per poter fare valutazione numerica fissiamo un certo limite a tale valore. Ci permettiamo di fare questo in quanto la maggior parte dei contributi significativi si troverà entro un certo limite.

$$\tilde{\mathcal{R}}(u, v) = \mathcal{F}\{\tilde{R}\} = \int_{x^2+y^2 < d^2} \tilde{R}(x, y) e^{-2\pi i(ux+vy)} dx dy \quad (2.78)$$

$$\tilde{\mathcal{T}}(u, v) = \mathcal{F}\{\tilde{T}\} = \int_{x^2+y^2 < d^2} \tilde{T}(x, y) e^{-2\pi i(ux+vy)} dx dy \quad (2.79)$$

Ne segue che la serie di convoluzioni con cui abbiamo definito la combinazione di multipli profili, nello spazio delle frequenze, è fornito di seguito. Similmente possono essere definiti anche i profili per la

trasmittanza, per ambo i lati (positivi e negativi).

$$\mathcal{R}_{12}^+ = \mathcal{T}_1^- \cdot (1 + \mathcal{R}_2^+ \mathcal{R}_1^- + (\mathcal{R}_2^+ \mathcal{R}_1^-)^2 + (\mathcal{R}_2^+ \mathcal{R}_1^-)^3 + \dots) \cdot \mathcal{R}_2^+ \cdot \mathcal{T}_1^+ + \mathcal{R}_1^+ \quad (2.80)$$

Osservando questa equazione possiamo notare come quella fra parentesi sia una *serie geometrica*. E possibile quindi riscrivere l'equazione nel seguente modo, utilizzando la forma chiusa della serie geometrica, assumendo che  $|\mathcal{R}_2^+ \mathcal{R}_1^-| < 1$ . Equivalentemente a come abbiamo fatto per la riflettanza, possiamo fare lo stesso per la trasmittanza.

$$\mathcal{R}_{12} = \mathcal{R}_1 + \frac{\mathcal{T}_1 \mathcal{R}_2 \mathcal{T}_1}{1 - \mathcal{R}_1 \mathcal{R}_2} \quad (2.81)$$

$$\mathcal{T}_{12} = \frac{\mathcal{T}_1 \mathcal{T}_2}{1 - \mathcal{R}_1 \mathcal{R}_2} \quad (2.82)$$

I profili reali di trasmittanza e riflettanza per materiali multi-layer vengono quindi ottenuti computando l'inversa della trasformata di Fourier, utilizzando per esempio un algoritmo come Fast Fourier Transform.

### 2.3.3 Modello Burley

Burley e Christensen presentano un modello BSSRDF particolarmente semplice basato su approssimazioni [7], rispetto alla maggior parte dei modelli precedenti che nascono come modelli physically-based derivati dalla teoria della Radiative Transfer Equation. Questo modello ottiene risultati migliori per quanto riguarda il match con simulazioni brute-force Monte Carlo, rispetto ai modelli physically-based. Christensen fornisce anche una serie di diverse parametrizzazioni del modello che permettono il confronto diretto con i modelli physically-based. Vedremo successivamente quali siano queste parametrizzazioni. Ci sono una serie di vantaggi nell'utilizzare un modello approssimato come quello presentato:

1. I modelli physically based sono parametrizzati da termini come coefficiente di scattering ( $\sigma_s$ ) e coefficiente di assorbimento ( $\sigma_a$ ). Il problema di questi parametri è il fatto che sono poco intuitivi, rendendone difficile il controllo. Per questo motivo vengono utilizzati parametri più intuitivi come surface albedo e mean free path length che poi vengono convertiti nei termini fisici mediante processi di inversione numerica chiamati *albedo inversion* [6]. Il modello di Burley permette di utilizzare direttamente i parametri intuitivi, senza dover fare la tale conversione.
2. I modelli physically-based in genere modellano il problema considerando separatamente il fenomeno di single scattering e multiple scattering. Il modello di Burley include direttamente il termine di single scattering.
3. Fornisce una semplice *funzione cumulativa* (cdf) da utilizzare per fare *importance sampling*. Tale cdf viene presentata come non analiticamente invertibile, necessitando quindi di utilizzare un altro metodo per invertirla. Ci sono varie opzioni, come:
  - (a) scegliere casualmente uno dei due esponenziali nella cdf, usare la sua inversa e poi pesare i risultati usando *multiple importance sampling* (MIS).
  - (b) pre-computare una tabella della cdf inversa fissando  $d = 1$ , andando poi a scalare tale tabella con il valore di  $d$  effettivamente utilizzato.

- (c) utilizzare algoritmi di *root finding*, come, per esempio, il metodo di Newton (approfondimento nella sezione Implementazione Tecniche SSS Golubev).

Importante far notare come successivamente fu smentita la non invertibilità della formula da parte di Golubev [13], che presentò quindi la soluzione analitica al problema.

4. L'implementazione è più semplice, e la valutazione è più rapida. Inoltre non necessita la pre-computazione di tabelle di lookup. Nella realtà dei fatti Christensen [7] le performance sono solo leggermente migliori, in quanto per i modelli physically based è possibile fare tutta una serie di ottimizzazioni, come la sopra citata pre-computazione di tabelle di lookup che riducono il gap. Nonostante non si abbia quindi un vantaggio netto a livello di performance, rimane significativo il fatto che sia molto più semplice da implementare.
5. Non è necessario alcun utilizzo di fattori di correzione ad hoc per fare in modo che il modello si avvicini di più al riferimento creato mediante simulazioni di MC.

L'ispirazione di Burley proviene dalla tecnica comunemente utilizzata di approssimare funzioni complesse con funzioni più semplici, similmente a come ha fatto Schlick per le equazioni di Fresnel. Questo è un approccio che anche altri hanno utilizzato per approssimare il profilo diffusivo, come per esempio d'Eon, che ha ridefinito la funzione come una somma di gaussiane di media zero [9]. Comunemente una somma con 6 gaussiane produce una buona approssimazione. Burley osserva come, guardando ai grafici dei profili diffusivi, questi assomiglino più a delle esponenziali, rispetto che a delle gaussiane. Propone quindi un modello basato sulla somma di due esponenziali, divisa poi per il raggio  $r$  (Eq. 2.83).

$$R(r) = \frac{e^{-r/d} + e^{-r/3d}}{8\pi dr} \quad (2.83)$$

Nel suo modello Burley utilizza un termine  $d$  per controllare altezza e lunghezza della curva. Questo termine non ha un significato fisico, ma può essere impostato in base a scelte artistiche oppure derivato a partire da termini fisici. Christensen fornisce una parametrizzazione per ottenere  $d$  tramite trasformazioni da termini fisici come mean free path (mfp) ( $\ell$ ) e diffuse mean free path (dmfp) ( $\ell_d$ ). Questa conversione avviene attraverso un termine di traduzione  $s$ , il quale dipende dall'albedo  $A$  della superficie, dove  $A = \int_0^\infty R(r) 2\pi r dr$ . Per ottenere i valori di  $s$ , Christensen propone una serie di soluzioni in base alla configurazione di illuminazione che si sta usando.

### Configurazione Searchlight

In questa configurazione consideriamo un beam di luce parallelo incidente perpendicolarmente su un medium omogeneo semi-infinito sotto una superficie perfettamente piana. Assumiamo che la superficie in questione esibisca inizialmente una perfetta trasmissione speculare, dopo la quale la luce subisce un evento di scattering ed eventualmente venga assorbita o esca fuori dal medium (Fig. 2.13). Inoltre assumiamo che il medium abbia una funzione di fase isotropa. I fotoni uscenti presentano, quindi, una distribuzione radialmente simmetrica, che descrive il profilo diffusivo  $R(r)$ , quando la luce è normalmente incidente o il medium presenta trasmissione diffusiva. In questo contesto vogliamo rappresentare il valore  $d$  in funzione di un parametro che rappresenti una quantità fisicamente significativa, ovvero il volume

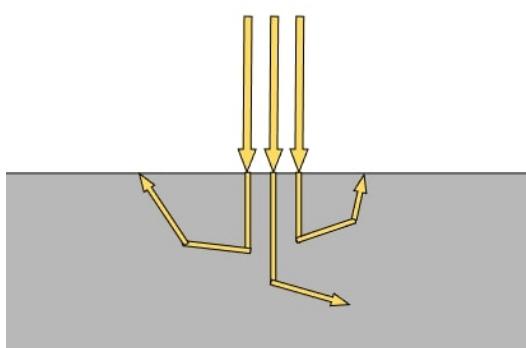


Figura 2.13: Configurazione searchlight in cui assumiamo la radianza arrivi sulla superficie perpendicolarmente lungo un beam. Presenta un fenomeno iniziale di perfetta trasmissione speculare

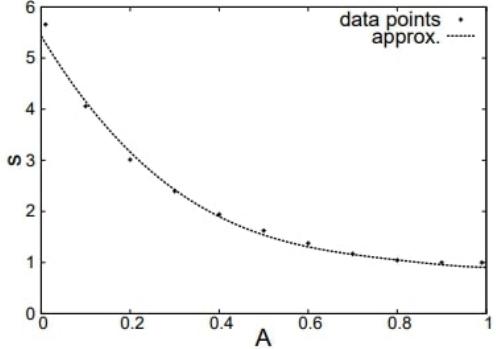


Figura 2.14: Plot del fit della curva che approssima  $s$ , rispetto a valori ottimali di  $s$  ottenuti mediante simulazioni di MC, per la configurazione searchlight con parametrizzazione mediante mean free path ( $\ell$ ) [24]

mean free path, che denominiamo con il termine  $\ell$ . Per descrivere questa relazione fra il valore  $d$  ed il valore  $\ell$ , introduciamo una variabile di “scaling”  $s$  nella loro relazione  $d = \ell/s$  ottenendo la seguente formula per il profilo diffusivo (Eq. 2.84).

$$R(r) = s \frac{e^{-rs/l} + e^{-rs/3l}}{8\pi lr} \quad (2.84)$$

Possiamo fissare  $\ell = 1$ , in quanto al variare di  $\ell$ , non cambia la forma della curva, ma solo lo stretch verticale ed orizzontale. Quindi tramite simulazione di Monte Carlo per i valori di  $[0.01, 0.02, \dots, 0.99]$  di albedo otteniamo i profili diffusivi da usare come riferimento. Utilizzando quindi una tecnica di brute-force random sampling dello spazio del parametro  $s$  andiamo a minimizzare l’errore relativo  $r_i$ , computato con l’equazione 2.85, ottenendo così i valori di  $s$  per il range di albedo che abbiamo utilizzato. Andando a computare l’errore di approssimazione rispetto ai profili diffusivi prodotti da Monte Carlo, otteniamo un errore relativo medio pari a 4.9%, utilizzando l’interpolazione per i punti intermedi.

$$r_i = \sum_i \frac{|R(r_i) - R_{MC}(r_i)|}{R_{MC}(r_i)} \quad (2.85)$$

Nella precedente equazione  $R_{MC}$  rappresenta il profilo diffusivo ottenuto mediante simulazioni di Monte Carlo, utilizzando, per esempio, un pacchetto software come MCML(Monte Carlo Multi-Layered).

È possibile però ottenere una funzione più semplice da valutare con, però, un errore leggermente più alto. Basta fare un fit manuale della curva a partire dai punti calcolati in precedenza(Fig. 2.14), ottenendo così una semplice espressione per determinare  $s$  a partire da  $A$  (Eq. 2.86). Utilizzando questa espressione per l’errore relativo medio rispetto ai riferimenti prodotti da Monte Carlo aumenta al 5.5%.

$$s = 1.85 - A + 7|A - 0.8|^3 \quad (2.86)$$

Rimanendo su questa stessa configurazione è possibile fornire una parametrizzazione alternativa della distanza di scattering. Invece di usare il mean free path nel volume, possiamo utilizzare il diffuse mean

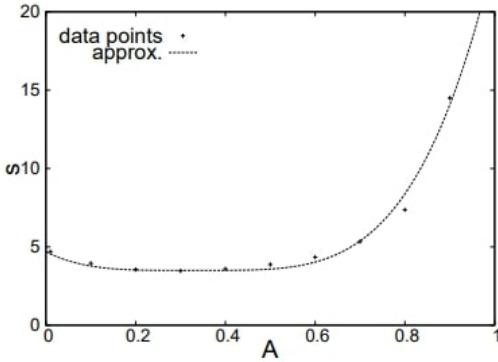


Figura 2.15: Plot del fit della curva che approssima  $s$ , rispetto a valori ottimali di  $s$  ottenuti mediante simulazioni di MC, per la configurazione searchlight con parametrizzazione mediante diffuse mean free path ( $\ell_d$ ) [24]

free path  $\ell_d$  sulla superficie. Per computare  $\ell_d$  a partire dal coefficiente di scattering  $\sigma_s$  e dal coefficiente di assorbimento  $\sigma_a$ , dobbiamo prima computare il coefficiente diffusivo (Eq. 2.87)

$$D = (\sigma_t + \sigma_a)/(3\sigma_t^2) \quad (2.87)$$

Dato  $D$  possiamo calcolare l'effective transport coefficient e quindi  $\ell_d = 1/\sigma_{tr}$ . Similmente a prima computiamo i nuovi riferimenti con simulazione di Monte Carlo e quindi troviamo i valori di  $s$  che minimizzino l'errore. L'errore, considerando i valori ottimali ottenuti dalla minimizzazione, risulta essere del 6.4%, mentre se andiamo a fare il fit di una curva (Fig. 2.15) ed utilizzare la relativa espressione ottenuta (Eq. 2.88) per determinare i valori di  $s$ , abbiamo un errore pari al 7.7%. Nonostante questa sia, come vedremo, la parametrizzazione con l'errore maggiore, è anche quella che utilizza il parametro più intuitivo. Per questo stesso motivo è anche quella più comunemente utilizzata.

$$s = 3.5 + 100(A - 0.33)^4 \quad (2.88)$$

## Configurazione Diffuse Surface Transmission

Nella configurazione precedente abbiamo considerato un fascio di luce che entra perpendicolarmente nella superficie, con perfetta trasmissione speculare. In contrasto questa configurazione assume una ideale trasmissione diffusiva sulla superficie (Fig. 2.16), considerando la luce proveniente da tutte le direzioni. Questa modellazione è adatta per rappresentare materiali ruvidi, come pelle asciutta, make-up, frutta e per situazioni in cui ignoriamo, o non sappiamo, la provenienza della luce.

In questo caso, consideriamo di nuovo una parametrizzazione con mean free path  $\ell$ , invece che diffuse mean free path. Andiamo quindi, come sempre fatto fino ad ora, a computare i riferimenti tramite simulazione di Monte Carlo, in questo caso con la nuova configurazione. Ripetiamo sempre lo stesso processo delle configurazioni precedenti per ottenere i valori di  $s$  ottimali. L'errore relativo medio di tali valori rispetto al riferimento è del 2.6%. Andando, quindi, a fare il fit manuale di una curva sui valori ottimali di  $s$  (Fig. 2.17), otteniamo l'espressione (Eq. 2.89). Considerando tale espressione

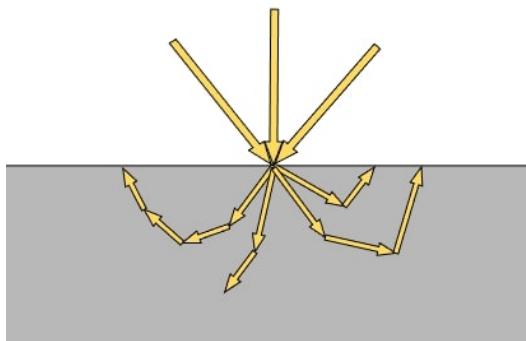


Figura 2.16: Configurazione diffuse surface transmission in cui consideriamo radianza proveniente da tutte le direzioni che assume immediatamente comportamento diffusivo entrando nel medium

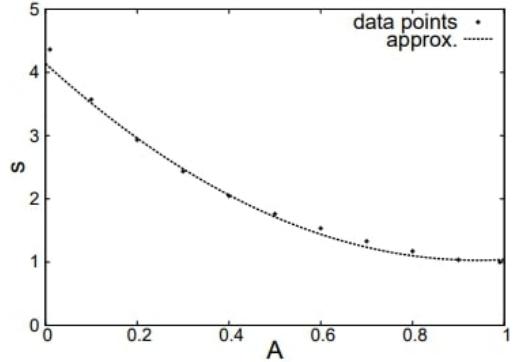


Figura 2.17: Plot del fit della curva che approssima  $s$ , rispetto a valori ottimali di  $s$  ottenuti mediante simulazioni di MC, per la configurazione diffuse surface transmission con parametrizzazione mediante mean free path ( $\ell$ ) [24]

l’errore relativo medio aumenta a 3.9% rispetto ai riferimenti prodotti con Monte Carlo.

$$s = 1.9 - A + 3.5(A - 0.8)^2 \quad (2.89)$$

## Considerazioni

Riassumendo, Christensen [7] ha presentato tre diverse parametrizzazioni utilizzabili per il modello di Burley:

1. Configurazione searchlight con mean free path
2. Configurazione searchlight con diffuse mean free path
3. Configurazione diffuse surface transmission con mean free path

Comunemente nelle varie implementazioni che fanno uso di questo modello viene utilizzata la configurazione con dmfp in quanto risulta essere quella più intuitiva. Nonostante le due configurazioni, ovvero searchlight e diffuse surface transmission, rappresentino due contesti di illuminazione diversi con tipi di materiali diversi, Christensen fa notare come, in realtà, i render prodotti risultino essere particolarmente simili [7].



# 3

## Implementazione Tecniche SSS

L'implementazione di tecniche di subsurface scattering (SSS) in tempo reale rappresenta una sfida non indifferente a livello computazionale. La complessità del subsurface scattering deriva dal fatto che coinvolge molteplici interazioni della luce all'interno del materiale, richiedendo una modellazione accurata dei percorsi che la luce può seguire. In un contesto di offline rendering tale processo viene comunemente implementato mediante una qualche forma di path tracing, dove il modello a cui è associato il materiale viene considerato come un volume. Nonostante negli ultimi anni si stiano diffondendo GPU aventi acceleratori hardware per il ray tracing, come la serie RTX di NVidia, il loro utilizzo per computare il SSS non è ancora praticabile. Questo perché, attualmente, in un contesto di rendering in tempo reale si possono "lanciare" solo una manciata di raggi per pixel all'interno di un frame. Avendo quindi questo budget limitato, in genere si preferisce utilizzare i raggi disponibili per effetti più significativi quali l'illuminazione globale (GI). Per tale motivo ancora oggi si continuano ad utilizzare i metodi classici sviluppati negli ultimi 20 anni.

In questo capitolo esploreremo le principali metodologie per l'implementazione del subsurface scattering in tempo reale. Analizzeremo, inoltre, i vantaggi e le limitazioni di ciascun approccio, considerando il compromesso tra fedeltà visiva e efficienza computazionale.

### 3.1 Approcci Real-Time

Nel contesto del subsurface scattering, le convoluzioni sono particolarmente utili per modellare la diffusione della luce all'interno di un materiale. Una convoluzione è un'operazione matematica che combina due funzioni per produrre una terza funzione. In termini di immagini, combina un'immagine di input con un nucleo per produrre un'immagine di output. Tale nucleo è una matrice che definisce il tipo di trasformazione che verrà applicata. Nel caso del SSS, rappresenta la diffusione della luce all'interno del materiale. Questo perché, come abbiamo visto in precedenza, la luce, entrando in un materiale traslucido, non si propaga in linea retta, ma viene diffusa in tutte le direzioni e questa diffusione può essere modellata come una convoluzione dove il nucleo rappresenta il profilo di diffusione. Quest'ultimo, in generale, assume una forma simile a quella di una gaussiana, tenendo però in considerazione che sostanzialmente nessun profilo diffusivo utile può essere approssimato con una sola gaussiana. La

convoluzione del profilo con l'immagine di input simula il processo di scattering interno, permettendo quindi di calcolare come la luce si diffonda sotto la superficie prima di riemergere.

Un vantaggio importante dell'utilizzare le convoluzioni per implementare il SSS è che le convoluzioni sono relativamente semplici da implementare e possono essere integrate facilmente nelle pipeline di rendering esistenti come un semplice passo di post-processing. Inoltre, utilizzare convoluzioni è computazionalmente molto più efficiente rispetto a simulazioni dirette del percorso della luce.

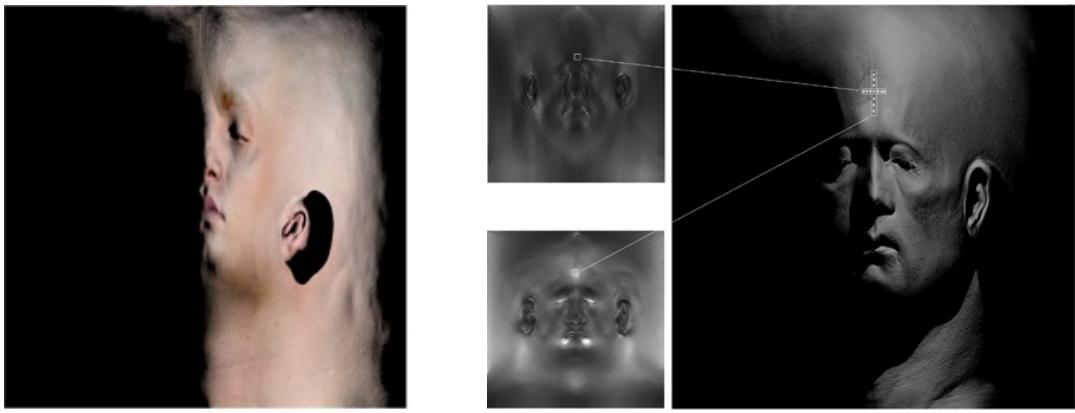
È presente un ulteriore separazione di approcci, usando le convoluzioni, in base alla fase in cui decidiamo di applicare il nucleo. Quest'ultimo può essere infatti applicato ad ogni irradiance map ottenuta per un modello soggetto a SSS, ovvero in **texture space**. Alternativamente il nucleo può essere applicato all'immagine ottenuta dal rendering della scena, ovvero in **screen space**. Di seguito analizzeremo i vantaggi e gli svantaggi di ciascuna di queste metodologie.

### 3.1.1 Texture Space

Le prime tecniche di diffusione in texture space vengono implementate durante la produzione del film "The Matrix Reloaded", grazie al lavoro di Borshukov e Lewis [4]. L'implementazione è basata sull'osservazione che lo scattering della luce è, generalmente, un fenomeno locale, e che quindi quest'ultimo possa essere approssimato applicando un'operazione di convoluzione, in particolare un blur, alla irradiance map utilizzando un determinato profilo diffusivo. Le convoluzioni vengono fatte sulle mappe ottenute facendo l'unwrapping delle mesh. Per effettuare le convoluzioni, Borshukov e Lewis propongono di utilizzare la somma di due profili analitici, uno che rappresenta lo scattering "stretto" del epidermide ed uno che rappresenta lo scattering "ampio" del derma. Il problema è che questi profili sono basati su parametri non fisici, con la conseguenza che gli artisti devono trovare ad occhio i valori corretti. Questa limitazione è uno dei motivi per cui successivamente verranno proposte tecniche che approssimano profili diffusivi, ottenuti da parametri fisici, mediante una somma di gaussiane [9][20].

Nonostante questa tecnica sia molto più vantaggiosa a livello di prestazioni rispetto ad una simulazione completa del trasporto della luce, in particolare nel contesto in cui è stata proposta, ovvero in quello del rendering offline, presenta una serie di problemi che vengono esposti di seguito [9][20][26].

1. È necessario produrre l'irradiance map per ciascuno dei modelli 3D renderizzati nella scena (Fig. 3.1a). Questo problema non è presente in screen space, in quanto tutti gli elementi soggetti a subsurface scattering vengono processati contemporaneamente in un unico post-processing pass.
2. In base all'orientamento della superficie, i texel in texture space potrebbero essere mappati sullo stesso pixel in screen space, andando così a sprecare convoluzioni.
3. Lavorando in texture space non possiamo sfruttare il depth culling, così come l'early-z rejection. Infatti è necessario fare le convoluzioni sull'intera texture, indipendentemente dal fatto che quella parte della superficie risulti occlusa o meno.
4. Indipendentemente dalla dimensione dell'oggetto da renderizzare nella scena è necessario fare lo shading di tutti i texel presenti nel relativo offscreen irradiance buffer. È possibile ottimizzare questo aspetto, ma risulta necessario gestire manualmente le mipmap e la loro selezione in base



(a) Irradiance texture che verrà convoluta in texture space per produrre l'effetto di SSS

(b) Stretch map orizzontali e verticali usate per modificare i pesi del profilo diffusivo. È possibile vedere come le dimensioni del nucleo verticale ed orizzontale non siano più le stesse, data la distorsione.

Figura 3.1: Texture che vengono utilizzate quando si implementa il SSS in texture space [9]

alla distanza dell'oggetto nella scena. Questo non è un problema in screen space in quanto è intrinseco alla soluzione stessa.

5. Generalmente l'UV mapping contiene molto stretching e warping, con la conseguenza che anche il profilo diffusivo subisce tali alterazioni, perdendo la sua simmetria radiale una volta che la texture viene mappata in world space sul modello. Una soluzione a questo problema consiste nel computare la stretch map e utilizzare poi i risultanti distortion factors per modificare i pesi del profilo diffusivo (Fig. 3.1b). In questo modo è possibile preservare la proprietà di radialità di quest ultimo.
6. Il funzionamento della diffusione in texture space parte dal presupposto che punti vicini sulla superficie lo siano anche nel UV mapping. Questo non risulta sempre vero, come nel caso dei bordi, che causano artefatti nel caso non siano gestiti con considerazioni particolari. Non funziona anche nel caso di superfici sottili, come le orecchie, in quanto probabilmente i punti sui due lati della superficie non sono vicini nello spazio UV. Una possibile soluzione a quest'ultimo problema è l'utilizzo delle translucency shadow maps [21].

### 3.1.2 Screen Space

Nonostante la maggior parte dei problemi derivanti dall'approccio in texture space vengano risolti o non siano presenti adottando, invece, l'approccio in screen space, quest'ultimo non è esente da limitazioni, che presentiamo di seguito:

1. Similmente all'approccio texture space non viene presa in considerazione back illumination, in quanto in screen space non abbiamo tale informazione, necessitando quindi di implementare tale componente separatamente. Jimenez ha proposto una soluzione utilizzando shadow map per determinare lo spessore e quindi poter determinare la trasmissione [21].

2. Necessario l'utilizzo di ulteriori texture per memorizzare lo stencil e la componente speculare. Lo stencil è necessario nel caso si voglia limitare l'applicazione del SSS solo su una determinata sezione di pixel.
3. Lavorando in screen space si è limitati ad avere solo l'informazione delle superfici che vengono rasterizzate. Quindi ogni volta ci siano delle occlusioni, ci saranno informazioni mancanti. Questo è un problema generale degli approcci che lavorano in screen space, come per esempio anche SSR (Screen Space Reflections).
4. Questo approccio fa uso di multipli passaggi in screen space, il che può provocare una serie di inconvenienti nel caso in cui il resto della pipeline si basi su un forward renderer.

## 3.2 Jimenez (2009)

Come abbiamo visto nella sezione relativa ai modelli per il subsurface scattering, è possibile ottenere i profili diffusivi per un materiale andando a specificarne alcuni parametri fisici. Nel caso usassimo il modello dipole di Jensen i parametri sarebbero riferiti ad un solo layer, mentre nel caso di multipole, sarebbe necessario specificare i parametri per ciascuno di essi. In ogni caso, ciò che otteniamo alla fine è il profilo diffusivo per quel materiale. Come abbiamo visto nella sezione precedente, questi profili diffusivi vengono usati per applicare le convoluzioni ad una certa immagine di input producendo così l'effetto di subsurface scattering. In teoria questo processo può essere fatto direttamente con il profilo diffusivo bidimensionale ottenuto. Il problema è dato dall'elevato costo computazione nell'utilizzo di questo approccio. Prendendo, per esempio, un profilo diffusivo avente dimensioni 32x32 una volta discretizzato, sarebbe necessario fare  $32 \times 32 = 1024$  accessi ad una texture per pixel. Considerando che il campionamento di una texture risulta una delle operazioni con costo maggiore, in particolare nel caso in cui i dati della texture non siano nella cache necessitando quindi accesso alla memoria della GPU, finisce spesso per essere una delle operazioni che influisce di più sulle performance di un shader. In particolare nel momento in cui lavoriamo in un contesto in tempo reale, convoluzioni con nucleo 2D di queste dimensioni risultano essere impraticabili. Sarebbe utile poter separare il nucleo in due passaggi, uno verticale ed uno orizzontale, facendo così solo  $32 + 32 = 64$  accessi per pixel, un numero molto minore di 1024. Il problema è che il profilo diffusivo è radialmente simmetrico ed in generale un profilo di questo tipo non risulta essere analiticamente separabile. Sappiamo però che una gaussiana, nonostante sia radialmente simmetrica, è un'eccezione, risultando quindi separabile. Osservando il profilo diffusivo è evidente la somiglianza ad una gaussiana, o comunque ad una somma di esse. Risulta plausibile, quindi, che sia possibile ottenere un'approssimazione tramite una somma di gaussiane. Possiamo quindi andare a definire una funzione che definisca l'errore (Eq. 3.1) ed utilizzare quindi un software come MatLab per risolvere il problema di ottimizzazione.

$$E = \frac{\sqrt{\int_0^{\infty} r(R(r) - G_{sum}(r))^2 dr}}{\sqrt{\int_0^{\infty} r(R(r))^2}} \quad (3.1)$$

I due parametri che varieranno durante l'ottimizzazione per ottenere il migliore fit sono:

1. La varianza: definisce quanto sia più, o meno, "piatta" la funzione. Una varianza bassa vuol dire che avremo un picco molto significativo, mentre una varianza alta sarà rappresentata da una funzione molto "ampia".
2. I pesi: definiscono quanto assorbimento ci sia per ciascuno dei canali RGB. Inoltre questi valori sono sottoposti ad un ulteriore vincolo, ovvero che la sommatoria dei pesi sia = 1, in quanto vogliamo garantire la conservazione di energia.

Da questo processo di ottimizzazione otterremo le gaussiane la cui somma approssima il profilo diffusivo originale. Maggiore è il numero di gaussiane che utilizziamo per l'approssimazione, maggiore sarà la precisione di quest'ultima, al costo, però, di dover fare più convoluzioni. Tutto questo processo è stato presentato da d'Eon nel 2007 in [9], andando però poi ad applicare il profilo diffusivo approssimato in texture space, con tutti i limiti che ne conseguono, come abbiamo visto nella relativa sezione 3.1.1.

Per risolvere la maggior parte delle problematiche relative all'implementazione in texture space, Jimenez nel 2009 propone una variante della tecnica che opera in screen space [20]. Il processo è concettualmente identico a quello presentato da d'Eon, ovvero andiamo a mantenere separate la componente speculare e la componente diffusiva, renderizzando tutta la scena nei 2 render target separati, applicando le convoluzioni solo alla componente diffusiva. A differenza dell'approccio in texture space, però, andiamo ad utilizzare due ulteriori render target, ovvero:

1. Matte: determinato durante la fase iniziale di rendering e utilizzato successivamente per effettuare lo stencil test, in modo da eseguire le convoluzioni solo sulle parti della scena che necessitano di subsurface scattering.
2. Depth: necessario per modulare le dimensioni e la forma del nucleo utilizzato per effettuare le convoluzioni. Più un elemento è distante, più sarà piccolo il nucleo. Da notare che per come abbiamo presentato l'algoritmo fino ad ora, i nucleo sono coplanari rispetto allo schermo, quando in realtà dovrebbero seguire la curvatura della superficie su cui vengono applicati. Per ottenere un'approssimazione di questo comportamento vengono computati dinamicamente degli pseudo stretch factor (Eq. 3.2) in base alla derivata della profondità del pixel relativamente alla profondità dei pixel facenti parte dello stesso quad. Questo calcolo della derivata è lo stesso utilizzato dalla GPU per determinare quale mipmap utilizzare quando viene fatto il texturing. È possibile fare questa computazione grazie al metodo con cui viene fatto lo shading, che non è per pixel, ma per quad, ovvero un insieme di pixel 2x2 [12]. Otteniamo quindi le seguenti formule, le quali producono dei valori che svolgono un lavoro simile a quello degli stretch factor utilizzati in texture space.

$$\begin{aligned} s_x &= \frac{\alpha}{d(x, y) + \beta \text{abs}(\nabla_x d(x, y))} \\ s_y &= \frac{\alpha}{d(x, y) + \beta \text{abs}(\nabla_y d(x, y))} \end{aligned} \quad (3.2)$$

Da notare la presenza di due parametri di controllo  $\alpha$  e  $\beta$  all'interno delle equazioni per gli stretch factors (3.2). Il parametro  $\alpha$  determina direttamente il livello globale di subsurface scattering, mentre  $\beta$  determina quanto il gradiente della profondità della geometria sottostante influenzi il livello di subsurface scattering. I valori per queste due variabili vengono determinati in maniera euristica e Jimenez suggerisce

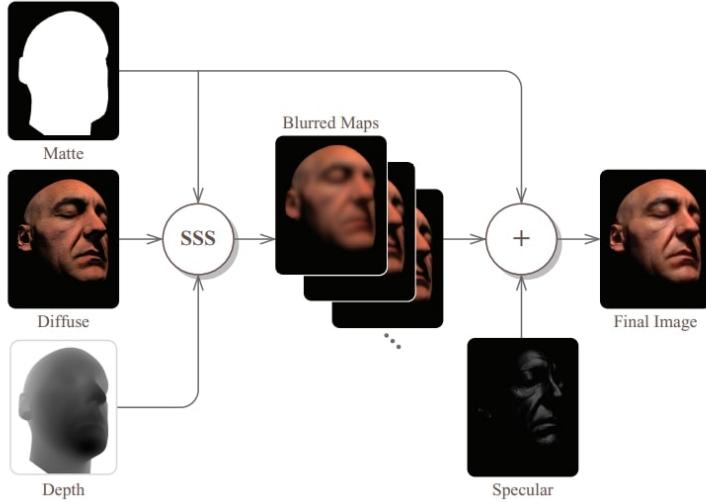


Figura 3.2: l'intero processo per produrre l'effetto di SSS con il metodo proposto da Jimenez [20]

$\alpha = 11$  e  $\beta = 800$ , almeno come punto iniziale da cui partire. Riassumendo il tutto, il processo è descritto in Fig. 3.2.

### 3.3 Jimenez (2015)

Jimenez presenta una nuova tecnica per il Screen-Space Subsurface Scattering (SSSS) non particolarmente dissimile nell'implementazione rispetto a [20], ma piuttosto nella determinazione dei nuclei da utilizzare per fare le convoluzioni. Le convoluzioni continuano, come abbiamo visto precedentemente, ad essere fatte in  $2N$  passaggi con un nucleo 1D. L'obiettivo è quello di avere  $N = 1$ , ovvero un solo nucleo separabile, in modo tale da fare solo 2 passi convolutivi, uno orizzontale ed uno verticale. Per poter ottenere ciò andiamo a rilassare uno dei vincoli relativi al profilo diffusivo approssimato presentato in precedenza, ovvero la sua simmetria radiale. Nonostante questo possa sembrare problematico per quanto concerne la qualità finale delle convoluzioni, vedremo come sotto certe condizioni di illuminazione e prendendo determinati accorgimenti riguardanti l'orientamento dei nuclei, la differenza rispetto all'utilizzo di un profilo diffusivo radialmente simmetrico sarà minima. Tenendo conto che stiamo lavorando con profili diffusivi discretizzati, possiamo vederli come matrici, ed è possibile quindi verificare come la maggior parte dell'energia diffusiva sia rappresentata dai primi valori singolari (Fig. 3.3). Questa informazione ci indica che è possibile ottenere una buona approssimazione di rango basso. Se però andiamo a determinare solo il primo componente della Singular Value Decomposition (SVD) di rango 1 è evidente come questo non approssimi correttamente il profilo diffusivo originale (Fig. 3.4).

Tenendo in conto, inoltre, che secondo il teorema di Eckart-Young la SVD produce la migliore possibile ricostruzione in termini di RMS [22]. Il problema è dato dal fatto che l'algoritmo non tiene conto di come percepiamo l'energia diffusiva del subsurface scattering, ovvero che risulta molto più significativa la coda rispetto al picco. Jimenez presenta quindi una serie di metodi con cui ottenere i nuclei approssimati.

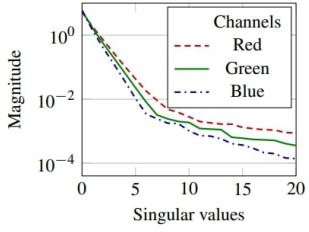


Figura 3.3: decomposizione in singular values del profilo diffusivo originale [22]



Figura 3.4: Profili diffusivi ottenuti mediante singular value decomposition con diversi valori di rango, rispetto al ground truth [22]

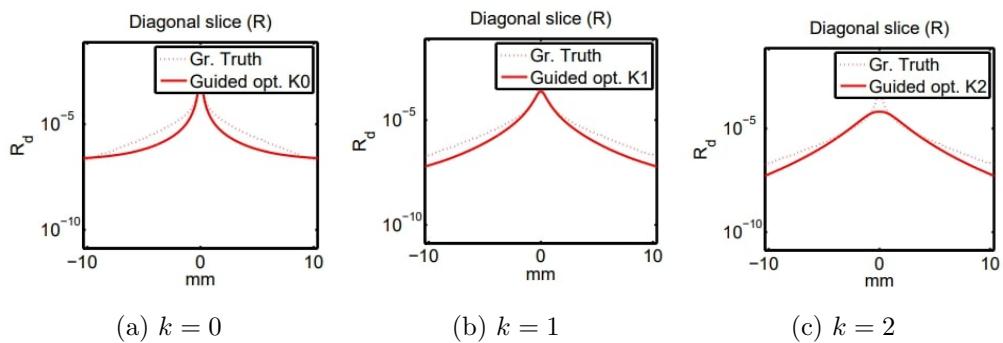


Figura 3.5: Grafici rappresentati i profili diffusivi, in questo caso per il canale rosso, ottenuti mediante guided optimization con i diversi  $k$  rispetto al ground truth. Con  $k = 0$  è evidente come l'ottimizzatore si concentreri di più nel ottimizzare il picco rispetto alla coda. Con  $k = 2$ , invece, il picco viene trascurato a favore della coda [22]

### 3.3.1 Guided Optimization

Con questo metodo, miriamo a risolvere un problema di minimizzazione in cui si cerca di ridurre al minimo la discrepanza tra il prodotto convolutivo  $a(x) * a(y)$  ed il profilo diffusivo originale  $R(x, y)$ , garantendo contemporaneamente la conservazione dell'energia (Eq. 3.3). Andiamo quindi a risolvere il problema di ottimizzazione soggetto al vincolo  $\|R_d\|_1 = \|a\|_1^2$  per garantire la conservazione di energia.

$$a_s = \operatorname{argmin}_a \int_{\mathbb{R}^2} \Gamma(x, y, k) (R_d(x, y) - a(x)a(y))^2 dxdy \quad (3.3)$$

$$\Gamma(x, y, k) = (x^2 + y^2)^{k/2} (1 - e^{-bx^2}) (1 - e^{-by^2}) \quad (3.4)$$

Purtroppo, nella sua forma base, ovvero senza la funzione  $\Gamma$  (o equivalentemente con il parametro  $k$  di tale funzione settato a 0), l'ottimizzazione tende a concentrarsi principalmente nell'approssimare il picco del profilo, trascurando la coda (Fig. 3.5). Per mitigare questo problema, introduciamo la *funzione guida*  $\Gamma$  (Eq. 3.4) che viene modulata da un parametro  $k$ . Aumentando questo parametro, la funzione guida porta progressivamente l'ottimizzatore a maggiore importanza alla coda del profilo, penalizzando il picco e, di conseguenza, trascurando maggiormente i dettagli (Fig. 3.5c). È fondamentale trovare un equilibrio ottimale, e secondo Jimenez, un valore di  $k = 2$  risulta essere un buon compromesso.

### 3.3.2 Analytic Pre-Integration

A differenza dell'approccio precedente qui non cerchiamo di rendere i nuclei separabili vicini al profilo diffusivo originale a livello matematico, ma piuttosto a rendere i risultati ottenuti dalla loro applicazione percettivamente simili. Nonostante non sia possibile riprodurre il comportamento di un profilo diffusivo mediante una soluzione analitica al problema nel caso generale, è possibile farlo se vincoliamo il problema ad un certo insieme di segnali in input. In particolare se consideriamo la classe di segnali di irradianza additivamente separabili, ovvero quelli che rispettano l'equazione  $E(x, y) = E_1(x) + E_2(y)$ . Tenendo in considerazione questo vincolo possiamo riscrivere l'integrale che definisce la radianza uscente  $M_e$  ottenendo il seguente risultato:

$$\begin{aligned} M_e(x, y) &= \int \int E(x', y') R_d(x - x', y - y') dx' dy' \\ &= \int E_1(x') R_d(x - x', y - y') dy' dx' \\ &\quad + \int E_2(y') R_d(x - x', y - y') dx' dy' \\ &= \int \int E(x', y') \frac{1}{\|a_p\|_1} a_p(x - x') a_p(y - y') dx' dy' \end{aligned} \tag{3.5}$$

Nell'equazione 3.5  $a_p$  rappresenta il nucleo 1D pre-integrato di  $R_d$ , dove data la simmetria radiale ne consegue che  $a_p(x) = a_p(y)$ . Inoltre per definizione abbiamo che  $\|a_p\|_1 = \|R_d\|_1$ . Grazie a questo processo di pre-integrazione possiamo quindi definire il nucleo  $A_p$  del profilo diffusivo originale, ossia:

$$A_p = \frac{1}{\|R_d\|_1} a_p(x) a_p(y) \tag{3.6}$$

Questo approccio produce un risultato identico al ground truth, in presenza di irradianza additivamente separabile. Risulta essere un'approssimazione sufficientemente buona anche rompendo il vincolo di separabilità additiva, in quanto la maggior parte dei segnali reali  $E(x, y)$  possono essere approssimati localmente con funzioni additivamente separabili [22]. Nonostante produca sostanzialmente gli stessi risultati del profilo diffusivo originale (quando si verificano i vincoli precedentemente definiti), questa tecnica è limitata dal fatto che il risultato non è modificabile direttamente. È necessario, invece, ottenere un nuovo profilo diffusivo con i parametri desiderati mediante simulazione di Monte Carlo utilizzando un software come MCML [30], per solo poi applicare il passaggio di pre-integrazione. Significa, quindi, che il suo utilizzo è limitato in quanto non è presente alcun controllo immediato per gli artisti, limitandone quindi la praticità in ambienti produttivi.

### 3.3.3 Artist Friendly Model

Gli ultimi due metodi presentati sono caratterizzati dalla mancanza di controllo da parte di un artista. Considerando, invece, i metodi basati su somma di gaussiane visti in precedenza [9][20], questi presentano molti gradi di libertà nella modifica del profilo, rendendo difficoltoso e poco intuitivo il controllo del risultato. L'ultimo modello presentato da Jimenez si pone l'obiettivo di fornire un controllo limitato a pochi parametri intuitivi. In particolare il modello è basato sulla somma pesata di due nuclei gaussiani 1D, dove uno rappresenta la distanza di scattering vicina, mentre l'altro la distanza di scattering lontana.

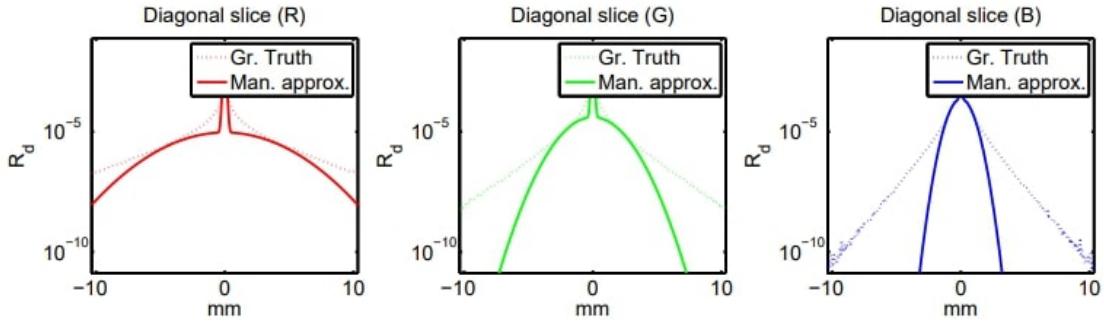


Figura 3.6: visualizzazione di un profilo diffusivo  $R_d$ , separato nei tre canali RGB, ottenuto dalla somma di due gaussiane controllate dai parametri  $\sigma_n$  e  $\sigma_f$ . Il picco è la conseguenza della gaussiana che modella la near scattering distance, in quanto si concentra nel preservare i dettagli ad alta frequenza [22]

Una somma pesata di queste due gaussiane, secondo un certo termine  $w$  (Eq. 3.7), produce la base del nucleo separabile che stiamo modellando, ovvero  $A_m(x, y) = a_m(x)a_m(y)$ .

$$a_m(x) = wG(x, \sigma_n) + (1 - w)G(x, \sigma_f) \quad (3.7)$$

Una gaussiana è definita dalla sua varianza, o equivalente deviazione standard  $\sigma$ , e da un valore medio  $\mu$ . In questo caso si assume la gaussiana centrata su zero, con la conseguenza che il valore medio risulta pari a zero. Possiamo quindi ignorare tale parametro, mentre quello che risulta significativo è la deviazione standard. Intuitivamente modifiche a questo parametro portano ad appiattire più o meno la curva (Fig. 3.6). Ed è proprio questo che manipolano i due parametri  $\sigma_n$  e  $\sigma_f$ , i quali controllano rispettivamente la distanza di scattering vicina e la distanza di scattering lontana in termini di deviazione standard.

I parametri  $\sigma_n$  e  $\sigma_f$ , in realtà, non sono i valori di deviazione standard direttamente specificati dall'artista, ma sono ottenuti a partire da questi ultimi mediante la formula  $\sigma_n = \sigma_n's$  (ed equivalentemente  $\sigma_f = \sigma_f's$ ). Il termine  $s$  in questione rappresenta la scala ed è definito in termini di unità in world space. Il suo utilizzo permette di definire i valori delle distanze di scattering delle due gaussiane indipendentemente dalla scala che verrà poi utilizzata nella scena. Da notare il fatto che la gaussiana (Eq. 3.8) sia 1D, con la conseguenza che il fattore di normalizzazione non è  $1/2\pi\sigma_n^2$  come nelle gaussiane 2D, ma risulta essere  $1/2\pi\sigma_n$ .

$$G(x, \sigma) = \frac{1}{2\pi\sigma} e^{-x^2/2\sigma^2} \quad (3.8)$$

Gli artisti possono quindi creare interattivamente i profili diffusivi modificando solo i parametri di near e far scattering per ciascun canale. I profili ottenuti vengono discretizzati al volo ottenendo i nuclei che verranno utilizzati per effettuare le convoluzioni, allo stesso modo con cui vengono applicate per le altre proposte di Jimenez in questa pubblicazione [22].

### 3.3.4 Artefatti

La scelta iniziale di non garantire il vincolo di radialità del profilo diffusivo, in modo da poter ottenere un nucleo separabile, implica anche una serie di problematiche. In particolare, queste si verificano quando il footprint del segnale in input (l'irradianza) diventa più piccolo della banda che il nucleo riesce a gestire. Gli artefatti che ne conseguono sono sostanzialmente due, nonostante la fonte che li causa sia la stessa.

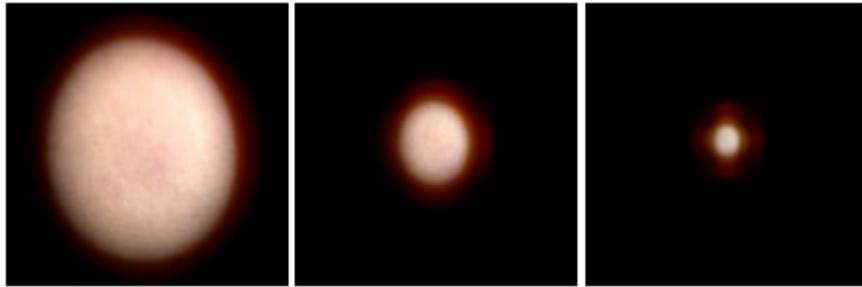


Figura 3.7: Artefatto causato dalla non radialità del profilo diffusivo. Diminuendo la dimensione della feature del segnale, l’artefatto diventa sempre più evidente. Fortunatamente questo è un caso estremo di illuminazione raramente presente nella realtà [22]

1. Alla progressiva riduzione della dimensione delle feature del segnale, si rende sempre più evidente come il filtro che è stato applicato non sia radialmente simmetrico (Fig. 3.7). Ciò che viene prodotto infatti ha una forma chiamata “a stella” (nonostante ricordi più una croce). Fortunatamente questo artefatto è un caso patologico, causato dalle condizioni di illuminazione estreme, ovvero qualcosa di raramente presente in scene “normali”.
2. L’altro tipo di artefatto è presente, invece, anche in condizioni “normali” di illuminazione. Si mostra come una griglia visibile su tutta la superficie del materiale (Fig. 3.8). È sempre lo stesso artefatto a forma di stella del caso precedente, ma in questo caso ripetuto su tutta la superficie. In questo caso la soluzione è quella di ruotare casualmente gli assi per ciascun insieme di campioni, in modo tale da rompere il pattern a croce [22][17]. Questa rotazione non verrà, però, applicata a tutti i campioni del nucleo, ma solo a quelli entro un certo range. Il motivo è che si vuole evitare il trashing della cache della GPU, causato dal campionamento di un texel di una texture che risulta essere “lontano” dal fragment corrente nel fragment shader. Questo fenomeno deriva dal fatto che quando un wave di thread è in esecuzione sulla GPU e sta operando su un blocco di frammenti, nella sua cache sono probabilmente presenti anche blocchi di frammenti vicini. Quando campioniamo una parte di una texture non presente nella cache, questa deve essere caricata dalla memoria video principale della GPU, con conseguente impatto sulle performance. Inoltre finisce anche per sostituire valori nella cache che sarebbero ancora stati utili, con conseguenti altri cache miss, portando a quello che viene chiamato cache trashing. L’idea è quindi quella di non ruotare tutti i campioni, ma solo quelli entro una certa distanza dal centro del nucleo. In particolare, Jimenez suggerisce una distanza minore del 10% della larghezza del nucleo, mentre del 30% nel caso volessimo rimuovere gli artefatti anche nei primissimi piani.

### 3.4 Golubev (2018)

Golubev osserva come la maggior parte dei modelli basati su Gaussian mixture non siano particolarmente intuitivi per gli artisti che li utilizzano. Gli artisti devono quindi o utilizzare direttamente i parametri misurati fisicamente o fare un fit manuale del modello ad occhio. Questo è reso problematico dal fatto che il modello presenta multipli gradi di libertà e che non necessariamente tutte le combinazioni producono risultati significativi. Nonostante sia possibile, comunque, produrre ottimi risultati, è necessario un

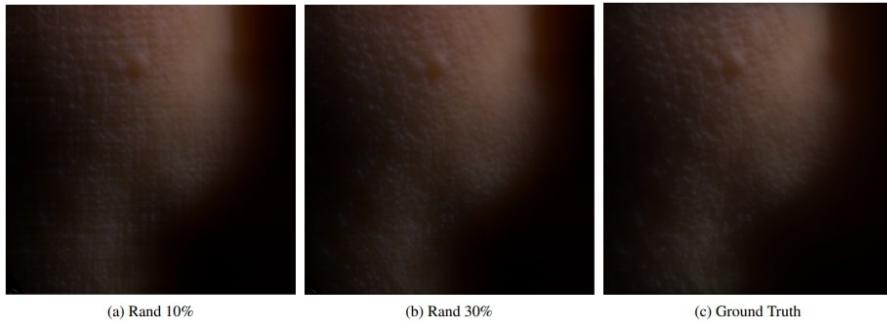
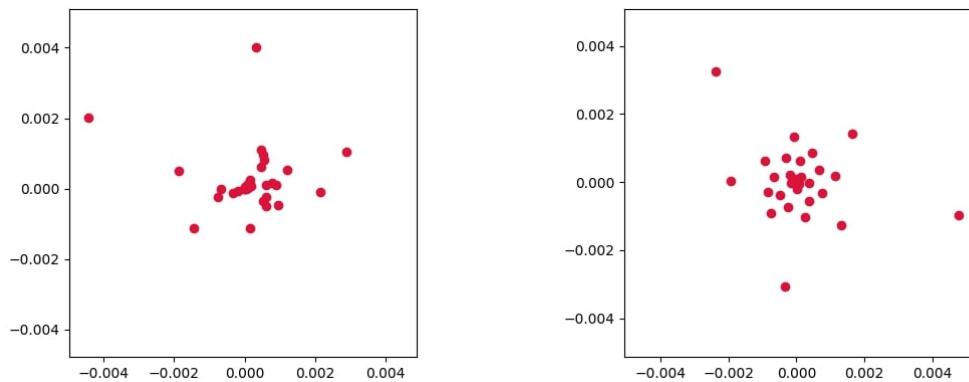


Figura 3.8: Altro artefatto causato dalla non radialità del profilo diffusivo. Nei primissimi piani, anche con la rotazione dei campioni ad una distanza minore del 10% della larghezza del nucleo, è ancora visibile la "griglia". In questo caso, ovvero da molto vicino, è necessario aumentare il threshold al 30% per far scomparire la "griglia" [22]

importante investimento temporale per ottenere tali risultati. Inoltre, modella contemporaneamente sia il fenomeno di single scattering che di multiple scattering. Golubev [13] propone quindi di implementare in screen space il modello diffusivo normalizzato presentato da Burley [7], che abbiamo introdotto in precedenza nella sezione 2.3. Come abbiamo visto, tale modello è un ottimo fit rispetto ai valori di riferimento prodotti tramite simulazioni di Monte Carlo. Prima di implementare effettivamente il SSS, bisogna tenere conto anche della BRDF diffusiva che si utilizza. Questo perché, se la distanza di scattering in screen space risulta essere minore di un pixel, la BSSRDF può essere approssimata dalla BRDF. Il vantaggio di questa osservazione è che ad una certa distanza possiamo disabilitare completamente il SSS, approssimando il fenomeno con la BRDF. Per fare ciò è però necessario che entrambi i modelli producano risultati simili dopo una certa distanza. Golubev consiglia di utilizzare la Disney Diffuse BRDF, sempre proposta da Burley. Nonostante non sia completamente fisicamente accurata, è comunque meglio rispetto ad utilizzare un semplice termine Lambertiano. Per implementare il SSS, possiamo adottare un approccio che lo concepisca come un tipo di energy conserving blur sulla superficie. Effettuiamo le convoluzioni in screen space, mentre utilizziamo il depth buffer per tener conto della superficie. Idealmente per fare le convoluzioni dovremmo campionare direttamente la superficie, ma questo approccio è estremamente dispendioso, non risultando adatto per un sistema real-time. Andiamo invece a distribuire i campioni su un disco, dove tali campioni sono stati pre-computati. Questi ultimi vengono distribuiti sul disco campionando l'angolo theta mediante la *sequenza di fibonacci*, in modo tale da avere una distribuzione uniforme (Fig. 3.9).

Usare la distribuzione sul disco per campionare una superficie non planare non è propriamente corretto, in quanto la proiezione dei punti sul disco sulla superficie distorce le distanze. Questa distorsione fa sì che l'iniziale distribuzione dei punti venga a perdere (Fig. 3.10). Un altro aspetto da considerare è l'orientamento del disco. La soluzione più semplice è quella di orientare il disco in maniera perpendicolare rispetto alla telecamere. Anche questo aspetto risulta essere un'approssimazione in quanto il disco dovrebbe essere tangente rispetto al punto sulla superficie che si sta considerando (Fig. 3.11). Golubev osserva come fortunatamente, in pratica, queste approssimazioni non producano risultati significativamente diversi.

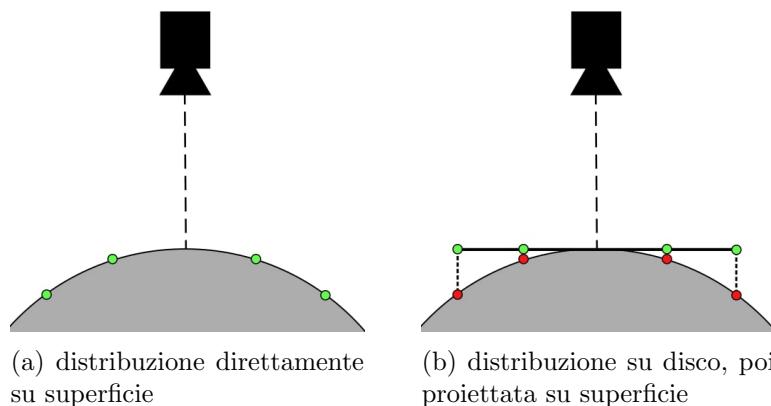
Da notare come in (Fig. 3.9) la distribuzione delle distanze dall'origine non sia né uniforme, né casuale. Questo perché, dato l'elevato costo di un campione, vogliamo minimizzare il numero di campioni



(a) distribuzione casuale dell'angolo rispetto all'origine

(b) distribuzione uniforme dell'angolo rispetto all'origine

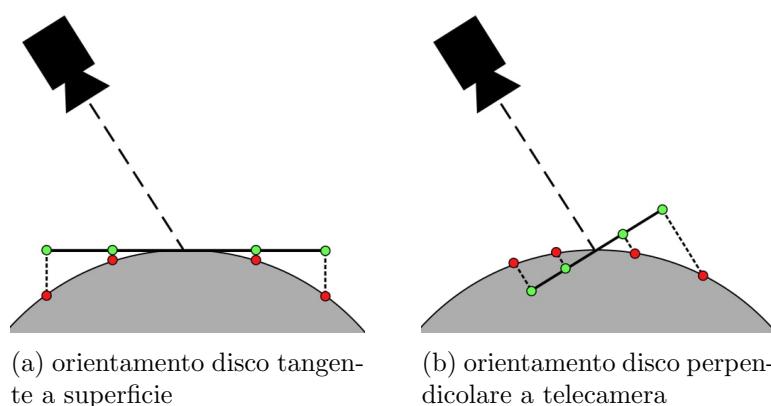
Figura 3.9: uno stesso insieme di distanze dall'origine, distribuite radialmente scegliendo l'angolo o a) casualmente, o b) sequenza di fibonacci



(a) distribuzione direttamente su superficie

(b) distribuzione su disco, poi proiettata su superficie

Figura 3.10: diversi tipi di distribuzione dei punti di campionamento



(a) orientamento disco tangente a superficie

(b) orientamento disco perpendicolare a telecamera

Figura 3.11: diversi tipi orientamento del disco

che utilizziamo. Vogliamo quindi selezionare i campioni nella maniera più efficace possibile. Per fare ciò viene adottata una tecnica chiamata importance sampling.

### 3.4.1 Importance Sampling

L'idea dell'importance sampling è basata sul fatto che certi campioni abbiano una maggiore "importanza" rispetto ad altri nell'influenzare il risultato, ovvero la stima finale. Scegliere quindi campioni "importanti", rispetto a sceglierli casualmente, porta ad una riduzione della varianza nella stima finale. Nell'ambito del rendering, la riduzione della varianza si traduce in una riduzione del rumore. La scelta di tali campioni viene quindi fatta in base ad una certa distribuzione, che rappresenti matematicamente l'importanza dei possibili campioni, come per esempio una funzione di densità di probabilità. Golubev osserva come il profilo diffusivo presentato da Burley e Christensen [7] sottenda un'area unitaria, risultando quindi essere in sostanza una funzione di densità di probabilità (PDF) ed integrandola possiamo ottenere la funzione cumulativa (CDF) che indicheremo con  $P$ , ossia:

$$P(r) = 1 - \frac{1}{4}e^{-sr} - \frac{3}{4}e^{-sr/3} \quad (3.9)$$

Il passaggio di integrazione lo facciamo in quanto uno dei metodi più comuni per ottenere campioni che rispecchino la distribuzione della PDF consiste nell'utilizzare l'inversa della CDF. Passando, poi, valori generati uniformemente a tale funzione invertita, otterremo campioni che seguono la probabilità della PDF. Tali campioni vengono poi utilizzati per stimare il contributo del subsurface scattering mediante integrazione con il metodo di metodo di Montecarlo. Non invertiamo la PDF perché potrebbe avere più valori per uno stesso  $y$ , a differenza della CDF che essendo monotonicamente crescente non presenta questo problema.

Nel momento in cui una funzione non è analiticamente invertibile, uno dei metodi più comuni è l'utilizzo di algoritmi di Root Finding. La maggior parte degli algoritmi per trovare una soluzione a questo tipo di equazioni sono basati su trial and error:

1. fai un guess per  $x_n$
2. fai un guess per  $x_{n+1}$  e  $f(x_n)$
3. Ripeti finché  $f(x_n)$  non raggiunge un certo threshold che rappresenta la vicinanza allo zero della funzione

Gli algoritmi root-finding sono definiti dalla loro convergenza (Eq. 3.10) dove  $x_0$  è il valore corretto (ovvero la soluzione),  $c$  è una costante e  $m$  l'ordine di convergenza. Se  $m = 1$  la convergenza si dice lineare.

$$\lim_{k \rightarrow \infty} (x_{k+1} - x_0) = c(x_k - x_0)^m \quad (3.10)$$

Uno degli algoritmi di root-finding più noti è il metodo di Halley. Questo è un metodo iterativo per risolvere equazioni non lineari. Si tratta di un'estensione del metodo di Newton-Raphson e utilizza una serie di approssimazioni successive per trovare le radici di una funzione. Il metodo di Halley è particolarmente utile perché può convergere più rapidamente rispetto ad altri metodi iterativi. La

formula di Halley per trovare la radice  $x$  di un funzione  $f(x)$  è la seguente:

$$x_{n+1} = x_n - \frac{2f(x_n)f'(x_n)}{2[f'(x_n)]^2 - f(x_n)f''(x_n)} \quad (3.11)$$

Una caratteristica utile del metodo di Halley è che ha una velocità di convergenza cubica rispetto alla velocità di convergenza quadratica del metodo di Newton-Raphson.

Questo era il metodo inizialmente utilizzato da Golubev per invertire la funzione [13] in quanto Burley e Christensen avevano dichiarato in [7] che la cdf non fosse analiticamente invertibile. Successivamente si scoprì che quella non fosse un'affermazione corretta, con conseguentemente Golubev che trova la soluzione analitica [14]. Utilizzando un software come Mathematica e dopo una serie di semplificazioni ottiene la seguente inversa della cdf

$$x = 3 \log \left( \frac{1 + G(u)^{-1/3} + G(u)^{-1/3}}{4u} \right) \quad (3.12)$$

dove

$$G(u) = 1 + 4u \left( 2u + \sqrt{1 + 4u^2} \right) \quad (3.13)$$

Il valore  $u$  deriva dal fatto che se proviamo ad ottenere l'inversa  $x = P^{-1}(y)$  quello che ne risulta è un'equazione estremamente complicata. La soluzione è quindi trovare l'inversa di  $P(x) = 1 - u$ , invece che di  $P(x) = y$ , ottenendo quindi la semplice equazione 3.12. Conoscendo quindi l'inversa è possibile determinare la distribuzione dei campioni sul disco che abbiamo visto in precedenza. Da lì è sufficiente, mediante integrazione di Monte Carlo, utilizzare tali campioni per stimare il contributo di essi dato uno specifico profilo diffusivo di Burley.

Tale profilo, come abbiamo visto nella relativa sezione 2.3, può essere controllato con il parametro di diffuse mean free path, il quale è un parametro fisico. Questo approccio permette quindi agli artisti di creare interattivamente il profilo con dei parametri intuitivi, potendosi basare anche su dei valori misurati nel mondo reale per una certa selezione di materiali.

### 3.5 Jimenez Traslucenza (2010)

Fino ad ora abbiamo visto solo tecniche che implementano il SSS solo con l'informazione presente immediatamente in screen space. Ciò significa che non abbiamo a disposizione l'informazione globale della scena, ovvero manca l'irradianza che influisce sulle superfici non visibili alla camera. L'ovvia conseguenza è la mancata modellazione di fenomeni di back lit translucency. Jimenez [21] propone quindi un modello fondato sull'approccio proposto da Simon Green, che sfrutta le shadow map per computare la distanza di trasmissione all'interno dell'oggetto (Fig 3.12).

Per quanto riguarda l'attenuazione della luce durante il fenomeno di trasmissione nell'approccio di Green, questa viene implementata semplicemente mediante la legge di Beer, ovvero  $T(s) = e^{-s\sigma_t}$ , che descrive la relazione presente tra la quantità di luce assorbita e lo spessore del materiale. Nel modello proposto da Jimenez, viene mantenuta la stessa tecnica utilizzata da Green per computare la distanza di trasmissione, mentre viene modificata l'equazione del coefficiente di trasmissione  $T(s)$ . Questo perché, la legge di Beer descrive il relativo fenomeno all'interno di un materiale composto da un unico layer. È

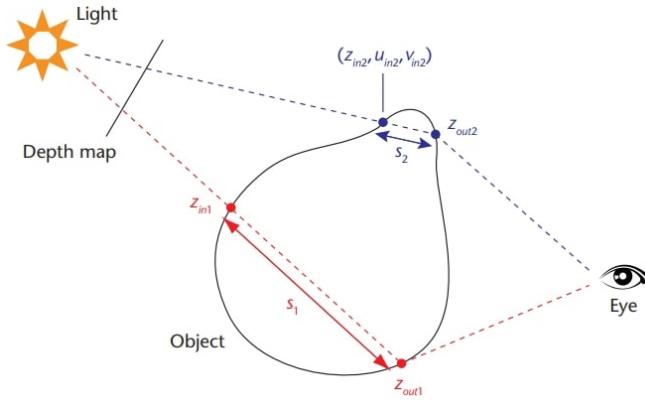


Figura 3.12: Raffigurazione di come viene determinata la distanza di trasmissione. Si usano le stesse depth map utilizzate per lo shadow mapping ed anche il metodo con cui vengono utilizzate non è particolarmente dissimile. Si determina la distanza di  $z_{out}$  dalla sorgente di luce e poi si usa la depth map per determinare la distanza di  $z_{in}$  dalla sorgente. La differenza fra queste due distanze produce  $s$ , ovvero la distanza di trasmissione [21]

noto, però, che la pelle sia composta da multipli strati, quali derma, epidermide, etc., i quali influenzano diversamente l'attenuazione della luce all'interno della pelle, necessitando quindi di un modello che ne tenga conto. La definizione di quest'ultima parte, però, da una serie di osservazioni che vanno a porre dei vincoli al problema, rendendolo più semplice da risolvere:

1. Per la maggior parte degli oggetti sottili è possibile approssimare la normale sul lato posteriore con l'opposta della normale sul lato anteriore. Nella maggior parte dei casi questa approssimazione risulta sufficiente ed è perfettamente corretta nel momento in cui i 2 lati della superficie sono paralleli. Da questo possiamo quindi assumere la seguente ridefinizione della normale  $N_A$

$$N_A = -N_C \quad (3.14)$$

dove A è il punto di entrata della luce sul lato posteriore, mentre C è il punto di uscita dal lato anteriore.

2. Il fenomeno di trasmissione ha la caratteristica di essere a bassa frequenza in presenza di materiali geometricamente spessi o aventi un alto valore di assorbimento. Una proprietà di questo tipo di materiali è che presentano un alto fattore di scattering, il quale porta alla rapida perdita di direzionalità della luce trasmessa rispetto alla direzione della luce entrante. Ne segue che anche nel caso in cui decidessimo di utilizzare delle normali ad alta frequenza per calcolare l'irradianza, la trasmissione risulterebbe comunque a bassa frequenza. Da qui possiamo utilizzare l'approssimazione per cui utilizziamo le normali dei vertici per computare la trasmissione, in modo tale da non richiedere la necessità di fare delle convoluzioni in più sull'irradianza.
3. Dato un oggetto illuminato da dietro, l'osservatore non ha una buona percezione dell'irradianza presente sul lato posteriore. Inoltre, considerando la pelle umana, l'albedo non varia particolarmente considerando l'interezza della superficie. Da queste due osservazioni possiamo fare l'assunzione che l'irradianza posteriore sia di un valore localmente costante e che l'albedo, sempre posteriore, possa essere approssimato dall'albedo visibile dall'utente al punto di uscita. Tutto ciò descritto in

formule corrisponde a

$$E(x, y) = E = \alpha_c \max(-N_c + L, 0.0) \quad (3.15)$$

L'obiettivo posto è quello di, dato un punto su di una slab planare identificato dalle coordinate  $(x, y)$ , computare la radianza trasmessa uscente  $M(x, y)$ . Partendo da un profilo diffusivo  $R(r)$  ottenuto tramite somma di gaussiane come visto in [9],  $M(x, y)$  si ottiene convolvendo l'irradianza con tale profilo diffusivo. Abbiamo quindi la seguente formulazione:

$$M(x, y) = \int \int E(x, y) R\left(\sqrt{r^2 + d^2}\right) dx dy \quad (3.16)$$

Ricordandoci della terza assunzione che abbiamo fatto, ovvero  $E(x, y) = E$ , possiamo semplificare il problema, ottenendo:

$$\begin{aligned} M(x, y) &= E \int \int R\left(\sqrt{r^2 + d^2}\right) dx dy \\ &= E \int_0^\infty R\left(\sqrt{r^2 + d^2}\right) dr \end{aligned} \quad (3.17)$$

In questo caso ciò che è stato fatto è l'aver riscritto l'integrale in termini del raggio  $r$ , grazie al fatto che il profilo diffusivo fosse radialmente simmetrico. Sostituendo il profilo diffusivo con la relativa somma di gaussiane definita da d'Eon [9], otteniamo il seguente risultato:

$$M(x, y) = E \sum_{i=1}^k w_i e^{-d^2/v_i} \quad (3.18)$$

L'integrale della seconda equazione in 3.17 scompare in quanto le gaussiane che abbiamo definito hanno risposta unitaria (il loro integrale è  $= 1$ ). A questo punto è possibile approssimare  $d$  con il valore  $s$  che rappresenta la distanza misurata usando le shadow map. Dopo tutte queste operazioni ciò che abbiamo ottenuto è una funzione per calcolare la radianza uscente trasmessa  $M(x, y)$  che dipende solo dall'irradianza  $E$  e dalla distanza di trasmissione  $s$ . In particolare ciò che possiamo fare è pre-computare il coefficiente di trasmissione  $T(s)$  in una texture di look-up o tabella di look-up, da usare poi quando si cerca di computare la radianza uscente derivante dalla trasmissione.

Da notare che possiamo fare la somma, invece che il blend, dei contributi di riflettanza e trasmittanza, in quanto usiamo la normale inversa per computare la trasmittanza. Questo implica che i due contributi non potranno mai essere simultanei.

## 3.6 Texturing

Tutti i modelli che abbiamo visto fino ad ora si basano su un'assunzione molto forte, ossia l'omogeneità del materiale. Ciò implica che, i risultati ottenuti utilizzando tali profili diffusivi avranno un colore perfettamente uniforme. Questo non è, però, il comportamento di molti materiali, quali la pelle, in cui le non uniformità nei vari strati del medium forniscono l'effetto di non omogeneità. Questo tipo di fenomeno è difficile da simulare, particolarmente in tempo reale, rendendo necessario l'utilizzo di un'approssimazione, ossia l'applicazione di texture contenenti i dettagli della superficie. Tale applica-

zione può essere effettuata in diverse fasi rispetto al momento in cui viene calcolato il contributo del subsurface scattering, producendo, di conseguenza, effetti diversi. Di seguito vengono indicate le tre diverse configurazioni comunemente adottate [9]:

- **Post-Scatter:** in questo caso le convoluzioni con il profilo diffusivo vengono effettuate prima di moltiplicare il risultato ottenuto con la texture diffusiva. Così facendo si preservano i dettagli in quest'ultima, in quanto non vi viene applicato il blur delle convoluzioni.
- **Pre-Scatter:** in questo caso, invece, le convoluzioni con il profilo diffusivo vengono effettuate dopo aver già moltiplicato l'irradianza della superficie con la texture diffusiva. Facendo così si guadagna una maggiore diffusione del colore sulla superficie, perdendo, però, un pò del dettaglio fornito dalla texture.
- **Pre-Scatter & Post-Scatter:** l'ultimo approccio è una combinazione dei due precedenti, risultando quindi essere una via di mezzo. Consiste nel moltiplicare l'irradianza con una frazione della texture diffusiva prima di effettuare la convoluzione e moltiplicare quindi tale risultato con la la restante frazione della texture. Il rapporto fra la quantità da applicare prima e dopo la convoluzione può essere scelto in maniera arbitraria, ma è comune utilizzare un valore pari a 0.5.

Importante far notare come tutti e tre gli approcci non siano fisicamente perfettamente accurati, facendo quindi ricadere la scelta a semplici preferenze. Infatti, per ciascuna delle tecniche di subsurface scattering che abbiamo implementato è stato utilizzato l'approccio indicato dall'autore nella relativa pubblicazione [13][20][22].



# 4

## Campionamento a 2 Livelli Basato sul Contrasto

In questo capitolo, presentiamo una tecnica originale per implementare il subsurface scattering in tempo reale, basata sull’analisi del contrasto locale nell’immagine. La nostra metodologia mira a ottimizzare l’efficienza computazionale riducendo il numero di campioni necessari per calcolare il contributo del profilo diffusivo, compromettendo il meno possibile la qualità visiva del rendering finale. Il principio alla base della tecnica proposta è semplice: utilizziamo il contrasto determinato localmente per pixel (dove il contrasto è stato computato riutilizzando i campioni ottenuti durante un passo base) come metrica per determinare dinamicamente se utilizzare o meno ulteriori campioni in un passo extra. In aree ad alto contrasto, dove i dettagli fini e le variazioni di irradianza sono predominanti, è essenziale utilizzare un maggior numero di campioni per catturare accuratamente gli effetti del subsurface scattering. Al contrario, nelle regioni a basso contrasto, la riduzione del numero di campioni risulta in un significativo risparmio computazionale, mantenendo comunque una rappresentazione visivamente soddisfacente. Una descrizione più dettagliata dell’algoritmo verrà fornita successivamente.

Questo approccio adattativo permette non solo di migliorare l’efficienza del processo di rendering, ma anche di mantenere un elevato livello di dettaglio dove più necessario. La nostra tecnica si distingue per la sua capacità di bilanciare efficacemente la qualità visiva e le risorse computazionali in base alle condizioni presenti nella scena, con un minimo costo aggiuntivo per computare il contrasto. Ulteriormente è particolarmente semplice da implementare, in quanto non richiede modifiche alla pipeline grafica in uso.

Nel prosieguo del capitolo, discuteremo in dettaglio i criteri di selezione dei campioni basati sul contrasto locale e i risultati sperimentali che dimostrano il livello di efficacia della nostra metodologia. Come prova ulteriore, nel capitolo 5 faremo una valutazione con utenti per mettere a confronto percettivamente la nostra tecnica con quelle presentate in precedenza.

### 4.1 Razionale delle Assunzioni

L’idea di questo metodo nasce da una serie di osservazioni riguardanti il funzionamento del subsurface scattering e come questo venga percepito. Per prima cosa possiamo notare come il fenomeno sia il più pronunciato in zone dove c’è un alto contrasto causato dall’ombreggiatura, come è possibile vedere in



(a) SSS disabilitato



(b) SSS abilitato

Figura 4.1: Configurazione di illuminazione in cui la direzione della luce non è perpendicolare alla superficie illuminata. Da notare come ci sia una chiara differenza fra l'abilitare o meno il SSS.



(a) SSS disabilitato



(b) SSS abilitato

Figura 4.2: Configurazione di illuminazione in cui la direzione della luce è approssimativamente perpendicolare alla superficie illuminata. Da notare come ci sia poca differenza fra l'abilitare o meno il SSS.

Fig. 4.1. Sia nel caso in cui questa ombreggiatura sia causata dai dettagli dei pori della pelle, sia nel caso sia causata da intere porzioni della superficie (e.g. il naso che crea l'ombra sulla guancia). Nella Fig. 4.1a, dove il SSS non è attivo, i pori producono un effetto molto "ruvido" che detrae dal realismo della pelle. Quest'ultimo viene quindi introdotto implementando una qualche tecnica di SSS, come possiamo vedere in Fig. 4.1b. Il prima citato effetto "ruvido" è causato dalla microgeometria della superficie nel momento in cui questa viene illuminata da una direzione non perpendicolare. Se, invece, illuminiamo la superficie da una direzione approssimativamente perpendicolare possiamo notare come l'effetto sia sostanzialmente assente (Fig. 4.2). In questo caso, infatti, ne consegue che percettivamente la differenza fra assenza di SSS e presenza di SSS sia minima. Da questa analisi empirica possiamo dedurre che il SSS abbia un impatto molto più importante nel caso in cui l'irradianza sulla superficie abbia un aspetto più simile a 4.1a rispetto che a 4.2a. Ne consegue che, nel secondo caso, possiamo utilizzare un numero minore di campioni.

## 4.2 L'algoritmo

L'algoritmo che proponiamo non è un nuovo metodo per calcolare il contributo del subsurface scattering, ma piuttosto un meccanismo che dinamicamente stabilisce il numero di campioni da utilizzare per pixel. Ciò significa che non è legato ad una specifica tecnica di SSS, potendo quindi essere implementato come

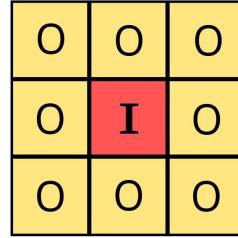


Figura 4.3: Configurazione comune di una finestra 3x3 utilizzata per computare il contrasto locale. In rosso è indicata la zona interna, mentre in giallo quella esterna. Vengono utilizzate anche configurazioni con dimensioni diverse, come, per esempio, 5x5 o 7x7.

estensione di un qualsiasi metodo di SSS basato su convoluzioni eseguite in un unico passo. Questo vuol dire che nella forma qui proposta non è adatto a tecniche che fanno uso di convoluzioni fatte in due passaggi (uno verticale ed uno orizzontale) o più, come per esempio quelle proposte da Jimenez [20][22]. Al contrario, il metodo proposto da Golubev [13] si distingue per l'esecuzione in un unico passaggio. Questa caratteristica lo rende particolarmente adatto alle nostre esigenze. Per questo motivo, abbiamo scelto di implementare il nostro meccanismo come un'estensione di tale metodo.

L'algoritmo, essendo un'estensione della tecnica di Golubev recupera gli stessi parametri di controllo per definire il profilo diffusivo utilizzato per il subsurface scattering. Con l'introduzione del meccanismo che proponiamo, ne vengono aggiunti altri due:

1. **Threshold:** se consideriamo che il contrasto locale computato produce un valore compreso nell'intervallo  $[0, 1]$ , con il parametro threshold andiamo ad indicare quale sia il valore di contrasto sufficientemente alto per attivare il campionamento extra. Maggiore è il threshold, maggiore sarà il numero di pixel che utilizzeranno i campioni extra. Nonostante attualmente sia un parametro modificabile per ogni singolo materiale, permettendo quindi di sperimentare più facilmente con i valori, in realtà può essere trasformato in una costante dello shader. In questo modo l'artista ha un aspetto in meno da considerare. Questo perché il parametro controlla, sostanzialmente, il bilanciamento fra performance e qualità, potendo quindi essere impostato a livello di progetto.
2. **Numero Campioni Extra:** il numero di campioni da utilizzare nel caso in cui venga attivato il campionamento extra al superamento del threshold per il contrasto. L'insieme di campioni è pre-computato allo stesso modo dei campioni utilizzati per il passo base, ovvero come viene fatto nella versione standard della tecnica di Golubev.

Il contrasto locale di un'immagine può essere computato utilizzando varie tecniche. Una delle più comuni implica l'uso di una finestra mobile. Di seguito vengono spiegati i passaggi che costituiscono un approccio di tale tipo:

1. **Definizione della finestra:** Scegli una finestra di dimensioni  $n \times n$  (Fig. 4.3). Una dimensione tipica potrebbe essere 3x3, 5x5 o 7x7 pixel. È possibile utilizzare la stessa finestra su tutta l'immagine, ma è anche possibile selezionarla dinamicamente in base a determinate condizioni [15].

2. **Scansione dell’immagine:** Muovi la finestra su tutta l’immagine, che nel nostro caso è quella ottenuta dal passo di illuminazione della pipeline, contenente quindi l’irradianza della scena. Per ogni posizione della finestra, la centriamo su un pixel specifico (il pixel centrale) e consideriamo i pixel che ricadono all’interno delle celle della finestra. Un approccio comune consiste nel suddividere la finestra in due aree, una interna ed una esterna, come in Fig. 4.3. I valori presenti in tali zone vengono utilizzati nel passo successivo per computare il contrasto.
3. **Calcolo del contrasto locale:** Per ogni posizione della finestra, si calcola il contrasto locale utilizzando una qualche formula. Nel tempo sono state presentate una moltitudine di metodi per computare il contrasto [3][25]. La maggior parte delle funzioni cercano di ottenere una stima il più accurata possibile. Ne consegue che finiscono per utilizzare una moltitudine di operazioni matematiche e/o ulteriori informazioni non locali alla finestra. Un approccio di tale tipo significherebbe un maggiore uso di risorse computazionali il cui costo andrebbe a ridurre i benefici introdotti dal usare un numero minore di campioni per pixel. Nel nostro caso, però, non necessitiamo di una stima estremamente accurata del contrasto, potendoci permettere, quindi, una funzione più semplice. In particolare utilizzeremo la seguente formula, presentata in [15], per ottenere il contrasto  $C$ , dove  $L_{inner}$  è la luminanza media della parte interna della finestra mentre  $L_{outer}$  è la luminanza media della parte esterna (Fig. 4.3).

$$C = \frac{|L_{inner} - L_{outer}|}{L_{inner} + L_{outer}} \quad (4.1)$$

4. **Assegnazione del valore di contrasto:** Infine, si assegna il valore di contrasto calcolato al pixel su cui è centrata la finestra. Il processo viene poi ripetuto per ogni posizione della finestra fino a coprire tutta l’immagine.

Uno degli aspetti fondamentali del nostro meccanismo consiste nello stabilire una metrica di contrasto locale in maniera computazionalmente efficiente. Come abbiamo già fatto notare, non vogliamo che ciò che guadagniamo usando meno campioni venga annullato da un sistema complesso per computare il contrasto. Fino a qui abbiamo stabilito che faremo uso di una finestra composta da due sezioni, una interna ed una esterna. Non abbiamo, però, ancora chiarito come campionare l’immagine in maniera efficiente per ottenere i valori  $L_{inner}$  e  $L_{outer}$ . Un possibile approccio consisterebbe nel utilizzare le mip-map dell’immagine contenente l’irradianza. Questo ci permetterebbe di determinare per un pixel la finestra che lo circonda, facendo uso dei texel delle mip-map più piccole come finestre. Infatti, più scendiamo nei livelli della mip-map, più i texel di tali livelli ricoprono un’area più grande dell’immagine originale. Inoltre, visto che per generare i vari livelli viene utilizzato un filtro bilineare, otterremmo direttamente i valori  $L_{inner}$  e  $L_{outer}$ , invece che dover prendere manualmente i singoli campioni secondo la finestra e poi fare la media. Il problema di questo approccio è la necessità di dover rigenerare le mip-map ad ogni frame, il che ha un costo non indifferente. Costo che, inoltre, cresce all’aumentare della risoluzione del render target principale.

Per questo motivo abbiamo virato su un approccio che non usa propriamente la finestra quadrata precedentemente definita (Fig. 4.3), ma una sua approssimazione circolare. Infatti, per computare il contrasto locale, andremo a riutilizzare i campioni che usiamo per il passo base nel calcolo del contributo

di subsurface scattering (Fig. 3.9b). Per determinare quali appartengano alla zona interna e quali alla zona esterna, li separiamo in base alla loro distanza dal centro. In [15] viene suggerito di mantenere un rapporto di 1:3 fra l'area interna e quella esterna della finestra. Utilizzando quindi questo rapporto e conoscendo la larghezza massima della finestra (determinata dal campione più distante), separiamo i campioni nelle rispettive sezioni. Ciò che facciamo, quindi, durante il campionamento nel passo base per il contributo del subsurface scattering, è accumulare i campioni in due variabili a seconda della loro zona di appartenenza. Alla fine del passo base dividiamo i valori accumulati per il numero di campioni utilizzati per ciascuna variabile, ottenendo così le medie, ossia i valori  $L_{inner}$  e  $L_{outer}$ . A questo punto possiamo applicare la formula 4.1 ottenendo così la misura del contrasto locale. Essendo il nostro meccanismo basato su 2 livelli e non su un intervallo continuo, è necessario mappare l'intervallo continuo del contrasto  $[0, 1]$  ai due singoli valori 0 e 1, i quali rappresentano la necessità a meno di attivare il passo di ulteriore campionamento, che viene determinato dal parametro di threshold. Tale conversione può essere effettuata con un paio di semplici operazioni nello shader (Eq. 4.2), dove  $threshold_{offset}$  è l'offset del threshold rispetto al valore 0.5.

$$C' = \text{round}(\text{saturate}(C + threshold_{offset})) \quad (4.2)$$

A questo punto, considerando che lavoriamo con campioni RGB, abbiamo ottenuto il valore di contrasto mappato  $C'$  per ciascuno dei canali. È sufficiente quindi verificare che, per almeno uno dei canali, il valore di  $C'$  sia uguale ad 1 e nel caso attivare l'ulteriore campionamento per il pixel corrente. Da qui, l'algoritmo procede come farebbe normalmente nella sua versione non estesa con il meccanismo proposto, quindi campionando e calcolando il contributo finale del subsurface scattering. Una panoramica generale dell'algoritmo completo viene presentata nell'Algoritmo 1, dove:

- $sssState$  è la struttura dati che contiene tutti i valori computati sia dalla funzione *baseStepSSS* che dalla funzione *extraStepSSS*.
- *baseStepSSS* è la funzione che si occupa di campionare la scena e calcolare il contributo del SSS. Dipende dall'algoritmo che abbiamo deciso di estendere. In questo caso, inoltre, accumula nelle due variabili *sssState.innerL* e *sssState.outerL* i campioni appartenenti alle due diverse zone della finestra. La funzione *extraStepSSS* fa la stessa cosa, eccetto per il fatto che non accumula le variabili per il contrasto, in quanto a questo punto non sono più necessarie.
- *numSamples* e *numSamplesExtra* definiscono rispettivamente il numero di campioni da usare per il passo base ed il numero da usare per l'eventuale passo extra.
- *sssParams* contiene tutti i parametri necessari per calcolare il contributo del SSS. Anche in questo caso dipende dall'algoritmo che abbiamo deciso di estendere.
- *computeSSS* è una funzione che utilizzando i valori accumulati nei passi precedenti calcola il contributo finale del subsurface scattering sulla superficie.
- *sssResult* è il valore finale del contributo dato dal fenomeno di subsurface scattering restituito dalla funzione *computeSSS*.

**Algoritmo 1** Implementazione SSS con meccanismo contrasto

---

```

1: sssState  $\leftarrow$  baseStepSSS(numSamples, sssParams)
2: C  $\leftarrow$   $|sssState.innerL - sssState.outerL|/(sssState.innerL + sssState.outerL)$ 
3: C'  $\leftarrow$  round(saturate(C + threshold_offset))
4: if any(C') then
5:   sssState  $\leftarrow$  extraStepSSS(numExtraSamples, sssParams, sssState)
6: end if
7: sssResult  $\leftarrow$  computeSSS(sssState)

```

---

Il metodo presenta, inoltre, un隐式的 meccanismo di Level-Of-Detail (LOD), in quanto il contrasto è computato a partire dai campioni del primo passo. All'aumentare della distanza della camera dalla superficie, l'area che verrà campionata attorno ad un certo pixel diventerà sempre più piccola. Ciò porta ad avere sempre più campioni che ricadono sugli stessi pixel, con la conseguente riduzione della varianza. Da una varianza minore consegue anche una diminuzione del contrasto, che implica il fatto che meno pixel supereranno il threshold impostato, non attivando quindi il campionamento extra.

### 4.3 Determinazione Valori Ottimali dei Parametri

L'estensione che è stata presentata ha introdotto due ulteriori parametri, ovvero il threshold ed il numero di campioni extra. Abbiamo quindi fatto una serie di esperimenti per stabilire quali siano dei valori ottimali per tali parametri, o almeno un buon punto di partenza per poi adattarli alle proprie esigenze di bilanciamento fra performance e qualità. Oltre ai nuovi parametri, verrà modificato anche il numero di campioni del passo base, in quanto, come abbiamo già spiegato, vogliamo campionare ad un tasso minore dove è meno necessario ed utilizzare i campioni extra solo nel caso in cui sia indispensabile.

Partiamo, quindi, andando a determinare un numero adatto di campioni base. Vogliamo che questo valore sia il più basso possibile, mantenendo, però, una buona qualità nelle zone in cui il contrasto è basso, ossia in quelle per cui ci aspettiamo che il subsurface scattering sia meno percepibile. Per stabilire ciò abbiamo fatto uso di una metrica chiamata DSSIM (Structural Dissimilarity Index), ovvero una misura utilizzata per quantificare la differenza tra due immagini. Si basa sulla metrica SSIM (Structural Similarity Index), che, al contrario, valuta la somiglianza strutturale tra due immagini [31]. Tale metrica non si concentra sulle differenze (o somiglianze se si parla di SSIM) fra i singoli pixel, ma sulle differenze in termini di strutture visive. Ossia, analizza le relazioni spaziali fra pixel per determinare come le componenti strutturali di un'immagine, quali forme, contorni e trame, influiscano sulla percezione umana. Inoltre, abbiamo deciso di utilizzare suddetta misura in quanto progettata per essere più allineata con la percezione umana rispetto a metriche come MSE (Mean Squared Error) o PSNR (Peak Signal-to-Noise Ratio), che possono non riflettere accuratamente come gli esseri umani percepiscono la qualità delle immagini. Essendo DSSIM definita in termini di SSIM, la formula che la descrive è la seguente

$$DSSIM(x, y) = \frac{1 - SSIM(x, y)}{2} \quad (4.3)$$

dove

$$SSIM(x, y) = \frac{(2\mu_x\mu_y + c_1)(2\sigma_{xy} + c_2)}{(\mu_x^2 + \mu_y^2 + c_1)(\sigma_x^2 + \sigma_y^2 + c_2)} \quad (4.4)$$

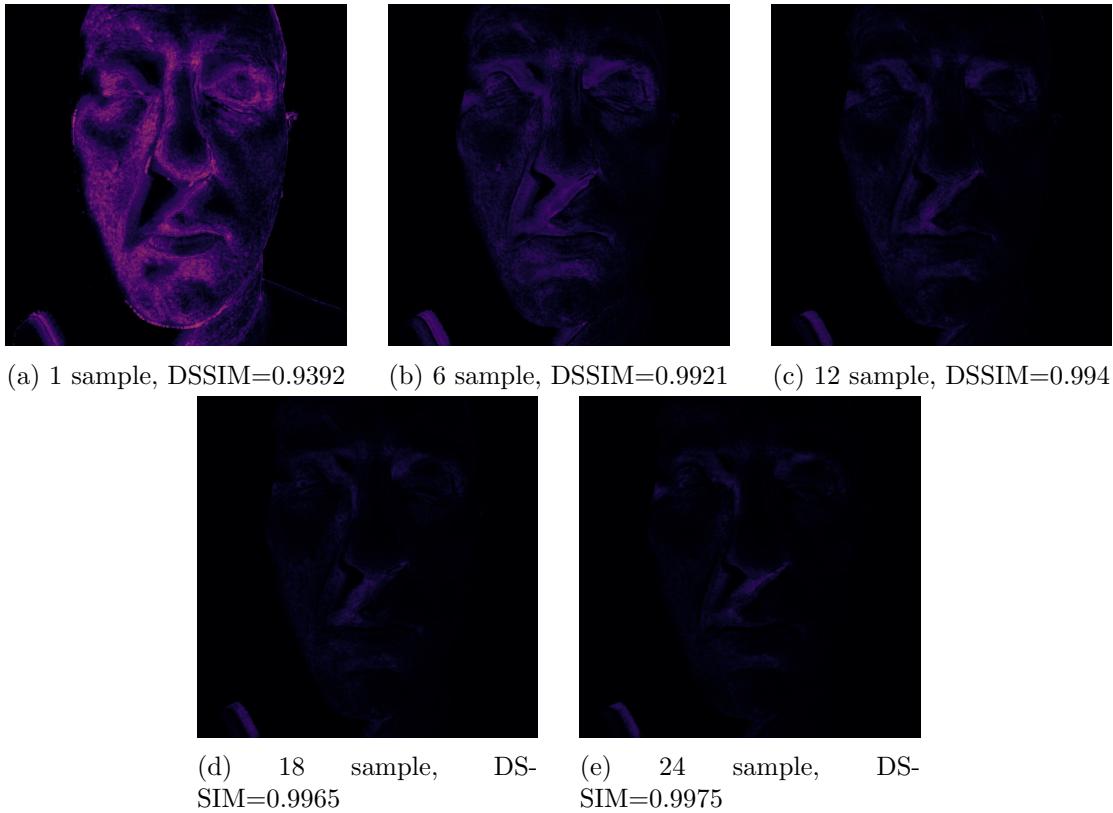


Figura 4.4: Immagini delle differenze dei render ottenuti con la versione standard del metodo di Golubev utilizzando diversi numeri di campioni, rispetto all’immagine di riferimento che di campioni ne usa 30. Essendo i valori di DSSIM in scala di grigi, per la visualizzazione abbiamo optato per una colormap percettivamente lineare. I valori così alti di DSSIM sono un pò ingannevoli, in quanto influenzati dal fatto che il background è perfettamente identico per tutte le immagini. Si nota comunque come i guadagni dettati da un numero maggiore di campioni tendano a diminuire.

è il valore della metrica di similarità. Sia in Eq. 4.4 che in Eq. 4.3  $x$  e  $y$  rappresentano due finestre della stessa dimensione NxN. Questo perché, ricordiamo, SSIM non lavora su singole coppie di pixel, ma su finestre. Con  $\mu$  viene indicata la media dei campioni delle rispettive finestre ed equivalentemente con  $\sigma^2$  le rispettive varianze. Invece,  $c_1$  e  $c_2$  sono dei parametri utilizzati per stabilizzare la divisione. Per computare le differenze dobbiamo per prima cosa avere un’immagine di riferimento. Quest’ultima la otteniamo utilizzando la versione standard del metodo di Golubev, impostando il numero di campioni a 30. Abbiamo deciso per tale valore in quanto dovrebbe essere sufficiente per ottenere una buona qualità alla distanza presa in considerazione, considerando che Golubev suggerisce di utilizzare 22 campioni per produrre buoni risultati [13]. I render che metteremo a confronto con quello di riferimento utilizzano rispettivamente 1, 6, 12, 18 e 24 campioni (Fig. 4.5). Mediante, quindi, un semplice programma scritto in python che utilizza l’equazione Eq. 4.3 otteniamo le immagini delle differenze, consultabili in Fig. 4.4. Considerando che DSSIM restituisce un valore appartenente ad una scala di grigi, per rendere più intuitiva e rapida l’interpretazione dei risultati abbiamo optato per un mapping mediante una colormap avente la caratteristica di essere percettivamente lineare. Tale proprietà fa riferimento al fatto che la transizione tra i colori nella colormap avviene in modo uniforme e costante relativamente alla percezione umana. In questo caso non ci interessa il valore globale della metrica per l’intera immagine, ma piuttosto quelli in specifiche parti dell’immagine. Se guardiamo i valori per le zone in

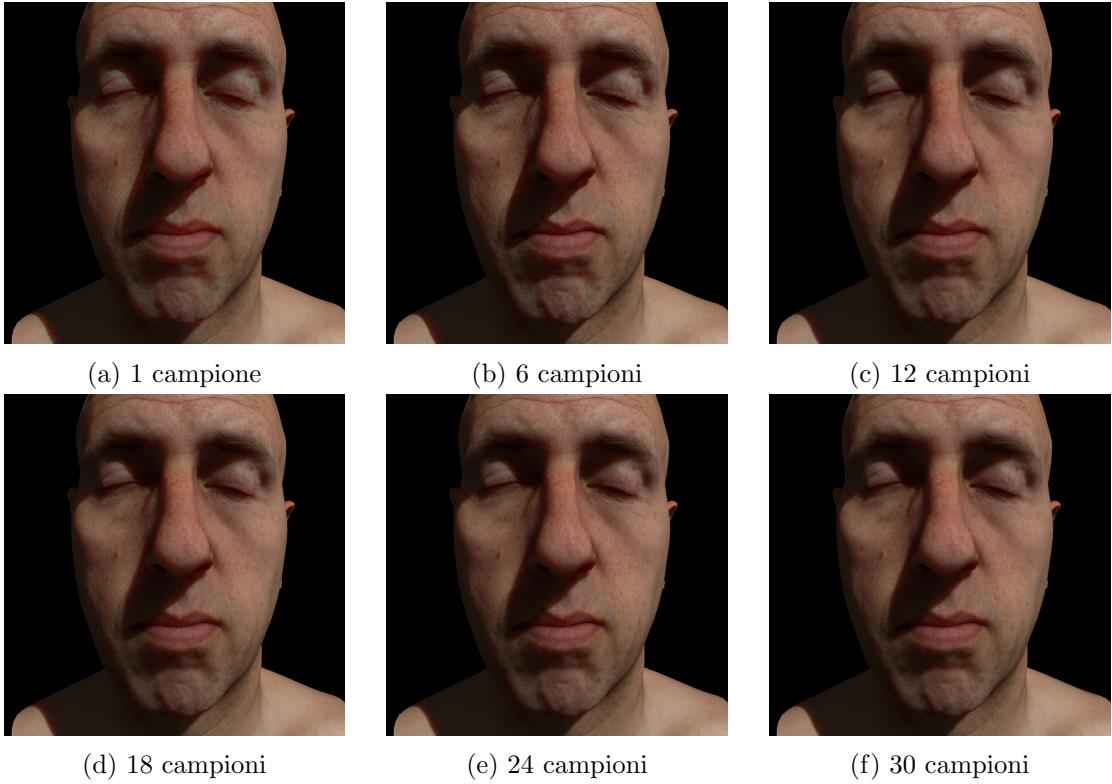


Figura 4.5: Immagini dei render ottenuti utilizzando la versione standard del metodo di Golubev con diversi numeri di campioni. È evidente il rumore nelle immagini con pochi campioni dedicati al subsurface scattering.

cui le superfici sono illuminate da una direzione approssimativamente perpendicolare (come la guancia nella parte destra dell'immagine) è evidente che ci siano rendimenti decrescenti (Fig. 4.4). Infatti, già dopo 12 campioni i guadagni per quanto riguarda le differenze percepite sono minimi. Ne concludiamo, quindi, che 12 è un buon numero di campioni da utilizzare per il passo base, potenzialmente scendendo anche un pò sotto tale numero. Da evitare, però, l'utilizzo di valori troppo bassi, in quanto introducono rumore nell'immagine, che rimarrà tale nelle zone in cui non sarà attivato il campionamento extra, ovvero ovunque ci sia un contrasto basso.

In merito al parametro chiamato threshold, abbiamo optato per stabilirne il valore in maniera empirica. Considerando quindi di utilizzare 12 campioni per il passo base abbiamo trovato come il threshold pari 0.25 sia un buon compromesso fra qualità e costo computazionale. Infatti, valori più alti tendevano ad attivare molti più pixel del necessario per il passo extra, mentre valori più bassi propendevano a sottostimare le zone che richiedevano ulteriori campioni per il subsurface scattering. Il contrasto locale calcolato con tale configurazione lo possiamo vedere in Fig. 4.6a. È promettente anche il fatto che se sottraiamo tale contrasto dall'immagine della differenza relativa al passo base con 12 campioni (Fig. 4.4c) otteniamo l'immagine in Fig. 4.6b. Ciò che possiamo evincere da quest'ultima figura è che il contrasto ha attivato una grande parte dei pixel appartenenti alle “zone calde”, ossia quelle con l'errore maggiore. La verifica di quanto ciò sia efficace a livello pratico, per quanto riguarda la qualità, la faremo nel capitolo 5.

Infine, per quanto concerne il numero di campioni da utilizzare per il passo extra, dato che abbiamo stabilito che 12 è un buon numero di campioni per il passo base e che Golubev, per la sua tecnica,

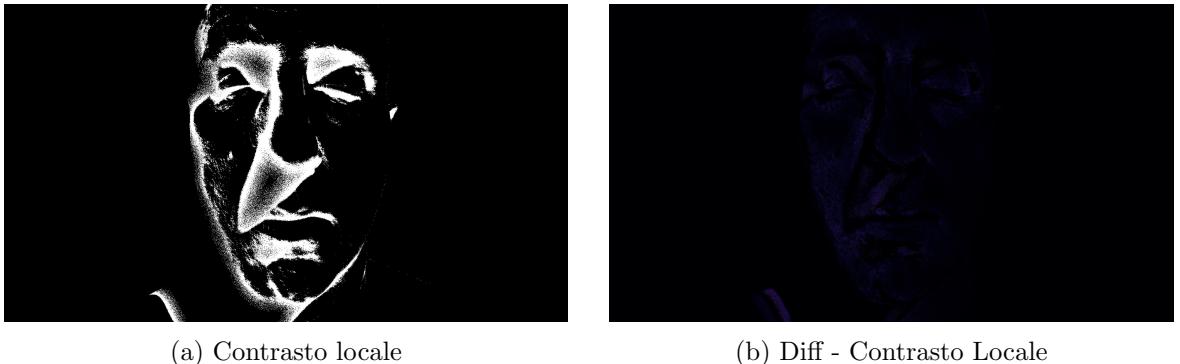


Figura 4.6: L’immagine (a) rappresenta i valori di contrasto locale ottenuti utilizzando 12 campioni durante il passo base ed un valore di threshold pari a 0.25. L’immagine (b) rappresenta, invece, ciò che si ottiene sottraendo il contrasto locale dall’immagine della differenza relativa all’uso di 12 campioni. Se confrontiamo il risultato ottenuto con l’immagine originale, si può notare come ci siano molte meno zone calde, ossia con errore alto.

suggeriva di utilizzare 22 campioni totali per ottenere un buon risultato [13], abbiamo optato per 10 ulteriori campioni. Ciò ci permette di raggiungere i 22 campioni dove necessario ed allo stesso tempo risparmiarne 10 dove non è altrettanto utile averne un numero così elevato. Ovviamente, per questo parametro si possono utilizzare valori ancora più alti, in particolare nel caso si voglia sfruttare il budget computazionale risparmiato dal uso di pochi campioni per le parti meno importanti e spenderlo, quindi, per usufruire di ancora più campioni dove maggiormente necessario.

## 4.4 Analisi performance

In questa sezione, ci concentreremo sull’analisi delle performance dell’estensione della tecnica di subsurface scattering (SSS) introdotta in questa tesi, ossia Golubev Plus, confrontandola direttamente con la sua versione standard. In particolare, non valuteremo la performance in termini di tempo di rendering (misurato in millisecondi), ma piuttosto nel numero di campioni per pixel utilizzati per computare il contributo del subsurface scattering. Questo perché non posso garantire di aver effettuato un ottimizzazione equivalente per entrambe le tecniche implementate, il che potrebbe produrre conclusioni errate. Per questo motivo utilizziamo il numero di campioni come misura, in quanto è indipendente dalla qualità dell’ottimizzazione. Inoltre, essendo il campionamento una delle operazioni più costose, dovendo spesso anche accedere alla memoria della GPU nel momento in cui i valori non si trovano nella cache, tale metrica risulta essere una buona indicazione delle performance generali dell’algoritmo.

La valutazione è stata condotta attraverso una serie di rendering relativi ad una scene standard di riferimento. Per tali test, abbiamo deciso di utilizzare tutte e tre le inquadrature che abbiamo visto in precedenza e, per ciascuna di esse, tre diverse angolazioni (Fig. 4.7). I parametri utilizzati per le due tecniche sono gli stessi di quelli utilizzati per i render presenti nel questionario, ossia Golubev con 22 campioni e Golubev Plus con 12 campioni base e 10 campioni extra. Per ciascuna tecnica, abbiamo quindi misurato il numero di campioni totali utilizzati per una certa inquadratura, facendo poi la media per ottenere un valore generale approssimativo della scena (Tab. 4.1). Possiamo osservare come, usando l’estensione proposta in questa tesi, ci sia un guadagno approssimativo, per quanto riguarda il numero

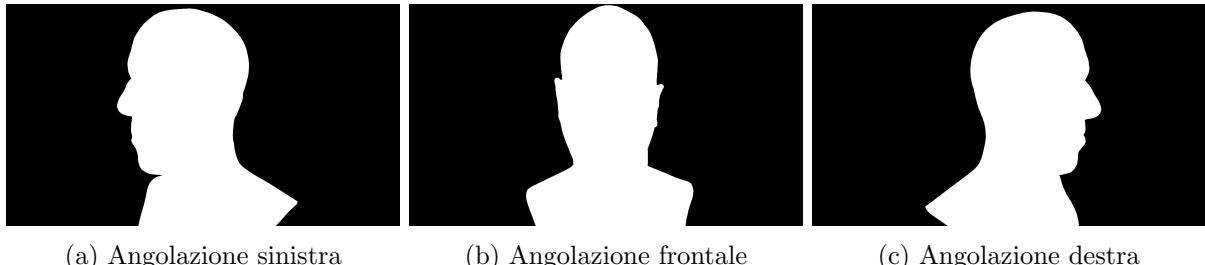


Figura 4.7: Le diverse angolazioni di una delle inquadrature utilizzate per la valutazione delle performance.

	Primissimo Piano	Primo Piano	Mezzo Primo Piano
Golubev	32 838 857	12 272 582	4 972 325
Golubev Plus	19 924 226	7 161 584	2 831 388
Riduzione Campioni	40%	42%	44%

Tabella 4.1: Numero di campioni in media utilizzati in una scena con una certa inquadratura. Nell'ultima riga viene indicato, in percentuale, quanti campioni in meno utilizza Golubev Plus rispetto a Golubev.

di campioni, che si aggira attorno al 40-45%. Questo è un buon risultato, considerando il fatto che, come vedremo nel capitolo 5.2, Golubev Plus produce render percettivamente equiparabili a Golubev.

Importante far notare, però, che il beneficio dettato dall'utilizzare un numero minore di campioni non si traduce in maniera lineare ad un guadagno, per quanto riguarda i tempi di rendering, a tutte le inquadrature possibili. Infatti, avvicinando la telecamera alla superficie che esibisce il fenomeno di subsurface scattering, una porzione di esse andrà a ricoprire una percentuale sempre maggiore dello schermo. Ciò comporta che i pixel vicini ad un certo momento saranno sempre più lontani avvicinandosi con l'inquadratura. Di conseguenza, durante il campionamento, saranno sempre più i campioni che dovranno accedere alla memoria principale della GPU, in quanto i texel necessari non saranno presenti nella cache.

# 5

## Analisi Percettiva Mediante Questionario

In questo capitolo, verrà eseguita un'analisi delle differenze percettive tra le varie tecniche presentate nei capitoli precedenti per renderizzare il subsurface scattering. Per valutare l'efficacia e l'impatto visivo di ciascuna tecnica, è stato sviluppato e somministrato un questionario ad un campione casuale di utenti. I risultati ottenuti permetteranno di comprendere meglio come le diverse metodologie vengano percepite in termini di realismo da parte dei partecipanti. Inoltre, di particolare interesse sarà vedere come si comporta la tecnica di subsurface scattering estesa con il meccanismo di campionamento dinamico basato su contrasto proposta in questa tesi (Cap. 4). Di seguito andremo a vedere come è stato costruito il questionario e successivamente analizzeremo i risultati ottenuti cercando di estrarre informazioni significative.

### 5.1 Struttura Questionario

Per la realizzazione del questionario utilizzato nella presente tesi, è stata impiegata la piattaforma online “Tally” che consente di creare e distribuire questionari in maniera semplice. La scelta è ricaduta su questo servizio in quanto non pone particolari limiti nella versione gratuita, quali il numero di domande o numero di invii al mese. Inoltre, è l'unico che permettesse di creare delle pagine in cui le immagini fossero sufficientemente grandi (ossia che ricoprissero la maggior parte dello spazio verticale della finestra), in quanto negli altri casi erano talmente piccole da risultare indistinguibili. Infatti, con la grande maggioranza degli altri servizi presi in considerazione, le immagini finivano per essere racchiuse in piccoli riquadri. Anche provando ad utilizzare lo zoom del browser per ingrandire la pagina i risultati non erano migliori, poiché, una volta raggiunta la dimensione adatta per l'immagine, risultava complesso navigare, siccome i vari bottoni dell'interfaccia finivano per sovrapporsi.

Bisogna far notare che non è stato possibile far svolgere il questionario in presenza, ossia in un ambiente controllato e sempre sullo stesso schermo, in quanto non sarebbe stato possibile raccogliere un numero sufficiente di feedback. Ciò implica che gli utenti abbiano utilizzato schermi con calibrazioni e dimensioni diverse, all'interno di ambienti con svariate condizioni di illuminazione. La varianza di tali variabili influenza inevitabilmente la percezione dei partecipanti nei confronti delle immagini messe a confronto.

All'inizio del questionario viene richiesto all'utente di fare un'autovalutazione riguardo al suo livello di familiarità con la computer grafica indicando il livello su una scala da "nessuna familiarità" ad "elevata familiarità". Per "livello di familiarità con la computer grafica" si intende un qualsiasi tipo di esposizione o esperienza personale con la computer grafica (CG), come film d'animazione, e non, in 3D, videogiochi con grafica 3D, serie TV o pubblicità che fanno uso di CG, interazione con modelli tridimensionali o qualsiasi altra esperienza legata alla grafica 3D. Questa informazione risulta utile successivamente per valutare se c'è stata una buona distribuzione dei vari livelli di familiarità, in quanto il questionario è stato distribuito casualmente, indipendentemente dal tipo di potenziale utente.

La parte principale del questionario è composto da 30 domande, dove per ciascuna di esse viene chiesto di indicare quale immagine della coppia presentata venga percepita come più realistica. La preferenza viene indicata su una scala a 5 valori (Fig. 5.1) dove il valore centrale, ossia lo 0 (zero), rappresenta nessuna preferenza. Per quest'ultima si intende che l'utente non percepisce alcuna differenza fra le due immagini. Andando a sinistra, invece, si indica una maggiore preferenza per l'immagine A, mentre andando a destra si indica una maggiore preferenza per l'immagine B. Le domande sono suddivise in 3 categorie in base alla distanza dal soggetto:

1. **Primissimo Piano:** Inquadra solo una parte del viso o un dettaglio molto piccolo, come gli occhi, le labbra o simili (Fig. 5.2a).
2. **Primo Piano:** Inquadra il viso del soggetto dalle spalle in su (Fig. 5.2b).
3. **Mezzo Primo Piano:** Inquadra il soggetto dalla metà del busto in su (Fig. 5.2c).

Questa separazione è importante, in quanto la rilevanza del subsurface scattering dipende anche dalla distanza del soggetto che esibisce tale fenomeno. Per ciascuna di queste categorie ci sono 5 immagini diverse rappresentanti le implementazioni del SSS prese in considerazione:

1. **No SSS:** in questo caso è disabilitata qualsiasi forma di subsurface scattering, inclusa la trasmissione attraverso superfici sottili.
2. **Jimenez Gaussian:** la tecnica descritta nella sezione 3.2. Da qui in avanti, in alcune circostanze, verrà usata l'abbreviazione "Gauss" per indicare questo metodo.
3. **Jimenez Separable:** la tecnica descritta nella sezione 3.3. Da qui in avanti, in alcune circostanze, verrà usata l'abbreviazione "Sep" per indicare questo metodo.
4. **Golubev:** la tecnica descritta nella sezione 3.4. In questa implementazione vengono usati 22 campioni per computare il contributo del subsurface scattering. Da qui in avanti, in alcune circostanze, verrà usata l'abbreviazione "Gol" per indicare questo metodo.
5. **Golubev Plus:** l'estensione della tecnica di Golubev con il meccanismo di campionamento dinamico basato sul contrasto presentato nel capitolo 4. In questa implementazione vengono usati 12 campioni per il passo base e 10 campioni per l'eventuale passo extra. Da qui in avanti, in alcune circostanze, verrà usata l'abbreviazione "Gol+" per indicare questo metodo.

Per ogni categoria ciascuna tecnica viene confrontata con tutte le altre appartenenti alla stessa categoria. Precisiamo, quindi, che tecniche appartenenti a categorie diverse, ovvero distanze dal soggetto diverse,



Figura 5.1: Scala utilizzata all'interno del questionario che permette all'utente di indicare la sua preferenza fra due immagini A e B.

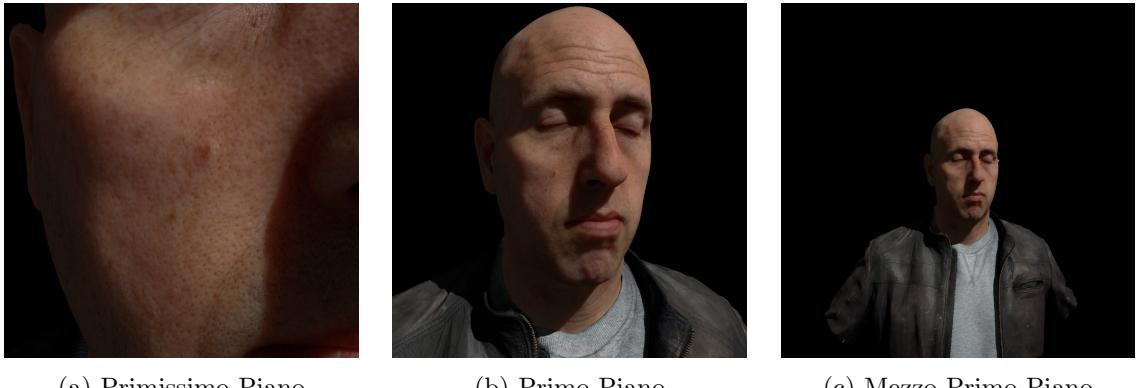


Figura 5.2: Le categorie definite per suddividere le immagini utilizzate all'interno del questionario in base all'inquadratura, o più precisamente, in base alla distanza dal soggetto.

non possono essere confrontate fra loro. Inoltre, le domande sui confronti hanno una distribuzione casuale. Non è presente alcun limite di tempo entro cui un utente deve completare il questionario. Il partecipante è libero di impiegare tutto il tempo che ritiene necessario per osservare le immagini a confronto e prendere una decisione.

## 5.2 Analisi Risultati

Nell'ambito di questo esperimento, sono state raccolte le risposte ai questionari da un totale di 20 partecipanti. Nonostante sia stato distribuito casualmente, la distribuzione dei livelli di familiarità con la computer grafica da parte dei partecipanti è stata buona, come possiamo vedere in Fig. 5.4. Eccezione unica per il livello “Elevata Familiarità” in cui abbiamo un unico utente.

L'analisi dei risultati è stata fatta separatamente per ogni categoria, dove alle varie tecniche è stato assegnato un punteggio in base alle risposte dei partecipanti. Prendendo in esempio una domanda in cui vengono confrontate due tecniche A e B, abbiamo che:

- se la risposta è 0, ovvero l'utente non esibisce una preferenza per un'immagine rispetto all'altra, non viene assegnato un **punto a nessuna** delle due tecniche.
- se la risposta è  $-1$ , ovvero l'utente presenta una leggera preferenza per l'immagine A, viene assegnato **un punto** alla tecnica A. Equivalentemente, se la risposta è 1, viene assegnato un punto alla tecnica B.

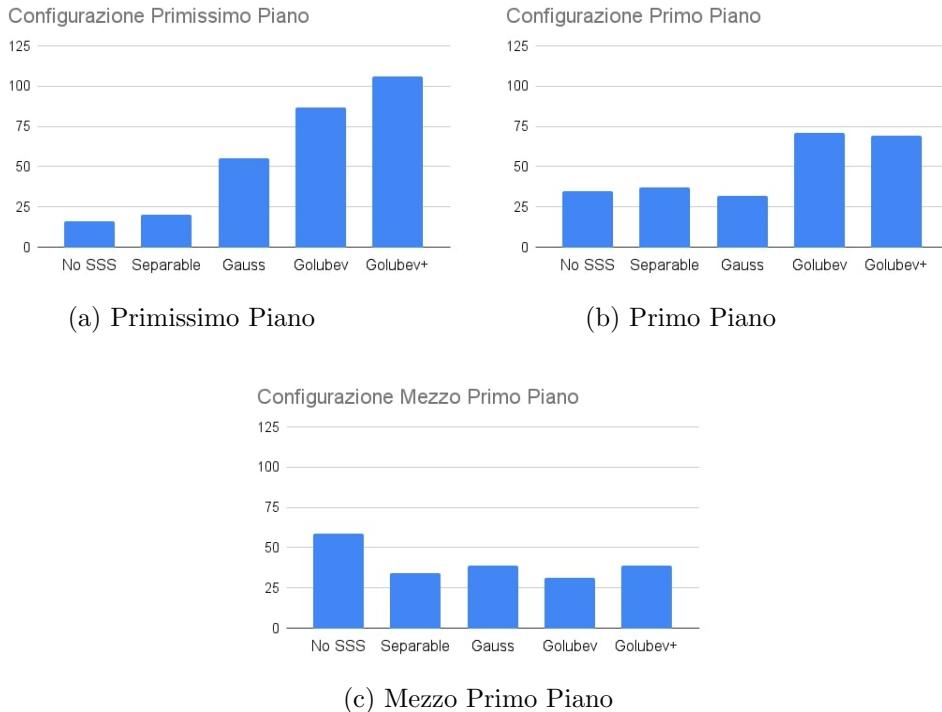


Figura 5.3: Distribuzione dei punteggi totali delle varie tecniche valutate per ciascuna categoria, indicante la distanza dal soggetto. Da notare il progressivo appiattimento della distribuzione all'aumentare della distanza.

- se la risposta è  $-2$ , ovvero l'utente presenta una leggera preferenza per l'immagine A, vengono assegnati **due punti** alla tecnica A. Equivalentemente, se la risposta è  $2$ , vengono assegnati due punti alla tecnica B.

A partire da questa regola possiamo accumulare i punteggi per le varie tecniche, ottenendo così le tabelle in Fig. 5.3. Un punteggio più alto significa che gli utenti hanno, in genere, indicato una preferenza maggiore per tale tecnica rispetto alle altre. Osservando i tre grafici sembra che ci sia una diminuzione del punteggio totale all'aumentare della distanza dal soggetto, il che sarebbe in linea con quello che ci aspettiamo. Infatti, a distanze maggiori da una superficie che esibisce un fenomeno di subsurface scattering, l'impatto di tale fenomeno diminuisce fino a ricadere completamente in un unico pixel. Difatti, in questa ultima condizione, il subsurface scattering può essere approssimato da una semplice BRDF, il che spiega perché dopo una determinata distanza il subsurface scattering possa essere disattivato. Se l'osservazione fatta prima fosse vera, significherebbe che i partecipanti esprimevano preferenze meno forti all'aumentare della distanza. Andando a plottare i punteggi totali per categoria su un grafico a linee otteniamo Fig. 5.5. È possibile vedere quindi come il punteggio diminuisca secondo un andamento lineare all'aumentare della distanza, confermando l'osservazione fatta in precedenza a partire dai grafici delle singole categorie (Fig. 5.3).

Fino questo punto abbiamo fatto un analisi generale a livello di categorie, considerandole nella loro interezza. Ora, invece, andremo ad indagare sui risultati delle specifiche tecniche di subsurface scattering, valutandole per singola inquadratura.

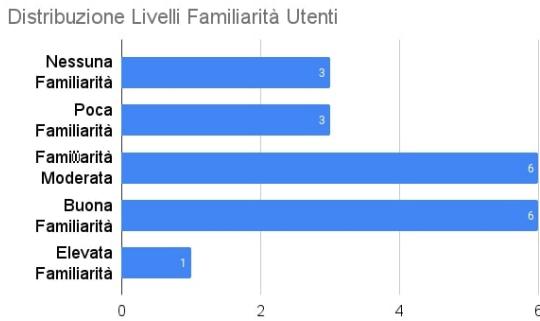


Figura 5.4: Distribuzione dei livelli di familiarità con la computer grafica da parte degli utenti che hanno partecipato al questionario.



Figura 5.5: Andamento del punteggio totale all'aumentare della distanza. Si noti come il punteggio diminuisca linearmente all'aumentare della distanza.

### 5.2.1 Primissimo Piano

Per questo raggruppamento sono state usate le immagini in Fig. 5.8. Partendo, quindi, dal primissimo piano, osservando il grafico in Fig. 5.3a possiamo notare come ci sia un chiaro ordinamento basato sulle preferenze indicate dai partecipanti. Non sorprende che, a questa distanza, il rendering che non utilizza alcuna forma di subsurface scattering si trovi in ultima posizione. Anche l'ordinamento delle restanti tecniche rispecchia più o meno quello che ci si poteva aspettare. Può trarre in inganno il fatto che in Fig. 5.3a “Separable” sia così vicino come punteggio a “No SSS”. Se, però, andiamo a confrontare direttamente la distribuzione delle preferenze degli utenti fra tali due metodi possiamo notare come ci sia una netta propensione verso “Separable” (Fig. 5.6a). La vicinanza nel punteggio generale deriva dal fatto che nel confronto con ciascuna delle altre tecniche, “No SSS” ha ricevuto delle preferenze da un paio di partecipanti (Fig. 5.6a). Lo stesso non vale per il metodo “Separable” che, nonostante il chiaro gradimento degli utenti rispetto a “No SSS”, non riceve neanche una singola votazione a favore quando confrontato con le altre tecniche (Fig. 5.6b).

Non è chiaro da cosa derivi la predilezione da parte di alcuni utenti per i render in cui è assente l'effetto di subsurface scattering. Si può ipotizzare siano dei valori anomali, anche considerando il fatto che a favore di “No SSS” non è presente neanche una singola “leggera preferenza” (che ricordiamo è indicata con -1, o 1 nel caso sia a favore dell'altra tecnica), ma esclusivamente multiple “preferenza forte”.

Un'altra cosa che stupisce è il fatto che la tecnica di Golubev estesa con il nostro meccanismo di campionamento dinamico basato sul contrasto (ossia “Golubev Plus”) sia favorita nel punteggio totale rispetto alla versione standard di Golubev (Fig. 5.3a). Anche andando ad approfondire la distribuzione delle preferenze nel confronto diretto fra i due metodi, è evidente una propensione non insignificante per “Golubev Plus” alla distanza presa in considerazione (Fig. 5.7). Preferenza che, però, sembra manifestarsi solo nella categoria “Primissimo Piano”, andando poi a riequilibrarsi all'aumentare della distanza, quindi sia in “Primo Piano” che in “Mezzo Primo Piano” (Fig. 5.7b). Questo fenomeno è sorprendente in quanto, a livello teorico, nel migliore dei casi “Golubev Plus” usa lo stesso numero di campioni di “Golubev”, producendo nella più favorevole delle circostanze (per quanto riguarda la qualità) lo stesso risultato ottenibile con “Golubev” stesso. Il motivo è che per quest'ultimo abbiamo utilizzato un numero

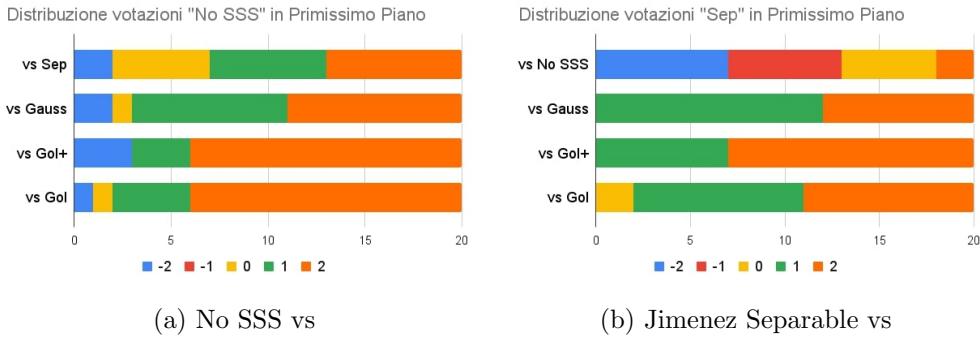


Figura 5.6: Distribuzione delle preferenze esibite dai partecipanti per (a) “No SSS” e per (b) “Separable”, rispetto a tutte le altre tecniche.

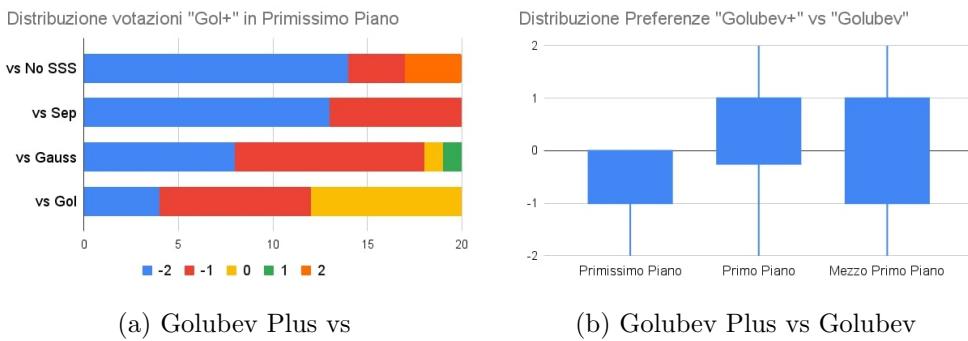


Figura 5.7: Nel grafico a barre è presentata la distribuzione delle preferenze esibite dai partecipanti per (a) “Golubev Plus”, rispetto a tutte le altre tecniche. Mentre nel box-plot è indicata la (b) distribuzione delle preferenze nel confronto fra “Golubev Plus” e “Golubev” alle varie distanze in considerazione.

di campioni pari a 22, mentre per “Golubev Plus” abbiamo impostato un passo base da 12 campioni ed un eventuale passo extra da 10 campioni. Ciò significa che “Golubev Plus” può usare al massimo 22 campioni, ossia lo stesso numero della versione standard senza campionamento dinamico. Per comprendere la motivazione di tale tendenza, se una motivazione è effettivamente presente, sarebbe necessario un ulteriore esperimento con utenti, in cui si confrontano esclusivamente le due tecniche in questione. Essendo tale studio ristretto ai soli due metodi citati sarebbe possibile diversificare ulteriormente le configurazioni da confrontare, richiedendo al partecipante, inoltre, di fornire una spiegazione per la sua selezione. In questo modo, alle scelte dei partecipanti saranno associate anche le motivazioni che hanno portato a tale decisione, permettendoci di trarre una qualche conclusione riguardante l’obiettivo posto dallo studio.

### 5.2.2 Primo Piano

Per la categoria corrente sono state utilizzate le immagini indicate in Fig. 5.10. Già a questa distanza si comincia a notare un appiattimento delle differenze percepite dagli utenti. Infatti, “No SSS”, “Separable” e “Gauss” finiscono per essere approssimativamente allo stesso livello. Nel contempo sia “Golubev” che “Golubev Plus” risultano decisamente le preferite, stando però a pressappoco lo stesso punteggio. Tale risultato è un buon segnale per quanto concerne alla bontà del meccanismo di campionamento dinamico proposto in questa tesi. Questo poiché, nonostante ci troviamo ad una distanza per cui sono ancora percepibili le differenze fra alcune tecniche, come evidenziato in Fig. 5.3b, nel confronto diretto

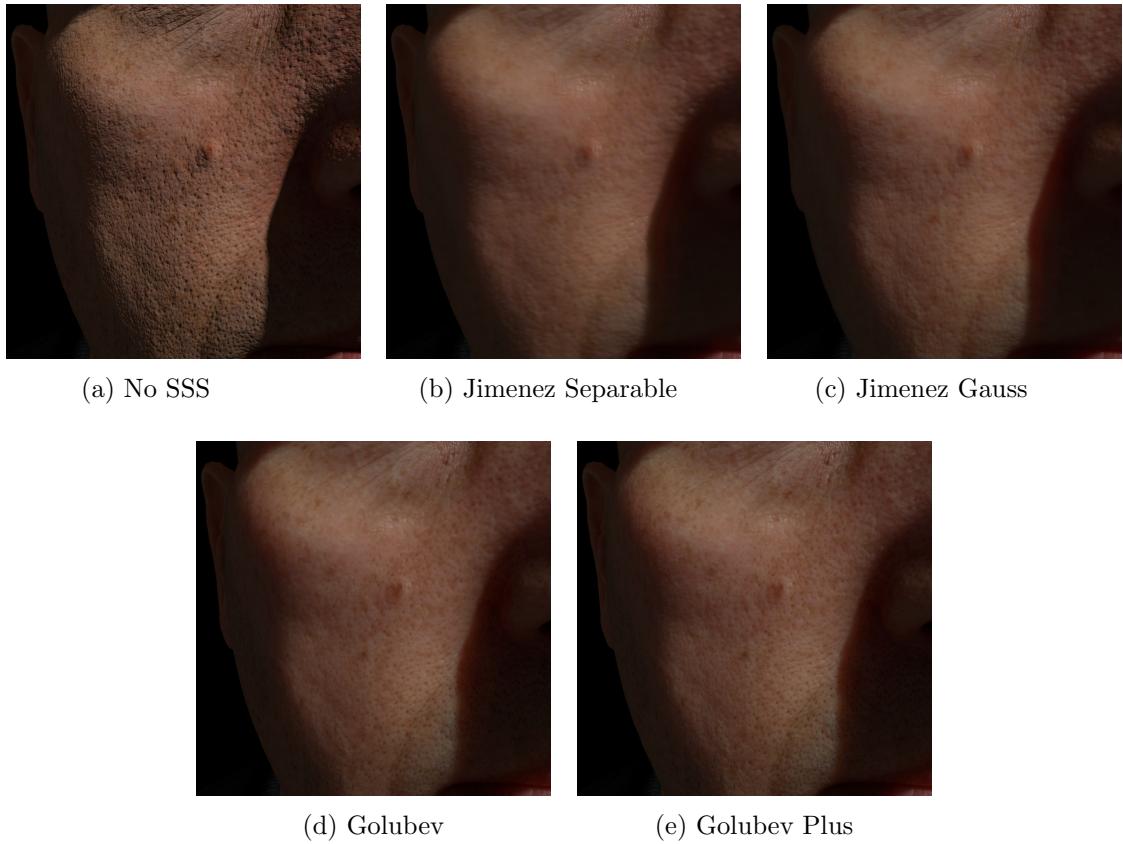


Figura 5.8: Immagini con le varie tecniche da confrontare utilizzate all'interno del questionario per quanto concerne la categoria "Primissimo Piano".

fra "Golubev Plus" e "Golubev" non emerge una chiara preferenza per una delle due tecniche (Fig. 5.7b). Era critico l'ottenere buoni risultati a questa distanza in quanto è la più vicina rispetto al soggetto, se consideriamo le inquadrature più comunemente utilizzate. Infatti, nonostante il "Primissimo Piano" sia importante per valutare la qualità delle tecniche di subsurface scattering, inquadrature di questo tipo non sono le più comuni, al contrario del "Primo Piano" il quale viene frequentemente adottato.

### 5.2.3 Mezzo Primo Piano

Per l'ultima categoria, ovvero il "Mezzo Primo Piano" sono state utilizzate le immagini indicate in Fig. 5.11. Questo raggruppamento ha la distanza dal soggetto maggiore e si può osservare una distribuzione

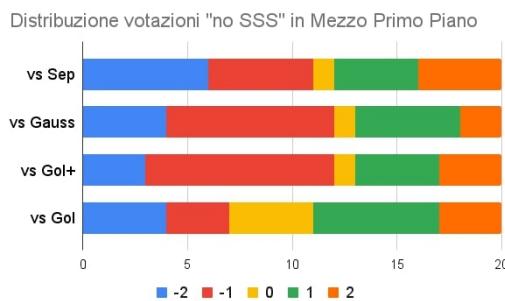


Figura 5.9: Distribuzione delle preferenze esibite dai partecipanti per il metodo senza subsurface scattering, ovvero "No SSS", rispetto a tutte le altre tecniche. I valori negativi indicano una preferenza per "No SSS", mentre quelli positivi per la tecnica della barra presa in considerazione.



Figura 5.10: Immagini con le varie tecniche da confrontare utilizzate all'interno del questionario per quanto concerne la categoria "Primo Piano".

dei punteggi relativamente uniforme fra tutte le tecniche (Fig. 5.3c), tranne che per "No SSS" che presenta una valutazione leggermente più alta. Non è esattamente chiaro da cosa derivi questa preferenza, in quanto il contributo del subsurface scattering a suddetta distanza non ricade ancora su un unico pixel ed è, quindi, ancora teoricamente percepibile. Inoltre, anche osservando la distribuzione delle preferenze da parte degli utenti rispetto alle restanti tecniche (Fig. 5.9) non ci sono particolari valori anomali che potrebbero spiegare questa propensione. Per speculazione si potrebbe ipotizzare che tale inclinazione derivi dal fatto che, per definizione, l'assenza di subsurface scattering preserva la massima quantità di dettagli forniti dalla normal map e dalla texture map. Tale fatto, alla distanza presa in considerazione, potrebbe essere percepito in qualche modo come più realistico. Non abbiamo però le informazioni necessarie per verificare tale affermazione; sarebbe utile un ulteriore studio specifico per smentire o convalidare l'ipotesi presentata.

I risultati della categoria corrente ci suggeriscono che, forse, a questa distanza potremmo ridurre di molto il numero di campioni usati per computare il subsurface scattering senza compromettere la percezione degli utenti. Questo poiché, come già detto, a questa distanza le preferenze espresse dai partecipanti sono risultate distribuite in maniera molto più uniforme (Fig. 5.3c). Sempre secondo gli stessi risultati, potremmo addirittura considerare di disabilitare completamente il subsurface scattering, nonostante la distanza sia tale per cui il contributo del SSS non è ancora circoscritto ad un unico pixel.



Figura 5.11: Immagini con le varie tecniche da confrontare utilizzate all'interno del questionario per quanto concerne la categoria "Mezzo Primo Piano".

### 5.3 Ranking tramite processo di gerarchia analitica

In questo capitolo, verranno valutate le varie tecniche di subsurface scattering in tempo reale che abbiamo presentato utilizzando il metodo Analytic Hierarchy Process (AHP) [28][29], sviluppato da Thomas L. Saaty negli anni '70. L'AHP è un metodo di decisione che permette di confrontare una serie di alternative in base ad una pluralità di criteri (e sottocriteri) e ricavare quindi una valutazione globale per le sopracitate alternative. Non applicheremo il processo nella sua interezza, in quanto utilizzeremo un solo criterio e non multipli come viene comunemente fatto. Infatti, il nostro obiettivo è utilizzare questo metodo per stabilire un ranking delle alternative in base alle preferenze che hanno esibito gli utenti durante il questionario. Tale ordinamento va ad affiancarsi a quello che abbiamo già ottenuto nella sezione 5.2, espandendo ulteriormente sulle conclusioni ottenute.

Come abbiamo già detto, utilizzeremo una versione parziale di AHP che può essere indicativamente suddivisa nei seguenti passi:

- Costruzione della matrice di confronto a coppie:** per ogni criterio di valutazione andrebbe realizzata la matrice di confronto. Come già detto, noi ne utilizzeremo solo uno, ossia il realismo del render percepito dall'utente. Ciascuna riga e ciascuna colonna identificano una delle tecniche di subsurface scattering che stiamo valutando. Il valore in una cella  $a_{ij}$  rappresenta il punteggio di preferenza dell'alternativa  $i$  rispetto all'alternativa  $j$ , indicato secondo una certa scala. Quella più comunemente utilizzata, in questi casi, è proprio quella di Saaty, che va da 1 a 9, dove il valore più piccolo indica che le due alternative sono equamente preferite, mentre il valore massimo indica

	No SSS	Separable	Golubev	Golubev+	Gauss
No SSS	1	0.5	0.3333	0.3333	0.5
Separable	2	1	0.5	0.3333	0.5
Golubev	3	2	1	0.5	3
Golubev+	3	3	2	1	2
Gauss	2	2	0.3333	0.5	1

(a) Primissimo Piano

	No SSS	Separable	Golubev	Golubev+	Gauss
No SSS	1	0.75	0.5	0.5	0.5
Separable	1.333	1	0.5	0.5	1
Golubev	2	2	1	1	2
Golubev+	2	2	1	1	2
Gauss	2	1	0.5	0.5	1

(b) Primo Piano

	No SSS	Separable	Golubev	Golubev+	Gauss
No SSS	1	2	1	2	2
Separable	0.5	1	1	1	0.75
Golubev	1	1	1	1	1
Golubev+	0.5	1	1	1	1
Gauss	0.5	1.333	1	1	1

(c) Mezzo Primo Piano

Tabella 5.1: Matrici di coerenza dei confronti fra le varie tecniche nelle tre diverse configurazioni. I singoli giudizi sono stati determinati prendendo la mediana fra le preferenze indicate dagli utenti.

una preferenza estremamente forte. Nel nostro caso, però, utilizzeremo un intervallo molto più piccolo, che va da 1 a 3, in modo da rispecchiare le opzioni di preferenza presenti nel questionario.

Un problema che ci si presenta è che  $a_{ij}$  è un valore singolo, mentre noi ne abbiamo una moltitudine, ciascuno proveniente da un utente, dovendo quindi trovare un metodo per ottenere un punteggio unico. L'approccio più semplice consiste nel determinare la mediana, andando a usare, però, il valore inverso del punteggio, ossia  $1/a_{jb}$  quando il giudizio di un utente  $b$  non va a favore di  $i$ , ma a favore di  $j$ . Facciamo questa distinzione, in quanto poi  $a_{ji}$ , che per definizione si ottiene con  $1/a_{ij}$ , avrà implicitamente preso in considerazione i giudizi a favore di  $j$  da parte di una serie di utenti. Abbiamo deciso di utilizzare la mediana, piuttosto che la media, in quanto quest'ultima è suscettibile alla presenza di valori estremi. Seguendo quindi ciò che abbiamo presentato fino ad ora, andiamo a costruire le matrici di confronto per ciascuna delle inquadrature che stiamo valutando (Tab. 5.1).

- Calcolo degli autovettori:** a partire dai risultati ottenuti nel passo precedente possiamo andare a determinare il ranking delle tecniche. Ciò viene fatto semplicemente determinando l'autovettore dell'autovalore maggiore della matrice in considerazione [28]. Seguendo quindi tale processo otteniamo gli autovettori per ciascuna delle inquadrature (Tab. 5.2), che una volta normalizzati, ci permettono di determinare un ordinamento delle alternative messe a confronto. Osservando i valori in Tab. 5.2 e mettendoli a confronto con quelli ottenuti in Fig. 5.3, possiamo notare una

	No SSS	Separable	Golubev	Golubev+	Gauss
Primissimo Piano	0.0841	0.1208	0.2766	0.3562	0.1623
Primo Piano	0.1181	0.1496	0.2836	0.2836	0.1652
Mezzo Primo Piano	0.2971	0.1592	0.1973	0.1680	0.1785

Tabella 5.2: Tabella in cui ciascuna riga corrisponde all’autovettore dell’autovalore maggiore della matrice corrispondente. Ciascuna di queste corrisponde ad una delle inquadrature.

n	1	2	3	4	5	6	7	8	9	10
Random Index (RI)	0.00	0.00	0.52	0.90	1.12	1.24	1.32	1.41	1.45	1.49

Tabella 5.3: Valori degli indici di consistenza casuali (RI) relativi a diverse dimensioni della matrice casuale in considerazione.

generale somiglianza delle tendenze, che producono sostanzialmente lo stesso ranking delle tecniche di subsurface scattering, validando ulteriormente le conclusioni tratte precedentemente.

3. **Verifica della coerenza:** l’ultimo passo consiste nella verifica del livello di consistenza delle matrici. Questo passaggio è opzionale, in quanto abbiamo già determinato l’ordinamento delle alternative in quello precedente. Calcolare, però, il *rapporto di consistenza* (Consistency Ratio - CR) ci permette di stabilire la coerenza dei giudizi forniti dagli utenti nel questionario. In questo caso, una matrice si dice coerente se rispetta la proprietà transitiva. Ossia se, per esempio, date tre alternative A,B e C, ed A è preferito 2 volte a B e B è preferito 3 volte a C, A dovrebbe essere preferito 6 volte a C per poter dire che la matrice è coerente. Per calcolare quindi tale indice utilizziamo la seguente formula

$$CR = \frac{CI}{RI} \quad (5.1)$$

dove *CI* è l’indice di consistenza della matrice in considerazione ed *RI* è l’indice di consistenza casuale (Random Consistency Index) di una matrice della stessa dimensione, generata però casualmente. L’indice di consistenza casuale dipende dal numero di alternative che stiamo considerando, ossia *n* (Tab. 5.3). L’indice di consistenza, invece, si calcola con la seguente equazione

$$CI = \frac{\lambda_{max} - n}{n - 1} \quad (5.2)$$

dove *n* è il numero di alternative in considerazione e  $\lambda_{max}$  è l’autovalore maggiore della matrice. Utilizzando tale equazione, otteniamo per ciascuna delle configurazioni i risultati indicati in Tab. 5.4. Considerando che CR ricade in un intervallo fra zero ed un valore positivo (che dipende dal numero di alternative), il significato dei valori prodotti, come indicato da Saaty [28], può essere suddiviso in:

	Primissimo Piano	Primo Piano	Mezzo Primo Piano
Indice di consistenza (CI)	0.0476	0.0130	0.0171
Rapporto di consistenza (CR)	0.0425	0.0116	0.0153

Tabella 5.4: Valori degli indici di consistenza (CI) e rapporti di consistenza (RI) relativi a ciascuna delle diverse configurazioni (ossia inquadrature). Dai risultati si evince come, in tutti i casi, le matrici si possano considerare moderatamente consistenti.

- (a)  $CR \approx 0$ : in questo caso la matrice è estremamente coerente.
- (b)  $CR < 0.1$ : in questo caso si può dire che la matrice è moderatamente coerente, indicando come i giudizi degli utenti siano sufficientemente affidabili.
- (c)  $CR \geq 0.1$ : in questo caso la matrice viene considerata non coerente, indicando come possano esserci dei problemi nei giudizi ottenuti. Sarebbe quindi consigliato modificare il metodo con cui sono state ottenute le preferenze e ripetere tutto il processo.

Nel nostro caso tutti i valori di CR in Tab. 5.4 ricadono sotto la soglia dello 0.1, indicando quindi una livello moderato di coerenza per tutte e tre le categorie che abbiamo analizzato.

# 6

## Considerazioni Finali

In conclusione, abbiamo introdotto una nuova tecnica per il subsurface scattering (SSS) basata su un campionamento a 2 livelli, dove quest'ultimo viene selezionato in base al contrasto locale. La tecnica è molto semplice da introdurre in una qualsiasi pipeline grafica, in quanto non necessita di alcuna modifica a quest'ultima. Può essere, infatti, implementata direttamente nello stesso shader in cui viene computato il contributo del SSS senza uso di ulteriori render target. Inoltre, non richiede l'utilizzo di un qualsivoglia sistema di anti-aliasing temporale (TAA) per eliminare il rumore come in [32]. Per quanto il TAA permetta di ridurre, per la maggior parte, gli artefatti causati dal sottocampionamento usando un sistema di accumulo, spesso finisce per sfocare l'immagine prodotta, in particolare quando la telecamera è in movimento.

Per determinare la bontà dell'algoritmo proposto, oltre che per fare una valutazione generale sulle varie tecniche di SSS, è stato effettuato un questionario in cui i partecipanti dovevano indicare per ogni coppia di immagini, ciascuna delle quali implementavano tecniche diverse, quale delle due percepissero come più realistica. Inoltre, ogni coppia veniva presentata in una di tre configurazioni diverse basate sulla distanza, ovvero, da vicino, da distanza media e da lontano. È stato osservato come la tecnica di Golubev in entrambe le sue varianti, ovvero sia classica che con il campionamento extra, risultasse nettamente la preferita nella configurazione da vicino. All'aumentare della distanza, però, le differenze tendevano ad appiattirsi, fino ad arrivare alla configurazione da lontano, in cui non c'è una preferenza netta da parte dei partecipanti di nessuna delle tecniche. Anzi, in quest'ultima configurazione, sembra che ci sia una leggera preferenza per l'assenza totale di subsurface scattering. In ogni caso, sembra che la tecnica proposta in questa tesi produca sostanzialmente risultati percettivamente indistinguibili da quelli ottenuti con la versione standard di Golubev, utilizzando, in media, circa 40-45% campioni in meno.

Per rafforzare ulteriormente le conclusioni tratte dall'analisi manuale del questionario, abbiamo fatto anche una valutazione usando un processo di gerarchia analitica (AHP) [28]. I risultati prodotti erano in linea con ciò che avevamo dedotto in precedenza ed, inoltre, abbiamo verificato anche la consistenza dei giudizi dei partecipanti.

Un'area promettente per futuri sviluppi della tecnica proposta consiste nell'uso di un numero variabile di campioni come in [32], piuttosto che due insiemi di campioni fissi, uno per il passo base ed uno per l'eventuale passo extra. Questo approccio potrebbe potenzialmente ridurre ulteriormente il numero

di campioni totali, non dovendo impostare il numero di campioni extra in base al caso peggiore. Inoltre, utilizzare un numero variabile potrebbe rendere più morbido il passaggio da zona a campionamento normale a zona ad alto campionamento. Nel caso, sarà da valutare se questo porterà a risultati percettivamente migliori. Uno svantaggio di questo approccio sarebbe, però, il non poter più pre-computare i campioni. Questi, infatti, dovranno essere determinati al volo nello shader in base al contrasto locale calcolato. Sarà da valutare se l'impatto sulle performance del dover calcolare i campioni al volo verrà comunque ripagato dal utilizzare un numero minore di campioni in totale. Un ultimo possibile sviluppo riguarda l'eventualità di rimappare il contrasto computato su una funzione non lineare. Per ora, infatti, c'è una relazione lineare fra il contrasto calcolato per un pixel e la sua rilevanza. Rimappando, invece, il contrasto su una funzione alternativa si potrebbe dare una distribuzione diversa di importanza ai valori di contrasto. Facendo ciò otterremmo una forma di miglioramento del contrasto, similmente a come viene fatto in [10][15]. Allo stato attuale, tale comportamento non è replicabile tramite il solo utilizzo del parametro threshold. L'introduzione di tale funzione potrebbe potenzialmente migliorare la selezione dei pixel che necessitano di un campionamento extra. È comunque da verificare se il miglioramento sarebbe percettivamente significativo, considerando anche il costo del rimappare una funzione.

# Bibliografia

- [1] The 76th Scientific & Technical Awards 2003 — 2004. <https://www.oscars.org/sci-tech/ceremonies/2004>, 2004.
- [2] Tomas Akenine-Moller, Eric Haines, e Naty Hoffman. *Real-time rendering*. AK Peters/crc Press, 2019.
- [3] Azeddine Beghdadi, Muhammad Ali Qureshi, Seyed Ali Amirshahi, Aladine Chetouani, e Marius Pedersen. A critical analysis on perceptual contrast and its use in visual information analysis and processing. *IEEE Access*, 8:156929–156953, 2020.
- [4] George Borshukov e John P Lewis. Realistic human face rendering for “the matrix reloaded”. In *ACM Siggraph 2005 Courses*, pp. 13–es. 2005.
- [5] Brent Burley e Walt Disney Animation Studios. Physically-based shading at disney. In *Acm Siggraph*, volume 2012, pp. 1–7. vol. 2012, 2012.
- [6] Matt Jen-Yuan Chiang, Peter Kutz, e Brent Burley. Practical and controllable subsurface scattering for production path tracing. In *ACM SIGGRAPH 2016 Talks*, pp. 1–2. 2016.
- [7] Per H Christensen. An approximate reflectance profile for efficient subsurface scattering. In *ACM SIGGRAPH 2015 Talks*, pp. 1–1. 2015.
- [8] Eugene d’Eon e Geoffrey Irving. A quantized-diffusion model for rendering translucent materials. *ACM transactions on graphics (TOG)*, 30(4):1–14, 2011.
- [9] Eugene d’Eon e David Luebke. Advanced techniques for Realistic Real-Time skin rendering In *GPU Gems 3*. A cura di Hubert Nguyen, capitolo 14. Addison-Wesley Professional, 2007.
- [10] Atam P Dhawan, Gianluca Buelloni, e Richard Gordon. Enhancement of mammographic features by optimal adaptive neighborhood image processing. *IEEE Transactions on Medical Imaging*, 5(1):8–15, 1986.
- [11] Craig Donner e Henrik Wann Jensen. Light diffusion in multi-layered translucent materials. *ACM Transactions on Graphics (ToG)*, 24(3):1032–1039, 2005.
- [12] Fabian Giesen. A trip through the graphics pipeline 2011. [www.fgiesen.wordpress.com/2011/07/09/a-trip-through-the-graphics-pipeline-2011-index/](http://www.fgiesen.wordpress.com/2011/07/09/a-trip-through-the-graphics-pipeline-2011-index/), Jul 2011.
- [13] Evgenii Golubev. Efficient screen-space subsurface scattering using burley’s normalized diffusion in real-time. *ACM SIGGRAPH Courses: Advances in Real-Time Rendering in Games Course*, 2018.

- [14] Evgenii Golubev. Sampling Burley's Normalized Diffusion Profiles. <https://zero-radiance.github.io/post/sampling-diffusion/>, Dec 2019.
- [15] Richard Gordon e Rangaraj M Rangayyan. Feature enhancement of film mammograms using fixed and adaptive neighborhoods. *Applied optics*, 23(4):560–564, 1984.
- [16] Jason Gregory. *Game Engine Architecture, Second Edition*. CRC Press, aug 15 2014.
- [17] Jing Huang, Tammy Boubekeur, Tobias Ritschel, Matthias Hollaender, e Elmar Eisemann. Separable approximation of ambient occlusion. 2011.
- [18] Akira Ishimaru e altri. *Wave propagation and scattering in random media*, volume 2. Academic press New York, 1978.
- [19] Henrik Wann Jensen, Stephen R Marschner, Marc Levoy, e Pat Hanrahan. A practical model for subsurface light transport. In *Proceedings of the 28th annual conference on Computer graphics and interactive techniques*, pp. 511–518, 2001.
- [20] Jorge Jimenez, Veronica Sundstedt, e Diego Gutierrez. Screen-space perceptual rendering of human skin. *ACM Transactions on Applied Perception (TAP)*, 6(4):1–15, 2009.
- [21] Jorge Jimenez, David Whelan, Veronica Sundstedt, e Diego Gutierrez. Real-time realistic skin translucency. *IEEE Computer Graphics and Applications*, 30(4):32–41, 2010.
- [22] Jorge Jimenez, Károly Zsolnai, Adrian Jarabo, Christian Freude, Thomas Auzinger, Xian-Chun Wu, Javier von der Pahlen, Michael Wimmer, e Diego Gutierrez. Separable subsurface scattering. In *Computer Graphics Forum*, volume 34, pp. 188–197. Wiley Online Library, 2015.
- [23] James T Kajiya. The rendering equation. In *Proceedings of the 13th annual conference on Computer graphics and interactive techniques*, pp. 143–150, 1986.
- [24] Morten S Mikkelsen. Skin rendering: Reflectance and integration. 2009.
- [25] Benhur Ortiz-Jaramillo, Asli Kumcu, Ljiljana Platisa, e Wilfried Philips. Content-aware contrast ratio measure for images. *Signal Processing: Image Communication*, 62:51–63, 2018.
- [26] Matt Pettineo. An introduction to real-time subsurface scattering. [www.therealmpj.github.io/posts/ssss-intro/](http://www.therealmpj.github.io/posts/ssss-intro/), Oct 2019.
- [27] Matt Pharr, Wenzel Jakob, e Greg Humphreys. *Physically based rendering: From theory to implementation*. MIT Press, 2023.
- [28] R.W. Saaty. The analytic hierarchy process—what it is and how it is used. *Mathematical Modelling*, 9(3):161–176, 1987.
- [29] Thomas L. Saaty. Decision-making with the ahp: Why is the principal eigenvector necessary. *European Journal of Operational Research*, 145(1):85–91, 2003.
- [30] Lihong Wang, Steven L. Jacques, e Liqiong Zheng. Mcml—monte carlo modeling of light transport in multi-layered tissues. *Computer Methods and Programs in Biomedicine*, 47(2):131–146, 1995.

- [31] Zhou Wang, A.C. Bovik, H.R. Sheikh, e E.P. Simoncelli. Image quality assessment: from error visibility to structural similarity. *IEEE Transactions on Image Processing*, 13(4):600–612, 2004.
- [32] Tiantian Xie, Marc Olano, Brian Karis, e Krzysztof Narkowicz. Real-time subsurface scattering with single pass variance-guided adaptive importance sampling. *Proceedings of the ACM on Computer Graphics and Interactive Techniques*, 3(1):1–21, 2020.