

روش‌های رندر سه بعدی

رسول کامکار

دانشجوی مهندسی کامپیوتر دانشگاه صنعتی اصفهان
rasool.kamkar20@gmail.com

الهام عموزاده

دانشجوی مهندسی کامپیوتر دانشگاه صنعتی اصفهان
elhamamoozade@ec.iut.ac.ir

ریحانه رضایی

دانشجوی مهندسی کامپیوتر دانشگاه صنعتی اصفهان
Rhnrezaei78@gmail.com

چکیده

در این مقاله به بررسی مفهوم رندر و دو نوع آن یعنی رندر از پیش انجام شده و رندر در زمان واقعی، تعریف مدل سه بعدی و ارتباط آن با رندر سه بعدی، روش‌های رندر سه بعدی شامل شطرنجی سازی، ردیابی پرتو و پیشروی پرتو با مطالعه مقالات و تحقیقات از پیش انجام شده در این زمینه پرداخته شده است. در نتیجه این بررسی، آشنایی با نحوه انجام رندر در سه روش مختلف، تفاوت‌های اصلی، مزیت‌ها و معایب هر روش نسبت به دیگری به دست آمد. شطرنجی سازی رایج‌ترین و پرکاربردترین روش با سرعت بالا و استفاده شده در هر دو نوع رندر است. ردیابی پرتو روشی با سرعت بسیار پایین اما قدرت و کیفیت بالاتر و معمولاً استفاده شده در رندر از پیش انجام شده است. پیشروی پرتو روشی نسبتاً جدید بر مبنای ردیابی، با سرعت اما محدودیت بیشتر و در عین حال ویژگی‌های خاص مانند جلوه‌های ویژه و زیبا در عین سادگی محاسبات است.

کلمات کلیدی: رندر سه بعدی، مدل سه بعدی، شطرنجی سازی، ردیابی پرتو، پیشروی پرتو

۱. مقدمه

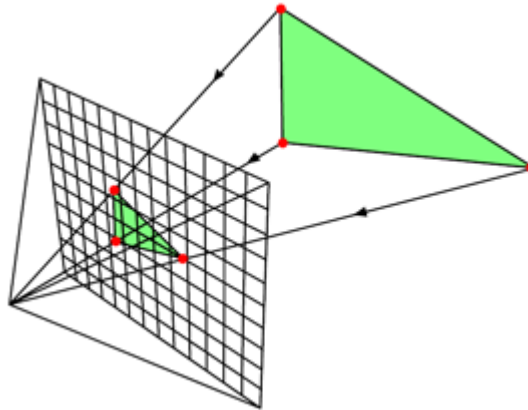
رندر (Render) فرایندی برای نمایش تصاویر بر روی صفحه نمایش است. اشکال هندسی، اجسام سه بعدی و جزییات دنیای مجازی داخل کامپیوتر هنگام رندر به صورت مجموعه‌ای از پیکسل‌ها درآمده و سپس بر صفحه نمایش نشان داده می‌شود. در طول زمان روش‌های زیادی برای انجام این کار مطرح شده‌اند، از جمله این روش‌ها می‌توان به پرتاب پرتو (Ray casting)، شطرنجی‌سازی (Rasterization)، ردیابی پرتو (Ray tracing) و پیشروی پرتو (Ray marching) اشاره کرد. روش پرتاب پرتو از روش‌های قدیمی و منسوخ شده است که با پیشرفت تکنولوژی دیگر استفاده نمی‌شود. در این مقاله در ابتدا به بررسی مفهوم رندر و مدل سه بعدی و سپس به سه روش ذکر شده دیگر، نحوه انجام آن‌ها، تفاوت‌ها، مزیت‌ها و معایب هر کدام پرداخته شده است.

۲. بخش اصلی

در این بخش در ابتدا به بررسی مفهوم رندر و مدل سه بعدی، و سپس به بررسی سه روش اصلی رندر می‌پردازیم.

۱-۲. رندر چیست [۱]

رندر فرایندی برای رسم اشکال و تصاویر بر روی صفحه نمایش است. هنگام رندر رنگ تک تک پیکسل‌های صفحه به گونه‌ای تنظیم می‌شوند تا تصویری قابل فهم و بامعنا به دید کاربر برسد.



شکل ۱ رندر یک مثلث بر روی صفحه [۲]

به طور کل رندر به دو دسته تقسیم می‌شود که به بررسی آن‌ها می‌پردازیم:

۱-۱-۲. رندر در زمان واقعی (real time) [۱]

این نوع رندر با سرعت خیلی بالا و به طور زنده انجام می‌شود و می‌تواند به تغییرات واکنش نشان دهد. مثال اصلی برای این نوع بازی‌های ویدیوای است که معمولا رندر با سرعت ۶۰ بار در ثانیه انجام می‌شود و می‌تواند با ورودی‌های بازیکن تغییر کند.

۲-۱-۲. رندر از پیش انجام شده (pre render) [۱]

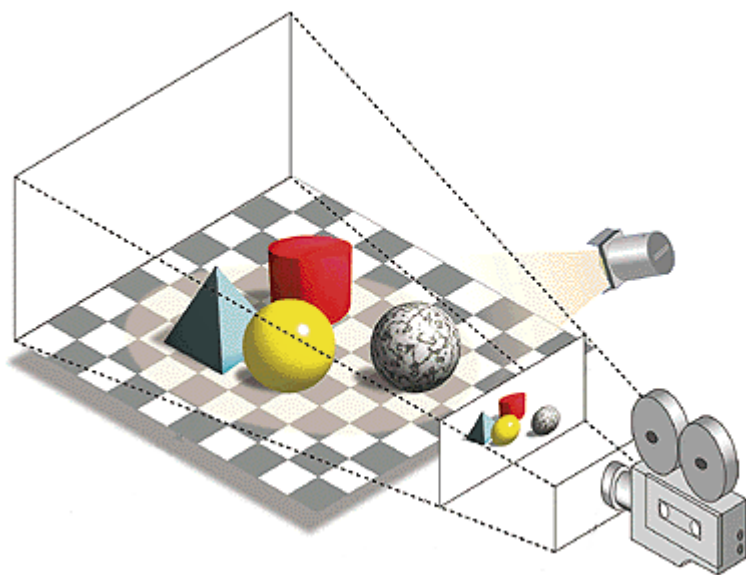
این نوع رندر نیازی به تغییر پویا ندارد و به همین علت می‌توان زمان خیلی بیشتری به آن اختصاص داد. نتیجه این نوع رندر معمولا به صورت یک تصویر یا مجموعه‌ای از تصاویر به شکل ویدیو، داخل یک فایل ذخیره شده و در اختیار ما قرار می‌گیرد که به راحتی و بدون محاسبات اضافه می‌توان از آن استفاده کرد. برای نمونه می‌توان به انیمیشن‌ها اشاره کرد.

۲-۲. مدل سه بعدی چیست [۳]

برای انجام رندر نیاز به محیطی داریم که آن را رندر کنیم و تصویر مورد نظر را به کاربر نشان دهیم. به این محیط، مدل سه بعدی گفته می‌شود. هر محیط سه بعدی سه مؤلفه اصلی دارد که در ادامه به بررسی آن‌ها می‌پردازیم.

۱-۲-۲. اشکال سه بعدی [۳]

مدل سه بعدی مورد نظر باید توسط اشکالی توصیف شود. این توصیف عموما ترکیبی از اشکال و ساختارهای سه بعدی است که در کنار هم قرار گرفته‌اند، مانند یک مداد، چند درخت یا حتی یک کره.

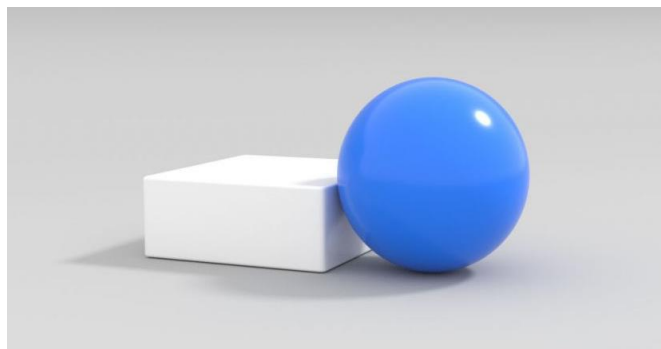


شکل ۲ تصویری از یک مدل سه بعدی، شامل دوربین، یک منبع نور و چندین شکل [۴]

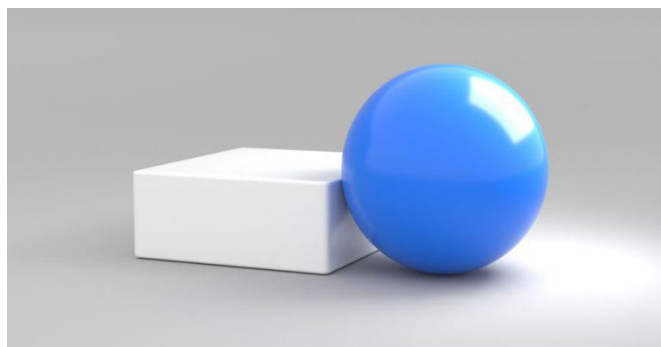
۲-۲-۲. منابع نور [۳]

برای اینکه رندر انجام شده بتواند یک تصویر دو بعدی مشخصی از فضای سه بعدی داشته باشد، نیاز به یک یا چند منبع نور است که باعث ایجاد سایه شود و بتوان عمق تصویر را درک کرد، این منابع نور در مدل سه بعدی تعریف می‌شوند.

منبع نور خود چندین دسته با ویژگی‌های مختلف دارد، مانند نور جهت‌دار، نور منطقه‌ای و نور نقطه‌ای. [۵]



شکل ۳ نور جهت‌دار [۵]



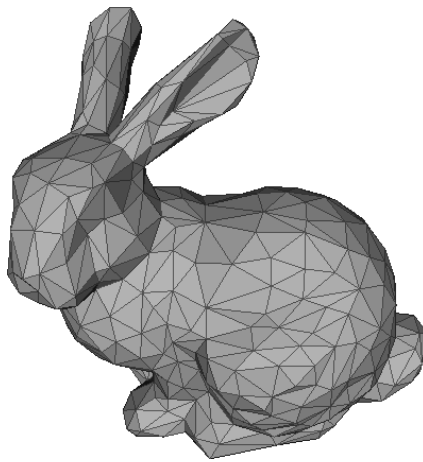
شکل ۴ نور منطقه‌ای [۵]

۲-۲-۳. چشم یا دوربین [۵]

برای اینکه تصویری از مدل سه بعدی خود داشته باشیم، نیاز است که بدانیم این مدل به چه شکلی دیده می‌شود. هر مدل نیاز به یک دوربین دارد که زاویه، موقعیت و پهنای دید بیننده نسبت به مدل را توصیف می‌کند و رندر باید نسبت به این دوربین انجام شود.

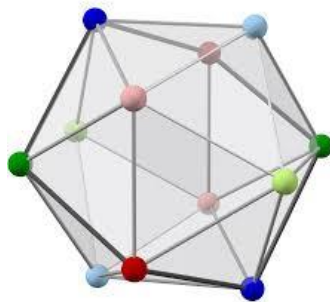
۲-۳. شطرنجی سازی [۶]

شطرنجی سازی (Rasterization) از پرکاربردترین و رایج‌ترین روش‌های رندر است. در این روش، اشکال در مدل سه بعدی متشکل از تعدادی مثلث است که به آن‌ها بافته (mesh) گفته می‌شود. کیفیت، وضوح و جزئیات یک شکل با تعداد مثلث‌های استفاده شده در آن رابطه مستقیم دارد.



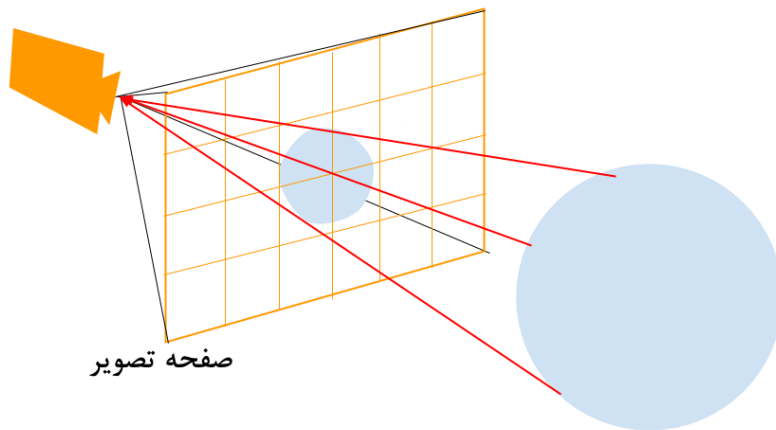
شکل ۵ یک نمونه بافته [۶]

هر مثلث خود متشکل از سه رأس است که این رئوس با سه مؤلفه مختصات (x, y, z) در فضا مشخص می‌شود.



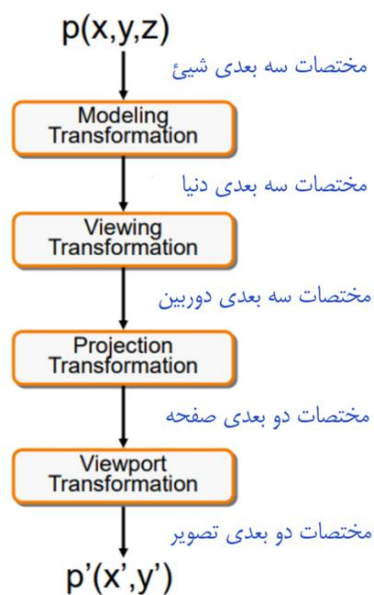
شکل ۶ رئوس مثلث در یک بافته [۶]

این روش به طور خلاصه رئوس سه بعدی را به رئوس دو بعدی تبدیل و سپس بر روی یک صفحه منعکس و رندر می‌کند. به صفحه مورد نظر صفحه تصویر (Image plane) گفته می‌شود که در فضایی بین دوربین و مدل قرار می‌گیرد.



شکل ۷ صفحه تصویر [۶]

تبدیل رئوس سه بعدی به دو بعدی به صورت ریاضیاتی و با استفاده از ماتریس‌های تبدیل انجام می‌شود که شامل ۴ مرحله اصلی است که در ادامه این چهار مرحله بررسی می‌شوند.



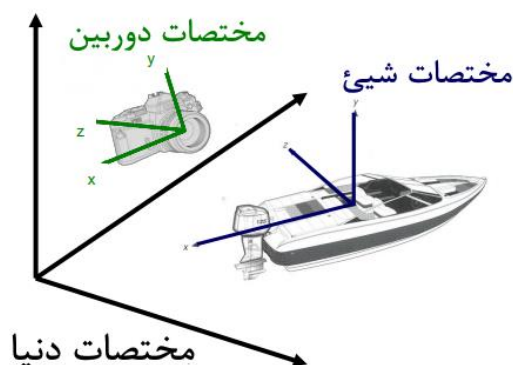
شکل ۸ مراحل تبدیل مختصات رئوس [۷]

۲-۳-۱. تبدیل مدل (Modeling Transformation) [۷]

زمانی که یک بافته برای استفاده آماده می‌شود، مختصات رئوس آن نسبت به خود آن بافته مقدار گرفته‌اند، به عبارت دیگر، مبدأ مختصات خود شکل است. در این مرحله مبدأ این مختصات تبدیل به مبدأ دنیای اصلی شده و رئوس وارد مدل سه بعدی می‌شوند.

۲-۳-۲. تبدیل دید (Viewing Transformation) [۷]

پس از تبدیل مدل، مبدأ باید دوباره تغییر کرده و رئوس نسبت به دوربین مختصات جدید بگیرند.

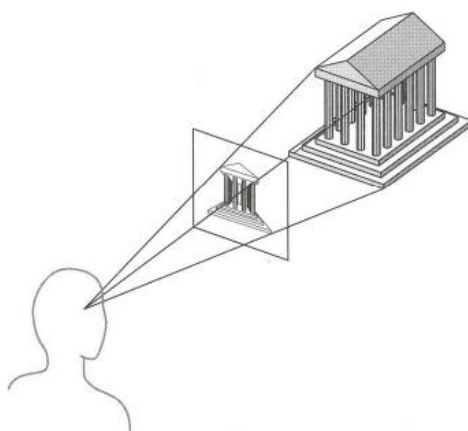


شکل ۹ تبدیل مدل از مختصات شیء به دنیا و تبدیل دید از مختصات دنیا به دوربین انجام می‌شود [۷]

۳-۳-۲. تبدیل نمایش (Projection Transformation) [۷]

این مرحله نقطه کلیدی رندر است. در ریاضیات به طور کل با اعمال این تبدیل یک بعد مختصاتی حذف می‌شود، برای درک بهتر این تبدیل تصور کنید یک کره را مقابل یک تخته گرفته و از سمت دیگر به آن نور بتابانید، سایه تشکیل شده نتیجه تبدیل را نشان می‌دهد.

در روش شطرنجی سازی، این تبدیل بعد سوم را از مختصات حذف کرده و ویژگی‌های دوربین مانند زاویه دید را بر روی تصویر اعمال می‌کند.



شکل ۱۰ تبدیل نمایش [۷]

۴-۳-۲. تبدیل چشم انداز (Viewport Transformation) [۷]

مرحله نهایی تبدیلی‌ها، رؤوس دو بعدی را از صفحه تصویر به صفحه نمایش منتقل می‌کند، این تبدیل به دلیل امکان متفاوت بودن اندازه دو صفحه ذکر شده مورد نیاز است.

۴-۲. ردیابی پرتو [۸]

برای درک این روش، روشنایی و روند دیدن توسط چشم انسان را در دنیای واقعی در نظر بگیرید. یک منبع نور، تعداد غیر قابل تجسمی از پرتوهای متشکل از فوتون‌ها را از خود منتشر می‌کند. برخی از این پرتوها با سطوح اجسام برخورد کرده، پس از برخورد مقداری از آن‌ها به سمت چشم بازتاب شده و در نتیجه ما می‌توانیم جسم را ببینیم، برخی دیگر به سمت

اجسام دیگر بازتاب شده که باعث انعکاس تصاویر می‌شود و برخی دیگر از جسم رد شده که شفافیت جسم را به ما نشان می‌دهد.



شکل ۱۱ رفتار نور، سایه و انعکاس [۸]

روش ردیابی پرتو رفتار فوتون‌های نور را شبیه‌سازی می‌کند. به دلیل نزدیک بودن این شبیه‌سازی به دنیای واقعی، کیفیت تصاویر رندر شده به ویژه کیفیت انعکاس و سایه بسیار بالاتر است، اما نسبت به روش قبلی سرعت بسیار پایین و محاسبات پیچیده‌تری دارد، به همین علت معمولاً این روش برای رندهای از پیش انجام شده استفاده می‌شود.

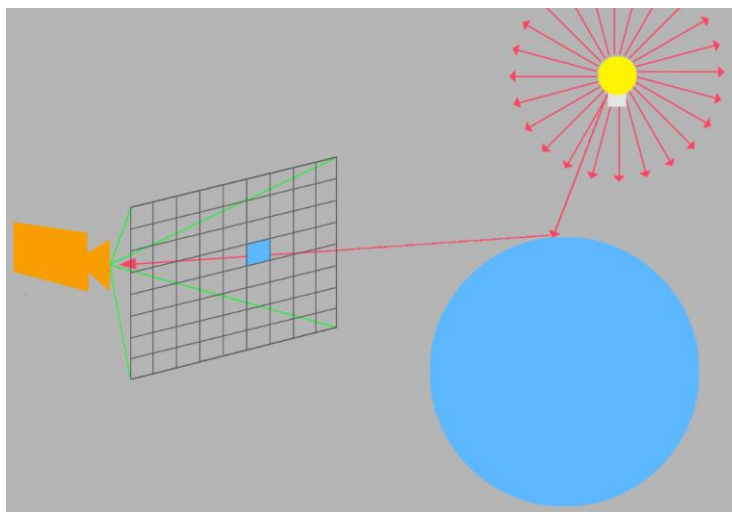
منظور از پرتو (Ray) در دنیای سه بعدی، خط مستقیمی است که دو نقطه در این فضا را به هم متصل می‌کند. [۱]

۲-۴-۱. انواع ردیابی [۸]

ردیابی پرتو به دو روش قابل انجام است که در ادامه آن‌ها را بررسی می‌کنیم.

۲-۴-۱-۱. ردیابی رو به جلو (forward) [۸]

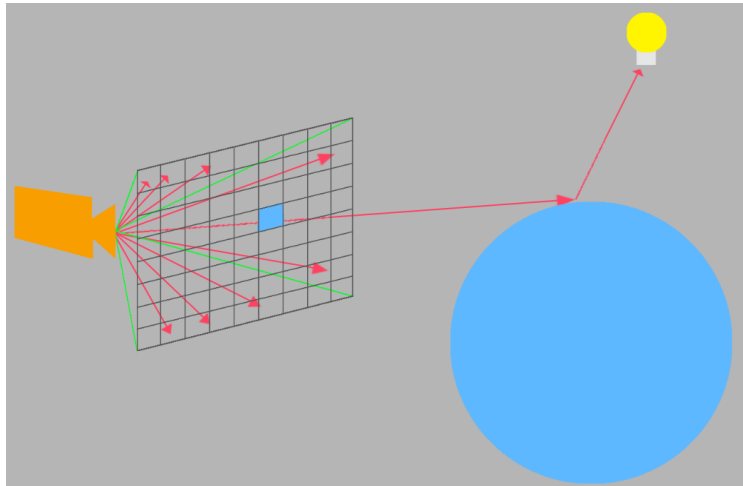
در این روش از ردیابی، تعداد بسیار زیادی پرتو از منابع نور به سمت مدل سه بعدی تابیده می‌شوند. سپس بررسی می‌شود که کدام پرتوها به دوربین رسیده و در نهایت تصویر رندر می‌شود. واضح است که در این روش تعداد زیادی از پرتوها مفید نیستند و باعث هدر رفت قدرت محاسباتی و حافظه می‌شوند.



شکل ۱۲ ردیابی رو به جلو [۸]

۲-۴-۲. ردیابی عقب‌گرد (backward) [۸]

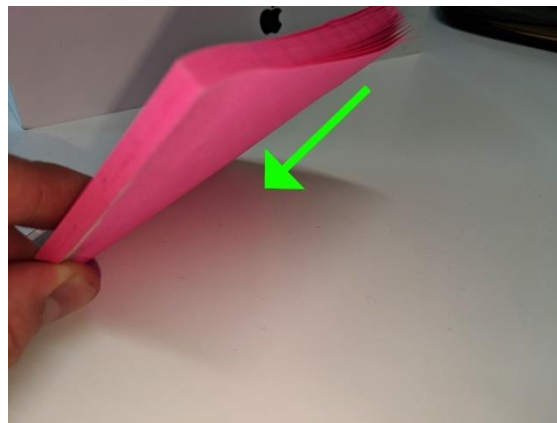
در این روش، به جای تابیدن پرتوها از سمت منبع نور به دوربین، مبدأ حرکت پرتو را دوربین در نظر می‌گیریم. از تمامی پیکسل‌های صفحه تصویر یک پرتو توسط دوربین تابیده شده و به سمت صحنه مورد نظر ادامه پیدا می‌کند. زمانی که حرکت پرتو به پایان رسید، تعداد برخوردها و بازتاب‌های آن و همچنین این مسئله که به منبع نور رسیده یا خیر، رنگ و مقدار روشنایی پیکسل مورد نظر در صفحه تصویر را مشخص می‌کند. این روش نسبت به ردیابی رو به جلو به طور چشم‌گیری سریع‌تر اما همچنان فرایندی سنگین و زمان‌بری است.



شکل ۱۳ ردیابی عقب‌گرد [۸]

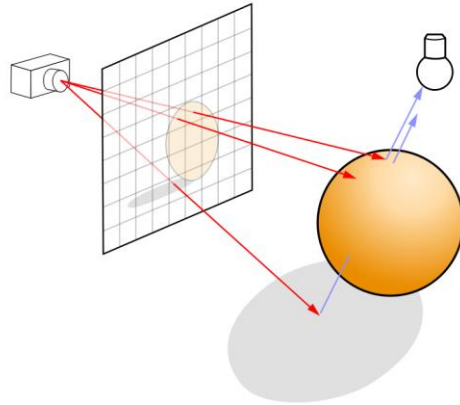
۲-۴-۲. سایه [۸]

سایه‌ها در دنیای واقعی کاملاً سیاه نیستند، بلکه پرتوهای نوری پس از چندین برخورد می‌توانند همچنان مقداری از انرژی خود را همراه خود داشته و همچنین رنگ بازتاب‌های قبلی را هم به سایه منتقل کنند.



شکل ۱۴ سایه در دنیای واقعی [۸]

در روش ردیابی هم یک پرتو پس از چندین برخورد با اجسام مختلف، اطلاعات جمع‌آوری شده را به پیکسل متناظر خود منتقل می‌کند تا رنگ آن تعیین شود. همچنین برای تعیین سایه، پس از برخورد یک پرتو با جسمی، بدون توجه به زاویه برخورد، پرتوهای سایه به سمت منابع نور منعکس می‌شوند، اگر در طول مسیر به جسمی برخورد کرده یا به منبع نور برسد، مشخص کننده مقدار سایه پیکسل است.



شکل ۱۵ سایه در ردیابی پرتو [۹]

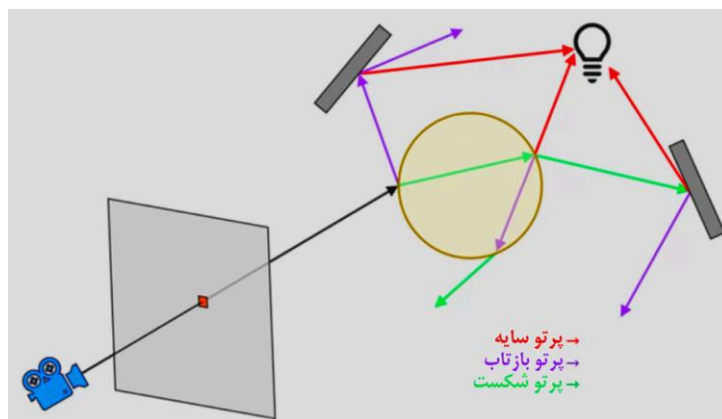
۲-۴-۳. انعکاس [۸]

انعکاس همواره مسئله دشواری برای روش شطرنجی سازی بوده است و همیشه سعی شده به صورت تقلبی اما قانع کننده انجام شود.



شکل ۱۶ انعکاس در بازی Battlefield V [۸]

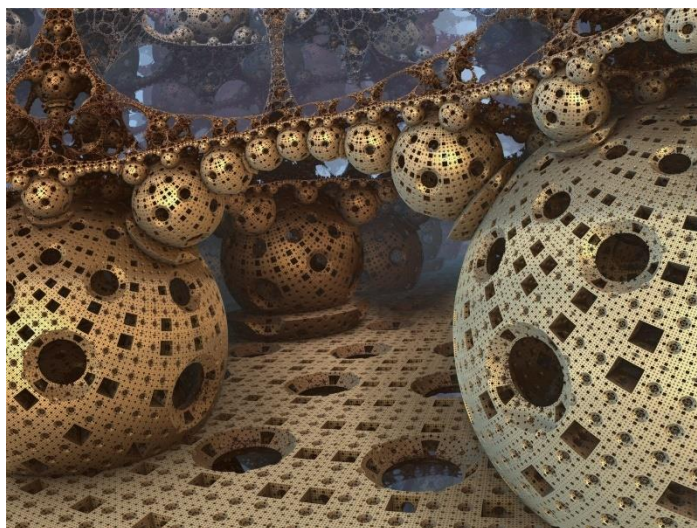
همانطور که در قسمت قبل ذکر شد، پرتو پس از چندین برخورد و بازتاب اطلاعاتی به دست می آورد. اگر در میان این اطلاعات جنس اجسامی که پرتو با آن‌ها برخورد داشته ذخیره شود، می توان انعکاس نور و اجسام مختلف را به طور طبیعی نشان داد. به طور مثال، اگر یک پرتو به فلزی با سطح صیقلی برخورد کند و پس از آن به سمت شعله آتشی منعکس شود، تصویر نسبتاً واضحی از آتش باید در آن فلز دیده شود.



شکل ۱۷ رفتار پرتوها [۱۰]

۵-۲. پیشروی پرتو [۱۱]

این روش یک نسخه تغییر یافته و خاص از ردیابی پرتو است که یک هدف ویژه دارد. این روش به طرز شگفت‌آوری نسبت به ردیابی سریع است اما نقطه ضعف آن محدودیت تصاویری است که می‌تواند رندر کند.

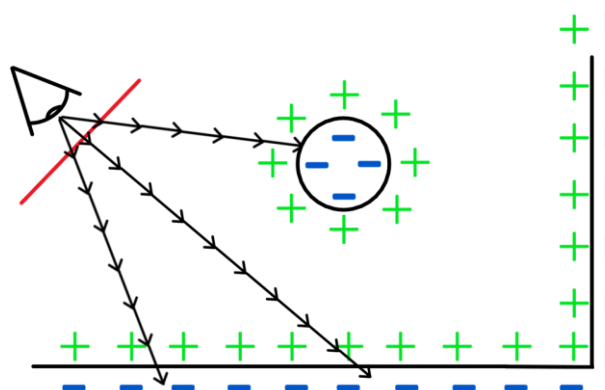


شکل ۱۸ تصویر رندر شده با روش پیشروی پرتو [۱۱]

۵-۲-۱. تابع فاصله علامت دار (signed distance functions) [۱۱]

پیشروی پرتو قادر به رندر کردن بافت‌های پیچیده مثلثی نیست و به جای آن، با روابط ریاضی خاصی به نام تابع فاصله علامت‌دار در ارتباط است و قادر است اشکال نسبتاً ساده‌ای که قابل توصیف توسط یک تابع ریاضی هستند را رندر کند. در این روش مدل سه بعدی از این توابع ریاضیاتی تشکیل می‌شود و اشکال پیچیده‌تر از ترکیب این حجم‌های ساده هندسی به دست می‌آیند.

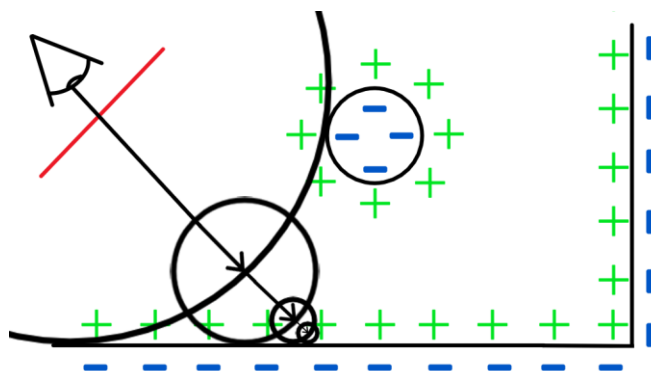
تابع فاصله علامت‌دار، یک نقطه در فضای سه بعدی را به عنوان ورودی دریافت و یک عدد علامت‌دار به عنوان خروجی برمی‌گرداند. اگر عدد مذکور مثبت باشد به این معنی است که نقطه مورد نظر، خارج از حجم است، اگر عدد صفر باشد، نقطه بر روی سطح حجم قرار دارد و در صورت منفی بودن، نقطه داخل حجم است.



شکل ۱۹ تابع فاصله در پیشروی پرتو [۱۲]

۲-۵-۲. رابطه پیشروی و ردیابی [۱۱]

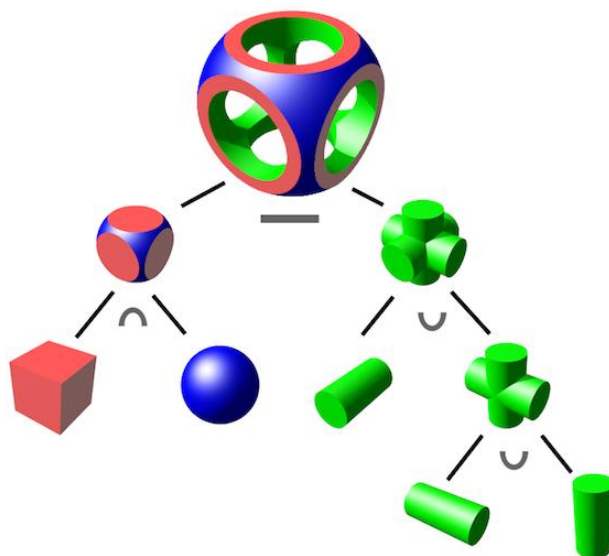
همانطور که پیشتر ذکر شد، پیشروی نوعی خاص از روش ردیابی است. در این روش، مانند حالت عقب‌گرد، پرتوها از سمت دوربین به سوی محیط تابیده می‌شوند، اما تفاوت اساسی، پیشروی پرتوها به صورت پاره‌خطی است. اندازه این پاره‌خطها معیار بسیار مهمی برای تعیین است، زیرا اگر طول پاره‌خطها بسیار کوچک باشد، محاسبات سنگین شده و سرعت کاهش می‌یابد و اگر بزرگ باشد، ممکن است از یک حجم رد شده و رندر به درستی انجام نشود. برای تعیین دقیق طول پاره‌خط و مقدار حرکت رو به جلو هر پرتو، از تابع فاصله ذکر شده استفاده می‌شود، به این صورت که از نقطه مورد نظر، کمترین مقدار بازگشتی میان همه توابع داخل مدل را یافته و به همان اندازه پرتو به جلو حرکت می‌کند.



شکل ۲۰ تعیین اندازه پاره‌خط [۱۲]

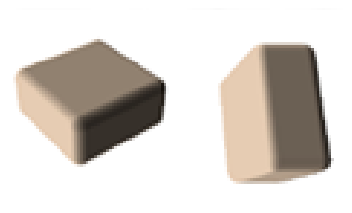
۲-۵-۳. اشکال پیچیده

توابع فاصله قابلیت ترکیب شدن با یکدیگر و تشکیل حجم‌های پیچیده‌تر با استفاده از اشتراک، اجتماع یا کم کردن از یکدیگر را دارند. [۱۳]

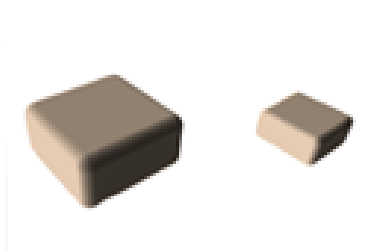


شکل ۲۱ ترکیب توابع فاصله و ایجاد اشکال پیچیده [۱۳]

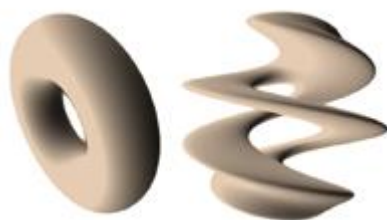
علاوه بر آن، می‌توان با اعمال ریاضیات ساده‌ای، چرخش (Rotation) و مقیاس (scale) اشکال را تغییر داد و حتی آن‌ها را پیچید (twist) یا خم (Bend) کرد. [۱۴]



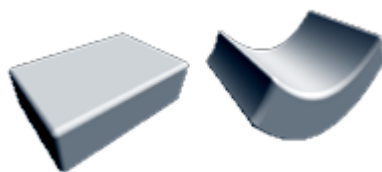
شکل ۲۲ چرخش [۱۴]



شکل ۲۳ مقیاس [۱۴]



شکل ۲۴ پیچیدن [۱۴]

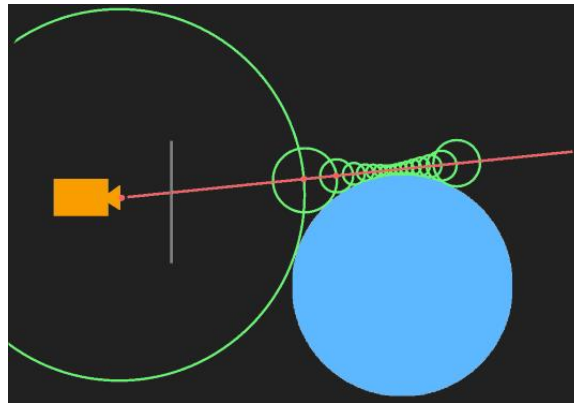


شکل ۲۵ خم کردن [۱۴]

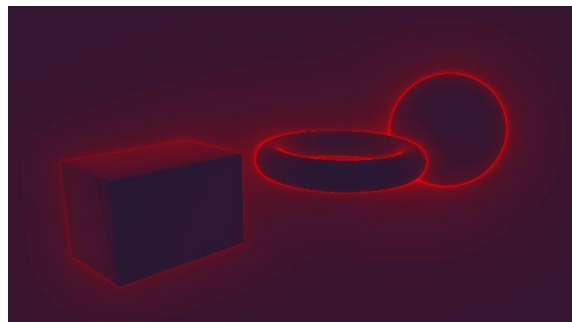
۲-۵-۴. جلوه‌های ویژه [۱۱]

به دلیل عملکرد خاصی که روش پیشروی دارد، می‌توان با حداقل محاسبات اضافه به جلوه‌های زیبایی دست یافت که در روش‌های دیگر نیاز به هزینه بیشتری دارند.

به طور مثال جلوه تابش را می‌توان با ذخیره کردن حداقل فاصله در حین حرکت پرتو اعمال کرد، به این صورت که اگر پرتو از فاصله بسیار کم یک شکل رد شده اما با آن برخوردی نداشته، نشان می‌دهد پیکسل مورد نظر می‌تواند رنگ روشن‌تری بگیرد که نتیجه آن نورانی شدن اطراف شکل می‌شود.



شکل ۲۶ نحوه عملکرد تابش [۱۱]

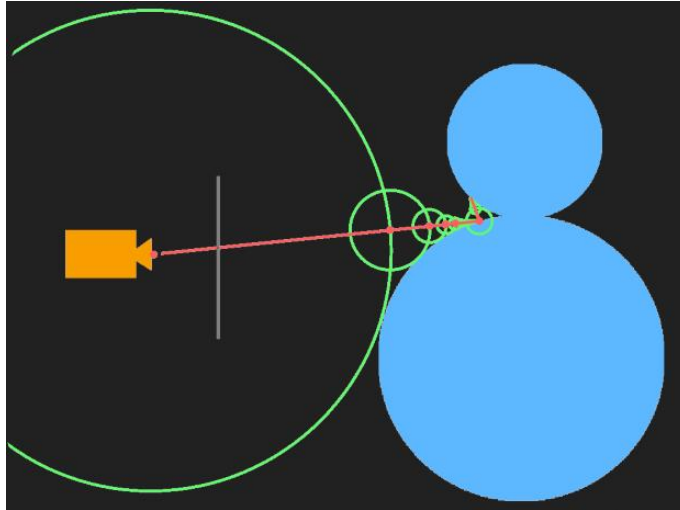


شکل ۲۷ جلوه تابش [۱۱]

جلوه دیگر انسداد محیطی است. این تکنیک گوشه و کنار اجسام را تاریک کرده و حس گم شدن و مسدود شدن نور را منتقل می‌کند. به این گونه که اگر یک پرتو با یک حجم برخورد کرد، عمود بر حجم در نقطه برخورد یک پرتو دیگر بازتاب شده و فاصله نزدیک‌ترین سطح (به غیر از سطحی که تازه با آن برخورد داشته) را به دست می‌آورد، اگر این فاصله به اندازه کافی کوچک باشد نشان می‌دهد انسداد محیطی باید بر آن پرتو و پیکسل متناظرش اعمال شود.

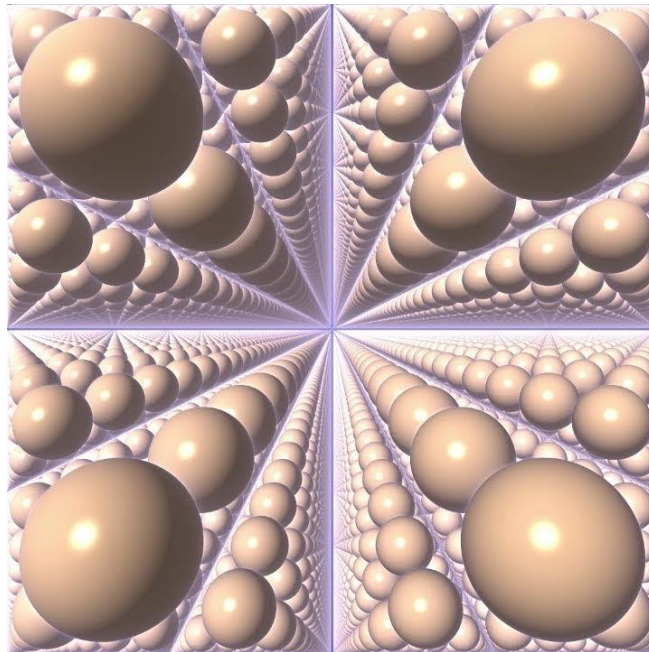


شکل ۲۸ جلوه انسداد محیطی [۱۱]



شکل ۲۹ نحوه عملکرد انسداد محیطی [۱۱]

یکی از مهم‌ترین جلوه‌های این روش، آئینه و تکرار است. به دلیل اینکه مدل در این روش مجموعه‌ای از روابط نسبتاً ساده ریاضی است، می‌توان با اعمال تبدیل‌های ریاضیاتی نچندان پیچیده‌ای، یک شکل را بارها و بارها تکرار کرد و تصویر میلیون‌ها جسم را با سرعت بسیار بالا و بدون محاسبات سنگین به دست آورد.



شکل ۳۰ نتیجه جلوه تکرار بر روی یک کره [۱۱]

۳. نتیجه‌گیری

نتیجه بررسی‌ها نشان می‌دهد که روش شطرنجی سازی، مبتنی بر ریاضیات است و ایده‌چندانی از دنیای واقعی نگرفته، کیفیت و جذابیت تصاویر آن نسبتاً پایین‌تر است اما سرعت بیشتری دارد، به همین دلیل به طور گسترده‌ای در صنعت و هر دو نوع رندر استفاده می‌شود، مانند بازی‌های ویدیویی یا طراحی‌های سه بعدی.

روش ردیابی پرتو، به دلیل تشابه بسیار با دنیای واقعی کیفیت تصاویر شگفت انگیزی به خصوص در زمینه سایه و انعکاس دارد اما در مقابل سرعت آن بسیار پایین تر است که موجب می شود بیشتر در رندرهای از پیش انجام شده استفاده شود.

پیشروی پرتو روشی مبتنی بر ردیابی پرتو است که بهبود سرعت چشم گیری نسبت به آن دارد، اما در مقابل نمی تواند اشکال پیچیده را رندر کند. در این روش مدل سه بعدی از روابط ریاضی که حجم های هندسی را نشان می دهند تشکیل می شود. در نتیجه تصاویر رندر شده در این روش ترکیبی پیچیده از این حجم های ساده هندسی هستند. مزیت این روش جلوه های ویژه و زیبا در عین سادگی محاسبات آن است.

مراجع

- [۱]- <https://si-ashbery.medium.com/learning-to-draw-with-a-pencil-made-of-code-aa21b2a7b963>
- [۲]- <https://www.scratchapixel.com/lessons/3d-basic-rendering/rasterization-practical-implementation>
- [۳]- Badler, Norman I., and Andrew S. Glassner. "3D object modeling." *SIGGRAPH 97 Introduction to Computer Graphics Course Notes 5* (۱۹۹۷).
- [۴]- <https://www.gamersnexus.net/guides/2429-gpu-rendering-and-game-graphics-explained>
- [۵]- <https://dreamfarmstudios.com/blog/the-ultimate-guide-to-lighting-fundamentals-for-3d/>
- [۶]- <https://si-ashbery.medium.com/rasterization-1328vcee9ab4>
- [۷]- <https://reillybova.github.io/COS426-Website/lectures/Lecture-14.pdf>
- [۸]- <https://si-ashbery.medium.com/raytracing-309fc44307e6>
- [۹]- [https://en.wikipedia.org/wiki/Ray_tracing_\(graphics\)](https://en.wikipedia.org/wiki/Ray_tracing_(graphics))
- [۱۰]- <https://www.techspot.com/article/1888-how-to-3d-rendering-rasterization-ray-tracing/>
- [۱۱]- <https://si-ashbery.medium.com/raymarching-3cdf86c637ba>
- [۱۲]- https://typhomnt.github.io/teaching/ray_tracing/raymarching_intro/
- [۱۳]- <http://jamie-wong.com/2016/07/15/ray-marching-signed-distance-functions/>
- [۱۴]- <https://iquilezles.org/articles/distfunctions/>