نمایش سریع حجمی با استفاده از همبستگی فضایی داده ها

داریوش مومنی
گروه مهندسی برق
دانشکده فنی دانشگاه تهران
قطب علمی کنترل و پردازش هوشمند
Momeni_dariush@yahoo.com

رضا آقائی زاده ظروفی
گروه مهندسی برق
دانشکده فنی دانشگاه تهران
قطب علمی کنترل و پردازش هوشمند
Zoroofi@ut.ac.ir

حمید سلطانیان زاده
گروه مهندسی برق
دانشکده فنی دانشگاه تهران
قطب علمی کنترل و پردازش هوشمند
Hzadeh@ut.ac.ir

چکیده : مجسم سازی حجمی[1] داده ها به علت پردازش تمامی واکسل های حجم بسیار زمان بر می باشد. این درحالی است که در کاربرد های پزشکی ، تحقیقاتی ، صنعتی و... پردازش جهت قابل رویت نمودن پارامترها یا اجزا تقریبا باید به صورت آنی باشد. در این مقاله با تکیه بر آلگوریتم برش و پیچش[2] و استفاده از همبستگی فضایی داده های حجمی بدست آمده روشی ارایه می شود که با سرعت مطلوبی قادر به پردازش حجمی داده ها جهت کاربردهای آنی خواهیم بود.

واژه های کلیدی: *مجسم سازی حجمی ، داده ساخت یافته ، عامل ساز برش و پیچش ، بازنمونه برداری*

۱- مقدمه

بررسی داده های حجمی می تواند به صورت مجسم سازی حجمی یا سطحی باشد. مجسم سازی حجمی در مقایسه با روش های بازسازی سطحی به دلیل استفاده ازتمامی واکسل های موجود درمجموعه داده ها بسیارزمانبر می باشند. عملیات هایی نظیر بخش بندی ،انتقال ، دوران ، بزرگنمایی و به طور کلی تبدیل داده های سه بعدی برای مدل سازی وپیاده سازی روش های مجسم سازی سه بعدی داده های حجمی مورد نیاز می باشد[۱][۲]. علاوه بر اینکه روش های نورپراکنی[3] ومحاسبه شفافیت یا کدریِ[4] واکسل های باز نمونه براداری[5] شده درمسیر پرتو نور به حجم محاسبات برای ایجاد تصویر پایه اضافه می شود.

تلاش به منظور سرعت بخشیدن به پردازش حجمی داده ها بر دو مسیر صورت گرفته است. در یک مسیر سعی شده است آلگوریتم های مجسم سازی پایه نظیر DVR[6] تنها با استفاده از یک پردازنده و پیاده سازی بسیار سطح پایین با استفاده ازامکاناتی نظیر حافظه های سریع ، سرعت پردازش بالا ، حافظه نهانی بالا، رجیسترهای MMX[7] و... به سرعت مطلوب برای کاربردهای آنی دست پیدا کنند[۳][۴]. بنابراین در این روش طراحی و پیاده سازی درسطح بسیار پایین پردازنده ها مد نظر می باشد.

در مسیر دیگر سعی شده است آلگوریتم های جدیدی ارایه شود که سرعت پردازش را به گونه ای بالا ببرند که کیفیت تصویر نهایی نسبت به روش های کلاسیک خیلی تغییر نکند. در اینجا آلگوریتم ها به دو دسته "مرتبه شیئی"[8] ودسته "مرتبه تصویری"[9] تقسیم بندی میشود[۵][۶]. آلگوریتمهایی را اصطلاحاً "مرتبه تصویری" می گویند که روش های کلاسیک نورپراکنی در آن ها استفاده می شود. در این آلگوریتم ها تصویر ایجاد شده از تقاطع صفحات فرضی با داده های حجمی همگذاره[10] می شوند. این دست از آلگوریتم ها ، مانندِ آلگوریتم مجسم سازی DVR [۱][۸][۱۱] ، دارای مزایائی نظیر ارایه مدل نوری مناسب و بازنمونه برداری با کیفیت بالا می باشند.

1-Volume Rendering *2-Shear & Warp* *3-Ray Casting* *4-Opacity*
5- Resampeling *6-Direct Volume Rendering* *7-MultiMedia eXtension* *8-Object Order*
9-Image Order *10-Composite*

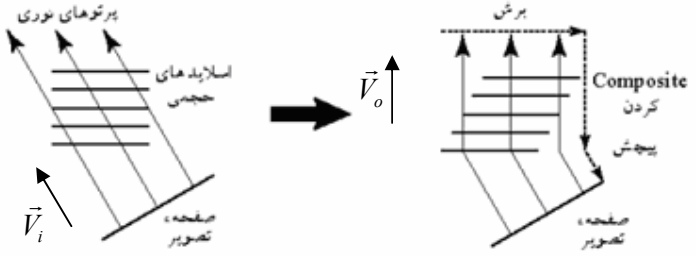
آلگوریتمهایی را "مرتبه شیئی" می نامند که از ساده ترین آدرس دهی برای دسترسی به واکسل ها استفاده میکنند. چرا که به داده های حجمی ، مطابق الگوی ذخیره شدن دسترسی پیدا می کنند. همین باعث شده است که این آلگوریتمها ، مانند "پخش کننده ها"[1] [۱][۲] از داده های ساخت یافته[2] و همبستگی فضائی داده های حجمی بیشترین استفاده را در افزایش سرعت داشته باشند.

آلگوریتمی که پیشنهاد می شود تمام مزایای فوق را در خود دارد. این آلگوریتم به کمک تجزیه عامل انتقال به عامل های پایه برش و اسکیل عمل مجسم سازی حجم را ساده تر می کند. این کار به ما امکان می دهد یک آلگوریتم مرتبه تصویری با همان مزیت های آلگوریتم های مرتبه شیئی ایجاد کنیم[۵]. در این آلگوریتم از داده های ساخت یافته برای تسریع در عملیات مجسم سازی استفاده می کند.

۲- عامل ساز برش و پیچش

به طور کلی هدف از مجسم سازی نگاشت دلخواه از فضای شی به فضای تصویر تعریف میشود که ذاتا حجم محاسباتی بالایی می برد. نیاز به فیلترهای عالی و تصویر کردن حجم در آلگوریتمهای مرتبه شیئی و نیاز به آدرس دهی اضافی در آلگوریتمهای مرتبه تصویری از مشکلات این نگاشت است[۱][۲]. این مشکل با انتقال حجم به یک فضای سیستم مختصات میانی قابل حل است. این سیستم مختصات میانی دارای این خاصیت مهم است که تمام پرتوهای دید کاملا موازی با یکی از محورهای سه گانه مختصات سیستم می باشد. این فضای میانی بویژه در نرم افزار VolPack برای ایجاد تصویر نهایی مورد استفاده قرار گرفته است[۵]. سیستم مختصات میانی را فضای شی برش یافته می نامیم. در روش کلاسیک مجسم نمایی برای ایجاد نما از زاویه دلخواه هر واکسل در امتداد نمای مورد نظر بازنمونه برداری و درون یابی[3] می شود[۱][۷]. در آلگوریتم های مبتنی بر برش و پیچش پس ازانتقال واکسل ها به فضای شی برش یافته واکسل ها با یکدیگراز جلو به پشت همگذاره می شوند و تصویر میانی را ایجاد می کنند. در نهایت تصویر میانی با کمک یک عملگر خطی بر فضای تصویر پایه نگاشت می شود.

شکل ۱ انتقال از فضای شی را به فضای شی برش یافته برای یک پروژکشن موازی نشان می دهد. از آنجائیکه برش اسلایدها با همدیگر موازی می باشد، ما می توانیم این برش را به سادگی با انتقال هرکدام از اسلایدها انجام دهیم.



الف ب

شکل ۱- الف. نورپراکنی اورتوگرافیک در روش کلاسیک ب. باانتقال هریک از اسلاید های حجم واکسل ها در امتداد پرتو های نوری موازی همگذاره می شوند.

ماتریسی که محل واکسلها را در امتداد نمای دلخواه نگاشت می کند M_{view} می نامیم. M_{view} یک ماتریس۴×۴ می باشد که هر یک از نقاط را در فضای شئ به فضای تصویر منتقل می کند و ترکیب خطی از ماتریس های دوران ، جابجایی ، اسکیل و... می باشد[۱]. درآلگوریتم های مبتنی بر برش و پیچشM_{view} به دو ماتریس تجزیه می شود[۵]

$$M_{View} = M_{Warp2D}.M_{Shear3D} \quad (۱)$$

1- Splatters 2-Data Structure 3-Interpolation

$M_{shear3D}$ ماتریسی است که انتقال را از فضای شی به فضای برش یافته ممکن می کند. چنانچه فعلا از ایجاد پرسپکتیو در تصویر نهائی صرف نظر کنیم با توجه به شکل ۱ و با درک ارتباط بردار یکه پرتو $\vec{V}_i$ در فضای شی و بردار یکه پرتو نگاشت یافته $\vec{V}_o$ در فضای شی برش یافته و اینکه این بردار ، موازی یکی از محورهای مختصات (محور Z) [۱] این فضا است ، ماتریس $M_{Shear3D}$ از روی درایه های ماتریس M_{View} قابل محاسبه است :

$$\mathbf{M}_{Shear3D}=\begin{bmatrix}1 & 0 & S_x & 0\\ 0 & 1 & S_y & 0\\ 0 & 0 & 1 & 0\\ 0 & 0 & 0 & 1\end{bmatrix} \qquad (۲)$$

که S_x و S_y برابرند با :

$$s_x=-\frac{\upsilon_{ox}}{\upsilon_{oz}}=\frac{m_{22}m_{13}-m_{12}m_{23}}{m_{11}m_{22}-m_{21}m_{12}} \quad , \quad s_y=-\frac{\upsilon_{oy}}{\upsilon_{oz}}=\frac{m_{11}m_{23}-m_{21}m_{13}}{m_{11}m_{22}-m_{21}m_{12}} \qquad (۳)$$

که m_{ij} درایه های ماتریس M_{View} می باشد. بعد از اینکه داده های حجمی به فضای برش یافته منتقل شدند ، می توانیم اسلایدهای برش یافته شده را با یکدیگر همگذاره کنیم. این عمل باعث ایجاد یک تصویر دو بعدی میانی می شود که باید به کمک ماتریس M_{Warp2D} تصویر میانی به تصویر صحیح نهایی نگاشت می شود. با توجه به رابطه ۱ ماتریس M_{Warp2D} چنین به دست می آید:

$$\mathbf{M}_{Warp2D}=\mathbf{M}_{View}.\mathbf{M}^{-1}_{Shear3D}=\mathbf{M}_{View}.\begin{bmatrix}1 & 0 & -s_x & 0\\ 0 & 1 & -s_y & 0\\ 0 & 0 & 1 & 0\\ 0 & 0 & 0 & 1\end{bmatrix} \qquad (۴)$$

توجه داشته باشیم که ماتریس M_{Warp2D} تنها بر روی یک ماتریس دوبعدی اعمال می شود و نسبت به ماتریس $M_{Shear3D}$ حجم محاسباتی بسیار کمتری دارد.

تصویر کردن حجم داده ها به فضای تصویر میانی چندین خاصیت جغرافیائی یا فضائی ایجاد می کند کـه آلگـوریتم مجسم نمایی را ساده تر می کند[۵].

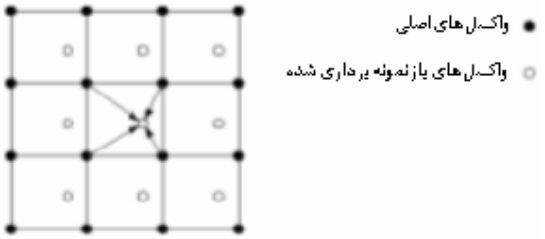
خاصیت اول : با توجه به شکل ۱ خط اسکن تصاویر در تصویر میانی کاملا موازی با خط اسکن واکسل ها در برابر حجم داده ها است.

خاصیت دوم : همه واکسل ها در یک اسلاید از داده های حجمی به همان میزان برش می یابند که میزان برش وابسته به مقدار Z هر اسلاید می باشد. چرا که با توجه به رابطه ۲ ، S_x و S_y مقادیر ثابتی هستند که فقط بر روی مقادیر Z واکسل ها اعمال می شوند.

خاصیت سوم : هر واکسل اسلاید هنگامیکه به تصویر میانی نگاشت می شود، دارای یک مقدار فاکتور اسکیل میباشد و این مقدار به جزء پرسپکتیو ماتریس وابسته است. در این پیاده سازی ، ما این مقدار را برابر با واحد در نظر میگیریم.

بنابراین برای هر خط اسکن داده های حجمی یک نگاشت یک به یک بین داده های حجمی و پیکسل های تصویر میانی صورت می گیرد. در عمل به کمک ماتریس $M^{-1}_{Shear3D}$ داده ها را از فضای شی بازنمونه برداری و درون یابی می کنیم. با توجه به شکل ۲ و خاصیت دوم هر واکسل در یک اسلاید هنگامیکه برش می یابد، دارای یک مقدار ضرایب بازنمونه برداری می باشد. بنابراین می توان یک مجموعه از ضرایب بازنمونه برداری را از قبل محاسبه کرد و برای هر کدام از واکسل های یک اسلاید استفاده کرد. از این ضرایب برای محاسبه کانولوشن فیلتر بازنمونه برداری با واکسل های اسلاید استفاده می شود.

۱- در پیاده سازی به کار رفته در این مقاله $\vec{V}_o$ موازی محور z فرض می شود.



شکل ۲ : مقدار ضرایب بازنمونه برداری برای واکسل های هر اسلاید مقدار ثابتی است

آلگوریتم برش و پیچش برای مجسم سازی داده های حجمی چنین عمل می کند :

در مرحله پیش پردازش :

الف- ایجاد مجموعه ساختاری از واکسلها که هر جزء این ساختار متشکل از مقدار اسکالر هر واکسل ، کدری ، بردار نرمال (به منظور نورپردازی و سایه زنی) و رنگ (در صورت انتخاب کاربر) می باشد. تخصیص رنگ و کدری با توجه به تابع انتقال رنگ و کدری صورت می گیرد. و کاربر می تواند با توجه به هیستگرام حجم و نوع داده حجمی تابع انتقال مناسب را انتخاب کند.

ب- با توجه به محدوده مقدار بردار نرمال و پارامترهای نورپردازی جدولی را برای پیش محاسبه درصد نور رسیده به هر واکسل ایجاد می کنیم. اینکار به منظور حذف محاسبات غیر ضروری است. پارامتر های نورپردازی[۱] ، تعیین تعداد منابع نوری و... در اختیار کار می باشد.

در مرحله پردازش با دخالت کاربر :

ج- ماتریس M_{View} با توجه به زاویه دید کاربر محاسبه و سپس ماتریسهای $M_{Shear3D}$ و M_{Warp2D} بدست می آیند.

د- به کمک ماتریس $M^{-1}_{Shear3D}$ واکسل های فضای برش یافته را از فضای شی بازنمونه برداری و درون یابی می کنیم. ضرایب فیلتر درون یابی برای هر اسلاید با توجه به میزان جابجایی آن محاسبه و برای همه واکسل های اسلاید به کار می رود.

ه- همه اسلایدهای بازنمونه برداری شده بایکدیگر به کمک جدول نورپردازی و تخمین نور رسیده به هر واکسلِ اسلاید به صورت جلو به پشت و با کمک اپراتور Over[۱] همگذاره میشوند[۱][۹]. در این مرحله داده های حجمی به یک تصویر دو بعدی میانی در فضای شئ برش یافته منتقل میشوند.

و- در این مرحله تصویرنهایی به کمک ماتریس M^{-1}_{Warp2D} از تصویر میانی بازنمونه برداری و درون یابی می شود.

ز- تکرار مراحل ج۰ تا و. در صورت تغییر نمای دید از طرف کاربر.

آلگوریتم در محیط برنامه نویسی Visual C++6 پیاده سازی شد. حلقه اصلی این آلگوریتم به صورت زیر طراحی و کدنویسی شده است :

برای هر Z_i=1 تا عمق حجم داده ها {

- محاسبه ضرایب فیلتر بازنمونه برداری برای اسلاید Z_i
- برای هر Y_i=1 تا ارتفاع تصویر میانی {

برای هر X_i=1 تا پهنای تصویر میانی {

(Xi , Yi) بازنمونه برداری = واکسل بازنمونه برداری

[جزء نرمال<- واکسل بازنمونه برداری]جدول_نورپردازی × جزء اسکالر<- واکسل بازنمونه برداری=نوررسیده به واکسل

(نوررسیده به واکسل , [Xi , Yi]پیکسل_تصویرمیانی) همگذاره = [Xi , Yi]پیکسل_تصویرمیانی

{

{

{

۱- عملگر *Over* به صورت $I_a Over I_b = I_a + (1-\alpha_a).I_b$ تعریف می شود که α_a مقدار کدری واکسلی است که شدت I_a به آن می رسد.

تابع "بازنمونه برداری" به کمک ماتریس $M_{Shear3D}^{-1}$ و فیلتر بازنمونه برداری که از درون یابی خطی استفاده می کند واکسل متناظر را از حجم داده ها بازنمونه برداری و مقدار آن را برمیگرداند. یادآوری می کنیم واکسل مورد بررسی ساختاری از مقدار اسکالر ، کدری ، بردار نرمال و رنگ می باشد. در اینجا مقادیر اسکالر ، بردارنرمال و رنگ درون یابی می شوند. برای درون یابی از روش درون یابی خطی استفاده می شود. روشهای دیگر درون یابی نظیر کانولوشن مکعبی ، BSpline ها نیز مورد استفاده قرار می گیرد[۱۰] .

تابع "همگذاره" مقادیر پیکسل تصویرمیانی و نور رسیده به واکسل را به کمک اپراتور Over همگذاره می کند. پوسته بیرونی این حلقه بر روی اسلاید های حجم تکرار می شود. دو حلقه بعدی داخلی بر روی پیکسل های اسلاید هاعمل می کند. توجه داشته باشیم که این آلگوریتم از یک فیلتر باز نمونه برداری دو بعدی استفاده می کند که این خود حجم محاسبات را نسبت به آلگوریتمهای قبلی در مرحله بازنمونه برداری کمتر می کند. نکته دیگر در مورد این آلگوریتم این است که حلقه بیرونی این آلگوریتم بر روی اسلایدهای داده های حجمی عمل می کند. در واقع حجم را به صورت یک آلگوریتمهای مرتبه شیئ طی میکند اما حلقه های درونی کاملا مشابه آلگوریتمهای مرتبه تصویری عمل می کند. یعنی به حجم داده ها مطابق آلگوریتمهای مرتبه شیئ دسترسی می یابیم و در عین حال از فیلترهای بازنمونه برداری که در آلگوریتمهای مرتبه تصویری پیاده سازی می شود، استفاده می کنیم. تنها تفاوت آن ، این است که بجای استفاده از فیلترهای بازنمونه برداری سه بعدی ، فیلترهای مورد استفاده فیلترهای بازنمونه برداری دو بعدی هستند که تاثیر کمی بر روی کیفیت تصویر نهائی می گذارد .

۳- سرعت بخشی به آلگوریتم برش و پیچش

۳-۱- رمز کننده طول عملیاتِ[۲] تصویر میانی

خاصیت یک از عامل ساز برش و پیچش به ما این امکان را می دهد ، هنگامیکه واکسل های حجم اسکن می شوند ، پیکسل های تصویر میانی نیز به صورت موازی اسکن شده و چنانچه پیکسلی بیش از ترشولد تعین شده کدر شده بود از آن پیکسل و واکسل های متناظر آن صرفنظر شود. این خاصیت "پایان دهی زودرس پرتو"[۱] نامیده می شود. آفستهایی که با پیکسل های کدر در تصویر میانی ذخیره می شوند این امکان را فراهم می کند که خاصیت"پایان دهی زودرس پرتو" براحتی پیاده سازی شود[۱][۵].

هدف از داده ساخت یافته برای تصویر میانی عبور از واکسل هایی است که قابل رویت نیستند (نه بخاطر شفاف بودن)، البته این داده ساخت یافته در حین عملیات مجسم سازی دائما تغییر داده می شود. هر پیکسل از تصویر میانی شامل رنگ، کدری و آفست در نظر گرفته می شود. آفست تعداد پیکسل هایی که باید عبور داده شود تا به گره بعدی برسیم را نشان می دهد (شکل ۳). آلگوریتم به این صورت عمل می کند که در ابتدا پیکسل های تصویر میانی همگی به صورت شفاف مقداردهی اولیه می شود. در هنگام اجرای آلگوریتم قبل از اینکه واکسلی بازنمونه برداری یا همگذاره شود آفست پیکسل متناظر تصویر میانی چک می شود چنانچه آفست پیکسل غیر صفر بود یعنی پیکسل کدر می باشد و بنابراین آلگوریتم نیازمند یک عملیات FIND برای پیدا کردن گره غیر کدر می باشد، ولی چنانچه مقدار آفست صفر بود و واکسل غیر شفاف بود عملیات همگذارگی تکمیل می شود. حال دوباره پیکسل ها بررسی می شود و چنانچه کدریِ پیکسلی از مقدار معینی بیشتر شد آفست ها ، مقدار دهی دوباره می شوند.

شکل ۳- چنانچه پیکسلی از تصویر میانی کدر شده بود می توان از آن پیکسل و واکسل های متناظر آن صرف نظرکرد.

1-Early Ray Termination *2-Run Length Encoding*

پیکسلی کدر تعریف می شود که کدری نسبت داده شده از ترشولد تعیین شده بیشتر باشد. واکسلی شفاف تعریف می شود که کدری نسبت داده شده از ترشولد تعیین شده کمتر باشد و ترشولد ها توسط کاربر قابل تغییر است . ولی معمولاً تغییرزیادی در مقدار ترشولد ها داده نمی شود.

۳-۲- رمز کننده طول عملیاتِ حجم داده ها

این داده ساخت یافته در مرحله پیش پردازش قابل محاسبه و ذخیره است و این امکان را فراهم می کند که از واکسل های شفاف به خوبی عبور کنیم و اجازه پردازش آنها طی فرآیند مجسم سازی به آنها داده نشود. در هنگام مجسم سازی ، ما از این داده ساخت یافته متوجه خواهیم شد که کدام قسمت از داده های حجمی شفاف هستند.

یک رمز کننده طول عملیات از یک سری عملیات هایی تشکیل شده است که بر روی یک خط اسکن از داده ها اعمال می شود. برای هر خط اسکن شده از داده ها اطلاعاتی نظیر تعداد واکسل های شفاف و غیر شفاف ، مقدار واکسل های غیر شفاف و محل آنها در این داده ساخت یافته ذخیره می شود.

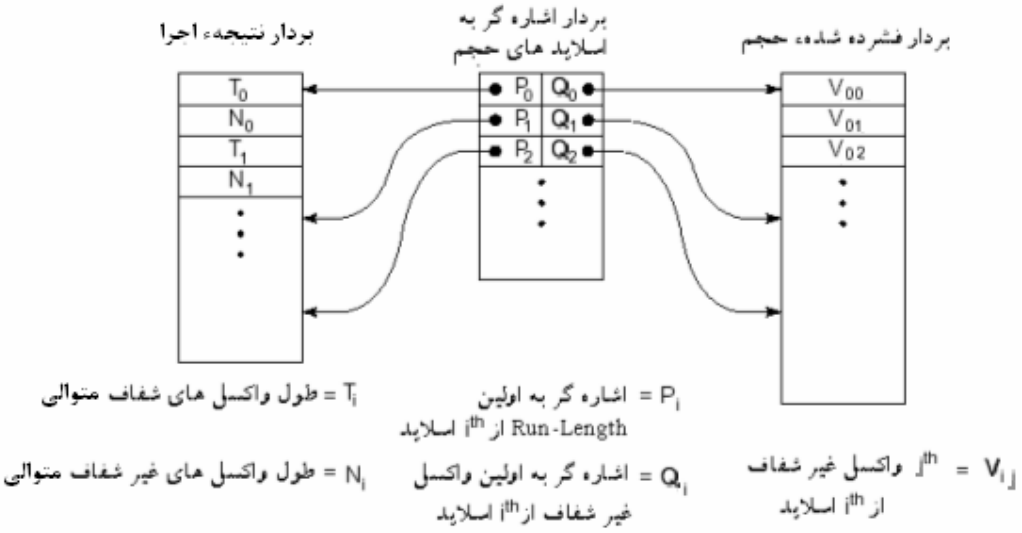
در طی ایجاد داده ساخت یافته رمز کننده طول عملیاتِ حجم ، سه بردار مقدار دهی می شود:

۱- یک بردار شامل تعداد واکسل های شفاف و غیر شفاف در هر خط اسکن

۲- یک بردار شامل فقط واکسل های غیر شفاف

۳- یک بردار که شامل اشاره گرهایی به اولین نتیجه اجرا در هر اسلاید از حجم اصلی ، در دو بردار اولی می باشد.

شکل ۴ چگونگی ایجاد این داده ساخت یافته را نشان می دهد.



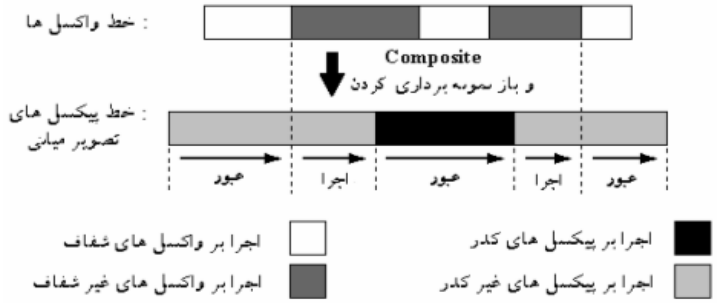
شکل ۴ : آرایه هایی که هنگام ایجاد داده ساخت یافته مربوط به حجم ایجاد می شوند.

به منظور دسترسی تصادفی به واکسل های غیر شفاف و نتیجه اجرا های یک اسلاید ، بردار سوم ایجاد شده است. نکته دیگر در ایجاد این داده ساخت یافته این است که اگر طول هر درایه از بردار اول را ۸ بیت فرض کنیم و چنانچه طول واکسل های شفاف یک خط اسکن از ۲۵۵ (بشترین مقدار درایه بردار) بیشتر شد در این صورت مقدار واکسل های غیر شفاف را صفر و مابقی از واکسل های شفاف اجرای قبلی را در دو درایه بعدی قرار می دهیم. همچنین فراموش نکنیم هر اجرا فقط بر روی یک خط اسکن از داده های حجمی صورت می گیرد.

بردار دوم فقط شامل واکسل های غیر شفاف است. در اینجا به نوعی با حذف واکسل های شفاف ، فشرده سازی صورت گرفته است. چراکه این واکسل ها در فرایند مجسم سازی داخل نمی شوند.

۴- آلگوریتم سریع برش و پیچش

با این دو داده ساخت یافته و خاصیت اول از عامل ساز برش و پیچش ، آلگوریتم مجسم سازی بسیار سریعی متولد می شود که بر اساس یک خط اسکن از داده های حجمی عمل می کند.



شکل ۵ : باکمک داده های ساخت یافته که همبستگی فضایی واکسل ها و پیکسل ها را نشان می دهد می توان از نواحی که شفاف و قابل رویت نیستند عبور کرد.

با توجه به شکل ۵ آلگوریتم داده های حجمی و پیکسل های تصویر میانی را به طور همزمان اسکن می کند و با کمک داده های ساخت یافته رمز کننده طول عملیاتِ حجم و تصویر میانی ، تنها واکسل هایی پردازش می شوند که *غیر شفاف* و *قابل رؤیت* هستند. در نظر داشته باشیم خاصیت دو و سه از عامل ساز برش و پیچش به ما اجازه می دهد مرحله بازنمونه برداری واکسل های بازمانده در هر اسلاید دارای ضرایب یکسانی برای باز نمونه برداری باشند.

پس عبور از مراحل پیش پردازش بیان شده در بند ۲ ، ایجاد و ذخیره رمز کننده طول عملیاتِ حجم ، محاسبه معکوس ماتریس های برش و پیچش و مقدار دهی اولیه متغیرها ساختار حلقه اصلی آلگوریتم سریع برش و پیچش به صورت زیر طراحی و کدنویسی شده است.

برای هر Z_i=1 تا عمق حجم داده ها {

- محاسبه ضرایب فیلتر بازنمونه برداری برای اسلاید Z_i
- برای هر Y_i=1 تا ارتفاع تصویر میانی {

برای هر X_i=1 تا پهنای تصویر میانی {

- اگر ( )==FALSE (تابع_عبور)

}

بازنمونه برداری (Xi , Yi) = *واکسل بازنمونه برداری*

[*جزء نرمال*<- *واکسل بازنمونه برداری*]*جدول_نورپردازی* × *جزء اسکالر*<- *واکسل بازنمونه برداری*=*نوررسیده به واکسل*

(*نوررسیده به واکسل* , [Xi , Yi]*پیکسل_تصویرمیانی*) *همگذاره* = [Xi , Yi]*پیکسل_تصویرمیانی*

{

- به روز در آوردن آفست های خط پیکسل تصویر میانی

{

{

{

تابع عبور با اسکن و دسترسی به رمز کننده طول عملیات حجم و تصویر میانی مقداری برمیگرداند که به کمک آن فقط واکسل های *غیر شفاف* و *قابل رؤیت* پردازش می شوند و از پردازش واکسلهای شفاف و واکسلهایی که در پشت اشیاء کدر دیده نمی شوند جلوگیری می کند.

۵- پیچش تصویر میانی

تصویری که در مرحله همگذارگی ایجاد شده است نیاز به پیچش برای ایجاد تصویر درست و نهایی دارد. پیاده سازی عملیات پیچش کاملا مشخص است واز نگاشت معکوس ویک فیلتر بازنمونه برداری خطی برای اینکار استفاده می شود.

۶-جمع بندی و نتیجه گیری

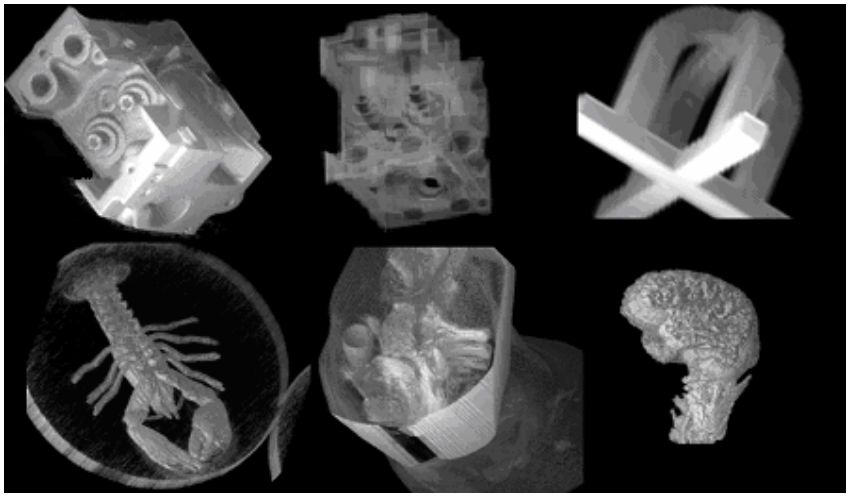
آلگورینم برش و پیچش به کمک خواصی که دارد توانسته است بدون اینکه از کیفیت تصویر نهایی بکاهد سرعت مجسم سازی را چند برابر کند. در این مقاله با بررسی و ارایه روشی مبتنی بر عامل ساز برش وپیچش و بدون نیاز به پیاده سازی بسیار سطح پایین به سرعت مطلوب جهت کاربرد های تقریباً آنی دست می یابیم. دراین روش با استفاده از داده های ساخت یافته ای نظیر رمز کننده طول عملیاتِ حجم وتصویر ، فقط قسمتهایی از حجم داده ها مورد پردازش قرار می گیرند که از نظر تابع انتقال غیر شفاف و قابل رویت باشد. این داده های ساخت یافته به ما کمک می کنند که پیوستگی و یا همبستگی بین داده های فضائی استخراج شود. علاوه براین در مرحله پیش پردازش با ایجاد مجموعه ساختاری از واکسلها که هر جزء این ساختار متشکل از مقدار اسکالر هر واکسل ، کدری ، بردار نرمال و رنگ(در صورت انتخاب کاربر) می باشد و نیز جدول نورپردازی ، از حجم محاسبات حلقه اصلی آلگوریتم کاسته شده است. آلگوریتم برش و پیچش در محیط برنامه نویسی Visual C++6 طراحی و پیاده سازی شده است.

در جدول ۱ مقایسه زمانی پیاده سازی آلگوریتم برش و پیچش تسریع یافته بر روی چند مجموعه داده نشان شده است. زمان اجرا مربوط به یک کامپیوتر PIII 733MHz خانگی با یک پردازنده اینتل نیم کَش ، ۲۵۶مگا بایت حافظه SDRAM ، هارد دور پایین و ۳۲ مگابایت حافظه کارت گرافیک می باشد. ترشولد کدری تصویر میانی ۹۵/. و ترشولد شفافیت واکسل ۰/۰۵ در نظر گرفته شده است. برای مقایسه بهتر زمانی از تخصیص رنگ به واکسل ها صرف نظر شده است.

جدول ۱- مقایسه زمانی پیاده سازی آلگوریتم برش و پیچش تسریع یافته بر روی چند مجموعه داده

مجموعه داده	اندازه (واکسل)	آلگوریتم DVR کلاسیک (sec.)	آلگوریتم برش و پیچش بدون استفاده از داده ساخت یافته (sec.)	آلگوریتم برش پیچش با کمک داده ساخت یافته (msec.)
قفل	۲۵۶×۲۵۶×۱۱۰	۹/۳۵	۲/۸۵	۳۷۰
مغز	۲۵۲×۲۵۲×۵۷	۳/۵۶	/۹۷	۱۴۷
شکم	۲۵۶×۲۵۶×۲۹۹	۲۲/۹۷	۶/۴۳	۹۸۱
خرچنگ دریای	۳۲۰×۳۲۰×۳۴	۳/۱۲	۱/۰۴	۱۳۳

شکل های ۶ برخی نتایج آلگوریتم برش و پیچش را بر روی مجموعه داده های جدول ۱ نشان می دهد.



شکل ۶ – نتایج آلگوریتم برش و پیچش را بر روی مجموعه داده های مختلف نشان داده شده است.
از تابع انتقال لِوّی برای تخصیص کدری استفاده شده است.

۷- مراجع

[1] B.Lichtenbelt, R.Crane,Shaz Naqvi: Introduction to volume rendering ,HP Company

[2] Rafael C. Gonzalez , Richard E. Woods : Digital Image Processing ,1993

[3] N.V Kazakova,M. Margala :A VHDL Implementation of a Shearing Unit for Shear Warp Factorization Volume Renderinf , IEEE Transaction on, 2000

[4] G. Knittel : The ULTRAVIS System , HP Laboratories,visual computing department

[5] Lacroute, P. & Levoy, M. [1994] : Fast volume rendering using a shear-warp factorization of the viewing transformation , Computer Graphics Proceedings , Annual ConferenceSeries (SIGGRAPH '94), Orlando, pp. 451–458.

[6] Jane Wilhelms , Allen van Gelder , Paul Tarantino, Jonathan Gibbs : Hierarchical and parallelizable Direct Volume Rendering for irregular and multiple grids , Computer science dept. University of California ,0-7803-3707-7/96 IEEE 1996

[7] Helwig Hauser, Lukas Mroz : Two-Level Volume Rendering , IEEE Transactions on Visualization and Computer Graphics,Vol 7, NO. 3, July 2001

[8] T. Porter & T. Duff : Compositing Digital Images , Computer Graphics Volume 18, Number 3 July 1984 pp 253-259

[9] J. Kajiya : The Rendering Equation , Computer Graphics Vol. 20 No. 4 (August 1986) pp. 143 - 150.

[10] G.Nielson, J.Tvedt : comparing methods of interpolation for scattered volumetric data"

[11] Jane Wilhelms and Allen Van Gelder :A coherent projection approach for direct volume rendering , Computer Graphics Vol. 25 No. 4 (July 1991) pp. 275 – 284.