



دانشگاه صنعتی امیرکبیر  
(پلی تکنیک تهران)  
دانشکده مهندسی پزشکی

پایان نامه کارشناسی  
گرایش بیوالکتریک

شبیه سازی واقعیت مجازی تعامل ابزار با بافت جهت مطالعه ی روند  
یادگیری اعمال نیروی دلخواه

نگارش  
رضا کریمزاده

استاد راهنما  
دکتر حامد آذرنوش

## صفحه فرم ارزیابی و تصویب پایان نامه - فرم تأیید اعضاء کمیته دفاع

در این صفحه فرم دفاع یا تأیید و تصویب پایان نامه موسوم به فرم کمیته دفاع - موجود در پرونده آموزشی - را قرار دهید.

### نکات مهم:

نگارش پایان نامه/رساله باید به **زبان فارسی** و بر اساس آخرین نسخه دستورالعمل و راهنمای تدوین پایان نامه های دانشگاه صنعتی امیرکبیر باشد. (دستورالعمل و راهنمای حاضر) رنگ جلد پایان نامه/رساله چاپی کارشناسی، کارشناسی ارشد و دکترا باید به ترتیب مشکی، طوسی و سفید رنگ باشد.

چاپ و صحافی پایان نامه/رساله بصورت **پشت و رو (دورو)** بلامانع است و انجام آن توصیه می شود.

اینجانب رضا کریمزاده متعهد می‌شوم که مطالب مندرج در این پایان نامه حاصل کار پژوهشی اینجانب تحت نظارت و راهنمایی اساتید دانشگاه صنعتی امیرکبیر بوده و به دستاوردهای دیگران که در این پژوهش از آنها استفاده شده است مطابق مقررات و روال متعارف ارجاع و در فهرست منابع و مآخذ ذکر گردیده است. این پایان نامه قبلاً برای احراز هیچ مدرک هم‌سطح یا بالاتر ارائه نگردیده است.

در صورت اثبات تخلف در هر زمان، مدرک تحصیلی صادر شده توسط دانشگاه از درجه اعتبار ساقط بوده و دانشگاه حق پیگیری قانونی خواهد داشت.

کلیه نتایج و حقوق حاصل از این پایان نامه متعلق به دانشگاه صنعتی امیرکبیر می‌باشد. هرگونه استفاده از نتایج علمی و عملی، واگذاری اطلاعات به دیگران یا چاپ و تکثیر، نسخه‌برداری، ترجمه و اقتباس از این پایان نامه بدون موافقت کتبی دانشگاه صنعتی امیرکبیر ممنوع است. نقل مطالب با ذکر مآخذ بلامانع است.

رضا کریمزاده

امضا

## تقدیم به :

ضمن تشکر و سپاس بی کران و در کمال افتخار تقدیم می‌نمایم به :

- محضر پدر و مادر عزیزم به خاطر همه‌ی تلاش‌های محبت‌آمیز که در تمامی دوران زندگی‌ام انجام داده‌اند.
- به دوستان فرزانه و فرهیخته‌ای که در راه کسب علم و معرفت مرا یاری نمودند.
- به اساتید گرامی و گرانقدری که به بنده آموزش دادند.

## تقدیر و تشکر

با تشکر فراوان از

خداوند متعال که هرچه داریم از اوست.

از استاد گرانقدرم، دکتر آذرنوش که در انجام این پروژه مرا یاری نمودند.

از استاد راهنمای گرامی، دکتر جعفری که در این چند سال هادی و راهنمای بنده در این راه بودند.

## چکیده

یکی از ابزارهایی که در آموزش و ارزیابی پزشکان و جراحان در سال‌های اخیر بسیار استفاده می‌شود شبیه‌سازهای عمل‌های جراحی است. با معرفی بحث واقعیت مجازی و اضافه کردن تکنولوژی هپتیک به این زمینه، امکان توسعه‌ی هرچه بهتر شبیه‌سازهای عمل جراحی فراهم شد. در این پروژه، کوشیده‌ایم تا با استفاده از تکنولوژی هپتیک و توسعه‌های نرم‌افزاری یک شبیه‌ساز عمل جراحی مغز طراحی شود. در نهایت با استفاده از امکاناتی که در شبیه‌ساز ارائه شد یک آزمایش طراحی شد سپس با انجام این آزمایش بر روی داوطلبان یک بررسی آماری بر روی روند یادگیری کاهش خطا در حین عمل جراحی صورت گرفت. در نهایت با تحلیل نتایج به دست آمده تاثیر مثبت شبیه‌ساز عمل جراحی بر روی افزایش مهارت کاربران بررسی شد.

## واژه‌های کلیدی:

واقعیت مجازی، هپتیک، شبیه‌ساز جراحی مغز، CHAI<sup>۳</sup>D، منعقدساز دوقطبی، روند یادگیری

۱ فصل اول مقدمه.....	۱
۱-۱ بیان مساله.....	۲
۱-۲ معرفی پروژه.....	۳
۱-۳ ابزارهای مورد استفاده در این پروژه.....	۵
۱-۴ شرکت کنندگان در مطالعه.....	۶
۱-۵ ساختار پایان نامه.....	۷
۲ فصل دوم مروری بر مفاهیم کلی هپتیک و پیشینه ی موضوع.....	۸
۲-۱ واقعیت مجازی.....	۹
۲-۱-۱ انواع واقعیت مجازی.....	۹
۲-۲ معرفی هپتیک و نحوه ی عملکرد.....	۱۲
۲-۲-۱ واژه شناسی.....	۱۲
۲-۲-۲ مفهوم کنونی.....	۱۲
۲-۲-۳ تاریخچه.....	۱۳
۲-۲-۴ نحوه ی عملکرد دستگاه هپتیک.....	۱۶
۲-۳ انواع دستگاه هپتیک.....	۱۸
۲-۳-۱ گیرهی منعکس کننده ی نیرو.....	۱۸
۲-۳-۲ ماوس بازخورد نیروی Logitech Wingman.....	۱۹
۲-۳-۳ دستگاه هپتیک Geomagic Touch.....	۲۰
۲-۳-۴ دستکش Cyber Grasp.....	۲۲
۲-۳-۵ غوطه وری در هپتیک.....	۲۳
۲-۴ کاربردهای هپتیک.....	۲۳
۲-۴-۱ بازی های رایانهای.....	۲۳
۲-۴-۲ ربات های دورکار.....	۲۴
۲-۴-۳ پزشکی.....	۲۴
۲-۴-۴ نظامی.....	۲۴
۲-۴-۵ آموزش.....	۲۵
۲-۵ محدودیت های هپتیک.....	۲۵
۲-۶ مروری بر سوابق پیشین.....	۲۵
۲-۷ جمع بندی.....	۲۶
۳ فصل سوم ساختار نرم افزار استفاده شده.....	۲۷
۳-۱ معرفی کتابخانه ی مورد استفاده.....	۲۸

۲۹	۲-۳- امکانات ارایه شده در CHAI۳D
۲۹	۱-۲-۳- شناسایی دستگاه هپتیک
۳۰	۲-۲-۳- دنیای مجازی
۳۲	۳-۲-۳- دوربین های نمایش محیط مجازی
۳۴	۴-۲-۳- نور و سایه
۳۵	۱-۴-۲-۳- نور محیطی
۳۶	۲-۴-۲-۳- نور پخش شونده
۳۶	۳-۴-۲-۳- نور اختصاصی
۴۱	۵-۲-۳- اشیای متشکل از مش
۴۳	۶-۲-۳- الگوریتم تشخیص برخورد
۴۴	۷-۲-۳- اجسام ساده‌ی اولیه
۴۵	۸-۲-۳- ابزارک ها
۴۷	۹-۲-۳- حلقه ی هپتیک
۴۸	۳-۳- الگوریتم و روشهای مورد استفاده
۴۹	۱-۳-۳- پیش پردازش
۵۴	۲-۳-۳- طراحی مدل ابزار منعقدساز دو قطبی
۵۶	۳-۳-۳- تغییر شکل بافت
۵۹	۴-۳-۳- مدل نیرویی بافت شبیه سازی شده
۶۲	۵-۳-۳- مدل منعقدساز رگ
۶۴	۶-۳-۳- محیط کلی شبیه سازی شده
۶۶	۴-۳- جمع بندی
۶۷	<b>۴ فصل چهارم تعریف آزمایش و تحلیل نتایج آن بر روی داوطلبان</b>
۶۸	۱-۴- طراحی آزمایش و داده گیری از داوطلبان
۶۸	۱-۱-۴- تعریف
۶۹	۲-۱-۴- روند آزمایش
۷۱	۳-۱-۴- شرایط و محیط انجام آزمایش
۷۱	۴-۱-۴- ذخیره سازی داده ها
۷۱	۵-۱-۴- شرکت کنندگان در آزمایش
۷۲	۲-۴- تحلیل نتایج بر روی داوطلبان
۷۲	۱-۲-۴- معرفی معیارهای سنجش عملکرد
۷۵	۲-۲-۴- تحلیل نتایج کلیه ی داوطلبان
۷۷	۱-۲-۲-۴- طول مسیر حرکت
۷۸	۲-۲-۲-۴- زمان انجام هر آزمایش
۷۹	۳-۲-۲-۴- میانگین نیرویی وارد شده به بافت
۸۰	۴-۲-۲-۴- انحراف معیار نیروی وارد شده
۸۱	۵-۲-۲-۴- درصد خطای نیروی اعمالی



۸۲.....	۶-۲-۴، درصد خطای منعقدسازی بافت مغزی .....
۸۳.....	۷-۲-۴، درصد خطای رگ منعقد نشده .....
۸۴.....	۸-۲-۴، میانگین میزان منعقد سازی .....
۸۵.....	۹-۲-۴، انحراف معیار میزان منعقد سازی .....
۸۶.....	۱۰-۲-۴، سایر تحلیلها .....
۸۹.....	۱۱-۲-۴، نتیجه گیری .....
۸۹.....	۳-۲-۴- تحلیل نتایج زیر گروه ها .....
۸۹.....	۱-۳-۲-۴، طول مسیر حرکت .....
۹۰.....	۲-۳-۲-۴، زمان انجام هر آزمایش .....
۹۱.....	۳-۳-۲-۴، میانگین نیروی وارد شده به بافت .....
۹۲.....	۴-۳-۲-۴، انحراف معیار نیروی وارد شده .....
۹۳.....	۵-۳-۲-۴، درصد خطای نیروی اعمالی .....
۹۴.....	۶-۳-۲-۴، درصد خطای منعقدسازی بافت مغزی .....
۹۵.....	۷-۳-۲-۴، درصد خطای رگ منعقد نشده .....
۹۶.....	۸-۳-۲-۴، میانگین میزان منعقد سازی .....
۹۷.....	۹-۳-۲-۴، انحراف معیار میزان منعقد سازی .....
۹۸.....	۱۰-۳-۲-۴، سایر تحلیل ها .....
۱۰۱.....	۱۱-۳-۲-۴، نتیجه گیری .....
۱۰۱.....	۳-۴- ارزیابی شبیه ساز توسط داوطلبان.....
۱۰۲.....	۴-۴- جمع بندی.....
۱۰۳.....	<b>۵ فصل پنجم جمع بندی و نتیجه گیری و پیشنهادها.....</b>
۱۰۴.....	۱-۵- پیشنهادها.....
۱۰۶.....	<b>منابع و مراجع.....</b>
۱.....	<b>پیوست ها.....</b>

## صفحه

## فهرست اشکال

شکل ۱-۱ شبیه ساز پرواز.....	۴
شکل ۲-۱ ابزار منعقد ساز خونریزی دوقطبی.....	۵
شکل ۳-۱ دستگاه Geomagic Touch.....	۶
شکل ۱-۲ دستگاه HMD.....	۱۰
شکل ۲-۲ واقعیت مجازی در صفحه‌ی نمایش.....	۱۰
شکل ۳-۲ دور حضوری.....	۱۱
شکل ۴-۲ شبیه ساز پرواز.....	۱۱
شکل ۵-۲ GROPE.....	۱۴
شکل ۶-۲ GROPE-III.....	۱۴
شکل ۷-۲ سیستم Sandpaper.....	۱۵
شکل ۸-۲ دستگاه Exoskeleton.....	۱۶
شکل ۹-۲ تعامل کاربر و محیط مجازی.....	۱۷
شکل ۱۰-۲ بلوک دیاگرام سیستم هپتیک.....	۱۸
شکل ۱۱-۲ گیرهی منعکس کننده‌ی نیرو.....	۱۹
شکل ۱۲-۲ ماوس بازخورد نیروی Logitech Wingman.....	۲۰
شکل ۱۳-۲ دستگاه Touch.....	۲۰
شکل ۱۴-۲ ویژگی‌های دستگاه‌های Touch.....	۲۱
شکل ۱۵-۲ دستکش Cyber Grasp.....	۲۲
شکل ۱۶-۲ غوطه وری در هپتیک.....	۲۳
شکل ۱-۳ محور مختصاتی تعریف شده برای دستگاه هپتیک.....	۳۰
شکل ۲-۳ گرهای والد و زیر مجموعه‌هایشان.....	۳۱
شکل ۳-۳ محیط مجازی ساخته شده با CHAI۳D.....	۳۱
شکل ۴-۳ صفحات برش دور و نزدیک و چگونگی نمایش صحنه.....	۳۳
شکل ۵-۳ تاثیر نور محیطی روی یک جسم سه بعدی.....	۳۵
شکل ۶-۳ تاثیر نور پخش شونده روی جسم سه بعدی.....	۳۶
شکل ۷-۳ تاثیر نور اختصاصی بر روی جسم سه بعدی.....	۳۷
شکل ۸-۳ ترکیب سه نوع نور (تصویر اول از سمت راست مجموع سه تصویر دیگر است).....	۳۷
شکل ۹-۳ منبع نقطه‌ای نور.....	۳۸
شکل ۱۰-۳ منبع نور جهتدار.....	۳۹
شکل ۱۱-۳ نورافکن.....	۴۰

شکل ۳-۱۲	جسم متشکل از ورتکس و مثلث	۴۱
شکل ۳-۱۳	جسم حجم سنجی شده	۴۴
شکل ۳-۱۴	اجسام اولیه در CHAI۳D	۴۵
شکل ۳-۱۵	ابزارهای CHAI۳D	۴۶
شکل ۳-۱۶	بافت مغز استفاده شده	۴۸
شکل ۳-۱۷	تصویر تغییر اندازه داده شده	۵۰
شکل ۳-۱۸	یک تصویر باینری دو قطعه‌ای	۵۰
شکل ۳-۱۹	تصویر قطعه بندی شده	۵۱
شکل ۳-۲۰	رگهای مورد نظر استخراج شده	۵۲
شکل ۳-۲۱	تصویر نهایی برای سه بعدی سازی رگ	۵۳
شکل ۳-۲۲	بافت شبیهسازی شده در محیط مجازی	۵۴
شکل ۳-۲۳	ابزار منعقدساز دو قطبی مدل شده	۵۵
شکل ۳-۲۴	ابزار منعقدساز دو قطبی	۵۵
شکل ۳-۲۵	چارچوب برای قرار گرفتن در اطراف بافت مغزی	۵۶
شکل ۳-۲۶	مدل تغییر شکل بافت در تعامل با ابزار	۵۸
شکل ۳-۲۷	تغییر شکل بافت در تعامل با ابزار منعقد ساز	۵۹
شکل ۳-۲۸	مدل استفاده شده برای محاسبه نیرو	۵۹
شکل ۳-۲۹	منعقد سازی رگ با مدل ارایه شده	۶۴
شکل ۳-۳۰	کلید فعال سازی منعقدکننده و استفاده از منعقدکننده	۶۵
شکل ۳-۳۱	محیط نهایی شبیهساز جراحی مغز	۶۵
شکل ۴-۱	رگ انتخاب شده برای منعقد سازی	۶۹
شکل ۴-۲	گوی قرمز و آبی و متن راهنما برای شروع و پایان	۷۰
شکل ۴-۳	کلید عملگر تعریف شده برای شبیهساز	۷۰
شکل ۴-۴	مسیر حرکت در سه بعد	۷۳
شکل ۴-۵	مسیر حرکت در دو بعد	۷۴
شکل ۴-۶	منحنی نرمال مقادیر T	۷۶
شکل ۴-۷	طول مسیر کل افراد در طی ده آزمایش	۷۸
شکل ۴-۸	مدت زمان انجام آزمایش کل افراد در طی ده آزمایش	۷۹
شکل ۴-۹	میانگین نیروی اعمالی افراد برای ده آزمایش	۸۰
شکل ۴-۱۰	انحراف معیار نیروی وارد شده در ده آزمایش	۸۱
شکل ۴-۱۱	درصد خطای نیروی اعمالی در ده آزمایش	۸۲
شکل ۴-۱۲	میانگین و واریانس درصد خطای سوزاندن بافت مغزی در ده آزمایش	۸۳
شکل ۴-۱۳	میانگین و انحراف معیار درصد خطای رگ منعقد نشده	۸۴

شکل ۴-۱۴	انحراف معیار و میانگین میزان منعقد سازی.....	۸۵
شکل ۴-۱۵	انحراف معیار میزان منعقد سازی.....	۸۶
شکل ۴-۱۶	نمودار شدت نیرویی.....	۸۷
شکل ۴-۱۷	نمودار مجموع نیرویی.....	۸۸
شکل ۴-۱۸	نمودار حرارتی میانگین منعقد سازی.....	۸۸
شکل ۴-۱۹	میانگین و انحراف معیار طول مسیر برای دو گروه.....	۹۰
شکل ۴-۲۰	میانگین و انحراف معیار زمان انجام هر آزمایش برای دو گروه.....	۹۱
شکل ۴-۲۱	میانگین و انحراف معیار میانگین نیروی وارد شده.....	۹۲
شکل ۴-۲۲	میانگین انحراف معیار نیرو برای دو گروه.....	۹۳
شکل ۴-۲۳	میانگین و انحراف معیار درصد خطای نیروی اعمالی.....	۹۴
شکل ۴-۲۴	درصد خطای منعقد سازی بافت مغزی توسط دو گروه.....	۹۵
شکل ۴-۲۵	انحراف معیار و میانگین درصد خطای رگ منعقد نشده برای دو گروه.....	۹۶
شکل ۴-۲۶	میانگین و انحراف معیار میانگین میزان منعقد سازی.....	۹۷
شکل ۴-۲۷	انحراف معیار میزان منعقد سازی.....	۹۸
شکل ۴-۲۸	نمودار شدت نیرویی دو گروه دو نمودار سمت چپ گروه اول- دو نمودار سمت راست گروه دوم.....	۹۹
شکل ۴-۲۹	میانگین نیروی اعمالی برای دو گروه دو نمودار بالا گروه اول- دو نمودار پایین گروه دوم.....	۱۰۰
شکل ۴-۳۰	میانگین منعقد سازی برای دو گروه دو تصویر بالا گروه اول- دو تصویر پایین گروه دوم.....	۱۰۰
شکل ۴-۳۱	امتیاز کسب شده در هر مورد.....	۱۰۲
شکل ۱	دستگاه Geomagic touch.....	۲
شکل ۲	ناحیهی مورد آزمایش.....	۳
شکل ۳	پنجرهی مجازی.....	۳
شکل ۴	دکمه‌ی عملگر.....	۴

## صفحه

## فهرست جداول

جدول ۱-۳ - قطعه کد ساخت دنیا و اضافه کردن منبع نور و دوربین.....	۳۲
جدول ۲-۳ تنظیم موقعیت دوربین و تعریف صفحات برش.....	۳۳
جدول ۳-۳ قطعه کد مربوط به نمایش توسط دوربین.....	۳۴
جدول ۴-۳ قطعه کد مربوط به فعالسازی منبع نور نقطه‌های.....	۳۸
جدول ۵-۳ قطعه کد فعالسازی منبع نور جهتدار.....	۳۹
جدول ۶-۳ قطعه کد فعالسازی نورافکن.....	۴۰
جدول ۷-۳ قطعه کد مربوط به تعریف ورتکس و مثلثها.....	۴۲
جدول ۸-۳ تعریف چند مش به صورت مجزا.....	۴۳
جدول ۹-۳ فعال سازی الگوریتم تشخیص برخورد.....	۴۴
جدول ۱۰-۳ قطعه کد ساخت کره.....	۴۵
جدول ۱۱-۳ ساخت برچسب اطلاعاتی.....	۴۶
جدول ۱۲-۳ حلقه‌ی هپتیکی.....	۴۷
جدول ۱۳-۳ قطعه کد تغییر اندازه‌ی تصویر.....	۴۹
جدول ۱۴-۳ قطعه بندی تصویر.....	۵۱
جدول ۱۵-۳ قطعه کد جدا سازی رگها.....	۵۱
جدول ۱۶-۳ قطعه کد ایجاد بافت در محیط مجازی.....	۵۳
جدول ۱۷-۳ قطعه کد مربوط به تغییر شکل بافت.....	۵۸
جدول ۱۸-۳ قطعه کد مدل سازی نیروی بافت مغزی.....	۶۱
جدول ۱۹-۳ قطعه کد اعمال نیروی اصطکاک.....	۶۱
جدول ۲۰-۳ قطعه کد ایجاد تغییر بصری منعقد ساز دوقطبی.....	۶۳
جدول ۱-۴ شرح کیفی برای مقادیر P.....	۷۷

صفحه

فهرست پیوست

پ- ۱	فرم معرفی پروژه برای داوطلبان.....	۱
پ- ۲	فرم رضایتنامه حضور داوطلب در آزمایش.....	۴
پ- ۳	فرم اطلاعات شخصی داوطلب.....	۶
پ- ۴	فرم ارزیابی داوطلب از آزمایش.....	۷
پ- ۵	کد کامل شبیه ساز عمل جراحی.....	۸
پ- ۶	کد متلب مربوط به قطعه بندی تصویر و جداسازی رگها.....	۶۹
پ- ۷	کد متلب استخراج معیارها برای مقایسه.....	۷۰
پ- ۸	کد متلب مقایسه ی معیارها برای دو گروه.....	۷۹

## فهرست علائم

### علائم لاتین

P مقدار پی در تست آماری تی

E مذبذب یانگ

F نیرو

A سطح مقطع

K ثابت فنر

### علائم یونانی

$\sigma$  تنش

$\epsilon$  کرنش

$\mu$  ضریب اصطکاک

## فصل اول

### مقدمه



## مقدمه

طبق مطالعه‌ای که در یک بازه‌ی سه ساله توسط سازمان رتبه‌بندی بهداشت و درمان آمریکا صورت گرفته است، نشان داده شد که خطاهای پزشکی موجب مرگ بالغ بر ۲۳۰ هزار نفر در بیمارستان‌های آمریکا شده است [۱]. اکثر این اشتباهات که قابل پیشگیری هستند به عنوان وقایع ناشدنی<sup>۱</sup> شناخته می‌شوند که در اثر سهل انگاری و یا استرس و ناآزمودگی جراح اتفاق می‌افتد به عنوان مثال انجام روند اشتباه در عمل جراحی و طی نکردن سلسله مراتب بدیهی عمل و یا جا گذاشتن ابزار و لوازم جراحی در بدن بیمار از نمونه‌های وقایع ناشدنی هستند.

خطاهای عمل جراحی گستره‌ی وسیعی از مشکلات را در بر می‌گیرد، ۵۹ درصد از قربانیان خطا در عمل‌های جراحی از آسیب‌های موقت رنج می‌برند، ۳۳ درصد دچار آسیب‌های دائمی جبران ناپذیر می‌شوند و ۸ درصد به مرگ منجر می‌شود. تعیین تعداد دقیق خطاهای عمل‌های جراحی امکان پذیر نیست زیرا اکثر بیمارستان‌ها از افشای اطلاعات مربوط به خطاهای عمل جراحی خودداری می‌کنند. همچنین بیماران از رخ دادن اکثر خطاها در حین عمل جراحی بی اطلاع هستند. به عنوان مثال اگر در یک جراحی مغز و اعصاب به جای تومور قسمتی از مغز هم برداشته شود و بیمار این مساله را نتواند بعد از عمل تشخیص دهد هیچ شخصی از این خطا آگاه نمی‌شود [۲].

## ۱-۱- بیان مساله

آموزش جراحان و آماده کردن آنها برای رویارویی با عمل جراحی عموماً در دوره‌ی دستیاری بعد از تکمیل تحصیلات ابتدایی پزشکی است. در این دوره کارآموز باید به خوبی با روند انجام عمل جراحی، کنترل استرس و ترس، کنترل لرزش و حرکت دست، اعمال نیروی متناسب با بافت مورد جراحی و ... آشنا شود.

از طرفی جراح متخصص مسوول عمل وقت و انرژی کافی برای انجام عمل و توضیح قسمت‌های مختلف برای دستیار را ندارد لذا یادگیری دستیار در حین عمل به طور میانگین پایین خواهد بود. همچنین در

<sup>۱</sup> Never Events

صورتی که بخواهیم دستیار را مورد ارزیابی قرار دهیم، باید یک بیمار را در اختیار او قرار داده تا عمل روی آن انجام دهد که می‌توان گفت بازی با جان یک فرد است زیرا یک فرد تقریباً ناآزموده را مسؤول یک عمل جراحی کرده‌ایم.

با این اوصاف در این روند، آموزش و ارزیابی توسط استاد جراح به صورت سلیقه‌ای اعمال می‌شود و از یک سیستم یکتا و کارآمد استفاده نمی‌شود و نتیجه‌ی آن بوجود آمدن جراحانی است که از نقاط ضعف خود در عمل جراحی آگاه نیستند و حتی ممکن است این ضعف باعث رخ دادن وقایع ناشدنی شود.

روش‌هایی مانند تمرین بر روی مدل‌های پلاستیکی، حیوانات بیهوش و یا اجساد انسان جهت کسب مهارت وجود دارند، اما مدل‌های پلاستیکی به سختی می‌توانند رفتار مکانیکی بافت‌های زنده را شبیه‌سازی کنند. جسد حیوانات و انسان نیز علاوه بر موانع اخلاقی و قانونی که بر سر راه استفاده از آن‌ها وجود دارد، معمولاً به سادگی و به تعداد مورد نیاز در دسترس نیستند و به دلایلی نظیر تفاوت آناتومی انسان و حیوان و وجود فشار خون در انسان زنده، استفاده از آن‌ها کارایی محدودی دارد [۳].

## ۱-۲- معرفی پروژه

در طول دهه‌های گذشته توجه به ایمنی بیمار در اتاق عمل بر روی عملگرهای پشتیبانی مانند دستگاه‌ها، داروها، کارکنان و روش‌های مدیریتی متمرکز بود و به ندرت به تکنیک‌ها و عملکرد جراح در اتاق عمل توجه می‌شد. اما امروزه تمرکز به سمت آموزش و ارزیابی هرچه بهتر جراح رفته است [۴].

یکی از روش‌های نوین برای آموزش دستیاران پزشکی و به حداقل رساندن صدمات به بیمار استفاده از تکنولوژی واقعیت مجازی<sup>۱</sup> است. واقعیت مجازی را می‌توان تکنولوژی دانست که در آن محیطی غیر از محیط واقعی ولی با مشخصات نزدیک به آن در جلوی چشمان کاربر قرار می‌گیرد و کاربر براساس حرکت سر و اعضای بدن با آن محیط تعامل برقرار می‌کند. در واقع واقعیت مجازی یک تکنولوژی و فناوری نوین است که به کاربر امکان می‌دهد تا با یک محیط شبیه‌سازی رایانه‌ای تعامل<sup>۲</sup> داشته باشد [۱].

<sup>۱</sup> Virtual Reality

<sup>۲</sup> Interaction

شبیه‌سازهای واقعیت مجازی مبتنی بر گرافیک رایانه‌ای تاثیر چشم‌گیری در افزایش سطح مهارت خلبانان داشته‌اند (شکل) و این موفقیت نوید بخش این مساله است که می‌تواند دستاوردی مشابه در حوزه‌ی پزشکی برای پزشکان داشته باشد. در طی ۳ دهه‌ی اخیر برای آموزش دستیاران جراحی استفاده از وسایل مبتنی بر رایانه رشد چشمگیری داشته است که از نمونه‌ی بارز آن می‌توان به استفاده‌ی روز افزون از شبیه‌سازهای واقعیت مجازی اشاره کرد.



شکل ۱-۱ شبیه ساز پرواز

صنعت بهداشت و درمان با وجود موسساتی که از تصاویر کامپیوتری برای تشخیص و درمان بیماری‌ها استفاده می‌کنند، یکی از بزرگ‌ترین استقبال‌کنندگان از تکنولوژی واقعیت مجازی است. شبیه‌سازی‌های واقعیت مجازی می‌توانند تصاویر تشخیصی درست و دقیقی را از اسکن‌ها و مدل‌های سه‌بعدی تهیه شده از آناتومی بیماران را ایجاد کنند. مدل‌های مجازی به پزشکان و جراحان تازه‌کار و باتجربه کمک می‌کنند تا بتوانند امن‌ترین و موثرترین راه برای پیدا کردن محل تومورها را ارائه دهند، ابزارهای جراحی را در جای درست قرار دهند یا امکان تمرین و تست درمان‌ها و عمل‌های جراحی پیچیده را فراهم بیاورند.

از دیگر کاربردهای در حال توسعه‌ی واقعیت مجازی می‌توان به موارد زیر اشاره کرد:

(۱) تفریح، ورزش و بازی

(۲) آموزش و پرورش

(۳) نظامی

(۴) میراث باستانی (بازسازی تمدن‌های تاریخی)

(۵) کسب و کار و مهندسی

با توجه به رشد روز افزون این تکنولوژی در زمینه‌ی پزشکی برآن شدیم که در این پروژه با استفاده از یکی از ابزارهای واقعیت مجازی یک فضای مجازی برای تعامل ابزار جراحی و بافت مغز توسعه دهیم. در این شبیه‌سازی هدف بررسی یادگیری و ارزیابی اعمال نیروی مناسب به بافت مغزی شبیه‌سازی شده است که در این پروژه به صورت خاص ابزار منعقد سازی خونریزی دوقطبی<sup>۱</sup> (شکل ۲) مورد بررسی قرار گرفته است. ابزار منعقد ساز خونریزی دوقطبی در حال حاضر به یک ابزار ضروری برای جراحان تبدیل شده است، این ابزار با استفاده از جریان الکتریسیته گذرنده از یک قطب به قطب دیگر مسیر عبور خون را منعقد می‌کند و از خونریزی حین جدا کردن یک بافت با منعقد کردن تمامی عروق خونی از خونریزی جلوگیری می‌کنند.



شکل ۲-۱ ابزار منعقد ساز خونریزی دوقطبی

### ۳-۱- ابزارهای مورد استفاده در این پروژه

<sup>۱</sup> Bipolar Electrocautery

در این پروژه برای تعامل محیط مجازی و کاربر از دستگاهی به نام هپتیک<sup>۱</sup> با نام تجاری Geomagic Touch که سابقاً با نام Phantom Omni شناخته می‌شد استفاده شده است. این دستگاه ساخت شرکت ۳D Systems است. تعامل این دستگاه با محیط مجازی به صورت بلادرنگ<sup>۲</sup> است و با وصل شدن به رایانه و گرفتن اطلاعات محیط مجازی نیروی اجسام و خواص فیزیکی آن‌ها را به دست کاربر منتقل می‌کند.



شکل ۱-۳ دستگاه Geomagic Touch

این دستگاه دارای یک کتابخانه یه نام CHAI<sup>۳D</sup> که به زبان ++C نوشته شده است و با اضافه کردن آن به محیط کدنویسی می‌توان فضای مجازی را تعریف نمود و از طریق دستگاه با محیط مجازی تعامل داشت. در فصل ۲ به تفصیل در مورد دستگاه‌های هپتیک بحث خواهیم کرد.

## ۴-۱- شرکت‌کنندگان در مطالعه

برای ارزیابی عملکرد شبیه‌ساز طراحی شده و مطالعه روی روند یادگیری داوطلبان، از ۲۰ نفر از دانشجویان دعوت شد تا در صورت تمایل در این مطالعه شرکت کنند.

<sup>۱</sup> Haptic

<sup>۲</sup> Realtime

## ۱-۵- ساختار پایان نامه

در فصل دوم به مفاهیم کلی تکنولوژی هپتیک، نحوه‌ی عملکرد آن، معرفی انواع هپتیک و کاربردهای این تکنولوژی پرداخته خواهد شد. در انتهای فصل سوابق پیشین مرتبط با این پروژه معرفی می‌گردد. در فصل سوم ابتدا به معرفی امکانات نرم‌افزاری هپتیک پرداخته شده‌است. در ادامه فصل با توجه به امکانات معرفی شده برای هپتیک و مدل‌سازی‌هایی که انجام شده است، یک شبیه‌ساز جراحی مغز برای منعقدسازی رگ مغزی طراحی شده‌است. در فصل چهارم برای ارزیابی روند یادگیری افراد در استفاده از این شبیه‌ساز، یک آزمایش طراحی شده است و با ثبت داده از نتایج این افراد در آزمایش، میزان پیشرفت ایشان در کاهش خطای منعقدسازی رگ بررسی گردیده است. در فصل آخر به جمع‌بندی مطالب و در نهایت پیشنهاداتی برای کارهای آینده در این زمینه ارایه شده است.

## فصل دوم

### مروری بر مفاهیم کلی هپتیک و پیشینه‌ی موضوع

## مقدمه

در این فصل قصد داریم نخست مروری بر کلیات و فضای واقعیت مجازی داشته باشیم، سپس تکنولوژی هپتیک و دستگاه استفاده شده در پروژه را شرح دهیم. به این منظور ابتدا به توضیح مفاهیم اولیه و معرفی این زمینه می‌پردازیم، سپس به نحوه‌ی عملکرد دستگاه و انواع مختلف آن اشاره می‌کنیم و در نهایت کاربردهای آن را بررسی می‌کنیم.

## ۱-۲- واقعیت مجازی

واقعیت مجازی یک فضای شبیه‌سازی شده‌ی تعاملی است که به صورت بلادرنگ<sup>۱</sup> با سنسورهای سیستم در ارتباط است و فضایی را برای کاربر ایجاد می‌کند که حس حضور در یک دنیای واقعی را تلقین کند. تعامل بین کاربر و سیستم واقعیت مجازی می‌تواند از یک نمایش سه بعدی ساده تا ایجاد تغییر در دنیای مجازی باشد [۵].

## ۱-۱-۲- انواع واقعیت مجازی:

در کل می‌توانیم کارهایی که در حوزه‌ی واقعیت مجازی صورت می‌گیرد را به ۴ دسته‌ی عمده تقسیم کنیم:

(۱) غوطه‌وری<sup>۲</sup> در محیط مجازی: به طور کامل نقطه‌ی دید کاربر را در محیط مجازی قرار می‌دهد. در این نوع از واقعیت مجازی عموماً از HMD<sup>۳</sup> استفاده می‌شود که بر روی سر قرار می‌گیرد و از قوه‌ی شنوایی و بینایی برای شبیه‌سازی استفاده می‌کند. این دستگاه به عنوان مثال می‌تواند این حس را به کاربر القا کند که در یک غار قرار گرفته‌است.

---

<sup>۱</sup> Real Time

<sup>۲</sup> Immersive

<sup>۳</sup> Head Mounted Display





شکل ۱-۲ دستگاه HMD

(۲) واقعیت مجازی در صفحه‌ی نمایش<sup>۱</sup>: با استفاده از یک صفحه‌ی نمایش و یک عینک سه بعدی ساز تصویر روی صفحه نمایش به صورت سه بعدی مشاهده می‌شود.



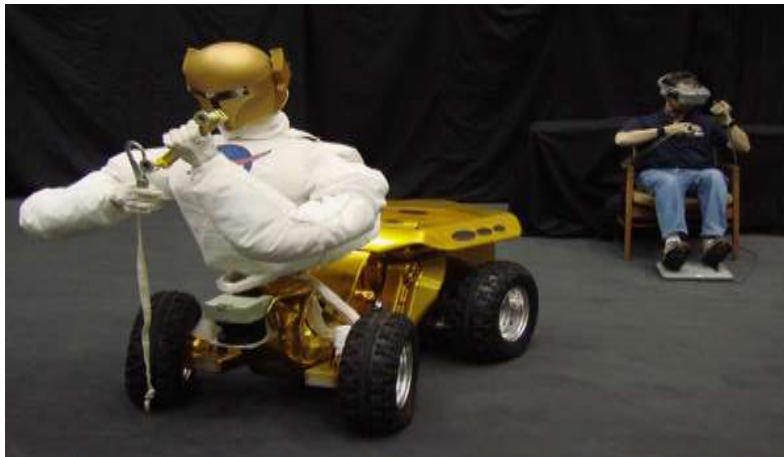
شکل ۲-۲ واقعیت مجازی در صفحه‌ی نمایش

(۳) دور حضوری<sup>۲</sup>: در این تکنولوژی سنسورهای کنترل از راه دور سیستم و حس‌های انسان به کنترل یک ربات در محیط واقعی منجر می‌شود. به عنوان مثال آتش نشان‌ها از این تکنولوژی برای دسترسی به جاهای پرخطر استفاده می‌کنند.

---

<sup>۱</sup> Window on World

<sup>۲</sup> Tele Presence



شکل ۲-۳ دور حضوری

۴) ترکیبی از موارد بالا: به عنوان مثال با ترکیب دور حضوری و غوطه‌وری در محیط مجازی می‌توان یک شبیه ساز پرواز طراحی نمود که فرد با گذاشتن کلاه HMD خود را در موقعیت یک خلبان می‌بیند و با توانایی اعمال تغییر به واسطه‌ی سیستم دور حضوری، می‌تواند هواپیما مجازی یا واقعی خود را کنترل کند.



شکل ۲-۴ شبیه ساز پرواز

همانطور که از توضیحات بالا دریافتیم واقعیت مجازی به صورت روز افزون در حال توسعه است، این توسعه باعث نزدیک شدن هرچه بیشتر محیط مجازی به واقعیت و غوطه‌وری کاربر در این محیط است. ادراک انسان از محیط از ۵ حس تشکیل شده است که شامل: بینایی، شنوایی، بویایی، چشایی و لامسه است؛ حال برای آنکه کاربر واقعیت مجازی حس حضور در محیط مجازی را داشته باشد باید به طریقی

گیرنده‌های هر حس را فریب داد. به این منظور برای چشم از عینک‌های واقعیت مجازی<sup>۱</sup>، برای گوش از هدست<sup>۲</sup> و برای لامسه از تکنولوژی بازخورد نیرویی یا هپتیک استفاده می‌شود. محققان در تلاش‌اند که با اضافه کردن هپتیک به واقعیت مجازی، حداکثر حس واقعی را به کاربر القا کنند.

## ۲-۲- معرفی هپتیک و نحوه‌ی عملکرد

در این قسمت به معرفی کامل هپتیک پرداخته می‌شود. سعی بر این شده‌است که اگر شخصی هیچ گونه آگاهی از تکنولوژی هپتیک ندارد با مطالعه‌ی این قسمت به طور کامل از معنای واژه هپتیک تا چگونگی کارکرد دستگاه هپتیک آگاهی پیدا کند.

### ۲-۲-۱- واژه شناسی

واژه هپتیک از ریشه‌ی یونانی "haptikos" مشتق شده که به معنای توانایی گرفتن و حس کردن اجسام است. پس معنای واژه هپتیک که لمس کردن و گرفتن است اسم بامسمایی برای این تکنولوژی که هدفش برقراری حس لامسه با محیط مجازی است، می‌باشد.

### ۲-۲-۲- مفهوم کنونی

هپتیک یک تکنولوژی است که حس لامسه را به کامپیوترها اضافه کرده‌است. یک دستگاه هپتیک حس لمس اشیا را با یک مولد کامپیوتری به افراد می‌دهد؛ بنابراین وقتی در یک شبیه سازی با اشیا مجازی برخورد ایجاد می‌شود با استفاده از این تکنولوژی به نظر واقعا آن‌ها لمس شده‌اند. هپتیک این اجازه را به انسان می‌دهد تا در سه بعد حس لامسه و توانایی درک جرم، حجم و شکل اجسام را داشته باشد. محققان به طور کلی واژه‌ی هپتیک را به آن دسته از زمینه‌های نسبت می‌دهند که پایه‌ی آن‌ها حس لامسه‌ی انسان است [۶].

---

<sup>۱</sup> VR

<sup>۲</sup> HeadSet

## ۲-۲-۳- تاریخچه

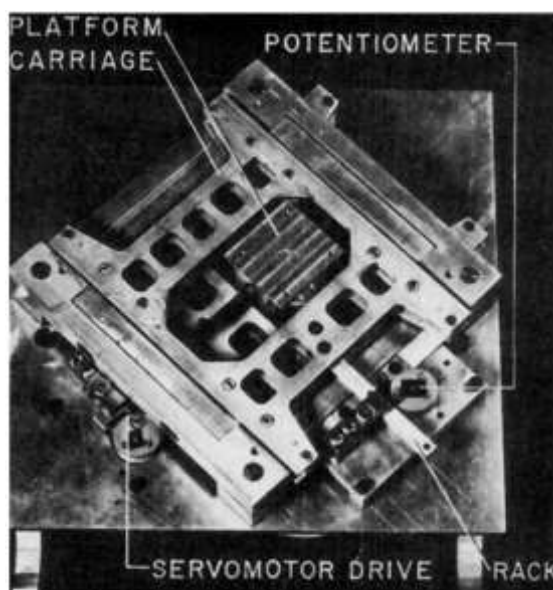
در دهه‌های ۱۹۷۰ و ۱۹۸۰ تلاش‌های تحقیقاتی چشم‌گیری در یک زمینه متفاوت رباتیک کلید خورده شد که توجهش به سمت دستکاری و حس اجسام از طریق لامسه بود. محققان پس از آغاز پروژه به سرعت دریافتند که ساخت یک دست رباتیکی که حس لامسه را به انسان منتقل کند بسیار پیچیده‌تر از امیدهای واهی اولیه‌ی آن‌هاست.

در اوایل دهه‌ی ۱۹۹۰ استفاده از واژه‌ی هپتیک رواج یافت و با ظهور چندین فناوری جدید مانند گرافیک رایانه‌ای نمایش اشیای شبیه سازی شده به انسان به صورت تعاملی امکان پذیر شد. به این ترتیب هپتیک کامپیوتری به وجود آمد که مبتنی به یک نمایشگر است که اشیای مجازی را در بر می‌گیرد و از طریق آن اشیای از نظر فیزیکی قابل حس می‌شوند.

این روش جدید انتقال اطلاعات به کاربر که از طریق حس لامسه صورت می‌گیرد با اعمال نیروی کنترل شده به کاربر خواص فیزیکی اجسام را منتقل می‌کند. این نیروها به خواص فیزیکی و میزان تماس با جسم بستگی دارد [۷].

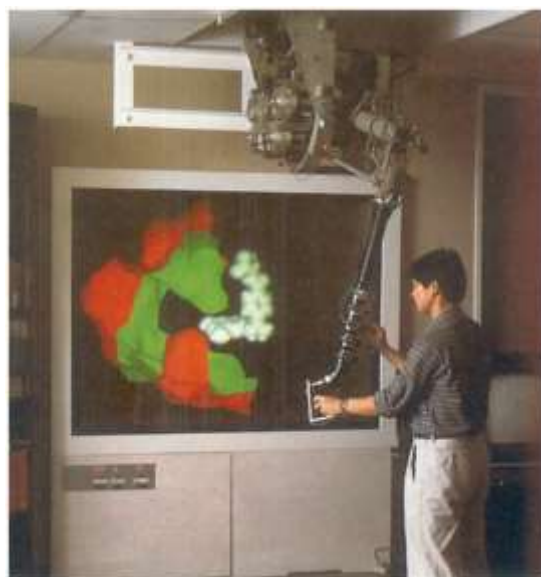
چندین ریشه‌ی پیدایش رابط کاربری هپتیک وجود داشت و آنها همگی به صورت همزمان در سال ۱۹۹۰ منتشر شدند. بروکس<sup>۱</sup> در سال ۱۹۶۷ یک پروژه برای نمایش بصری و هپتیک تعامل نیروی بین مولکول‌های پروتئین آغاز کرد. اولین سیستم هپتیک، GROPE-۱ یک پلتفرم حرکت دو بعدی را شبیه‌سازی کرد. دستگیره‌ای کوچک به پلت فرم متصل شده است که می‌تواند در یک صفحه افقی با وسعت ۲ اینچ مربع موقعیت دهی شود. پتانسیومترها موقعیت X و Y را می‌سنجند سپس سرو موتورهای نیروی X و Y را اعمال می‌کنند که هر دو به کامپیوتر متصل هستند و یک نمایش بصری را به این طریق شبیه‌سازی می‌کنند. این نخستین دستگاه هپتیک بود که محیط مجازی را شبیه‌سازی کرد (شکل ۵-۲ GROPE).

<sup>۱</sup> Brooks



شکل ۵-۲ GROPE

بعدها ARM Argonne معرفی شد، یک عملگر از راه دور<sup>۱</sup> که نیرو را در ۶ جهت آزادی منتقل می‌کرد. این دستگاه با یک صفحه نمایش بزرگ برجسته کن<sup>۲</sup> هماهنگ شده بود. در نهایت با این تکنولوژی سیستم GROPE-III به بازار عرضه شد (شکل ۶-۲ GROPE-III).

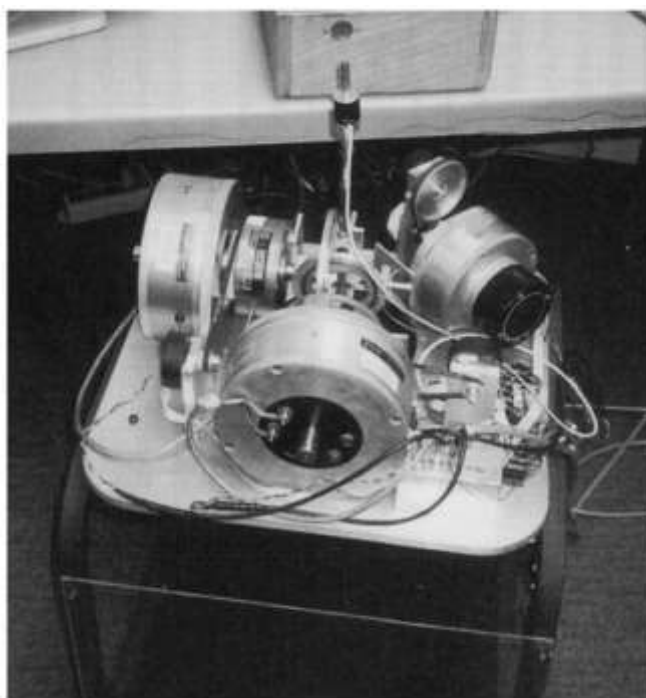


شکل ۶-۲ GROPE-III

<sup>۱</sup> Teleoperator

<sup>۲</sup> Stereoscop

مینسکی<sup>۱</sup> سیستمی را طراحی کرد که با نام sandpaper شناخته شد؛ که برای آزمایش با بافت‌های دارای حس لامسه ساخته شده بود (شکل ۷-۲ سیستم Sandpaper). این نرم افزار باعث ایجاد حالت-های فیزیکی مجازی بسیار کوچک می شود که دست کاربر را نسبت به مناطق کم ارتفاع می کشند و نسبت به مناطق بلند هل می دهند بنابراین یک نقشه از عمق بافت ایجاد می کند.



شکل ۷-۲ سیستم Sandpaper

در سال ۱۹۸۸، محققان شروع به طراحی یک رابط کاربری برای تعامل طبیعی در محیط مجازی کردند. بنابراین در سال ۱۹۹۰ اولین گام برای انتقال نیرو از روی صفحه نمایش در محیط مجازی به اندام انسان برداشته شد. دستگاه طراحی شده به این صورت بود که نیرو را به نوک انگشتان و کف دست منتقل می کرد، شکل ۸-۲ دستگاه Exoskeleton نمای کلی دستگاه را نشان می دهد این اولین تعامل بین محیط مجازی و حقیقی بود. این دستگاه‌ها با نام اسکلت بیرونی<sup>۲</sup> شناخته شدند [۶].

<sup>۱</sup> Minsky

<sup>۲</sup> Exoskeleton



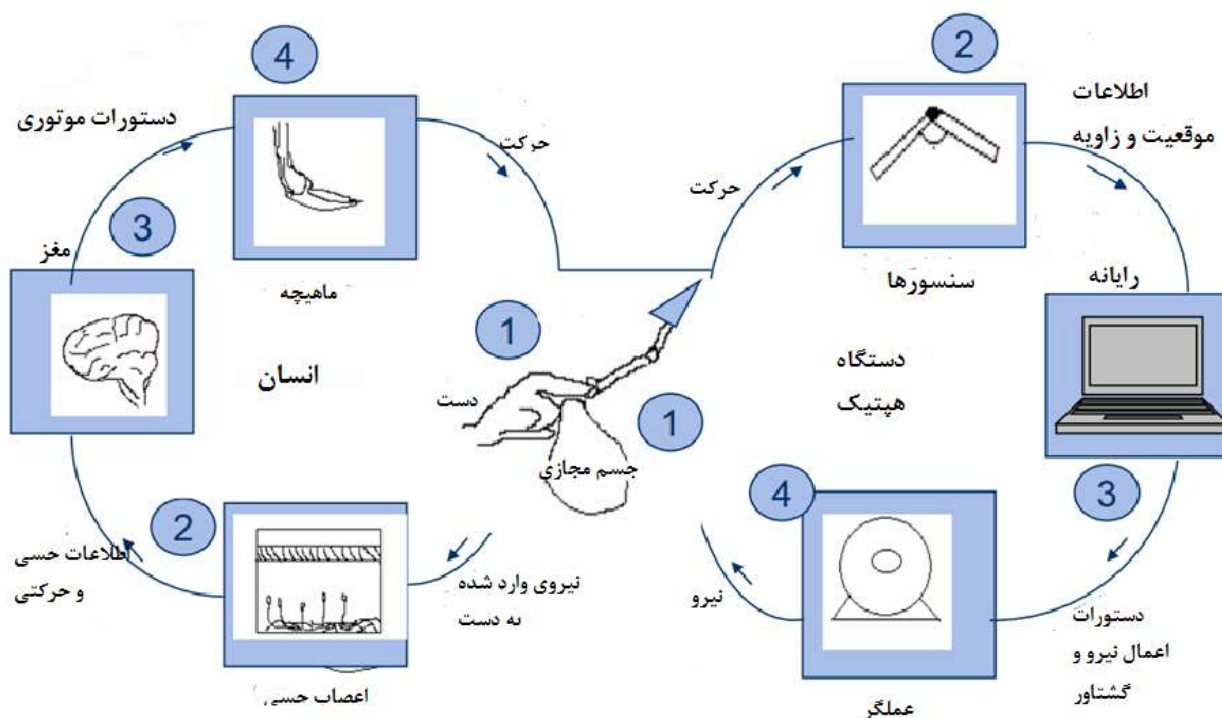
شکل ۸-۲ دستگاه Exoskeleton

## ۴-۲-۲- نحوه‌ی عملکرد دستگاه هپتیک

واقعیت مجازی این اجازه را به کاربر می‌دهد تا با محیط مجازی تعامل داشته باشد؛ لذا برای داشتن این تعامل به یک رایانه که پردازنده‌ی گرافیکی کارآمدی داشته باشد برای شبیه سازی نیاز است. پس از شبیه سازی محیط مجازی نیاز به یک دستگاه واقعیت مجازی است تا ارتباط بین کاربر و رایانه را برقرار سازد و این تعامل برقرار شود. در ادامه به توضیح اجمالی در خصوص چگونگی ارتباط کاربر و محیط مجازی پرداخته خواهد شد.

همانطور که در شکل ۹-۲ تعامل کاربر و محیط مجازیمشاهده می‌کنید، ارتباط کاربر و محیط مجازی را می‌توانیم به دو حلقه‌ی بازخوردی که یک سمت کاربر و سمت دیگر دستگاه هپتیک است تقسیم نمود. فرض کنید کاربر بازوی هپتیک را جابجا می‌کند، این جابجایی توسط سنسورهای درون هپتیک حس می‌شود و اطلاعات مربوط به موقعیت و زاویه‌ی بازو به رایانه منتقل می‌شود. با توجه به این اطلاعات رایانه امکان برخورد نشانگر و جسم مجازی در محیط شبیه سازی شده را بررسی می‌کند و در صورت برخورد دستورات اعمال گشتاور و نیرو را به موتورهای درون دستگاه هپتیک انتقال می‌دهد. به این صورت در صورت برخورد نشانگر هپتیک و جسم مجازی، نیرو بازخورد می‌شود.

در سمت دیگر کاربر این نیروی پدید آمده توسط موتورهای هپتیکی را به وسیله‌ی اعصاب لمسی خود حس می‌کند و این اطلاعات را به مغز منتقل می‌کند، مغز پس از دریافت اطلاعات با استفاده از اعصاب موتوری با ماهیچه‌های دست دستور تغییر مسیر یا توقف را می‌دهد. به این ترتیب کاربر کاملاً یک حس واقعی از برخورد با یک جسم را دریافت می‌کند [۸].

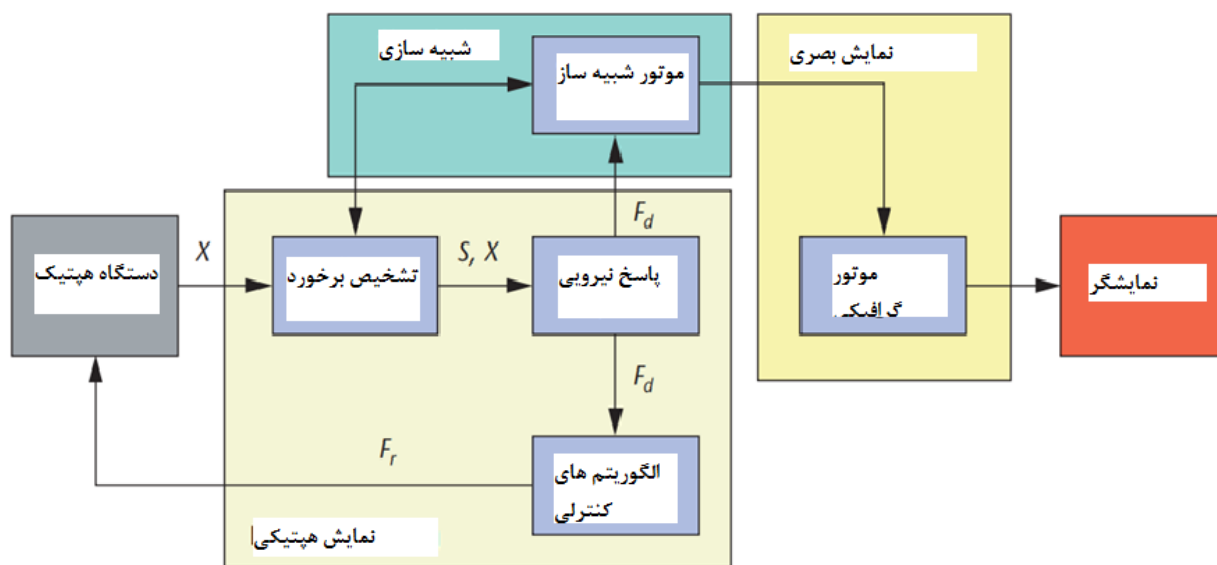


شکل ۹-۲ تعامل کاربر و محیط مجازی

اگر بخواهیم نگاهی دقیق‌تر به ساز و کار محیط مجازی بیندازیم بلوک دیاگرام شکل ۲-۱۰ بلوک دیاگرام سیستم هپتیک را در نظر بگیرید، در این دیاگرام حلقه‌ی نمایش هپتیکی به سه قسمت تقسیم شده است: الگوریتم‌های تشخیص برخورد، الگوریتم‌های پاسخ نیرویی و الگوریتم‌های کنترلی. الگوریتم‌های تشخیص برخورد اطلاعاتی در مورد برخورد نشانگر بازوی هپتیک و اجسام مجازی را بدست می‌آورد. الگوریتم‌های پاسخ نیرویی، نیروی ایده‌آل تعامل بین ابزار و جسم را محاسبه می‌کند و به الگوریتم‌های کنترلی می‌دهد، در نهایت الگوریتم‌های کنترلی از طریق دستگاه هپتیک نیروی ایده‌آل اعمالی را به صورت تقریبی به کاربر منتقل می‌کند.



دو بخش دیگر که با رنگ‌های آبی و زرد رنگ در شکل مشخص شده‌اند مربوط به موتور شبیه‌ساز مجازی و حلقه‌ی گرافیکی است که اطلاعات مربوط به موقعیت و نیرو را می‌گیرد و در صفحه‌ی نمایشگر تغییرات بصری را نمایش می‌دهد [۷].



شکل ۱۰-۲: بلوک دیاگرام سیستم هپتیک

## ۳-۲- انواع دستگاه هپتیک

یک تقسیم‌بندی ساده برای معرفی دستگاه‌های هپتیک طبقه‌بندی بر اساس میزان پیچیدگی و آزادی عمل کاربر برای تعامل با محیط مجازی است، لذا بر این اساس می‌توانیم انواع هپتیک را به صورت زیر معرفی کنیم [۹].

### ۳-۲-۱- گیره‌ی منعکس‌کننده‌ی نیرو

این دستگاه ساده‌ترین نوع دستگاه بازخورد نیرویی است که تقریباً مانند یک گیره عمل می‌کند. عملکرد آن به این صورت است که با یک فنر نیروی جسم مدل شده است و با توجه به جابجایی گیره‌ها نسبت به هم و ثابت سختی فنر که برای جسم مورد نظر تعریف شده، نیرو به دست کاربر منتقل می‌شود. این دستگاه یک درجه آزادی دارد (شکل ۱۱-۲) گیره‌ی منعکس‌کننده‌ی نیرو).



شکل ۱۱-۲ گیره‌ی منعکس کننده‌ی نیرو

## ۲-۳-۲- ماوس بازخورد نیروی Logitech Wingman

این ماوس در سال ۱۹۹۹ به بازار عرضه شد که برای بازی طراحی شده بود و در دو دوجبهت روی صفحه بازخورد نیرویی داشت (شکل ۱۲-۲) ماوس بازخورد نیروی Logitech Wingman). از ویژگی‌های این موس می‌توان به موارد زیر اشاره کرد:

- (۱) حس بازخورد نیرویی را برای اکثر بازی‌ها فراهم می‌کرد تا کاربر با تعامل بیشتری بازی کند.
- (۲) برای بازی‌های آنلاین اینترنتی هم جوابگو بود.
- (۳) برای سایر نرم‌افزارها هم نیروی بازخوردی تولید می‌کرد مثلاً رول‌های مکان‌نما<sup>۱</sup> و یا دکمه‌های مجازی منوها.
- (۴) در آن زمان به جای درگاه PS۲ از USB استفاده می‌کرد که سرعت تبادل اطلاعات را حدود ۳۰۰ برابر بالا برده بود.

---

<sup>۱</sup> Curser Rolls



شکل ۱۲-۲ ماوس بازخورد نیروی Logitech Wingman

### ۲-۳-۳- دستگاه هپتیک Geomagic Touch

همانطور که قبلاً اشاره شد این دستگاه سابقاً با نام Phantom omni شناخته می‌شد که امروزه با نام تجاری Geomagic Touch شناخته می‌شود و یکی از محصولات شرکت ۳D Systems است. این دستگاه یک نمونه‌ی میانی از دستگاه‌های هپتیک است که بیشتر در کارهای تحقیقاتی و مدل کردن سه بعدی استفاده می‌شود. دستگاه Touch به کاربر این اجازه را می‌دهد تا آزادانه با اجسام مجازی سه بعدی تعامل داشته باشد به این ترتیب امکان توسعه شبیه‌سازهای پزشکی و علمی فراهم شده است [۱۰].



شکل ۱۳-۲ دستگاه Touch

دستگاه Touch یک دستگاه موتوری است که نیروهای بازخوردی را از طریق این موتورها به دست کاربر منتقل می‌کند و حس لمس اجسام مجازی همانطور که در واقعیت هستند، به این طریق منتقل می‌شود. از کاربردها موردی آن می‌توانیم به موارد زیر اشاره کنیم:

(۱) شبیه‌سازهایی که به بازخورد نیرویی نیاز دارند.

(۲) آموزش و تمرین

(۳) ارزیابی مهارت

(۴) کنترل رباتیکی

(۵) طراحی رابط ماشینی

(۶) توانبخشی

(۷) طراحی مدل‌ها سه بعدی و نقشه برداری از اجسام

چون در این پروژه از این دستگاه استفاده کرده‌ایم، به طور خاص به بررسی ویژگی‌های آن می‌پردازیم. به این منظور قسمتی از راهنمای کاربری<sup>۱</sup> در شکل ۲-۱۴ ویژگی‌های دستگاه Touch آورده شده است.

Force Feedback Workspace	~ 6.4 W x 4.8 H x 2.8 D in	> 160 W x 120 H x 70 d mm
Footprint (physical area the base of the device occupies on a surface)	~ 6 5/8 W x 8 D in	~ 168 W x 203 D mm
Weight (device only)	3 lbs 15 oz	~1.42 kg
Range of Motion	Hand movement pivoting at wrist	
Nominal Position Resolution	> 450 dpi	~0.055 mm
Backdrive Friction	< 1 oz	< 0.26 N
Maximum Exertable Force (at nominal orthogonal arms position)	.75 lbf	3.3 N
Continuous Exertable Force (24 hrs)	> 0.2 lbf	> .88 N
Stiffness	X axis > 7.3 lbs./in	X axis > 1.26 N/mm
	Y axis > 13.4 lbs./in	Y axis > 2.31 N/mm
	Z axis > 5.9 lbs./in	Z axis > 1.02 N/mm
Inertia (apparent mass at tip)	~ 0.101 lbm	~ 45 g
Force Feedback	X, Y, Z	
Position Sensing	X, Y, Z (digital encoders)	
Stylus gimbal	Pitch, roll, yaw ( $\pm 5\%$ linearity potentiometers)	
Interface	RJ45 compliant on-board Ethernet Port or USB Port	
Supported platforms	Intel or AMD-based PCs	
OpenHaptics® SDK compatibility?	Yes	

شکل ۲-۱۴ ویژگی‌های دستگاه Touch

<sup>۱</sup> UserManual

وزن دستگاه حدود ۱/۵ کیلوگرم است و بازوی آن قابلیت حرکت در یک فضای ۱۶\*۱۲\*۷ سانتی مترمکعب را داراست. قدرت تفکیک مکانی حدود ۵۵ میکرومتر است که بسیار دقت بالایی است. این دستگاه چون دارای ۶ مفصل است به کاربر امکان ۶ درجه آزادی را می‌دهد. در فصل چهارم به تفصیل درمورد چگونگی ایجاد محیط مجازی و کدزنی برای دستگاه بحث شده است.

## ۲-۳-۴- دستکش Cyber Grasp

این دستکش که به صورت اسکلت بیرونی<sup>۱</sup> طراحی شده است بر پشت دست کاربر قرار می‌گیرد و برای هر انگشت یک نیروی بازخوردی جدا ایجاد می‌کند. این دستکش برای هر انگشت ۴ درجه آزادی فراهم می‌کند و با توجه به طراحی مناسبی که دارد برای هر فرد چه دارای دست بزرگی باشد چه کوچک به خوبی بر روی دست قفل می‌شود (شکل ۲-۱۵ دستکش Cyber Grasp).



شکل ۲-۱۵ دستکش Cyber Grasp

<sup>۱</sup> Exoskeleton

## ۲-۳-۵- غوطه‌وری<sup>۱</sup> در هپتیک

در این نوع هپتیک، دو دست کاربر را به دستگاه هپتیک متصل می‌کنند و با استفاده از عینک واقعیت مجازی او را در محیط مجازی قرار می‌دهند. به این ترتیب فرد در محیط مجازی غوطه‌ور می‌شود که هم نیروهای مجازی به وی بازخورد می‌شود و هم دید او به این محیط محدود می‌شود (شکل ۲-۱۶ غوطه‌وری در هپتیک).



شکل ۲-۱۶ غوطه‌وری در هپتیک

## ۲-۴- کاربردهای هپتیک

در این قسمت قصد داریم کاربردهای هپتیک را به صورت موردی بررسی کنیم.

### ۲-۴-۱- بازی‌های رایانه‌ای

اکثر سازنده‌ی بازی‌های رایانه‌ای از هپتیک‌های غیرفعال<sup>۱</sup> استفاده می‌کنند، به عنوان مثال دسته‌های بازی لرزشی در صورتی که کاربر در طی بازی خطایی انجام دهد و یا به مانعی برخورد کند این دسته‌ها

---

<sup>۱</sup> Immersion

می‌لرزند و حس قرار گیری در فضای بازی را به کاربر منتقل می‌کند. پیش بینی می‌شود در آینده هپتیک‌هایی مخصوص بازی‌های رایانه‌ای ساخته شود که کاربر در طی بازی، اجسام را بتواند حس و دستکاری کند.

## ۲-۴-۲- ربات‌های دورکار<sup>۲</sup>

در یک ربات دورکار، کنترل‌کننده در مسافت دورتری نسبت به عملگر قرار دارد و بدون اینکه شخص کنترل‌کننده نیرویی به دستش بازخورد شود بازو یا دوربین ربات را تحت کنترل می‌گیرد؛ حال با اضافه شدن قابلیت هپتیک به این نوع ربات‌ها کنترل‌کننده می‌تواند حس لمس اشیا و جابجایی آن‌ها را داشته باشد و در نتیجه ربات را با دقت بیشتری کنترل کند.

## ۲-۴-۳- پزشکی

در کاربردهای پزشکی می‌توان به شبیه‌سازهای جراحی که در این پروژه به این موضوع پرداخته شده، اشاره کرد. از دیگر کاربردها در این حوزه، عمل جراحی از راه دور<sup>۳</sup> است که جراح در محل دیگری از بیمارستان با استفاده از این تکنولوژی بافت‌های بیمار را حس می‌کند و با توجه به بازخورد نیرویی دقت عمل خود را بالا می‌برد.

## ۲-۴-۴- نظامی

در صنعت نظامی هپتیک برای آموزش افراد به کار برده می‌شود، به عنوان مثال در شبیه‌سازهای پرواز یا شبیه‌سازهای کنترل ماشین‌آلات جنگی با استفاده از این تکنولوژی شرایط بحرانی را شبیه‌سازی می‌کنند و افراد کنترل‌کننده‌ی این تجهیزات را آموزش می‌دهند.

---

<sup>۱</sup> Passive

<sup>۲</sup> TeleRobots

<sup>۳</sup> TeleOperation

## ۲-۴-۵- آموزش

جنبه‌ی دیگر استفاده از تکنولوژی هپتیک، آموزش است. با استفاده از این تکنولوژی هر نوع مدل و هر جسمی که برای یادگیری بهتر باید لمس شود را می‌توان مدل کرد و نیروهای آن را به کاربر منتقل کرد. به عنوان مثال در یادگیری شیمی، بحث نیروهای بین ملکولی، می‌توان با استفاده از این تکنولوژی انواع این نیروها را به دانش‌جو نشان داد تا با درک بهتری آن‌ها را به خاطر بسپارد [۱۱].

## ۲-۵- محدودیت‌های هپتیک

هپتیک هم مانند تمام تکنولوژی‌های ساخته شده، دارای محدودیت‌هایی است که از مهمترین آن‌ها می‌توان به موارد زیر اشاره کرد [۱۱].

- (۱) هزینه‌ی زیاد برای تهیه دستگاه و توسعه‌ی نرم‌افزاری
- (۲) هپتیک‌هایی که اسکلت بیرونی هستند و کاربر باید آن‌ها را بپوشد، حجم و وزن زیادی دارند.
- (۳) دستگاه‌های هپتیک نیروی محدودی را می‌توانند اعمال کنند.
- (۴) الگوریتم‌های اعمال نیروی هپتیک به صورت گسسته عمل می‌کنند و اگر سرعت محاسبه کمی پایین باشد کاربر این گسستگی را درک می‌کند.

## ۲-۶- مروری بر سوابق پیشین

در این قسمت به مرور سوابق پیشین و کارهای مرتبط با این پروژه خواهیم پرداخت. چون در این پروژه کاربرد پزشکی هپتیک مورد بررسی قرار گرفته لذا مقالاتی را که در زمینه‌ی شبیه‌سازهای پزشکی و اعتبار سنجی‌هایی که در این راستا صورت گرفته است، مورد بررسی قرار خواهد گرفت.

یکی از توسعه‌های صورت گرفته در حوزه‌ی پزشکی، توانبخشی افراد آسیب دیده و ناتوان است. به عنوان مثال، طراحی متدی برای کنترل دست کاربر آسیب دیده جهت آموزش نوشتن، این آموزش برای افرادی که کارایی خود را از دست داده‌اند و یا کودکانی که می‌خواهند نوشتن آموزش ببینند می‌تواند مفید باشد [۱۲].



استفاده‌ی پزشکی دیگر، انجام عمل‌های از راه دور است، به دلیل داشتن بازخورد نیرویی، پزشک جراحی که از راه دور عمل جراحی را اداره می‌کند به خوبی می‌تواند بافت را حس کند و فشار بیش از حد وارد نکند و عمل جراحی با آسیب بسیار کمتری انجام شود [۱۳].

در سال ۲۰۱۰ یک مرکز شبیه‌سازی عمل جراحی مغز و اعصاب و هوش مصنوعی در بیمارستان مغز و اعصاب مونترال کانادا تاسیس شد که سهم عظیمی در پیشرفت شبیه‌سازهای عمل جراحی دارد [۱۴]. در ادامه برخی از کارهای این مرکز را بررسی می‌کنیم.

در سال ۲۰۱۲ یک شبیه‌ساز عمل جراحی مغز و اعصاب با هدف آموزش و ارزیابی عمل جراحی مغز باز عرضه شد. در این شبیه‌ساز بافت مغزی و تومورهایی در مغز مدلسازی شده است و ابزار پزشکی مانند، قیچی جراحی و منعقدساز به محیط جراحی اضافه شده است. از ویژگی‌های منحصر به فرد این شبیه‌ساز اضافه کردن امکان خونریزی در حین عمل جراحی است. از این شبیه‌ساز برای آموزش و ارزیابی جراحان استفاده می‌شود [۱۵].

از دیگر کارهای این گروه، ارزیابی مهارت جراحی است. به عنوان مثال در یک مطالعه، از چند نفر جراح و دستیار جراح برای انجام یک عمل برداشت تومور مغزی در شبیه‌ساز طراحی شده، درخواست همکاری شده بود. برای این عمل، سه تومور مغزی با مشخصات فیزیکی متفاوت تعریف شده بود و هر یک در نقطه‌ی خاصی از مغز قرار داده شده بود و از داوطلبان خواسته شده بود تا این سه تومور را از بافت مغزی با نهایت دقت خارج کنند. با انجام این عمل، تمام اطلاعات و روند مربوط به عمل جراحی ثبت و تحلیل شده بود. در نهایت بین جراحان و دستیاران یک مقایسه صورت گرفته بود که نشان دهنده‌ی دقت عملکرد و ظرافت بیشتر جراحان نسبت به دستیاران بود [۱۶].

## ۷-۲- جمع بندی

در این فصل مفاهیم اولیه‌ی لازم برای آشنایی با هپتیک گفته شد که معرفی آن، تاریخچه، نحوه‌ی عملکرد، انواع هپتیک، کاربردها و محدودیت‌های این تکنولوژی بررسی شد. در نهایت مروری بر سوابق گذشته‌ی مرتبط با پروژه بحث شد. در فصل بعدی به معرفی امکانات نرم‌افزاری و توسعه‌ای که در این پروژه صورت گرفته است، پرداخته می‌شود.

## فصل سوم

### ساختار نرم‌افزار استفاده شده

## مقدمه

توسعه‌ی نرم‌افزاری که در این پروژه برای شبیه‌ساز عمل جراحی صورت گرفته است، به زبان C++ و در محیط کد نویسی Visual Studio نوشته شده است. در این بخش قصد داریم کتابخانه‌ی مورد استفاده، قسمت‌های مختلف کد و الگوریتم‌هایی که برای مدلسازی استفاده شده است را شرح دهیم.

### ۳-۱- معرفی کتابخانه‌ی مورد استفاده

برای توسعه‌های نرم‌افزاری و سهولت در کدزنی، توابع و ابزار پر استفاده در زمینه‌های مختلف رایانه‌ای را به صورت کتابخانه‌هایی ارائه می‌دهند که در کار ما کتابخانه‌ای با نام CHAI<sup>۱</sup> ارائه شده است. این کتابخانه، یک کتابخانه‌ی متن‌باز<sup>۲</sup> به زبان C++ است که مجموعه‌ای از الگوریتم‌های هپتیک رایانه‌ای، نمایش بصری و شبیه‌سازی تعامل بلادرنگ را گردآوری کرده است.

نوشتن این کتابخانه در آزمایشگاه هوش مصنوعی دانشگاه استنفورد با هدف ارائه‌ی یک چارچوب<sup>۳</sup> ساده برای معرفی هپتیک به جامعه‌ی محققین کلید خورد سپس با همکاری دانشگاه EPFL در سوویس و دانشگاه سیینا<sup>۴</sup> در ایتالیا، اولین نسخه‌ی عمومی آن در سال ۲۰۰۴ عرضه شد.

CHAI<sup>۱</sup> برای توسعه‌ی آسانتر و قابل حس‌تر نرم‌افزارهایی که مدل سازی سه بعدی را با قابلیت پس-خورد نیرویی ارائه می‌دهند طراحی شده است. این کتابخانه توانایی پشتیبانی از انواع مختلف هپتیک را دارا می‌باشد. بعد از گذشت چندین سال از آغاز نوشتن این کتابخانه، CHAI<sup>۱</sup> به یکی از محبوب‌ترین و پرکاربردترین چارچوب‌های متن‌باز نمایش هپتیک شده است [۱۷].

CHAI<sup>۱</sup> تعداد زیادی از الگوریتم‌های پس‌خورد نیرویی را شامل می‌باشد که به کاربر این اجازه را می‌دهد تا از قابلیت دستگاه هپتیک بیشترین استفاده را ببرد. همچنین این چارچوب با استفاده از کتابخانه‌های نمایش گرافیکی مانند OpenGL یک زیر ساخت برای نمایش بصری اجسام سه بعدی در محیط

<sup>۱</sup> Computer Haptics and Active Interface

<sup>۲</sup> Open Source

<sup>۳</sup> Framework

<sup>۴</sup> Siena

مجازی طراحی شده ارایه می‌دهد. این کتابخانه همچنین امکان اضافه کردن فایل‌های سه بعدی که در نرم‌افزارهای حرفه‌ای طراحی مانند AutoDesk، ۳D MAX و ... طراحی شده‌اند را به محیط مجازی امکان پذیر می‌سازد.

## ۳-۲- امکانات ارایه شده در CHAI۳D

پیش از اینکه به تشریح قسمت‌های نرم‌افزار توسعه داده شده در این پروژه بپردازیم، لازم است با امکانات و کلاس‌هایی که در کتابخانه‌ی CHAI۳D ارایه شده است، آشنایی پیدا کنیم. در ادامه به این موضوع می‌پردازیم.

### ۳-۲-۱- شناسایی دستگاه هپتیک

دستگاه هپتیک یک واسط دوجانبه بین انسان و محیط مجازی است که توسط کاربر برای تعامل با محیط شبیه‌سازی شده استفاده می‌شود. یک دستگاه هپتیک می‌تواند انواع مختلفی داشته باشد مانند: ماوس، جوی استیک<sup>۱</sup>، پد انگشتی، دستکش هپتیکی، دستگاه Geomagic Touch و ... باشد که می‌تواند اجسام را با جرم، فنر، دمپر، اصطکاک و خاصیت سختی شبیه‌سازی کند و نیروهای پس‌خوردی را به دست کاربر اعمال کند

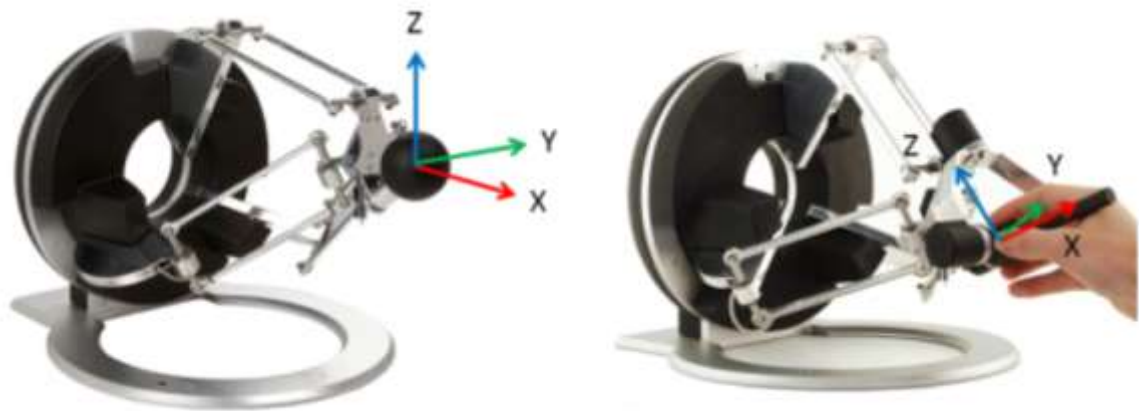
حال برای شناسایی دستگاه هپتیکی که به رایانه متصل می‌شود، CHAI۳D یک کلاس با نام cGenericHapticDevice ارایه داده است که دارای یک سری متدها برای ارتباط با اکثر هپتیک‌های سه بعدی است. موقعیت اثرگر<sup>۲</sup> یا دستگیره‌ی<sup>۳</sup> دستگاه هپتیک توسط متدهای این کلاس مانند: getPosition()، getRotation() و getGripperAngleDeg() قابل دستیابی است. همچنین اگر کلیدهای روی اثرگر توسط کاربر فشرده شوند با استفاده از متد getUserSwitch() قابل شناسایی است. نیرو و گشتاور توسط متدهای setForce()، setForceAndTorque() و setForceAndTorqueAndGripperForce() می‌تواند به دستگاه هپتیک منتقل شود.

<sup>۱</sup> joystick

<sup>۲</sup> End Effector

<sup>۳</sup> Handle

تمامی مقادیری که در این کتابخانه استفاده شده است از واحدهای متریک<sup>۱</sup> استفاده می‌کند و در مختصات دکارتی تعریف شده‌اند، محور X این مختصات به سمت کاربر، محور Y به سمت راست او و محور Z به سمت بالا تعریف شده است. همچنین مبدا مختصات در مرکز فضای کاری دستگاه قرار دارد (شکل ۱-۳ محور مختصاتی تعریف شده برای دستگاه هپتیک).



شکل ۱-۳ محور مختصاتی تعریف شده برای دستگاه هپتیک

### ۲-۲-۳-۲- دنیای مجازی

در CHAI<sup>۳</sup>D گره‌ها<sup>۳</sup> و اشیاء<sup>۴</sup> به عنوان بلوک‌های سازنده‌ی پایه‌ای برای سایر امکانات محیط مجازی هستند. شی دنیا به عنوان گره‌ی اصلی برای ساخت سایر گره‌ها شناخته می‌شود. دنیا یک صحنه مجازی ارائه می‌دهد که با زیر مجموعه‌هایی که دارد تعیین می‌کند که چه محتوایی در صفحه‌ی گرافیکی نمایش داده شود.

هر گره که در محیط مجازی تعریف می‌شود با توجه به موقعیت والد موقعیتش مشخص می‌شود. هر گره والد در صورت تغییر، تمام زیرمجموعه‌هایش تحت تاثیر آن تغییر قرار می‌گیرند به عنوان مثال اگر یک

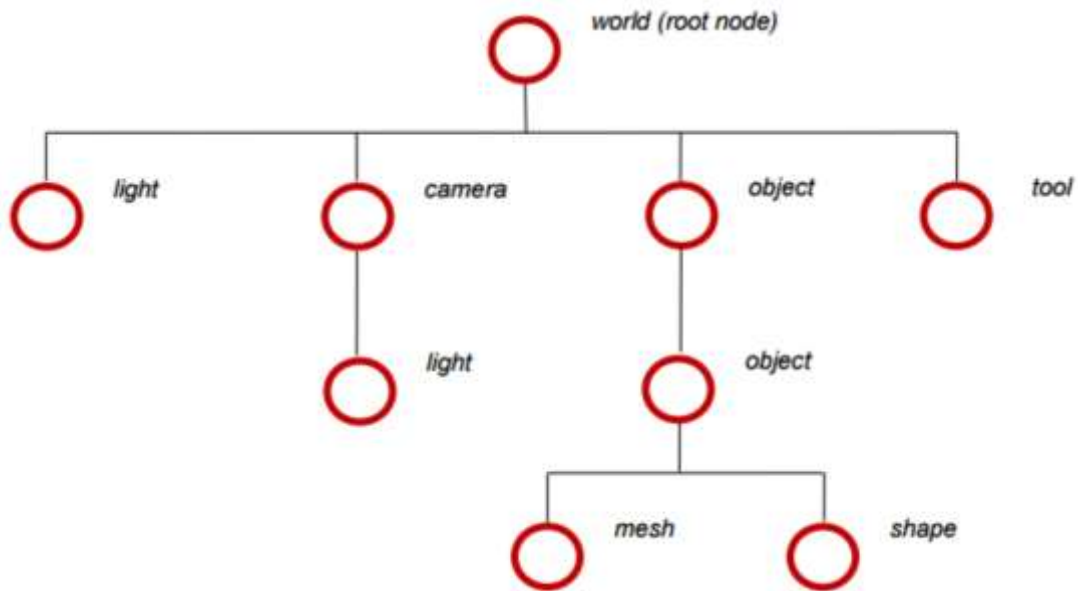
<sup>۱</sup> Metric

<sup>۲</sup> World

<sup>۳</sup> Nodes

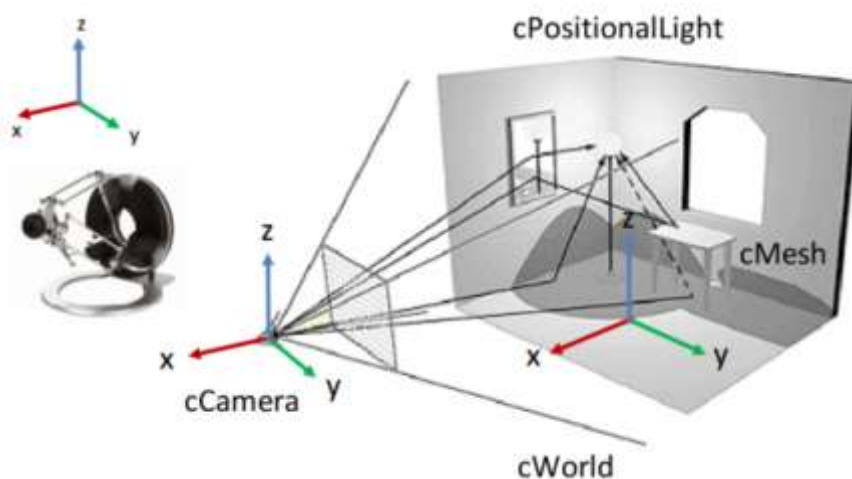
<sup>۴</sup> Objects

چرخش به گره والد اعمال شود تمامی فرزندان یا زیر مجموعه‌هایش تحت تاثیر آن چرخش قرار می‌گیرند. شکل ۲-۳ گره‌های والد و زیر مجموعه‌هایشان را در تعریف یک محیط مجازی نشان می‌دهد.



شکل ۲-۳ گره‌های والد و زیر مجموعه‌هایشان

به عنوان اولین گام در ساخت محیط مجازی ابتدا باید یک دنیا ساخت و سپس به آن دنیا یک منبع نور و یک دوربین برای نمایش اضافه کرد (شکل ۳-۳ محیط مجازی ساخته شده با CHAI۳D).



شکل ۳-۳ محیط مجازی ساخته شده با CHAI۳D

در قطعه کد زیر چگونگی ساخت دنیا و اضافه کردن منبع نور و دوربین به عنوان زیرمجموعه‌ی دنیا مشخص شده است.

جدول ۳-۱- قطعه کد ساخت دنیا و اضافه کردن منبع نور و دوربین

```
using namespace chaird;
// create a new world.
world = new cWorld();
// create a camera and insert it into the world
camera = new cCamera(world);
world->addChild(camera);
// create a directional light source and insert it into the world
light = new cDirectionalLight(world);
world->addChild(light);
```

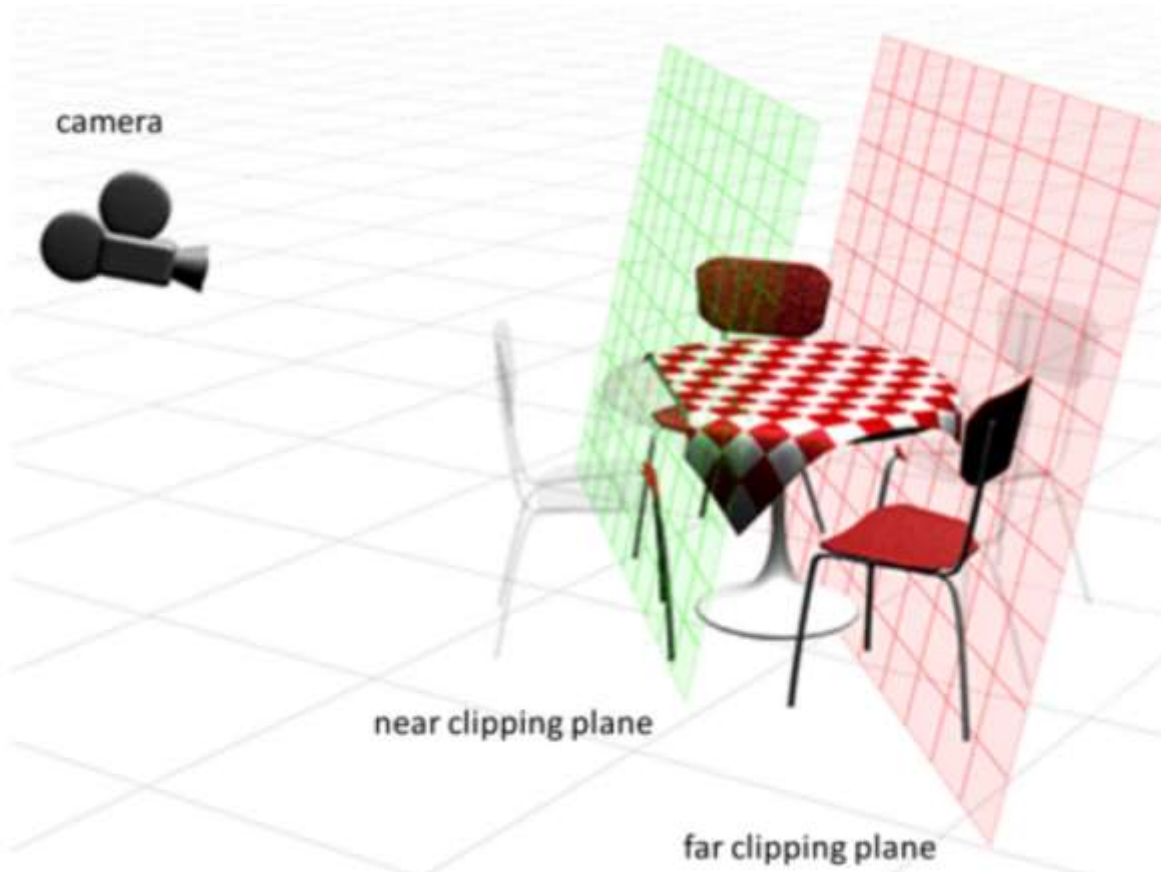
### ۳-۲-۳- دوربین‌های نمایش محیط مجازی

دوربین‌ها اصلی‌ترین ابزار نمایش سه بعدی هستند، آن‌ها در دنیای مجازی قرار می‌گیرند و به صورت یکتا نام‌گذاری می‌شوند. امکان تعریف چندین دوربین برای یک صحنه وجود دارد که هر یک به نقطه دید خاصی نسبت داده می‌شوند. دوربین‌ها می‌توانند به یک هدف خاص نسبت داده شوند یا توسط دستگاه هپتیک کنترل شوند.

در قطعه کد بعدی یک دوربین مجازی را در دنیای مجازی قرار خواهیم داد. موقعیت دوربین، موقعیت هدف مطلوب برای نمایش و یک بردار که مشخص می‌کند جهت بالا کجاست، برای ایجاد دید بصری از صحنه و چرخش دوربین متناسب با صحنه استفاده می‌شود. یک دوربین می‌تواند به عنوان فرزند (زیر مجموعه) یک شی دیگر یا ابزار عملگر هپتیک استفاده شود. صفحات برش<sup>۱</sup> دور و نزدیک برای نمایش اجسام مجازی بین این دو صفحه تعریف می‌شوند. اجسامی که خارج از فاصله‌ی این دو صفحه قرار گیرند نمایش داده نخواهند شد، بنابراین لازم است قبل از ایجاد اجسام سه بعدی در محیط مجازی

<sup>۱</sup> Clipping Planes

اطمینان پیدا کنیم که صفحات برش دور و نزدیک به درستی تعریف شده‌اند در شکل ۳-۴ صفحات برش دور و نزدیک و چگونگی نمایش صحنه با استفاده از دوربین قابل مشاهده است.



شکل ۳-۴ صفحات برش دور و نزدیک و چگونگی نمایش صحنه

قطعه کد زیر چگونگی تنظیم موقعیت دوربین و تعریف صفحات برش را نشان می‌دهد.

جدول ۳-۲ تنظیم موقعیت دوربین و تعریف صفحات برش

```
using namespace chai3d;
// define a camera position
cVector3d position(1,0, 0,0, 0,5);
// define a target position
cVector3d lookat(0,0, 0,0, 0,2);
// define the orientation of the camera by a vector pointing upward
cVector3d up(0,0, 0,0, 1,0);
// position and orient the camera
camera->set(position, lookat, up);
```



```
// set the near and far clipping planes of the camera.
camera->setClippingPlanes(۰,۰, ۱۰,۰);
```

در نمایش رایانه‌ای سه بعدی، یک پنجره‌ی دو بعدی مستطیلی تعریف می‌شود که در صفحه نمایش رایانه نمایش داده می‌شود. این پنجره، محیط مجازی و اجسام درون آن را به نمایش می‌گذارد. برای ساخت این پنجره‌ی اغلب از چارچوب‌های گرافیکی مانند FreeGLUT استفاده می‌کنند که در CHAI۳D هم از این چارچوب بهره برده شده است.

برای نمایش یک صحنه که توسط دوربین مجازی فیلم برداری می‌شود، کفایت طول و عرض پنجره‌ی دوبعدی برحسب تعداد پیکسل را به متد `renderView()` بدهیم. قطعه کد زیر چگونگی این عمل را نشان می‌دهد.

جدول ۳-۳ قطعه کد مربوط به نمایش توسط دوربین

```
using namespace chaird;
// update shadow maps (if any)
world->updateShadowMaps(false, mirroredDisplay);
// render world
camera->renderView(windowW, windowH);
// swap buffers
glutSwapBuffers();
// wait until all GL commands are completed
glFinish();
```

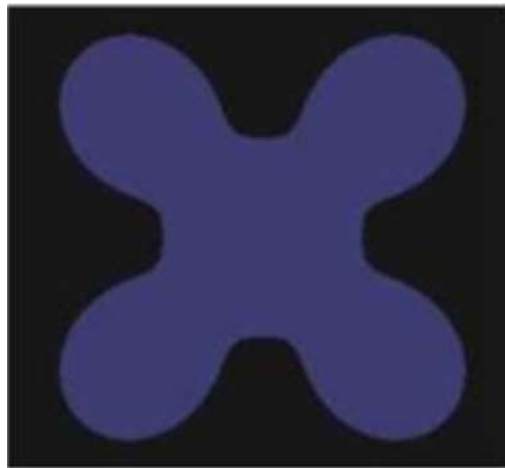
### ۳-۲-۴ نور و سایه

نور مهمترین عامل در بینایی انسان است در واقع چیزی که ما از اجسام می‌بینیم برپایه‌ی پرتوهای نوری است که از یک منبع نور به سطح اجسام می‌خورد و بازتاب آن به چشم ما می‌رسد. پرتوهای نور در اصل از یک منبع انرژی مانند خورشید یا چراغ درون اتاق منشا می‌گیرد. باید توجه داشت، یک پرتوی نوری به صورت تیوری در یک خط مستقیم سیر می‌کند که با برخورد به اجسام یا جذب می‌شود یا بازتابش می‌شود که اگر نور بازتاب شده به چشم برسد آن جسم قابل رویت می‌شود.

برای نمایش سه بعدی و طراحی محیط مجازی ضروری است تا انواع نور را بشناسیم و تاثیر هر یک را بر سطح اجسام مجازی سه بعدی بدانیم البته این دسته بندی به این معنی نیست که این نورها در طبیعت وجود دارند بلکه با توجه به نیاز نمایش گرافیکی این دسته بندی انجام شده است. در ادامه به معرفی این نورها می پردازیم.

### ۳-۲-۴-۱- نور محیطی<sup>۱</sup>

یک جسم سه بعدی با نور محیطی، دو بعدی به نظر می آید. این نور در واقع میانگین نورهایی است که از منابع نوری اطراف جسم انتشار پیدا می کند. زمانی که پرتوهای خورشید از پنجره ای اتاق عبور می کنند، به دیوارها برخورد می کنند، بازتابیده و در همه ی جهات پراکنده می شوند، در نتیجه کل اتاق را روشن می کنند. این نور با این کیفیت را نور محیطی می گویند. نور محیطی در یک محیط مجازی به تنهایی نمی تواند باعث سه بعدی به نظر رسیدن اجسام شود زیرا به تمامی قسمت های جسم نور یکسانی می رسد و همه جا به یک اندازه روشن می شود. بنابراین جسم به صورت دو بعدی به نظر می رسد و وقتی از این نور در محیط مجازی استفاده شود اجسام به صورت مسطح بر روی صفحه نمایش پدیدار می شوند. شکل ۳-۵ تاثیر این نوع نور روی یک جسم سه بعدی را نشان می دهد.

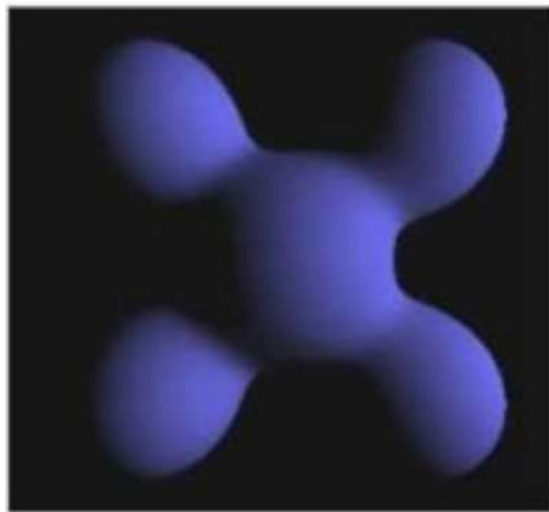


شکل ۳-۵ تاثیر نور محیطی روی یک جسم سه بعدی

<sup>۱</sup> Ambient Light

۳-۲-۴-۲- نور پخش شونده<sup>۱</sup>

نور پخش شونده را می توان اینگونه توصیف کرد، نوری که دارای موقعیت در فضا است و از یک جهت خاص تابیده می شود. به عنوان مثال یک چراغ قوه که کمی بالاتر از جسم قرار گرفته و به آن نور می تاباند می تواند به عنوان نور پخش شونده معرفی شود. در شکل زیر یک نور پخش شونده در سمت راست قرار گرفته که وقتی به جسم برخورد می کند، پراکنده می شود و بازتابش پیدا می کند.



شکل ۳-۶ تاثیر نور پخش شونده روی جسم سه بعدی

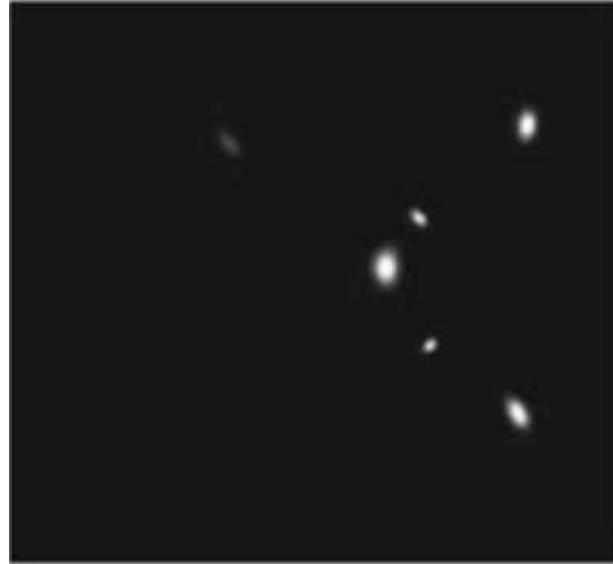
۳-۲-۴-۳- نور آینه وار<sup>۲</sup>

همانند نور پخش شونده، نور آینه وار هم جهت دار است و از یک جهت خاص تابیده می شود. تفاوت بین این نور و نور پخش شونده از این جهت است که نور آینه وار از سطح جسم به صورت شدیدتر و محلی تر بازتاب می شود. نمایش نور آینه وار به زاویه ی بین ناظر و منبع نور وابسته است. از نقطه دید ناظر نور آینه وار یک محدوده ی روشن تر ایجاد می کند که شدت بازتاب آن بسته به نوع جسم و قدرت منبع نوری

<sup>۱</sup> Diffuse Light

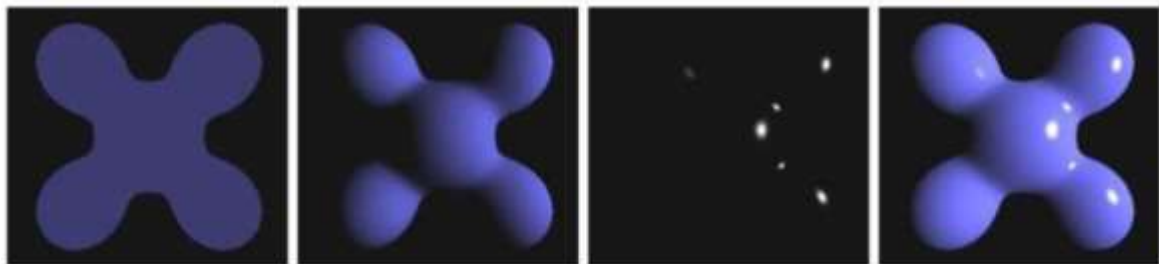
<sup>۲</sup> Specular Light

است که نور آینه‌وار از آن تابش می‌یابد شکل ۷-۳ تاثیر نور آینه‌وار بر روی جسم سه بعدی را نمایش می‌دهد.



شکل ۷-۳ تاثیر نور اختصاصی بر روی جسم سه بعدی

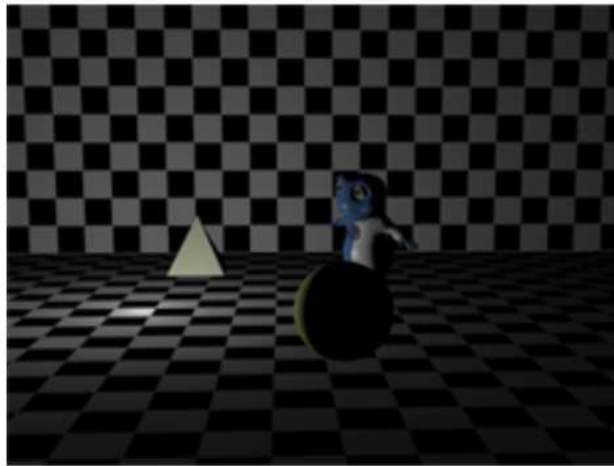
حال اگر سه نور معرفی شده را با هم داشته باشیم جسم با توجه به این نور پردازی در دید ما سه بعدی به نظر می‌آید، شکل زیر ترکیب این سه نوع نور را با هم نشان می‌دهد که سه بعدی به نظر رسیدن جسم کاملاً واضح است.



شکل ۸-۳ ترکیب سه نوع نور (تصویر اول از سمت راست مجموع سه تصویر دیگر است)

برای ایجاد سایه در محیط مجازی و سه بعدی به نظر رسیدن اجسام، CHAI<sup>3</sup>D باید شدت و جهت نوری که به اجسام تابانده می‌شود را بداند. برای تامین این خواص در محیط شبیه‌سازی شده لازم است که منابع نوری تعریف شوند. شدت و رنگ نور در CHAI<sup>3</sup>D برای تمام منابع یکسان است اما جهت تابش نور بسته به نوع منبعی است که انتخاب می‌شود. سه نوع منبع نوری در CHAI<sup>3</sup>D تعریف شده است که در ادامه به معرفی هر یک می‌پردازیم.

(۱) منبع نور نقطه‌ای<sup>۱</sup>: یک منبع نور نقطه‌ای در یک نقطه از فضا تعریف می‌شود و پرتوهای نور را در تمام جهات منتشر می‌کند. جهتی که پرتوی نور به جسم برخورد می‌کند با وصل کردن مختصات منبع نور به جسم بدست می‌آید. در شکل زیر یک منبع نقطه‌ای نور را ملاحظه می‌کنید.



شکل ۹-۳ منبع نقطه‌ای نور

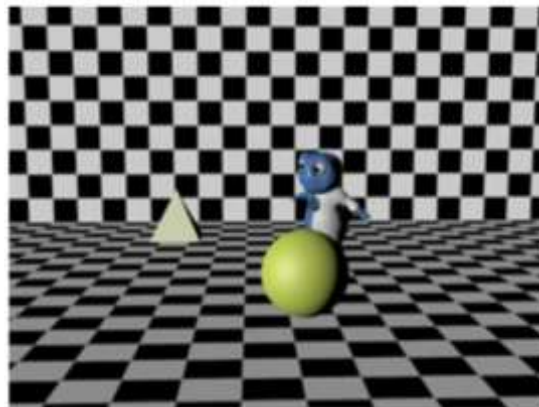
برای تعریف این نوع منبع در محیط کد نویسی از قطعه کد زیر استفاده می‌کنیم که ابتدا منبع نور نقطه‌ای تعریف می‌شود، سپس به عنوان یک فرزند یا زیر مجموعه به دنیای مجازی اضافه می‌شود و پس از روشن کردن منبع موقعیت مکانی آن تعیین می‌شود.

جدول ۴-۳ قطعه کد مربوط به فعالسازی منبع نور نقطه‌ای

```
using namespace chai3d;
// create a light source
light = new CPositionalLight(world);
// attach light to camera
world->addChild(light);
// enable light source
light->setEnabled(true);
// position the light source
light->setLocalPos(۱,۰, ۱,۰, ۰,۵);
```

<sup>۱</sup> Positional Light

(۲) منابع نور جهت‌دار<sup>۱</sup>: این نوع از منابع موقعیت مکانی خاصی ندارند و به صورت موازی پرتوهای نور را ساطع می‌کنند. منابع نور جهت‌دار نشان دهنده‌ی یک منبع نور بزرگ و بسیار دور از صحنه‌ی تصویربرداری است که در محیط واقعی می‌توانیم خورشید و ماه را برای این نوع منبع مثال بزنیم. در یک محیط شبیه‌سازی شده برای ایجاد سایه بدون اینکه مشخص کنیم منبع نور در کجا قرار دارد مفید است. شکل ۳-۱۰ یک منبع نور جهت‌دار را نشان می‌دهد.



شکل ۳-۱۰ منبع نور جهت‌دار

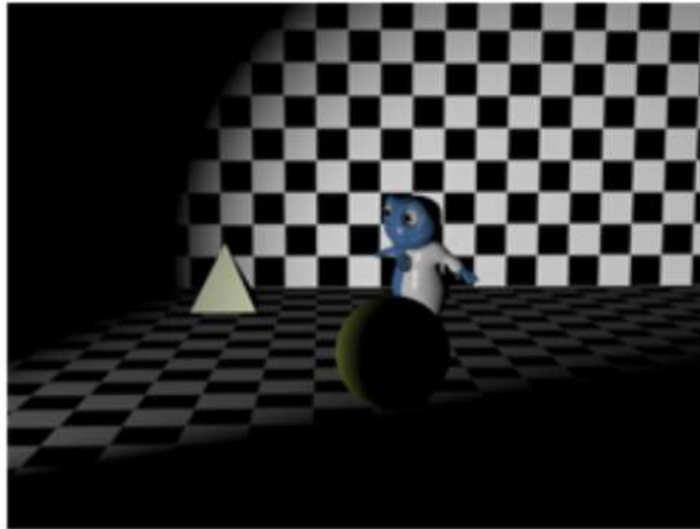
برای تعریف این نوع منبع نور در محیط مجازی طبق قطعه کد زیر، تنها کافیست جهت تابش آن را مشخص کنیم.

جدول ۳-۵ قطعه کد فعال‌سازی منبع نور جهت‌دار

```
using namespace chair3d;
// create a directional light source
light = new cDirectionalLight(world);
// insert light source inside world
world->addChild(light);
// enable light source
light->setEnabled(true);
// define direction of light beam
light->setDir(-1,0, 0,0, 0,0);
```

<sup>۱</sup> Directional Lights

۳) نورافکن<sup>۱</sup>: مانند منابع نقطه‌ای نور، نورافکن‌ها هم موقعیت مکانی خاصی در فضا دارند که با تعیین موقعیت و جهت تابش، یک محدوده‌ی مخروطی شکل از پرتوهای نوری را پدید می‌آورند. شکل ۱۱-۳ یک منبع نورافکن را نشان می‌دهد.



شکل ۱۱-۳ نورافکن

برای فعال سازی این نوع منبع باید موقعیت، جهت و حداکثر میزان زاویه‌ای که پرتوهای نوری می‌توانند داشته باشند را مشخص کنیم. قطعه کد زیر چگونگی این عمل را نشان می‌دهد.

جدول ۳-۶ قطعه کد فعال‌سازی نورافکن

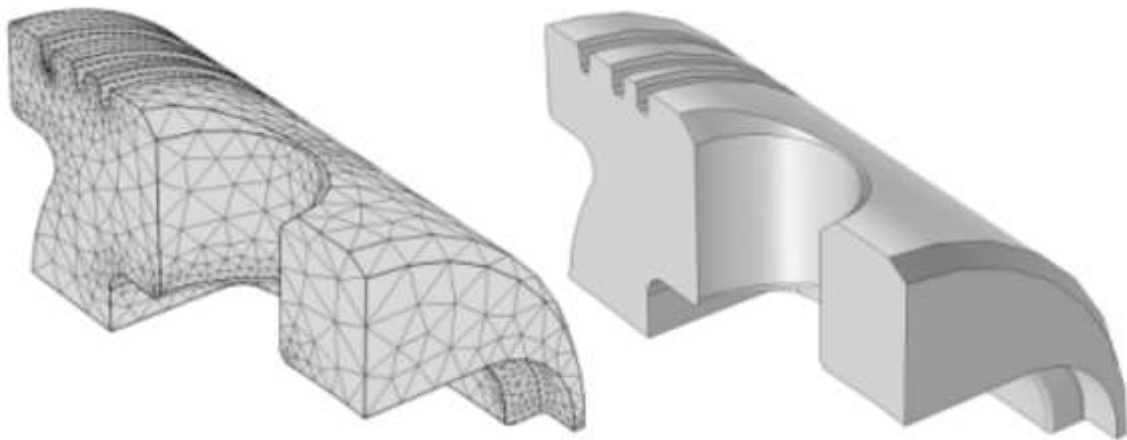
```
using namespace chaird;
// create a light source
light = new cSpotLight(world);
// attach light to camera
world->addChild(light);
// enable light source
light->setEnabled(true);
// position the light source
light->setLocalPos(۰,۶, ۰,۶, ۰,۵);
// define the direction of the light beam
```

<sup>۱</sup> Spot Lights

```
light->setDir(-۰,۰,-۰,۰,-۰.۰۵);
// set light cone half angle
light->setCutoffAngleDeg(۳۰);
// enable this light source to generate shadows
light->setShadowMapEnabled(true);
```

### ۳-۲-۵- اشیای متشکل از مش<sup>۱</sup>

در CHAI<sup>۳</sup>D یک مش چند وجهی که یک شی را می سازد از مجموعه ای از ورتکس ها<sup>۲</sup> (راس) و مثلث-ها تشکیل شده است که یک شی چند وجهی را می سازد. ورتکس شامل یک موقعیت مکانی و دیگر اطلاعات مانند رنگ است. دو ورتکس به وسیله ی یک ضلع به هم متصل می شوند و سه ورتکس با اتصال به هم از طریق سه ضلع یک مثلث را می سازند که ساده ترین چند ضلعی در فضای اقلیدسی است. از ترکیب این مثلث ها اجسام چند وجهی پدید می آیند. شکل زیر یک جسم متشکل از ورتکس را نشان می دهد.



شکل ۳-۱۲ جسم متشکل از ورتکس و مثلث

<sup>۱</sup> Mesh

<sup>۲</sup> Vertex



اشیا ساخته شده توسط مش‌ها قابلیت انتقال، چرخش، تغییر اندازه و گرفتن ویژگی‌های بافتی خاص را دارا می‌باشند. قطعه کد زیر چگونگی ساخت یک بافت مربعی شامل ۴ ورتکس و دو مثلث را نشان می‌دهد.

جدول ۳-۷ قطعه کد مربوط به تعریف ورتکس و مثلث‌ها

```
using namespace chai3d;
// create a mesh
object = new cMesh();
object->m_texture = cTexture3d::create();
object->m_texture->loadFromFile("myTexture.jpg");
object->setUseTexture(true);
object->m_material->setWhite();
int vertex0 = object->newVertex();
int vertex1 = object->newVertex();
int vertex2 = object->newVertex();
int vertex3 = object->newVertex();
object->m_vertices->setLocalPos(vertex0, -0.5, -0.5, 0.0);
object->m_vertices->setLocalPos(vertex1, 0.5, -0.5, 0.0);
object->m_vertices->setLocalPos(vertex2, 0.5, 0.5, 0.0);
object->m_vertices->setLocalPos(vertex3, -0.5, 0.5, 0.0);
// create two triangles by assigning their vertex IDs
object->m_triangles->newTriangle(vertex0, vertex1, vertex2);
object->m_triangles->newTriangle(vertex0, vertex2, vertex3);
```

اشیای ساخته شده از یک نوع مش تنها می‌توانند یک خاصیت فیزیکی داشته باشند، حال اگر یک جسم داشته باشیم که هر نقطه دارای خاصیت فیزیکی متفاوتی باشد باید یک کلاس داشته باشیم که توانایی ایجاد این شی را داشته باشیم لذا CHAI3D یک کلاس با عنوان cMultiMesh معرفی کرده است که امکان تعریف خواص فیزیکی برای هر شی متشکل از مش را به صورت مجزا فراهم کرده است. قطعه کد زیر چگونگی تعریف چند مش مجزا را نشان می‌دهد.

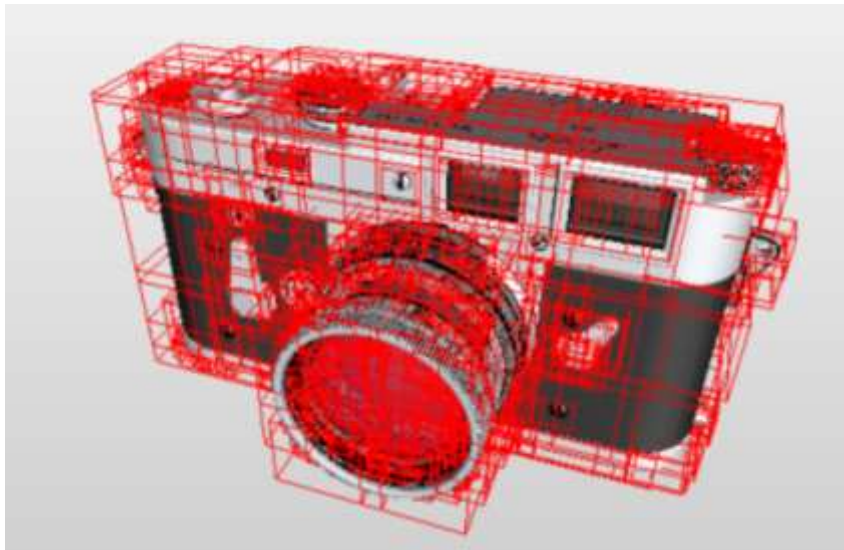
جدول ۸-۳ تعریف چند مش به صورت مجزا

```
using namespace chaird;
// create a multi mesh object
cMultiMesh* multiMesh = new cMultiMesh();
// add multi mesh object to world
world->addChild(multiMesh);
// create a first mesh
cMesh* mesh1 = multiMesh->newMesh();
// create a second mesh
cMesh* mesh2 = multiMesh->newMesh();
```

### ۳-۲-۶ الگوریتم تشخیص برخورد

تشخیص برخورد برای محاسبه‌ی تصادم بین عملگر و مش‌هاست. برای محاسبه‌ی سریع این روند CHAI<sup>۳</sup>D از سلسله مراتب حجم سنجی<sup>۱</sup> استفاده می‌کند. این سلسله مراتب یک ساختار درختی بر روی اشیا هندسی می‌سازد. تمام اشیای هندسی به حجم محدودی بسته می‌شوند که گره‌های برگ درخت را تشکیل می‌دهند. سپس این گره‌ها به صورت مجموعه‌های کوچک گروه‌بندی شده و در حجم بزرگتر محصور می‌شوند. در CHAI<sup>۳</sup>D، سلسله مراتب حجم سنجی باید پس از تعریف مثلث‌ها ایجاد شود. در نهایت با حجم بندی یک جسم و مقایسه‌ی موقعیت عملگر و درخت بدست آمده، برخورد با جسم به این صورت تشخیص داده می‌شود که اگر نشانگر در محدوده‌ی حجم گره‌ها قرار گرفته باشد برخورد با جسم اعلام می‌گردد. شکل زیر یک جسم حجم سنجی شده را نشان می‌دهد.

<sup>۱</sup> bounding volume hierarchy



شکل ۳-۱۳ جسم حجم سنجی شده

قطعه کد زیر چگونگی فعال سازی الگوریتم تشخیص برخورد را نشان می دهد.

جدول ۳-۹ فعال سازی الگوریتم تشخیص برخورد

```
// create axis aligned bounding volume hierarchy  
multiMesh->createAABBCollisionDetector(toolRadius);  
multiMesh->isInContactWith(object);
```

### ۳-۲-۷ اجسام ساده ی اولیه

در CHAI<sup>3D</sup> یک امکان برای ساخت اجسام سه بعدی ساده مانند کره، استوانه، مکعب و ... فراهم شده است (شکل ۳-۱۴) تا سرعت توسعه ی نرم افزار افزایش پیدا کند و شخص برای طراحی این اجسام زمان مصرف نکند. همچنین با ترکیب این اجسام می توانیم جسم های پیچیده تر را بدون آنکه نیاز به تعریف مش باشد بسازیم.



شکل ۳-۱۴ اجسام اولیه در CHAI۳D

به عنوان مثال در صورتی که بخواهیم یک کره بسازیم طبق قطعه کد زیر با دادن شعاع آن به کلاس سازنده، جسم کروی ساخته خواهد شد.

جدول ۳-۱۰ قطعه کد ساخت کره

```
// create a sphere and define its radius
object = new CShapeSphere(۰,۳);
// add object to world
world->addChild(object);
// set haptic properties
object->m_material->setStiffness(۵۰۰);
// create a haptic surface effect
object->createEffectSurface();
```

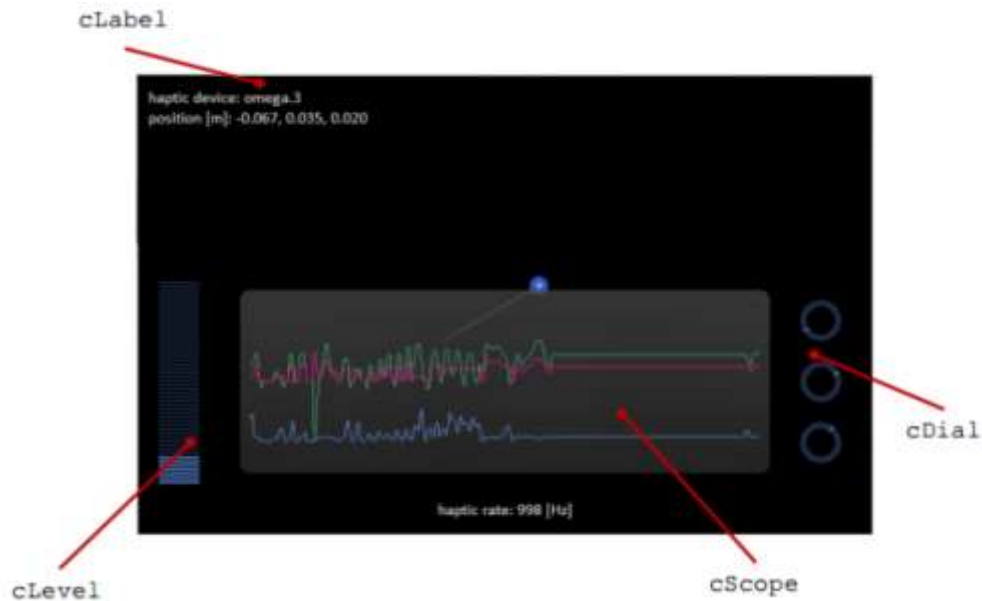
### ۳-۲-۸- ابزارک‌ها

در CHAI۳D ابزارهایی مهیا شده است که اطلاعات و وضعیت را می‌توان به صورت گرافیکی به کاربر نشان داد. از جمله این ابزارها می‌توان به برچسب‌های اطلاعاتی<sup>۱</sup>، نمایشگر سیگنال<sup>۲</sup> و نشانگر پله‌ای<sup>۱</sup>

<sup>۱</sup> Label

<sup>۲</sup> Scope

اشاره کرد. کلاس‌هایی که این ابزارها را فراهم می‌کنند عبارتند از: cLabel, cLevel, cDial, cScope. شکل زیر این ابزارها را نشان می‌دهد.



شکل ۳-۱۵ ابزارهای CHAI۳D

به عنوان مثال برای ساخت برجسب اطلاعاتی همانند قطعه کد زیر عمل می‌کنیم.

جدول ۳-۱۱ ساخت برجسب اطلاعاتی

```
// create a font
cFont* font = NEW_CFONTCALIBRI۲۰();
// create a label
cLabel* label = new cLabel(font);
// add label to front layer of camera
camera->m_frontLayer->addChild(label);
// assign color to label
label->m_fontColor.setBlack();
// assign text to label
label->setText("my message");
// position widget on screen
```

```
label->setLocalPos(۱۰۰, ۱۲۰);
```

### ۹-۲-۳- حلقه‌ی هپتیکی

زمانی که دستگاه هپتیک شناسایی شد و عملگر آن در دنیای مجازی شناخته شد لازم است که با یک سرعت مناسب تمام حرکات عملگر دریافت شود و با توجه به این اطلاعات نیروهای بازخوردی در صورت برخورد با اجسام مجازی به دستگاه اعمال شود، به همین دلیل یک حلقه لازم داریم تا این تعامل را برقرار سازد. قطعه کد زیر چگونگی ایجاد این حلقه و دستورات لازم برای اعمال بازخورد نیرویی به عملگر را نشان می‌دهد.

جدول ۱۲-۳ حلقه‌ی هپتیکی

```
void updateHaptics(void)
{
// simulation is now running
simulationRunning = true;
// main haptic simulation loop
while(simulationRunning)
{
// compute global reference frames for each object
world->computeGlobalPositions(true);
// update position and orientation of tool
tool->updateFromDevice();
// compute interaction forces
tool->computeInteractionForces();
// send forces to haptic device
tool->applyToDevice();
}
}
```

در این قسمت تقریباً تمام امکاناتی که از کتابخانه‌ی CHAI۳D در پروژه استفاده شده است را به صورت موردی توضیح دادیم. این کتابخانه امکانات دیگری مانند صداها، اجسام دارای حجم، بافت و ماده، داده-

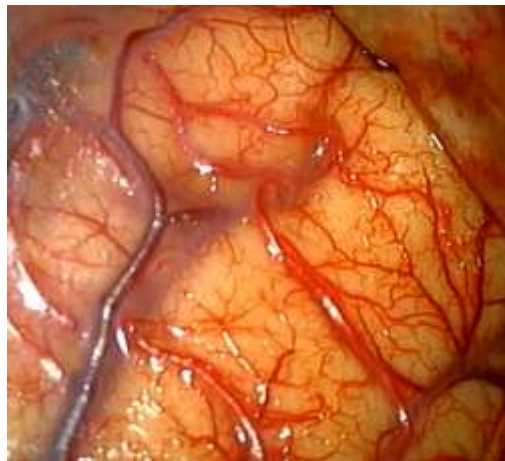
های ابری<sup>۱</sup> و ... را دارا می‌باشد که به دلیل استفاده نکردن در این پروژه از توضیح دادن آن‌ها صرف نظر می‌کنیم.

### ۳-۳- الگوریتم و روش‌های مورد استفاده

در ادامه به توضیح الگوریتم و روش‌های استفاده شده برای ایجاد شبیه‌ساز جراحی مغز در این پروژه می‌پردازیم.

در ابتدا، همانطور که در قسمت قبل گفته شد باید یک دنیای مجازی بسازیم و در آن دوربین و نور را برای نمایش محیط اضافه کنیم. سپس دستگاه هپتیک را شناسایی و عملگر آن را در محیط شبیه‌سازی شده قرار دهیم. این قسمت از کد پروژه که یک قالب کلی است و همچنین قسمت‌های دیگر که به این شکل می‌باشند در پیوست قرار داده شده‌اند.

پس از ساخت دنیای مجازی باید بافت مغز مدل‌سازی شود، برای این کار با استفاده از یک صفحه‌ی دو بعدی که از تعدادی ورتکس تشکیل شده است یک بافت به صورت مش تعریف شد. برای شبیه‌سازی بافت واقعی مغز ویدیوی عمل جراحی مغز یک انسان تهیه گردید و از بافت آن یک تصویر تهیه کردیم مطابق شکل ۳-۱۶. حال برای مدل‌سازی نیاز به یک سری پیش پردازش است که با کمک نرم‌افزار متلب صورت گرفت. در ادامه پیش پردازش صورت گرفته را شرح می‌دهیم.



شکل ۳-۱۶ بافت مغز استفاده شده

<sup>۱</sup> Point Cloud

### ۳-۳-۱- پیش پردازش

صفحه‌ی ورتکسی که تعریف کردیم، در حالت اولیه ابعاد ۵۰۰ در ۵۰۰ داشت اما چون سرعت به روز رسانی افت می‌کرد و شبیه‌سازی با سرعت کمی به روز می‌شد مجبور به کاهش ابعاد به ۱۲۰ در ۱۲۰ شدیم.

در اینجا لازم به ذکر است برای نمایش محیط مجازی دو حلقه پیوسته در حال به روز رسانی شدن است. یکی حلقه‌ی هپتیک و دیگری حلقه‌ی گرافیکی، حلقه‌ی هپتیک را در قسمت امکانات کتابخانه‌ی CHAI<sup>3D</sup> توضیح دادیم که مربوط به محاسبه‌ی موقعیت و اعمال پس‌خورد نیرویی به عملگر دستگاه هپتیک است. این حلقه باید فرکانس به روز رسانی بیش‌تر از ۱۰۰۰ هرتز را داشته باشد تا نیرو در نظر کاربر پیوسته و یکنواخت جلوه کند. در صورتی که این فرکانس کم‌تر شود عملگر در صورت برخورد به بافت شروع به لرزش می‌کند و بازخورد نیرویی صحیح مختل می‌شود.

حلقه‌ی گرافیکی مربوط به سرعت به روز رسانی صفحه نمایش است، انسان تغییراتی که کمتر از ۱۰۰ میلی ثانیه طول بکشد را به صورت پیوسته مشاهده می‌کند. لذا برای یک حلقه‌ی گرافیکی لازم است فرکانس به روز رسانی بیش‌تر از ۱۰ هرتز باشد که در کارهای نمایشی معمولاً سعی بر این است تا این سرعت را به ۶۰ هرتز یا ۶۰ فریم<sup>۱</sup> در ثانیه برسانند.

حال همانطور که اشاره شد، یکی از مشکلاتی که در این پروژه با آن روبرو بودیم مصالحه بین سرعت به روز رسانی و کیفیت بافت مورد استفاده بود که در صورت افزایش تعداد ورتکس‌ها سرعت حلقه‌ی هپتیک و گرافیکی به شدت افت می‌کرد و نیروها به صورت ضربه‌ای به دست بازخورد می‌شد.

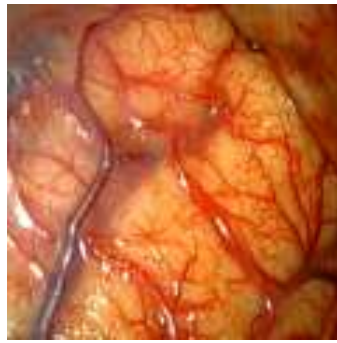
بنابراین اولین پیش پردازش انجام شده کاهش ابعاد تصویر بافت به ۱۲۰ در ۱۲۰ بود. این کار با قطعه کد زیر صورت گرفت و نتیجه به صورت شکل ۳-۱۷ حاصل شد.

جدول ۳-۱۳ قطعه کد تغییر اندازه‌ی تصویر

```
pic=imread('capture.jpg');
resiz_pic=imresize(pic,[۱۲۰ ۱۲۰]);
```

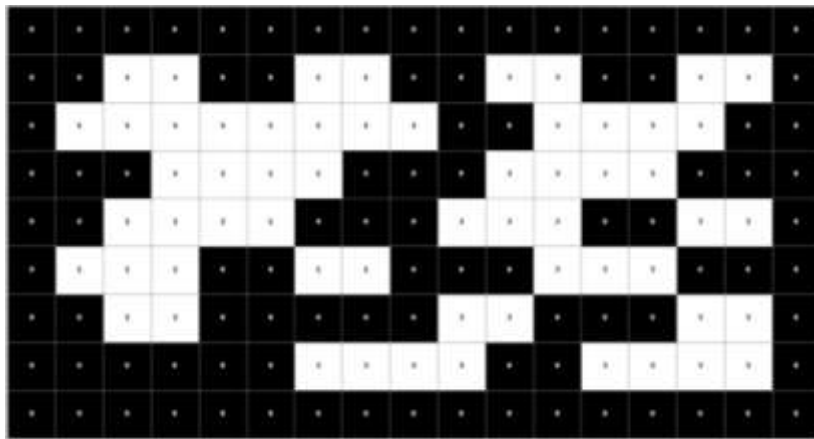
<sup>۱</sup> Frame





شکل ۳-۱۷ تصویر تغییر اندازه داده شده

در قسمت بعد پیش پردازش لازم بود برای سه بعدی سازی، رگ ها قطعه بندی شوند تا بتوان در محیط شبیه سازی شده رگ ها به صورت سه بعدی نمایش داده شوند. لذا برای این کار باید از یک الگوریتم قطعه بندی تصویر استفاده کنیم. الگوریتمی که در اینجا از آن استفاده شده است الگوریتم برچسب گذاری یک تصویری باینری است. در این الگوریتم روند به این صورت است که پیکسل هایی که مقدار ۱ را دارند و در همسایگی یکدیگرند به عنوان یک قطعه شناخته می شوند. به عنوان مثال تصویر باینری زیر دارای دو قطعه است که با استفاده کردن این الگوریتم مقدار پیکسل هر یک از این قطعات بسته به این که قطعه ای اول باشد یا دوم مقدار ۱ یا ۲ را می گیرد.



شکل ۳-۱۸ یک تصویر باینری دو قطعه ای

حال برای استفاده از این الگوریتم لازم است ابتدا تصویر بافت مغزی را به یک تصویر باینری تبدیل کنیم و با استفاده از این الگوریتم رگ ها را مشخص کنیم. قطعه کد زیر چگونگی این روند را نمایش می دهد، ابتدا تصویر را باینری می کند و در نهایت رگ های قطعه بندی شده را به نمایش می گذارد.

جدول ۳-۱۴ قطعه بندی تصویر

```

gray_pic=resiz_pic(:,:,۲);
se = strel('disk',۱۰);
tophat=imtophat(gray_pic,se);
bin_pic=imcomplement(im2bw(tophat,۰,۰.۵));
se2=strel('disk',۲);
dilated=imdilate(bin_pic,se2);
segmented=bwlabel(dilated);
figure;imshow(segmented)

```

نتیجه‌ی این کد به صورت شکل زیر است که همانطور که مشاهده می‌کنید هر قطعه دارای یک شدت مختص به خود است.



شکل ۳-۱۹ تصویر قطعه بندی شده

حال برای آنکه قسمت‌های مورد نظر خود را از تصویر استخراج کنیم و فقط رگ‌ها را به نمایش بگذاریم، باید با روش صحیح و خطا قسمت‌های مورد نظر را مشخص کنیم و در نهایت فقط آن قسمت‌ها را نمایش دهیم. کد زیر این عمل را انجام می‌دهد.

جدول ۳-۱۵ قطعه کد جدا سازی رگ‌ها

```

[m n=size(segmented);
vessel=zeros(m,n);
for i=۱:m
    for j=۱:n

```

```

        if(segmented(i,j)==۱ || segmented(i,j)==۸ || segmented(i,j)==۵
|| segmented(i,j)==۲ || ...
            segmented(i,j)==۱۵ || segmented(i,j)==۱۶ ||
segmented(i,j)==۱۸ ) %۲ ۵ ۸ ۱۵ ۱۶ ۱۸
            vessel(i,j)=۱;
        end
    end
end
figure;imshow(vessel);

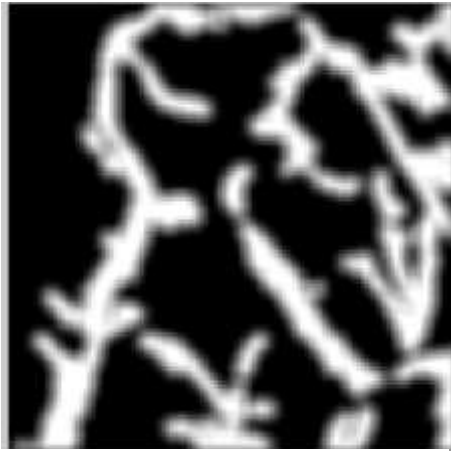
```

نتیجه‌ی بدست آمده تصویر زیر است.



شکل ۳-۲۰ رگ‌های مورد نظر استخراج شده

در سه بعدی سازی رگ از شدت پیکسل‌های تصویر استفاده کردیم، اما این استفاده‌ی مستقیم یک مشکل اساسی داشت، تمام رگ ارتفاع یکسانی نسبت به سطح مغز داشت و به صورت دایره‌ای مدل نشده بود لذا قبل از استفاده از تصویر بالا به عنوان تعیین ارتفاع رگ یک فیلتر میانگین‌گیر استفاده کردیم. در نتیجه شدت پیکسل‌های رگ به صورت پلکانی افزایش یافت و رگ حالت دایره‌ای سه بعدی به خود گرفت. شکل زیر تصویر بدست آمده‌ی نهایی برای این کار را نشان می‌دهد.



شکل ۳-۲۱ تصویر نهایی برای سه بعدی سازی رگ

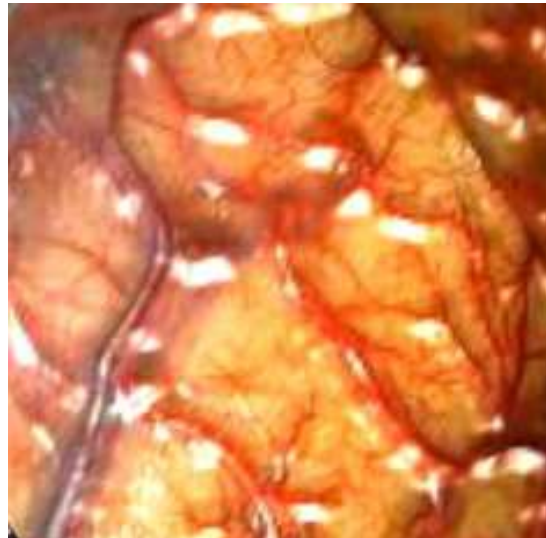
پس از این پیش پردازش دو تصویر بدست آمده را به محیط کدنویسی در Visual Studio اضافه کردیم و رنگ و رتکس‌ها و موقعیت آن‌ها را با توجه به این دو تصویر تعیین کردیم. قطعه کد زیر چگونگی انجام این عمل را نشان می‌دهد.

جدول ۳-۱۶ قطعه کد ایجاد بافت در محیط مجازی

```
index = object->newVertex();
// get color of image pixel
cColorb      color;
image.getPixelColor(x, y, color);
// compute vertex height by averaging the color components RGB and
scaling the value.
double vessel_h = ۰;
cColorb      colorseg;
segmented.getPixelColor(x, y, colorseg);
if ((double)colorseg.getLuminance() != ۰)
{
vessel_h = ۰,۰۰۰۰۶*(double)colorseg.getLuminance();
}
double px = scalex * (double)x - offsetX;
double py = -scaley * (double)y + offsetY;
// set vertex position
object->m_vertices->setLocalPos(index, px, -py, -۰,۳+ vessel_h);
```

```
//set vertex color
object->m_vertices->setColor(index, color);
```

نتیجه‌ی بافت مدل شده، به شکل زیر می‌باشد.

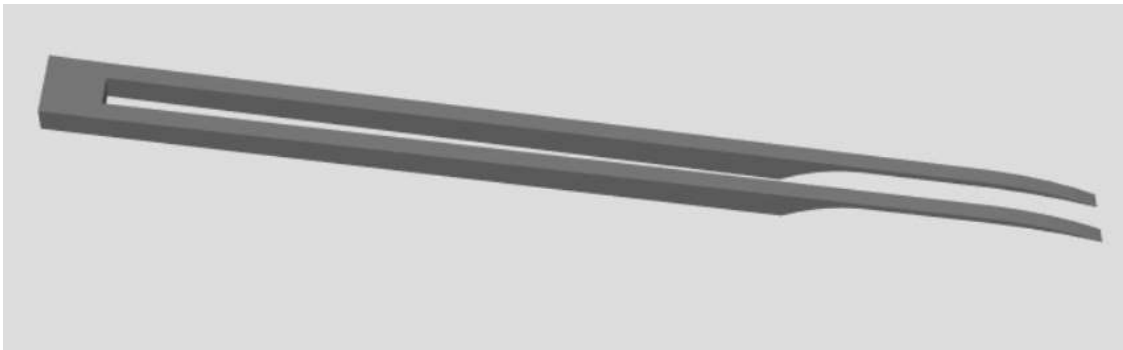


شکل ۳-۲۲ بافت شبیه‌سازی شده در محیط مجازی

پس از این مدل سازی الگوریتم تشخیص برخورد را فعال می‌کنیم تا در صورت تعامل با عملگر، بافت تغییر حالت دهد و نیروهای مدل شده به دست کاربر منتقل شوند.

### ۳-۳-۲- طراحی مدل ابزار منعقد ساز دو قطبی

برای اینکه کاربر بتواند ارتباط خوبی با محیط شبیه‌سازی شده برقرار کند لازم است که ابزارها و وسایلی که در محیط واقعی مورد استفاده قرار می‌گیرند، در محیط مجازی هم استفاده شوند. لذا لازم دانستیم که ابزار منعقد ساز دو قطبی را مانند مدل واقعی آن به محیط مجازی اضافه کنیم. برای این کار با استفاده از نرم افزار Solid Works به طراحی مدل سه بعدی آن پرداختیم که نتیجه در شکل ۳-۲۳ نشان داده شده‌است. ابزار منعقد ساز که در عمل‌های جراحی استفاده می‌شود به صورت شکل ۳-۲۴ است که همانطور که مشاهده می‌کنید سعی بر آن شده است که تا حد امکان ابزار طراحی شده معادل ابزار واقعی باشد.

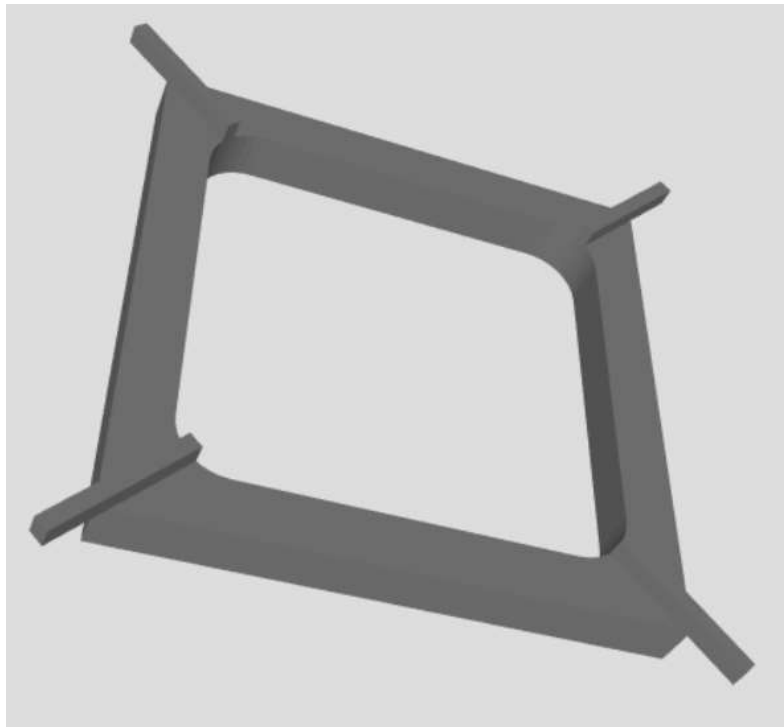


شکل ۳-۲۳ ابزار منعقدساز دو قطبی مدل شده



شکل ۳-۲۴ ابزار منعقدساز دو قطبی

ابزار دو قطبی را به عملگر هپتیک مربوط کردیم که با حرکت دادن عملگر، ابزار مدل شده هم در همان جهت حرکت می کند و با برخورد به اجسام نیروهای شبیه سازی شده به عملگر منتقل می شوند. این ابزار را با یک ابزار چارچوب مانند (شکل ۳-۲۵) که برای قرار گرفتن در اطراف بافت مغزی طراحی شده است را در محیط مجازی اضافه کردیم.



شکل ۳-۲۵ چارچوب برای قرار گرفتن در اطراف بافت مغزی

### ۳-۳-۳- تغییر شکل بافت

یکی از چالش‌های اصلی این پروژه ارایه‌ی الگوریتمی برای تغییر شکل بافت در هنگام برخورد عملگر با بافت بود. همانطور که پیشتر اشاره شد برای تعامل کاربر با محیط مجازی دو حلقه همواره در حال به روز رسانی هستند، حلقه‌ی هپتیکی و حلقه‌ی گرافیکی، در حلقه‌ی گرافیکی اطلاعات بصری سازی شده به کاربر نمایش داده می‌شود و در حلقه‌ی هپتیکی محاسبات مربوط به برخورد با اشیا، نیروهای پسخوردی و تغییر شکل محاسبه می‌شود.

در تلاش اول برای ایجاد تغییر شکل ابتدا نقطه‌ی برخورد عملگر با بافت را محاسبه کردیم و سپس با توجه به فاصله‌ی ورتکس‌ها از نقطه‌ی برخورد، یک فرو رفتگی با فرمول زیر تعریف کردیم. که  $d$  فاصله‌ی نگاشت شده بین صفر و یک است.

$$0.5 + 0.5 * \cos(d * \pi)$$

این الگو به این صورت عمل می کند که اگر فاصله ی نقطه ی برخورد با ورتکس برابر صفر باشد ( $d=0$ ) میزان فرورفتگی ورتکس برابر یک و اگر بیشتر از شعاع تعریف شده باشد  $d$  برابر یک می شود و مقدار عبارت صفر خواهد شد در نتیجه میزان فرورفتی برابر صفر خواهد بود.

حال اگر بخواهیم میزان فرورفتگی بافت متناسب با موقعیت عملگر در تعامل با بافت باشد باید موقعیت آن را بخوانیم و در عبارت فوق ضرب کنیم در نتیجه میزان فرورفتگی نقطه ای که دقیقاً با عملگر متعامل است برابر موقعیت عملگر و هر چه از آن دورتر شویم به صفر نزدیکتر می شود.

$$posZtool * (0.5 + 0.5 * \cos(d * \pi))$$

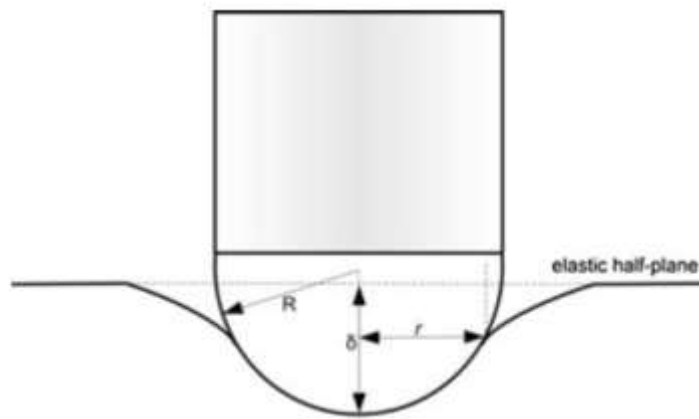
اما این مدل برای تغییر شکل یک مشکل اساسی داشت. به علت اینکه حلقه ی هپتیکی فاصله ی تمامی ورتکس ها از عملگر را می سنجید و با توجه به فاصله ی آن ها مختصات عمودیشان را تغییر می داد، سرعت به روز رسانی به ۴۰۰ هرتز رسید و نیروهایی که به دست کاربر منتقا می شد به صورت گسسته حس می شد و دستگاه شروع به لرزش می کرد.

برای بهبود این الگوریتم و حل مشکل این راه حل ارایه شد که به جای آنکه فاصله ی تمام ورتکس ها از نوک عملگر سنجیده شود، با توجه به نوک عملگر شماره ی ورتکس هایی که در حوزه ی تغییر شکل قرار دارند را محاسبه کنیم و با توجه به این شماره ها مختصات آن ها را جابجا کنیم تا تغییر شکل حاصل شود. در این الگو اضافه بر استفاده از تابع کسینوسی، برای همواری بیشتر تغییر شکل از یک تابع نمایی میرا شونده نیز مطابق فرمول زیر استفاده کردیم.

$$posZtool * \exp(-4 * d) * (0.5 + 0.5 * \cos(d * \pi))$$

با توجه به شکل زیر می توان عملکرد این مدل را به خوبی دریافت. هرچه عملگر در درون بافت بیشتر فرو رود شعاع تغییر حالت بیشتر خواهد شد و با توجه به تابع فوق موقعیت عمودی هر ورتکس بدست می آید.





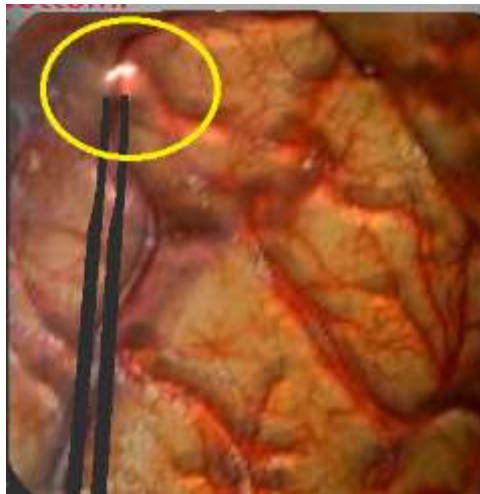
شکل ۳-۲۶ مدل تغییر شکل بافت در تعامل با ابزار

قطعه کدی که برای این کار نوشته شد به صورت زیر است. در ابتدا ورتکس‌هایی که در محدوده‌ی تغییر شکل قرار می‌گیرند، استخراج می‌شوند و سپس با توجه به موقعیت عملگر درون بافت و تابع تعریف شده تغییر موقعیت اعمال می‌شود.

جدول ۳-۱۷ قطعه کد مربوط به تغییر شکل بافت

```
cVector2d posVertex;
double relativeDistance = sqrt(i*i + j*j) / (deformation_radius);
float clampedRelativeDistance = cClamp(0,1)(relativeDistance);
double w=posTool.z()*exp(-clampedRelativeDistance * 4) * (0.5 + 0.5*cos(relativeDistance * C_PI));
if ( nv >= 0 && nv < max && nv<((column + i)*(sizeX)) && nv>((column + i - 1)*(sizeX)))
{
desired= mapposition[2][(int)nv];
posVertex = object->m_vertices->getLocalPos(nv);
posVertex.z(w + desired);
object->m_vertices->setLocalPos(nv, posVertex);
}
```

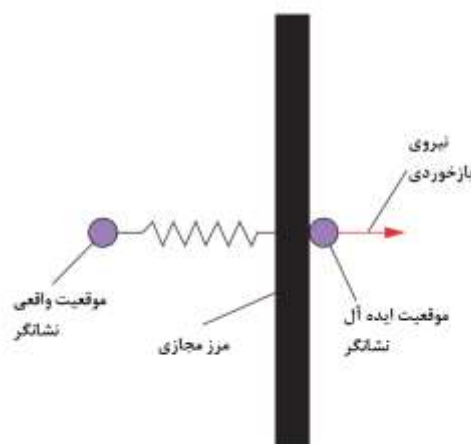
در شکل زیر تغییر شکل بافت در هنگام تعامل ابزار با بافت را مشاهده می‌کنید که یک فرورفتگی نیم-کره‌ای در بافت به وجود آمده که در صورت حرکت دادن ابزار این نیم‌کره نیز هم جهت با ابزار تغییر موقعیت می‌دهد و حس بصری دستکاری بافت به کاربر منتقل می‌شود.



شکل ۳-۲۷ تغییر شکل بافت در تعامل با بزار منعقد ساز

### ۳-۳-۴- مدل نیرویی بافت شبیه سازی شده

به طور معمول برای مدل سازی نیروی اجسام از مدل جرم-فنر-دمپر استفاده می شود. برای مدل سازی نیروی بافت مغزی به علت محدودیت محاسبات و کم نشدن سرعت به روز رسانی حلقه‌ی هپتیکی فقط از مدل فنری استفاده شد. در این مدل موقعیت نشانگر و مرز تعریف شده برای بافت محاسبه می شود و در صورتی که نشانگر از مرز مجازی بافت عبور کند با محاسبه‌ی فاصله‌ی بین مرز و موقعیت نشانگر و ضرب کردن ثابت فنر بافت مدل شده، نیرو بدست می آید. شکل زیر به خوبی مفهوم این مدل سازی را نشان می دهد.



شکل ۳-۲۸ مدل استفاده شده برای محاسبه‌ی نیرو

مدول یانگ یک بافت مغزی به طور متوسط ۳ کیلو پاسکال است که با استفاده از این مدول باید ضریب سختی فنر برای بافت مغز محاسبه شود و در مدل نیرویی شبیه ساز قرار گیرد. برای محاسبه ی این ضریب مطابق فرمول های تنش و کرنش عمل می کنیم.

$$\sigma = E\varepsilon \quad (۱)$$

که در آن  $\sigma$  تنش،  $E$  مدول یانگ و  $\varepsilon$  کرنش است. برای تنش و کرنش به صورت مجزا فرمول های زیر موجود است.

$$\sigma = \frac{F}{A}, \quad \varepsilon = \frac{\Delta L}{L} \quad (۲)$$

$F$  نیرو،  $A$  سطح مقطع،  $\Delta L$  تغییر طول و  $L$  طول اولیه است. با جایگذاری فرمول های ۲ در ۱ نتایج زیر حاصل شد:

$$\frac{F}{A} = \frac{\Delta L}{L} * E$$

$$F = K * \Delta L$$

$$K = \frac{AE}{L} \quad (۳)$$

در رابطه ی ۳،  $K$  ضریب سختی فنر بدست آمده از طریق مدول یانگ است لذا با توجه به اطلاعاتی که از بافت مغزی و مدلی که در محیط مجازی، شبیه سازی شده است به محاسبه ی ضریب سختی معادل می پردازیم. بافت مغزی مدل شده در محیط مجازی دارای ۶ سانتی متر عرض و ۶ سانتی متر طول است و اگر مغز را با یک نیم کره ی ساده مدل کنیم، شعاع آن به طور تقریبی ۸ سانتی متر و مدول یانگ آن ۳ کیلو پاسکال است. بنابراین با توجه به اطلاعات فوق و رابطه ی ۳ داریم:

$$K = \frac{۰.۰۶ * ۰.۰۶ * ۳۰۰۰}{۰.۰۸} = ۱۳۵ \text{ Nm}$$

بنابراین ثابت فنری برای بافت مغزی بدست آمد، اکنون با استفاده از این ثابت و محاسبه‌ی میزان فرورفتگی منعقدکننده‌ی دو قطبی در بافت، در حلقه‌ی هپتیکی هربار نیرو محاسبه و به عملگر دستگاه اعمال می‌شود. قطعه کد زیر این عمل را انجام می‌دهد.

جدول ۱۸-۳ مدل سازی نیروی بافت مغزی

```
cVector3d desiredPosition;
cVector3d force(0, 0, 0);
desiredPosition.set(posTool.x(), posTool.y(), mapposition[۲][index]);
double Kp = ۱۳۰; // [N/m]
// compute linear force
forceField = Kp * (desiredPosition - posTool);
force.add(forceField);
hapticDevice->setForce(force);
```

گام دیگر مدل سازی نیرویی، بحث اصطکاک بافت مغزی است. همانطور که می‌دانیم، نیروی اصطکاک از ضرب ضریب اصطکاک جنبشی در نیروی عمودی تکیه‌گاه بدست می‌آید.

$$F = \mu_k * N$$

در رابطه‌ی اخیر  $F$  نیروی اصطکاک،  $\mu$  ضریب اصطکاک جنبشی و  $N$  نیروی عمودی تکیه‌گاه است.

نیروی عمودی تکیه‌گاه همان نیرویی است که در قسمت قبل با مدل فنر بدست آمد، حال لازم است که این نیرو در جهت مخالف حرکت نشانگر به آن اعمال شود. برای این کار نیروی عمودی روی محورهای افقی با توجه به زاویه‌هایی که نشانگر با بافت دارد، توزیع می‌شود و با یک ضریب به عملگر هپتیک در جهت عکس حرکت آن اعمال می‌شود. قطعه کد زیر در حلقه‌ی هپتیکی این عمل را انجام می‌دهد.

جدول ۱۹-۳ قطعه کد اعمال نیروی اصطکاک

```
cMatrix3d AngleTool = tool->getDeviceGlobalRot();
cVector3d Angle = AngleTool.getCol0();
double friction_const = 0.۱;
forceFieldfriction.set(-friction_const*sin(Angle.z()*C_PI_DIV_۲)*
friction_const*sin(Angle.y()*C_PI_DIV_۲)*forceField.z(), -
friction_const*sin(Angle.z()*C_PI_DIV_۲)*
friction_const*cos(Angle.y()*C_PI_DIV_۲)*forceField.z(), 0);
```

```
force.add(forceFieldfriction);
hapticDevice->setForce(force);
```

### ۳-۵- مدل منعقدساز رگ

در این شبیه ساز عمل جراحی مغز بر آن شدیم که یک قسمت از گام های اولیه ی عمل جراحی را شبیه سازی کنیم. بنابراین منعقدساز دو قطبی رگ پیشنهاد شد. در قسمت های قبل به نحوه ی عملکرد و مدل سه بعدی ارایه شده برای منعقدساز دوقطبی به تفصیل اشاره شد. در این قسمت قصد داریم مدلی برای تعامل منعقدساز و بافت مغزی ارایه دهیم که در صورت تعامل منعقدساز و بافت ویژگی های بصری بافت چگونه تغییر پیدا کند.

برای آگاهی کافی از چگونگی تغییرات بصری بافت مغزی از دکتر علیرضا خوشنویسان، متخصص جراح مغز و اعصاب بیمارستان شریعتی تهران، در این مورد سوال شد و با توجه به توضیحات ایشان، تغییر رنگ بافت مغزی در هنگام منعقدسازی به این صورت است که ابتدا به رنگ روشن تر در می آید و با نگه داشتن بیش از حد منعقد ساز و یا اعمال نیروی بیشتر، بافت تیره می شود. با توجه به این توضیحات و ویدیوهایی که در این رابطه تهیه شد همین روند که دکتر توضیح دادند مشاهده گردید. بنابراین با توجه به این اطلاعات مدل تغییر بصری بافت ارایه شد.

می دانیم که برای نمایش رنگی یک تصویر از ترکیب سه رنگ قرمز، سبز و آبی استفاده می شود. در قسمت های قبل گفته شد که ورتکس های بافت مدل شده از یک تصویر برای رنگ بندی استفاده می کند بنابراین هر ورتکس دارای سه عنصر رنگی قرمز، سبز و آبی است و برای آنکه تغییر بصری در آن به وجود آید باید این سه را دستکاری کنیم.

با توجه به توضیحات دکتر خوشنویسان برای روشن تر شدن بافت در لحظه ی اول استفاده از منعقد ساز دوقطبی، باید شدت هر عنصر رنگی را بالا ببریم. یک مدل برای این کار، اضافه کردن شدت های عناصر رنگی برای رسیدن به مقدار نهایی ( مقدار ۲۵۵) برای روشن تر شدن بافت و سپس کاهش مقدار آن ها برای تیره شدن بافت در هنگام نگه داشتن زیاد منعقدساز است. این مدل یک اشکال اساسی داشت، با اضافه یا کم شدن شدت ها تناسب های رنگی به هم می ریخت و پس زمینه ی بافتی، رنگ های نامتوازن مانند بنفش یا آبی به خود می گرفت.

برای حل این مشکل و حفظ پس‌زمینه‌ی بافتی از یک ضریب تناسبی استفاده کردیم. این ضریب تناسبی از تقسیم مقدار ۲۵۵ بر بزرگترین عنصر رنگی ورتکس بدست آمد. به عنوان مثال فرض کنید یک ورتکس دارای شدت‌های ۱۴۷، ۱۰۰ و ۸۶ برای سه عنصر رنگی قرمز، سبز و آبی باشد بنابراین ضریب تناسبی برای این ورتکس از تقسیم ۲۵۵ بر ۱۴۷ بدست می‌آید که برابر ۱/۷۳۴ است. حال در لحظه‌ی اولی که منعقدساز با بافت تعامل برقرار می‌کند، این ضریب در همه‌ی عناصر رنگی ضرب می‌شود و بافت بدون تغییر پس زمینه رنگ روشنتری پیدا می‌کند و در صورتی که منعقد ساز ثابت بماند و نیرویی به بافت اعمال کند این ضریب کاهش می‌یابد تا شدت‌های ورتکس با ضرب شدن در این ضریب کاهش یابند و بافت تیره رنگ شود.

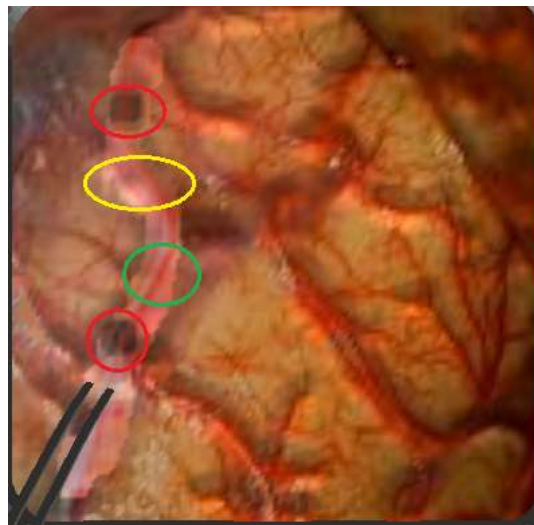
پیشتر گفته شد که تغییر بصری بافت توسط منعقدساز به زمان ثابت نگه داشتن آن در یک نقطه و نیرویی وارد شده به آن نقطه بستگی دارد، لذا برای دخیل کردن این تاثیرات، می‌بایست کاهش ضریب تناسبی را به این دو عامل وابسته کرد. همانطور که گفتیم حلقه‌ی هپتیکی با سرعت تقریباً ۱۰۰۰ هرتز به روز رسانی می‌شود بنابراین اگر از مقدار ضریب تناسبی مقداری را کم کنیم هر ثانیه هزار بار آن مقدار از ضریب تناسبی کم می‌شود در نتیجه وابستگی به زمان تامین می‌شود. برای ایجاد وابستگی به نیرو، مقداری را که از ضریب تناسبی کم می‌کنیم را به نیروی وارد شده از سمت کاربر وابسته می‌کنیم. قطعه کد زیر این روند را نشان می‌دهد.

جدول ۳-۲۰ قطعه کد ایجاد تغییر بصری منعقد ساز دوقطبی

```
if (forceField.z() > ۰)
{
color_coef = (forceField.z() / ۲۰۰۰);
if (array_color_coef[(int)nv] > coef_ther && nv != nv۳)
array_color_coef[(int)nv] = array_color_coef[(int)nv] - color_coef;
if (array_color_coef[(int)nv۲] > coef_ther && nv۲ != nv۴)
array_color_coef[(int)nv۲] = array_color_coef[(int)nv۲] - color_coef;
if (array_color_coef[(int)nv۳] > coef_ther) array_color_coef[(int)nv۳] =
array_color_coef[(int)nv۳] - color_coef;
if (array_color_coef[(int)nv۴] > coef_ther) array_color_coef[(int)nv۴] =
array_color_coef[(int)nv۴] - color_coef;
}
cColorb color;
```

```
color.set(vertexcolorArray[0][nv]*array_color_coef[nv],vertexcolorArray[1][nv]*array_color_coef[nv],vertexcolorArray[2][nv]*array_color_coef[nv]);
object->m_vertices->setColor(nv, color);
```

نتیجه‌ی نهایی منعقد سازی در شکل زیر نشان داده شده است، ناحیه‌هایی که تیره رنگ هستند نشان دهنده‌ی نگه داشتن بیش از حد منعقد ساز در آن نقطه‌اند که با نمایان‌گرهای قرمز رنگ در تصویر مشخص شده‌اند و ناحیه‌هایی که روشن‌تر هستند نشانگر نگه‌داشتن مدت زمان کم است که با نمایان‌گر زرد رنگ مشخص شده‌است و نمایانگر سبز رنگ بیانگر منعقد سازی متعادل است.



شکل ۳-۲۹ منعقد سازی رگ با مدل ارایه شده

### ۳-۳-۶- محیط کلی شبیه‌سازی شده

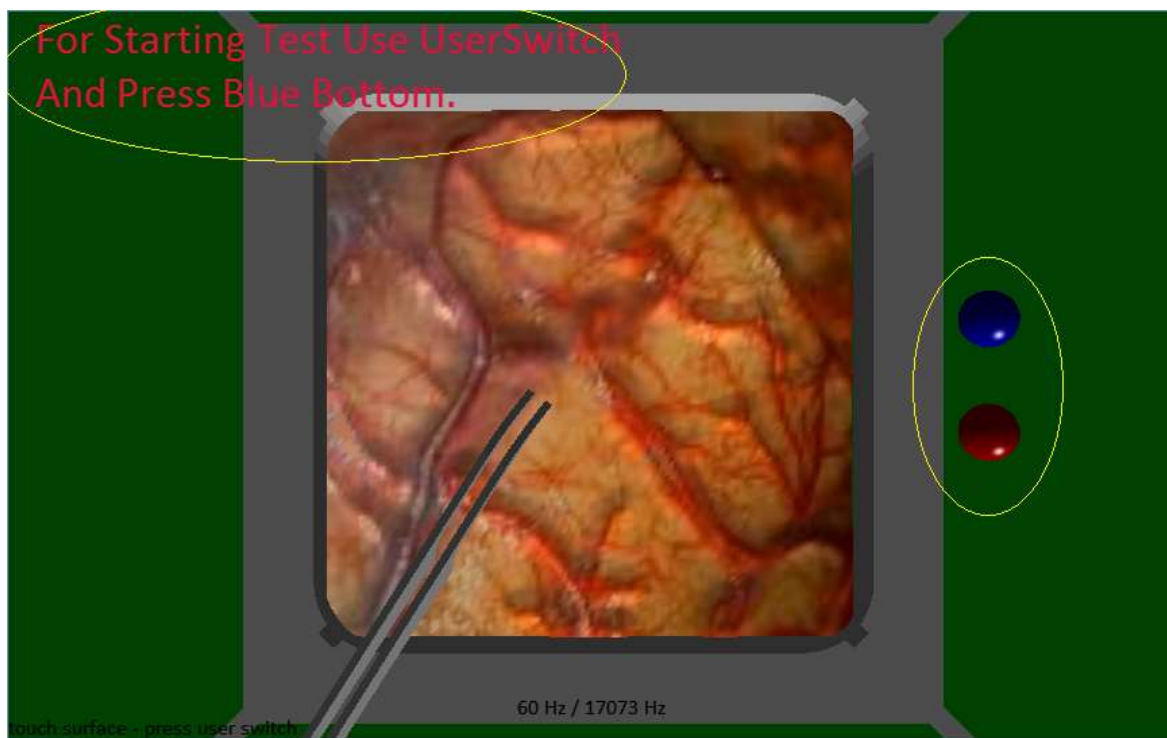
پس از اضافه کردن امکانات فوق و پیاده سازی الگوریتم‌های گفته شده، برای بهتر کردن محیط شبیه‌سازی شده رنگ اطراف بافت به سبز پررنگ تغییر داده شد تا حس پارچه‌های مورد استفاده در اتاق عمل القا شود. همچنین برای شروع عمل منعقد سازی و پایان آن از دو گوی با دو رنگ آبی و قرمز استفاده شد که این دو گوی در محیط مجازی جای داده شدند که یک خاصیت مغناطیسی به آن‌ها نسبت داده شد که در صورت نزدیک شدن ابزار به آن، ابزار به گوی بچسبد و کاربر راحت‌تر بتواند با زدن دکمه‌ی تیره رنگ روی عملگر (شکل ۳-۳۰) این امکان را فعال سازی کند. با زدن دکمه، گوی روشن می‌شود و کاربر از فعال شدن امکان منعقد سازی مطلع می‌شود.

پس از فعال سازی امکان منعقد سازی برای آنکه در محل مورد نظر منعقد سازی اتفاق بیفتد همان دکمه ی تیره رنگ روی عملگر دستگاه در نظر گرفته شد. بنابراین پس از فعال سازی با زدن این دکمه در هر جای بافت عمل انعقاد توسط منعقد ساز دوقطبی اتفاق می افتد.



شکل ۳-۳۰ کلید فعال سازی منعقد کننده و استفاده از منعقد کننده

برای راهنمایی کاربر نیز متنهایی در سمت بالا چپ، برای انجام آزمایشی که در فصل بعد بررسی خواهد شد، اضافه کردیم. در نهایت محیط شبیه سازی شده به صورت شکل ۳-۳۱ ارائه شد.



شکل ۳-۳۱ محیط نهایی شبیه ساز جراحی مغز



در نهایت این شبیه ساز را برای اعتبار سنجی، به محل کار دکتر علیرضا خوشنویسان، بیمارستان شریعتی تهران، برده شد و ایشان رضایت کلی خود را از شبیه ساز و نیروهایی که به صورت بازخوردی به دست اعمال می شود، اعلام کردند و نظرات سازنده ای در جهت رفع خطاها و پیشرفت پروژه پیشنهاد دادند.

### ۳-۴- جمع بندی

در این قسمت با امکانات ارایه شده برای دستگاه هپتیک و چگونگی استفاده از این امکانات آشنا شدیم. در نهایت با استفاده از این امکانات محیط مجازی مربوط به این پروژه که یک شبیه ساز منعقد ساز رگ مغزی بود، طراحی شد و گام به گام مدل های استفاده شده شرح داده شد. در فصل بعد به معرفی آزمایش تعریف شده و تحلیل نتایجی که از انجام این آزمایش بر روی داوطلبان حاصل شد پرداخته می شود.

## فصل چهارم

### تعریف آزمایش و تحلیل نتایج آن بر روی داوطلبان

## مقدمه

با توجه به فصول قبل دریافتیم که در این پروژه یک شبیه‌ساز منعقد ساز رگ در عمل جراحی مغز طراحی شده‌است. قصد داریم برای ارزیابی روند یادگیری افراد مختلف از این شبیه‌ساز استفاده کنیم. برای این کار باید با توجه به امکانات شبیه‌ساز یک آزمایش طراحی کنیم و از داوطلبان بخواهیم که این آزمایش را انجام دهند. با انجام این آزمایش تمام اطلاعات قابل ثبت برای هر داوطلب ذخیره می‌شود و در نهایت به تحلیل این داده‌ها می‌پردازیم.

### ۴-۱- طراحی آزمایش و داده‌گیری از داوطلبان

در این قسمت با توجه به امکانات شبیه ساز یک آزمایش<sup>۱</sup> برای ارزیابی افراد طراحی شده است که در ادامه به توضیح قسمت‌های این آزمایش و روند انجام آن پرداخته می‌شود.

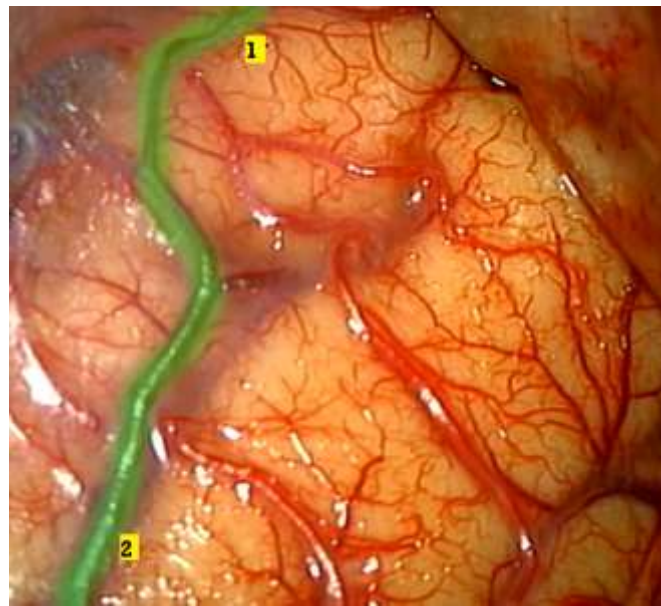
#### ۴-۱-۱- تعریف

آزمایشی که در نظر گرفته شد به این صورت است که در بافت مغزی شبیه سازی شده یک رگ انتخاب شد و با دستورالعمل‌هایی که قبل از انجام آزمایش به کاربر ارائه گردید مقرر شد، رگ مورد نظر را به صورت کامل با عملگر هپتیک منعقد کنند. عمل منعقدسازی رگ مورد نظر برای هر نفر ده مرتبه تکرار می‌شود و برای هر آزمایش داده‌های کاربر ذخیره می‌شود تا روند یادگیری او در طی ده آزمایش بررسی شود.

شکل زیر رگ انتخاب شده از بافت شبیه سازی شده، که با رنگ سبز در تصویر مشخص شده‌است، را نشان می‌دهد. کاربر باید از نقطه‌ی ۱ به سمت نقطه‌ی ۲ شروع به منعقد سازی کند و کل رگ را منعقد کند.

---

<sup>۱</sup> Task



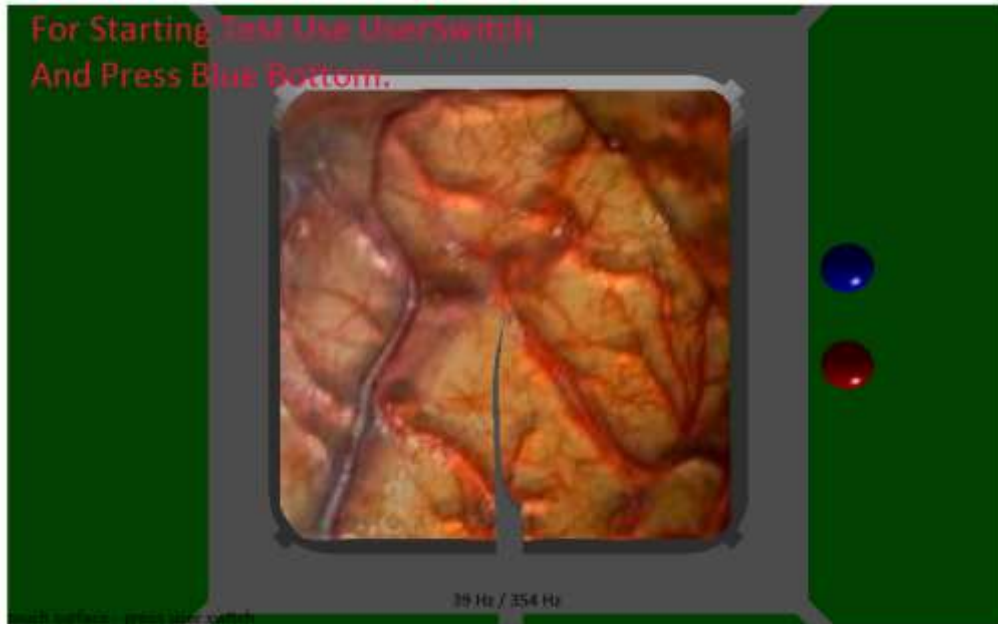
شکل ۴-۱ رگ انتخاب شده برای منعقد سازی

پیشتر اشاره شد که میزان منعقد سازی، به نیرویی که کاربر وارد می‌کند و زمانی که منعقد ساز را در آن نقطه نگه می‌دارد بستگی دارد و در صورتی که کاربر بیش از حد نیاز عمل منعقد سازی را انجام دهد، بافت به رنگ سیاه در می‌آید که نشانگر منعقد سازی بیش از حد و آسیب به بافت مغزی است. در دیگر سمت اگر کاربر کم‌تر از حد لازم منعقد سازی کند، بافت روشن‌تر از شدت اولیه می‌شود. بنابراین کاربر یک بازخورد بصری برای تنظیم میزان منعقد سازی دارد. در دستورالعمل و به صورت شفاهی برای هر داوطلب این مساله توضیح داده شد.

#### ۴-۱-۲- روند آزمایش

همانطور که گفته شد این آزمایش برای هر داوطلب ده مرتبه تکرار می‌شود، اما برای آموزش هرچه بهتر و نبود خطایی در طی آزمایش‌ها یک آزمایش اضافه برای آموزش داوطلب در نظر گرفته شد که با راهنمایی ناظر آزمایش این کار انجام می‌شد و پس از آن، ناظر در طی ده آزمایش دیگر دخالتی نداشت. روند انجام آزمایش به این صورت است که داوطلب طبق شکل زیر برای شروع آزمایش ابتدا گوی آبی رنگ را با عملگر حس می‌کند و با فشردن کلید تیره رنگ روی عملگر، آزمایش شروع می‌شود. لازم به ذکر است راهنمای متنی برای نمایش شماره‌ی آزمایش و فشردن کدام گوی برای شروع یا پایان

آزمایش، در گوشه‌ی سمت چپ تصویر تعبیه شده است. پس از شروع آزمایش، کاربر می‌بایست رگ را منعقد کند و پس از اتمام گوی قرمز رنگ را با همان کلید روی عملگر فعال کند تا آزمایش تمام شود.



شکل ۴-۲ گوی قرمز و آبی و متن راهنما برای شروع و پایان

لازم به ذکر است که بر روی عملگر دستگاه دو کلید برای دستکاری یا اعمال فرمان‌هایی به محیط مجازی وجود دارد که برای این شبیه‌ساز کلیه‌ی کارها اعم از منعقد سازی رگ و فعالسازی و اتمام آزمایش، برای کلید تیره رنگ مطابق شکل زیر تعریف شده است.



شکل ۴-۳ کلید عملگر تعریف شده برای شبیه‌ساز

#### ۴-۱-۳- شرایط و محیط انجام آزمایش

برای اینکه بتوانیم داوطلبان را ارزیابی کنیم باید برای آزمایش شرایط یکسانی را برای تمامی افراد مهیا می‌کردیم. بنابراین محیط آزمایش برای کلیه داوطلبان به صورت شکل زیر در نظر گرفته شد. داوطلب بر روی صندلی می‌نشست و با دست راست و مسلط بر صفحه‌ی رایانه با شبیه‌ساز شروع به کار می‌کرد.



رسم توضیحی ۴-۱-۳ شرایط و محیط آزمایش

#### ۴-۱-۴- ذخیره سازی داده‌ها

در طی آزمایش برای هر داوطلب داده‌هایی از قبیل: زمان، موقعیت عملگر، میزان منعقد سازی برای هر ورتکس و نیرو با سرعت نمونه‌برداری ۵۰ هرتز، از زمانی که داوطلب گوی شروع آزمایش را روشن می‌کند تا زمانی که گوی اتمام را روشن می‌کند، ثبت می‌شد. این داده‌ها برای هر آزمایش و هر فرد به صورت مجزا ذخیره می‌گردید تا در مرحله‌ی بعدی مورد بررسی قرار گیرد.

#### ۴-۱-۵- شرکت کنندگان در آزمایش

برای بررسی روند یادگیری توسط شبیه‌ساز طراحی شده ۲۰ نفر از دانشجویان مورد آزمایش قرار داده شدند. روند پذیرش و انجام آزمایش به این صورت بود که شخص ابتدا فرم رضایت‌نامه، که در پیوست

می‌باشد را مطالعه می‌کرد. در این فرم هدف کلی پروژه، مزایای شرکت در آزمایش برای فرد و محرمانه ماندن اطلاعات شخصی برای داوطلب شرح داده شد و در نهایت رضایتمندی خود را با آگاهی از حقوق خود برای شرکت در آزمایش اعلام می‌کرد.

پس از فرم رضایت‌نامه، داوطلب در یک فرم دیگر، اطلاعات شخصی خود مانند نام، شماره‌ی همراه، ایمیل، حرفه‌ی خود و میزان ساعاتی که به بازی رایانه‌ای یا نواختن موسیقی در یک هفته می‌پردازد را وارد کرد.

در فرم سوم برای آنکه داوطلب آگاهی از پروژه داشته باشد و با دانش قبلی به انجام آزمایش بپردازد، به معرفی پروژه، دستگاه هپتیک و چگونگی روند آزمایش پرداخته شد. پس از مطالعه‌ی این فرم یک بار هم به صورت شفاهی با انجام یک آزمایش، ناظر، توضیحات را مرور می‌کرد. به این صورت داوطلب به طور کامل با محیط آزمایش و قسمت‌های مختلف آن آشنا می‌شد و در نهایت به انجام آن می‌پرداخت. پس از اتمام آزمایش فرم چهارمی با عنوان ارزیابی داوطلب از شبیه‌ساز و نظرخواهی برای بهبود آن در اختیارشان گذاشته شد.

## ۴-۲- تحلیل نتایج بر روی داوطلبان

همان‌طور که گفته شد، داده‌هایی که در طول آزمایش ذخیره می‌شود عبارت است از:

- (۱) زمان
- (۲) موقعیت عملگر
- (۳) میزان منعقدسازی هر ورتکس
- (۴) نیرو

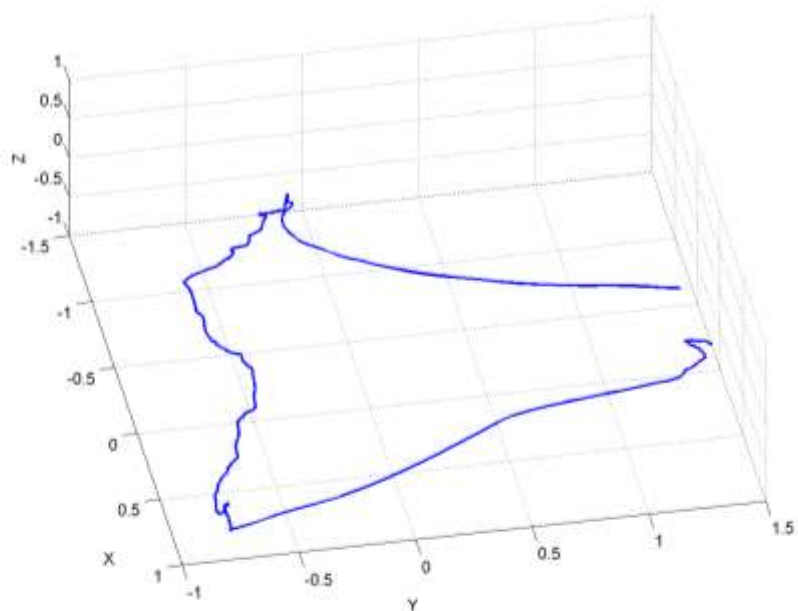
حال با استفاده از این داده‌ها که برای هر آزمایش و هر فرد به ثبت رسید، به تحلیل عملکرد داوطلبان می‌پردازیم.

## ۴-۲-۱- معرفی معیارهای سنجش عملکرد

برای آنکه مقایسه‌ای بین داوطلب‌ها و میزان بهبود عملکرد آن‌ها صورت بگیرد لازم است که معیارهایی معرفی شوند، لذا برای این منظور معیارهای زیر را در نظر می‌گیریم.

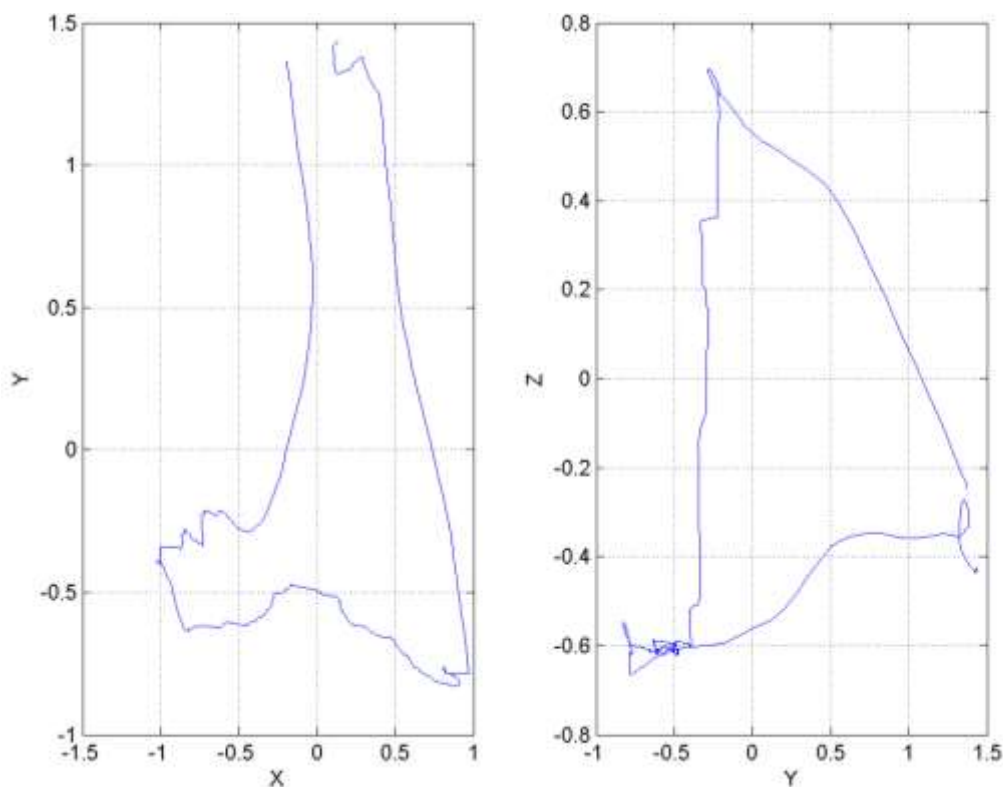
(۱) طول مسیر حرکت: هر داوطلب در ده آزمایشی که انجام می‌دهد، مسیری را از نقطه‌ی آغاز تا نقطه‌ی پایان برای انجام آزمایش می‌پیماید. این مسیر برای هر فرد به صورت سه بعدی و دو بعدی چاپ شد، اما برای این که مقایسه‌ای بین آزمایش‌هایی که هر فرد انجام داده یا مقایسه با افراد دیگر داشته باشیم، طول مسیر طی شده محاسبه شد.

به عنوان مثال مسیر طی شده توسط یک داوطلب به صورت زیر است. که در سه نمودار زیر نشان داده شده است. نمودار اول مسیر حرکت در سه بعد و نمودار دوم و سوم مسیر حرکت در دو بعد را نشان می‌دهد. طول مسیر پیموده شده برای این فرد در این آزمایش ۲۸ سانتی متر است.



شکل ۴-۴ مسیر حرکت در سه بعد





شکل ۴-۵ مسیر حرکت در دو بعد

- ۲) زمان انجام هر آزمایش: برای هر آزمایشی که داوطلب انجام می‌داد، از زمانی که آزمایش آغاز تا زمانی که پایان می‌یافت، زمان محاسبه می‌شد.
- ۳) میانگین نیرویی وارد شده به بافت: در هر آزمایش برای هر فرد، میانگین نیرویی که کاربر به بافت مغزی شبیه‌سازی شده وارد کرده، محاسبه شد.
- ۴) انحراف معیار نیروی وارد شده: پس از محاسبه‌ی میانگین نیروی وارد شده به بافت، انحراف معیار آن نیز بدست آورده شد.
- ۵) درصد خطای نیروی اعمالی: این معیار به این صورت تعریف شد: نیرویی که از سمت داوطلب به بافت غیر رگ مشخص شده وارد می‌شود به کل نیروهایی که به بافت وارد می‌شود.
- ۶) درصد خطای منعقدسازی بافت مغزی: داوطلبان در آزمایش می‌بایست فقط رگ را منعقد می‌کردند و در صورتی که به اشتباه بافت مغزی را منعقد کرده باشند، درصد خطای آن‌ها به این صورت تعریف می‌شود: میزان منعقد سازی بافت مغزی غیر از رگ به بافت منعقد شده‌ی کل.
- ۷) درصد خطای رگ منعقد نشده: داوطلبان می‌بایست کل رگ شبیه‌سازی شده را منعقد می‌کردند که اگر قسمتی از آن منعقد نشده باقی می‌ماند به صورت خطا محاسبه شد. بنابراین درصد

خطای رگ منعقد نشده به این صورت تعریف شد که مساحت قسمت‌هایی از رگ که منعقد نشده به کل رگ.

- (۸) میانگین میزان منعقد سازی: همان‌طور که پیش‌تر گفتیم، میزان منعقد سازی به مدت زمان و نیرویی که داوطلب برای هر نقطه صرف می‌کند وابسته است، لذا میزان منعقد سازی در هر نقطه متفاوت است، در نتیجه برای داشتن یک معیار مقایسه‌ای میانگین آن محاسبه شد.
- (۹) انحراف معیار میزان منعقد سازی: با داشتن میانگین میزان منعقد سازی انحراف معیار نیز به عنوان یک معیار محاسبه شد.

## ۴-۲-۲- تحلیل نتایج کلیه داوطلبان

اکنون با معرفی معیارهای سنجش عملکرد، آمادگی لازم برای تحلیل نتایج بدست آمد. بنابراین در این قسمت با توجه به معیارهای گفته شده روند یادگیری داوطلبان مورد بررسی قرار داده می‌شود.

پیش از آن که به تحلیل نتایج پرداخته شود، لازم است یکی از ابزارهای آماری به نام تست تی<sup>۱</sup> معرفی شود. تست تی نوعی آمار استنباطی است که می‌توان برای تعیین وجود یا عدم وجود تفاوت‌های معنادار بین میانگین دو گروه از آن استفاده کرد؛ تفاوت‌هایی که می‌توانند نشان‌دهنده وجود ویژگی‌های خاص باشند.

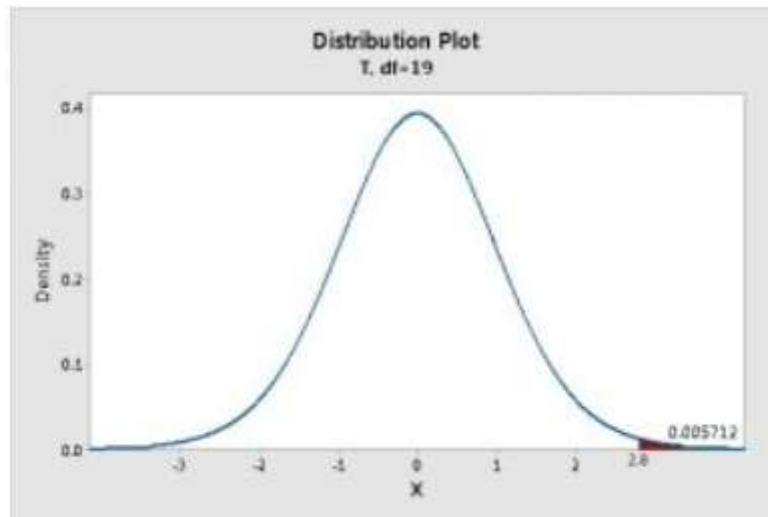
تست تی از نظر ریاضیات از هر دو مجموعه‌ی داده نمونه می‌گیرد و با در نظر گرفتن فرضیه صفری که دو میانگین آن برابر باشد، وضعیت مسئله را مشخص می‌کند. بر اساس این فرمول‌های تعمیم‌پذیر، مقادیر مشخصی محاسبه و با مقادیر استاندارد مقایسه می‌شوند. سپس فرضیه‌ی صفر مفروض بر همین اساس تایید یا رد خواهد شد. اگر قرار باشد فرضیه صفر رد شود، این یعنی مشاهدات داده‌ای درست بوده و بر طبق شانس و احتمال رخ نداده است.

مقدار پی<sup>۲</sup> از تست تی بدست می‌آید. تی و پی به طور جدایی ناپذیری با هم مرتبط است. آن‌ها به صورت خیلی مشابه در کنار هم نتایج تجزیه و تحلیل آماری ظاهر می‌شوند. بیشترین مقدار مورد انتظار برای  $t$  محلی است که قله نمودار شکل زیر قرار دارد (یعنی مقدار صفر). این بدان معنا است که در بیشتر واقع

<sup>۱</sup> T-test

<sup>۲</sup> P-value

انتظار می‌رود که مقدار  $t=0$  شود. علت این امر آن است که وقتی یک نمونه به صورت تصادفی از یک جامعه برداشته می‌شود انتظار می‌رود که اختلافی بین میانگین نمونه با میانگین جامعه وجود نداشته باشد یعنی به احتمال زیاد اختلاف بین میانگین نمونه و میانگین جامعه نزدیک به صفر باشد.



شکل ۴-۶ منحنی نرمال مقادیر T

احتمال اینکه مقدار تی (چه در جهت مثبت و چه در جهت منفی) مقدار بزرگی شود خیلی کم است. یعنی آنکه هر چه از مقدار صفر در هر دو جهت دور می‌شویم احتمال رخداد چنین وضعیتی به صورت طبیعی کاهش می‌یابد. به عنوان مثال ناحیه قرمز مشخص شده در منحنی فوق احتمال اینکه مقدار تی برابر با ۲٫۸ و بیشتر از آن باشد را نشان می‌دهد. احتمال این امر ۰٫۰۰۵۷۱۲ محاسبه شده است که اگر آن را گرد کنیم برابر با ۰٫۰۰۶ می‌شود که به این، مقدار پی گفته می‌شود.

به عبارت دیگر، احتمال به دست آوردن مقدار تی برابر با ۲٫۸ و یا بالاتر، زمانی که نمونه برداری از جمعیت یکسان (در مثال، یک جمعیت با میانگین ۵ در نظر گرفته شده)، حدود ۰٫۰۰۶ است. مقادیر T و P به طور جدایی ناپذیری مرتبط هستند و به سادگی می‌توان از آنها برای تصمیم در خصوص درست یا نادرست بودن یک فرض استفاده کرد. مقدار یکی از آنها بدون تغییر در دیگری، تغییر نخواهد کرد. مقادیر بزرگتر قدرمطلق T منجر به مقادیر کوچکتر P می‌شود که این امر سبب کاهش احتمال پذیرش فرض صفر می‌شود. به طور معمول مطالعات آماری در سطح اطمینان ۹۵٪ (یعنی آلفای برابر با ۰٫۰۵) انجام می‌شود. در جدول زیر میزان سطح اطمینان به صورت کیفی برای مقادیر مختلف P شرح داده شده است [۱۸].

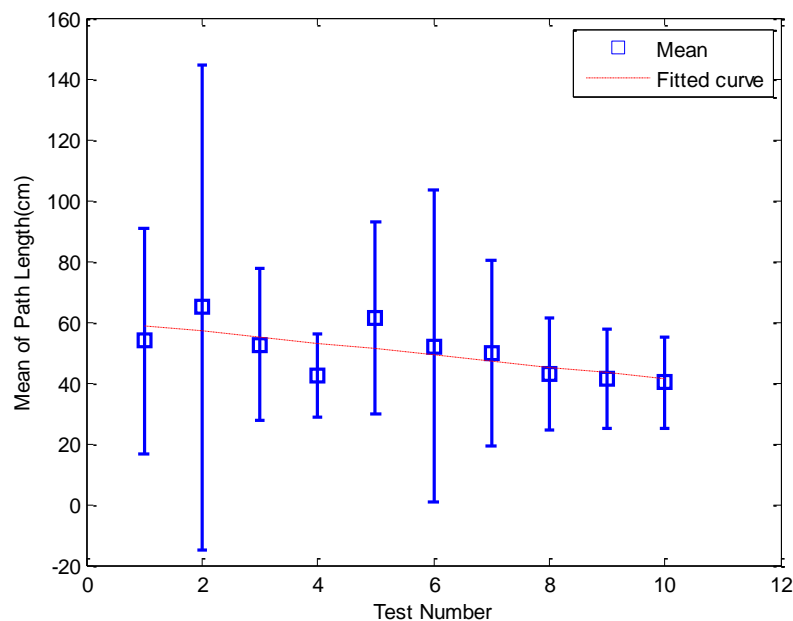
جدول ۱-۴ شرح کیفی برای مقادیر P

شرح کیفی	مقدار P
دلیل خیلی قوی برای رد فرض صفر	$< 0,01$
دلیل قوی برای رد فرض صفر	$0,01 - 0,05$
دلیل ضعیف برای رد صفر	$0,05 - 0,1$
فرض صفر را نمی‌توان رد کرد	$> 0,1$

برای بررسی هر یک از معیارهای گفته شده در بالا، باید روندی در نظر گرفته شود. برای تعریف این روند، همان‌طور که گفته شد، ۲۰ نفر در آزمایش شرکت کردند که هر یک ۱۰ بار عمل منعقد سازی را انجام دادند. حال برای بررسی یادگیری، میانگین و واریانس آزمایش اول تا دهم تمام افراد به صورت کلی محاسبه شد. یعنی به عنوان مثال برای طول مسیر آزمایش اول، از طول مسیری که تمام افراد در آزمایش اول طی کرده بودند میانگین و واریانس گرفته شد و به عنوان نتیجه‌ی آزمایش اول طول مسیر گزارش شد.

#### ۴-۲-۱- طول مسیر حرکت

یکی از معیارهایی که برای بررسی روند یادگیری معرفی شد طول مسیر حرکت عملگر در طی عمل منعقد سازی است. هرچند که این معیار برای بررسی صحت انجام وظیفه‌ای که به داوطلب سپرده شده و منعقد سازی درست رگ مغزی معیار خوبی نیست اما انتظار داریم طول این مسیر در طی تکرار آزمایش کمتر شود. با میانگین و انحراف معیارگیری برای تمام افراد در هر آزمایش نمودار زیر به دست آمد.

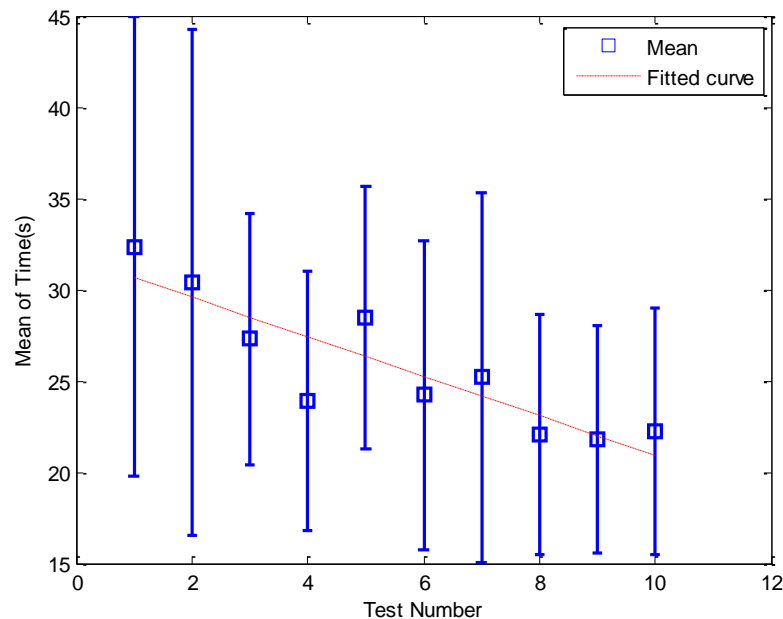


شکل ۴-۷ طول مسیر کل افراد در طی ده آزمایش

با توجه به نمودار بالا انحراف معیار و میانگین در آزمایش‌های اول نسبت به آزمایش‌های آخر بیش‌تر است و با توجه به خط قرمز رنگ برازش شده بین میانگین آزمایش‌ها، افراد روند انتخاب مسیر بهینه را پیش گرفته‌اند. برای اثبات اختلاف معنادار بین آزمایش اول و آخر از تست تی استفاده شد و مقدار  $P$  برای این دو آزمایش  $0.063$  بدست آمد. با توجه به این مقدار می‌توان نتیجه گرفت که بین میانگین و انحراف معیار آزمایش اول و آخر اختلاف معناداری است و افراد در آزمایش آخر بهتر عمل کرده‌اند.

#### ۴-۲-۲-۲- زمان انجام هر آزمایش

این معیار هم مانند معیار طول مسیر برای ارزیابی صحت عملکرد داوطلبان معیار مناسبی نیست اما باز هم انتظار داریم با کسب تجربه در انجام آزمایش، مدت زمان برای آزمایش‌های آخر کاهش یابد. نمودار زیر انحراف معیار و واریانس برای هر آزمایش را نشان می‌دهد.

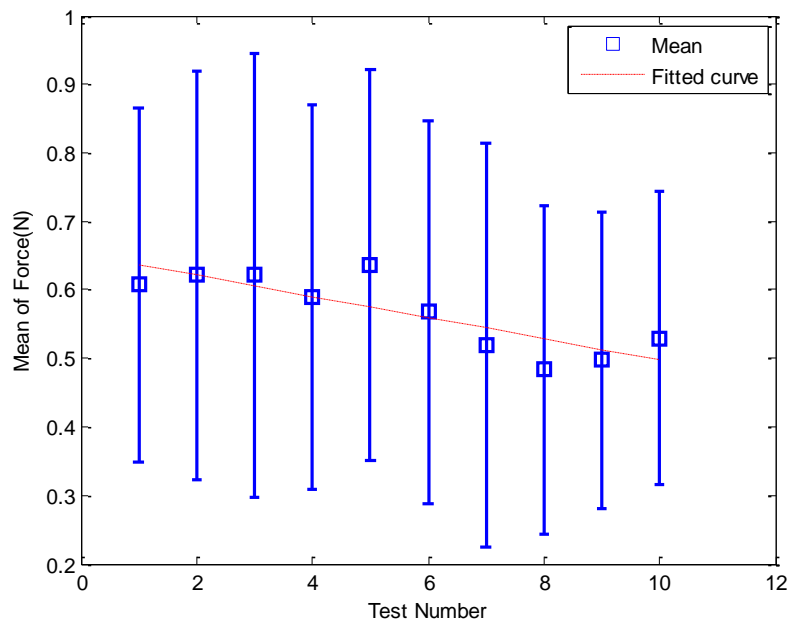


شکل ۴-۸ مدت زمان انجام آزمایش کل افراد در طی ده آزمایش

همانطور که مشاهده می‌شود، میانگین و انحراف معیار به صورت کلی در حال کاهش است. لازم به ذکر است خط قرمز رنگ بر میانگین‌های زمانی برازش شده است. برای مشاهده‌ی اختلاف معنادار بین آزمایش اول و آخر مقدار  $P$  برابر  $0.0022$  به دست آمد که دلیل بسیار قوی برای رد فرض صفر مبنی بر برابری میانگین‌ها است. در نتیجه در مجموع زمان صرف شده برای هر آزمایش رو به کاهش است و با توجه به نمودار در آزمایش‌های آخر تقریباً به عدد ثابتی همگرا شده‌است.

#### ۴-۲-۲-۳- میانگین نیرویی وارد شده به بافت

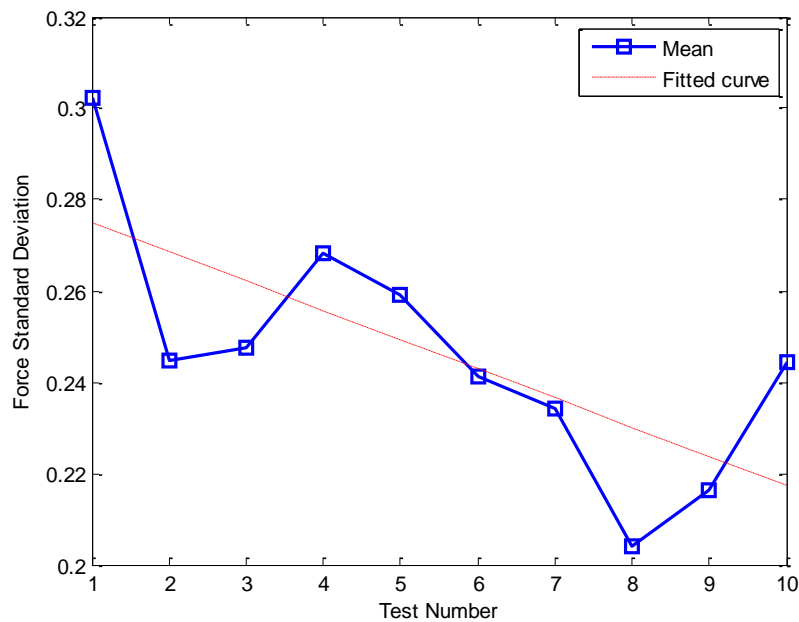
بافت مغزی از حساس‌ترین و مهمترین قسمت‌های بدن است بنابراین در حین عمل جراحی باید با نیروی کم و ملاطفت بالایی با این بافت برخورد کرد. انتظار می‌رود که داوطلبان با نیرو و انحراف معیار کم با بافت برخورد کنند. با توجه به نمودار زیر این مهم، کم و بیش بر آورده شده و اعمال نیروی روند کاهشی داشته است و نیروی اعمالی حدود  $0.6$  نیوتن بوده است. مقدار  $P$  بدست آمده آزمایش اول و آخر این برای معیار برابر  $0.00338$  است که نمایانگر تفاوت چشمگیر در دو آزمایش است و یادگیری افراد در اعمال نیروی کمتر به بافت مغزی روندی صعودی داشته است.



شکل ۴-۹ میانگین نیروی اعمالی افراد برای ده آزمایش

#### ۴-۲-۲-۴-۴ انحراف معیار نیروی وارد شده

همان طور که در بالا گفته شد انتظار داریم میانگین و انحراف معیار نیروی وارده به مغز در طی روند یادگیری کم شود لذا برای این امر انحراف معیار نیروی وارد شده در هر آزمایش توسط داوطلبان را به صورت اختصاصی مورد بررسی قرار دادیم. نمودار زیر میانگین انحراف معیارها برای افراد در طی ۱۰ آزمایش را نمایش می دهد.



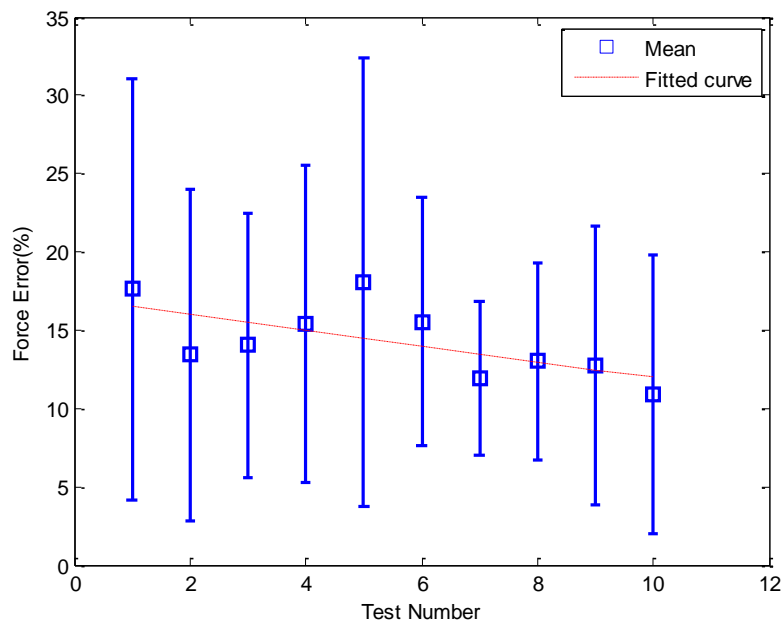
شکل ۴-۱۰ انحراف معیار نیروی وارد شده در ده آزمایش

با توجه به نمودار بالا و خط برازش شده، انحراف معیار به صورت نزولی در حال کاهش است و افراد سعی در وارد کردن نیرویی با انحراف معیار کم و یکنواختی بیشتر داشته‌اند.

#### ۴-۲-۵- درصد خطای نیروی اعمالی

پیش‌تر درصد خطای نیروی اعمالی را اینگونه تعریف کردیم: نیرویی که از سمت داوطلب به بافت غیر رگ مشخص شده وارد می‌شود به کل نیروهایی که به بافت وارد می‌شود. با توجه به این تعریف انتظار داریم داوطلبان در طی آزمایش‌ها این خطا را کم کنند و فقط به رگی که برای آزمایش برایشان تعریف شده نیرو وارد کنند. نمودار زیر انحراف معیار و واریانس درصد خطای نیروی اعمالی را نشان می‌دهد.



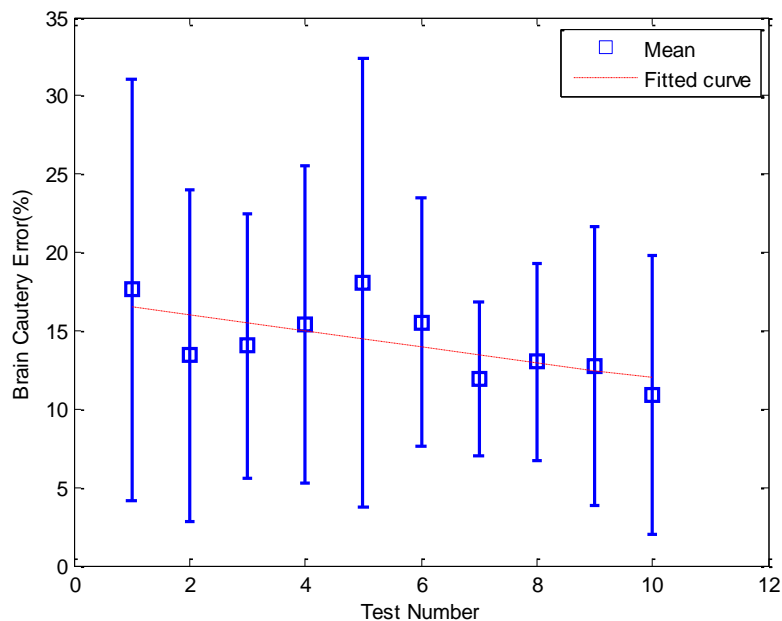


شکل ۴-۱۱ درصد خطای نیروی اعمالی در ده آزمایش

با توجه به نمودار، خط برازش شده دارای یک شیب منفی است که نشانگر کاهش خطاست همچنین انحراف معیار این خطا نیز در حال کاهش است. مقدار  $P$  برای آزمایش اول و آخر در این معیار برابر  $0.0350$  بدست آمد که گویای تفاوت قابل ملاحظه بین این دو آزمایش و در نتیجه روند کاهش خطا است.

#### ۴-۲-۲-۶ درصد خطای منعقدسازی بافت مغزی

در این مورد نیز انتظار داریم داوطلب بافت مغزی را منعقد نکند و خطای این عمل را کاهش دهد. نمودار زیر میانگین و واریانس درصد خطای منعقدسازی بافت مغزی را نشان می‌دهد.

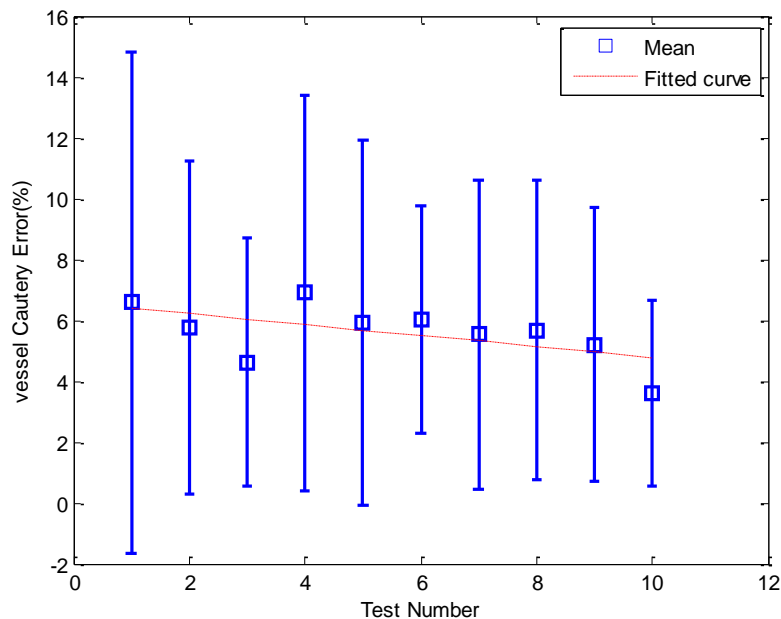


شکل ۴-۱۲ میانگین و واریانس درصد خطای سوزاندن بافت مغزی در ده آزمایش

همانطور که ملاحظه می‌شود میانگین و واریانس این خطا روبه کاهش است و مقدار  $P$  بدست آمده برای آزمایش‌های اول و دهم برابر  $0.0274$  است که می‌توان گفت اختلاف معناداری بین آزمایش اول و دهم وجود دارد و افراد در طی روند یادگیری این خطا را کاهش داده‌اند.

#### ۴-۲-۷- درصد خطای رگ منعقد نشده

انتظار داریم داوطلب کل رگ را منعقد کند و در صورتی که قسمتی از آن را منعقد نکند در تحلیل‌ها به صورت خطا ظاهر شود. این خطا باید با توجه به روند یادگیری داوطلب در نهایت کم شود. نمودار زیر میانگین و انحراف معیار درصد خطای رگ منعقد نشده را نشان می‌دهد.

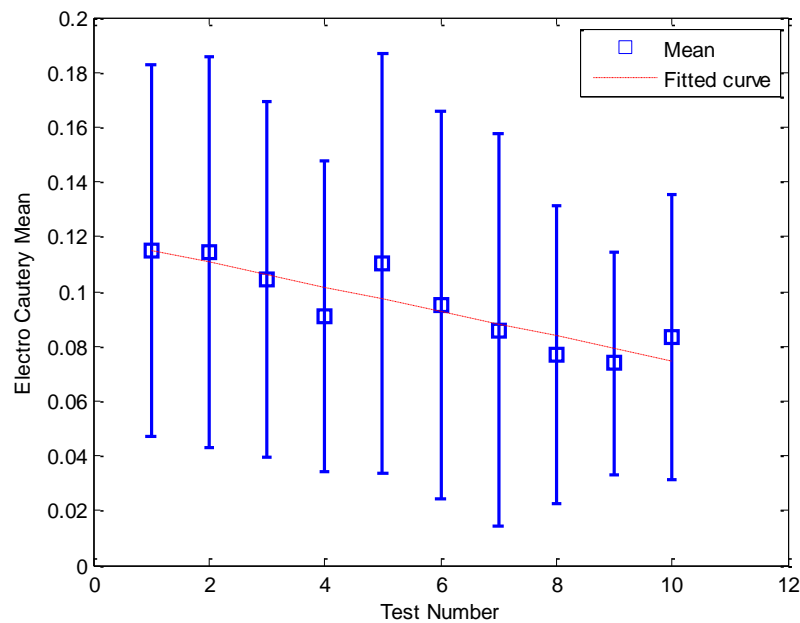


شکل ۴-۱۳ میانگین و انحراف معیار درصد خطای رگ منعقد نشده

خط برازش شده برای میانگین‌ها شیب منفی دارد و در کل انحراف معیار این خطا در حال کاهش است بنابراین می‌توان گفت داوطلب سعی در کم کردن این خطا داشته است. همچنین مقدار  $P$  برای آزمایش اول و آخر برابر  $0,0081$  است که بر این نتیجه صحه می‌گذارد.

#### ۸-۲-۲-۴- میانگین میزان منعقدسازی

به علت این که میزان منعقد سازی به صورت بصری با تغییر رنگ بافت به افراد بازخورد می‌شد. بنابراین هر فرد در این مورد به صورت سلیقه‌ای برخورد کرده است و نمی‌توان به عنوان معیاری که روند یادگیری را نشان می‌دهد با آن برخورد کرد اما انحراف معیار و میانگین آن برای تمام افراد در ده آزمایش در زیر نشان داده شده است.

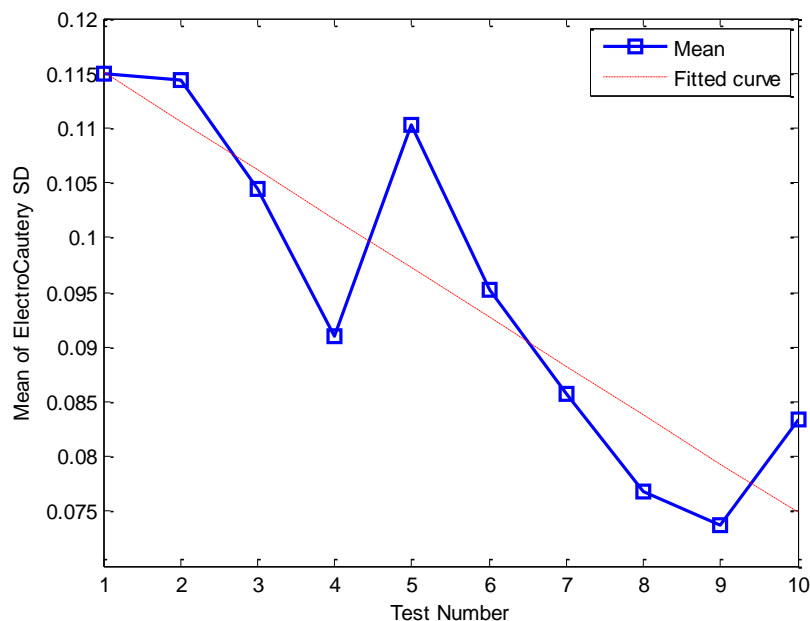


شکل ۴-۱۴ انحراف معیار و میانگین میزان منعقد سازی

در نمودار دیده می‌شود که میانگین‌ها و انحراف معیار میانگین افراد در کل، در حال کاهش است و می‌توان گفت کم شدن انحراف معیار میانگین هر فرد نشان دهنده‌ی یکنواخت‌تر منعقد کردن بافت است. مقدار  $P$  بدست آمده برای آزمایش اول و آخر این معیار برابر  $0.0054$  است که گویای تفاوت چشمگیری در میانگین و انحراف معیار آزمایش اول و آخر است.

#### ۹-۲-۲-۴- انحراف معیار میزان منعقد سازی

انحراف معیار میزان منعقد سازی برای هر فرد در هر تست محاسبه شد و سپس برای هر تست بین ۲۰ نفر داوطلب میانگین گرفته شد، این نتیجه به صورت نمودار زیر بدست آمد.

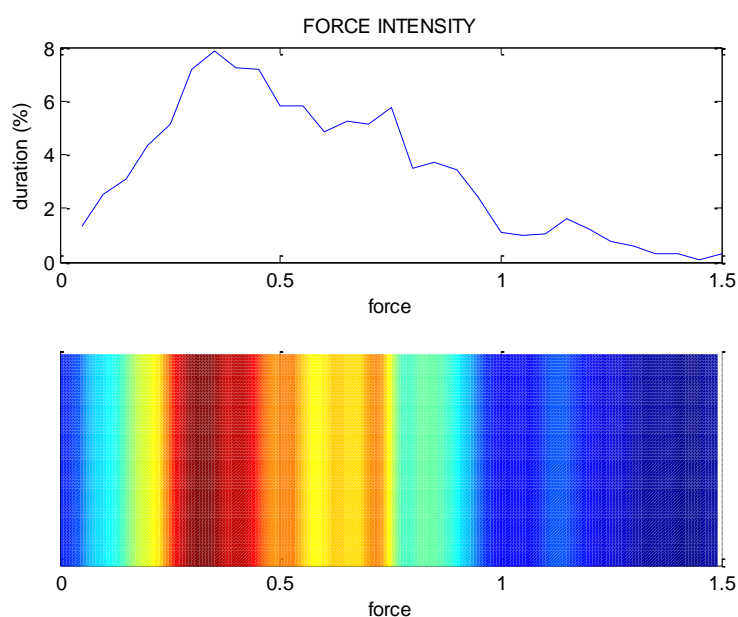


شکل ۴-۱۵ انحراف معیار میزان منعقد سازی

همانطور که در نمودار پیداست، با توجه به خط برازش شده این انحراف معیار با شیب خوبی در حال کاهش است که یعنی داوطلبان در آزمایش‌های آخر رگ را یکنواخت‌تر منعقد کرده‌اند. بنابراین داوطلبان روند یادگیری خوبی در یکنواخت منعقد کردن رگ، پیش گرفته‌اند.

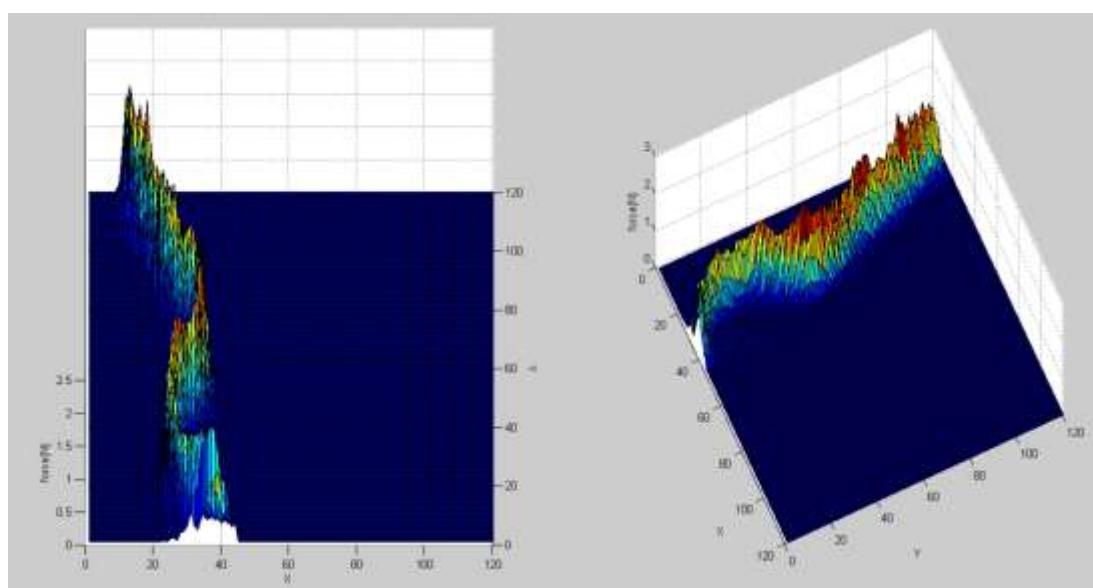
#### ۴-۲-۲-۱۰ سایر تحلیل‌ها

تحلیل دیگری که می‌توان روی داده‌ها داشت، نمودار نیرویی است که نشان می‌دهد به طور میانگین هر نیرو چه میزان استفاده شده است. در شکل زیر، نمودار بالا میزان استفاده از هر نیرو را به این صورت که محور افقی مقدار نیرو و محور عمودی درصد استفاده از آن نیرو است را نشان می‌دهد. نمودار پایین نمودار حرارتی استفاده از هر نیرو است به این ترتیب که محور افقی مقدار نیرو و رنگی که در نمودار استفاده شده میزان استفاده از آن نیرو را نشان می‌دهد، هرچه قرمزتر باشد میزان استفاده از آن نیرو بیشتر است. با تحلیل این نمودار ملاحظه می‌شود که داوطلبان اکثر نیروهای خود را در کمتر از یک نیوتن به بافت وارد کرده‌اند و می‌توان گفت عملکردشان در اعمال نیرو به بافت مغزی با دقت و ملاحظت بوده است.



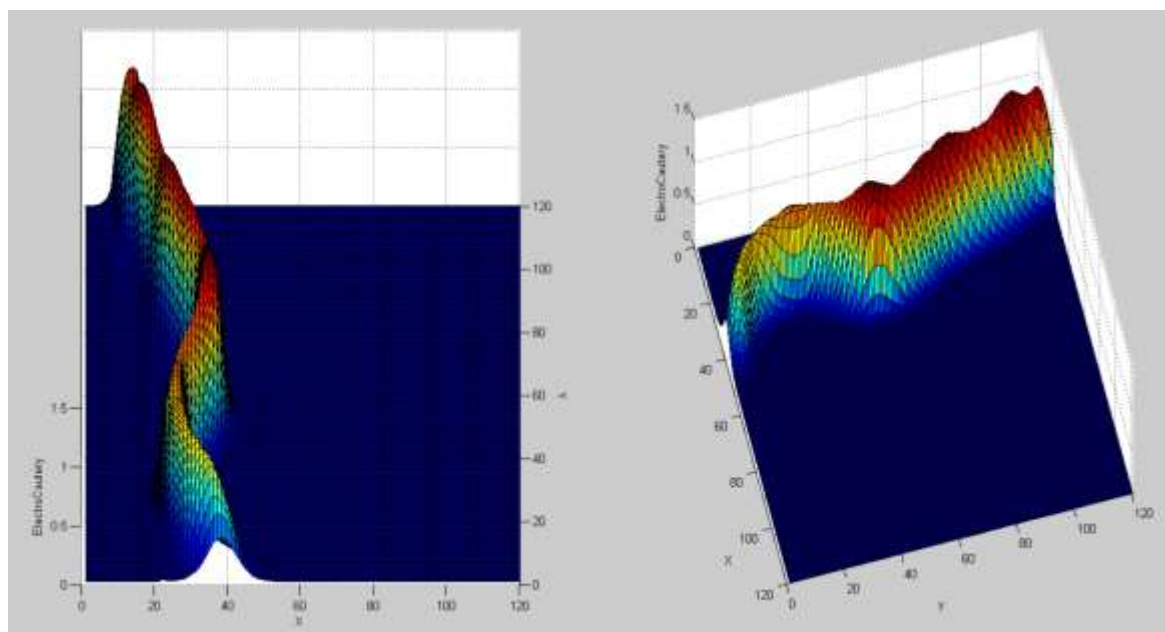
شکل ۴-۱۶ نمودار شدت نیرویی

نمودار دیگری که می‌توان استخراج کرد، نمودار مجموع نیرویی است که از میانگین گیری کل نیروهایی که داوطلبان به بافت وارد کرده اند به دست آمد. این نمودار به صورت نمودار حرارتی در دو زاویه دید مختلف در شکل زیر رسم شده است. در شکل ۴-۱ رگ انتخاب شده برای منعقد سازیمشاهده می‌شود که تقریباً تمام نیروها به رگ اعمال شده و تقریباً شدت نیرویی در طول رگ یکسان است.



شکل ۴-۱۷ نمودار مجموع نیرویی

تحلیل دیگری که می‌توان استخراج کرد نمودار حرارتی میانگین منعقدسازی توسط کلیه داوطلبان است که در شکل زیر نشان داده شده‌است. مشاهده می‌شود که داوطلبان رگ را به عنوان هدف منعقدسازی قرار داده‌اند اما یکنواختی انجام این عمل در مجموع درست اعمال نشده‌است.



شکل ۴-۱۸ نمودار حرارتی میانگین منعقدسازی

#### ۴-۲-۱۱- نتیجه گیری

با تحلیل‌هایی که در بالا بر روی معیارهای گفته شده داشتیم، مشاهده شد که همه‌ی خطاها رو به کاهش و عملکرد در حال میل به سمت بهترین حالت انجام آزمایش بود. بنابراین همانطور که انتظار می‌رفت، شبیه‌ساز جراحی در نقش یک ابزار آموزشی و ارزیابی عملکرد، به خوبی ظاهر شد و روند یادگیری افراد با شرکت در این آزمایش برای منعقدسازی رگ مغزی به خوبی دیده شد.

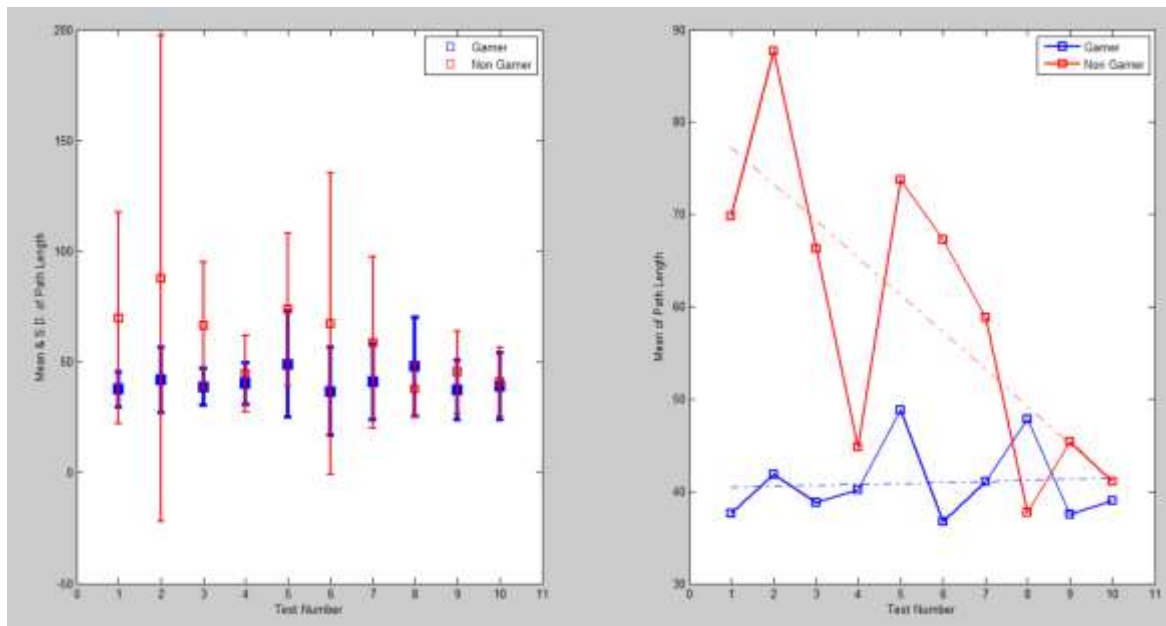
#### ۴-۲-۳- تحلیل نتایج زیر گروه‌ها

در پرسش‌نامه‌ای که به داوطلبان شرکت در آزمایش داده شد دو سوال در مورد این که بازی‌های رایانه‌ای انجام می‌دهند یا ساز موسیقی می‌نوازند پرسیده شد به این ترتیب ۲۰ نفر داوطلب در آزمایش به دو گروه تقسیم شد: گروه اول، افرادی که ساز موسیقی می‌نوازند یا بازی رایانه‌ای انجام می‌دهند و گروه دوم، افرادی که این دو را انجام نمی‌دهند. تعداد افراد گروه اول و دوم، هر یک ۱۰ نفر است. بنابراین برآمدیم تا کلیه‌ی تحلیل‌هایی که برای همه‌ی داوطلبان انجام دادیم را برای این دو گروه بررسی کنیم.

#### ۴-۲-۳-۱- طول مسیر حرکت

نمودار زیر میانگین و انحراف معیار طول مسیر برای دو گروه را در کنار هم نشان می‌دهد. گروه اول افرادی که بازی رایانه‌ای انجام می‌دهند یا ساز می‌نوازند با رنگ آبی و گروه دوم افرادی که بازی نمی‌کنند و ساز نمی‌نوازند با رنگ قرمز نشان داده شده‌است.



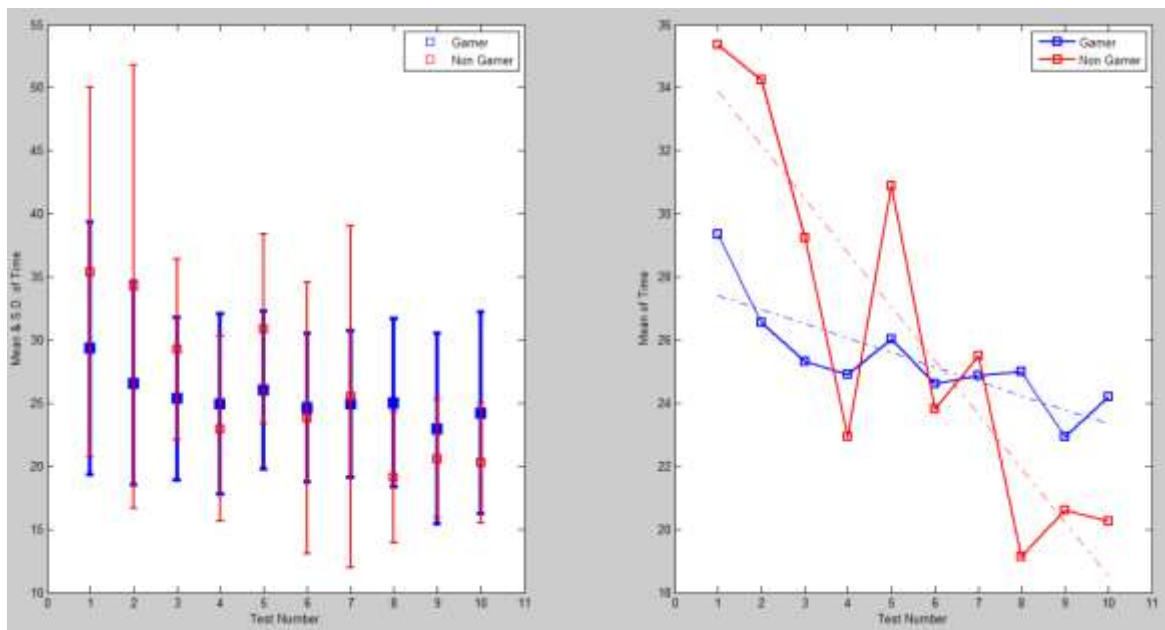


شکل ۴-۱۹ میانگین و انحراف معیار طول مسیر برای دو گروه

ملاحظه می‌شود که گروه اول با میانگین و انحراف معیار کم‌تری نسبت به گروه دوم شروع کرده‌اند و تقریباً با همین رود ادامه داده‌اند اما گروه دوم روند کاهشی برای میانگین و انحراف معیار داشته است و در نهایت با گروه اول هم‌گام شده است. مقدار  $P$  برای این دو گروه برابر  $0.00069$  است بنابراین با اطمینان می‌توان گفت که دو گروه تفاوت چشمگیری در میانگین و انحراف معیار طول مسیر داشته‌اند. همانطور که در قسمت اول تحلیل‌ها گفته‌شد، معیار طول مسیر برای ارزیابی صحت و دقت عملکرد داوطلبان نمی‌تواند معیار خوبی باشد ولی افراد گروه اول از همان ابتدا طول مسیر کمتری را پیموده‌اند.

#### ۴-۲-۳-۲- زمان انجام هر آزمایش

میانگین و انحراف معیار زمان انجام هر آزمایش برای دو گروه در قالب نمودار زیر نمایش داده شده‌است.

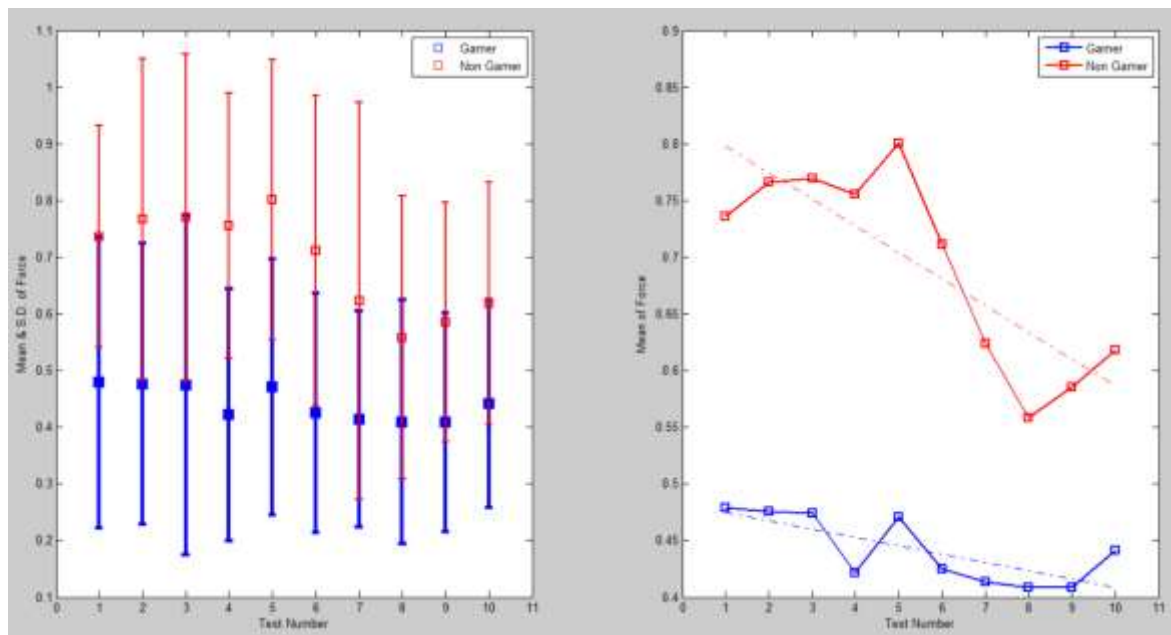


شکل ۴-۲۰ میانگین و انحراف معیار زمان انجام هر آزمایش برای دو گروه

مشاهده می‌شود که گروه اول با انحراف معیار تقریباً ثابت و میانگین کمتر از گروه دوم پیش رفته است اما گروه دوم با میانگین و انحراف معیار بیشتری نسبت به گروه اول شروع کرده است ولی با میانگین و انحراف معیار کمتری آزمایش را به اتمام رسانده است که نشانگر روند کاهش زمان بهتر نسبت به گروه اول است. مقدار  $P$  بدست‌آمده برای این دو گروه در این معیار برابر ۰,۵۲۸۰ است که با توجه به این عدد نمی‌توان تفاوت معناداری برای این دو گروه در این معیار قایل شد.

#### ۴-۳-۳-۲- میانگین نیروی وارد شده به بافت

میانگین و انحراف معیار میانگین نیروی وارد شده به بافت در هر آزمایش برای دو گروه در نمودار زیر نشان داده شده است.

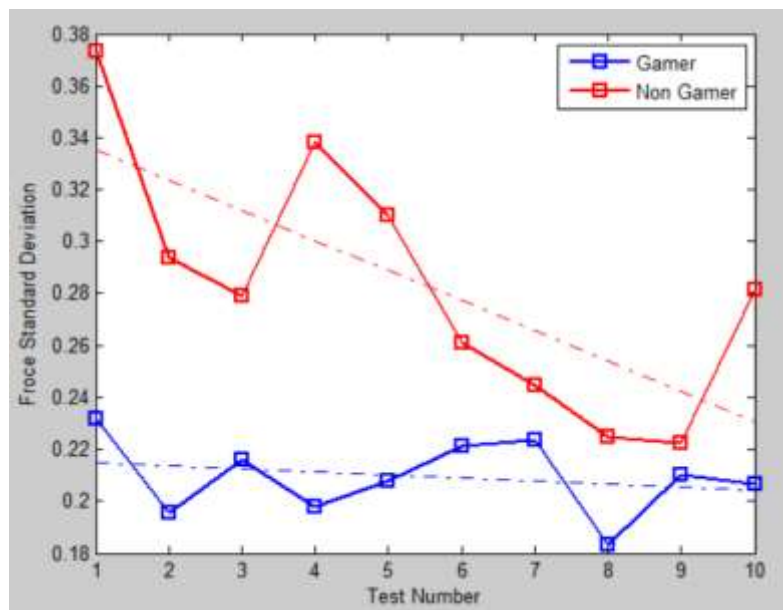


شکل ۴-۲۱ میانگین و انحراف معیار میانگین نیروی وارد شده

با توجه به نمودار، گروه اول با میانگین و انحراف معیار کمتر و تقریباً ثابتی نسبت به گروه دوم آزمایش را انجام داده‌اند. مقدار  $P$  برای این دو گروه در حدود صفر است که نشانگر تفاوت قابل ملاحظه در میانگین دو گروه است بنابراین می‌توان این نتیجه را گرفت که گروه اول با نیروی کمتر و ملاطفت بیشتر نیرو به بافت مغزی وارد کرده‌اند همانطور که در بازی رایانه‌ای یا نواختن یک ساز موسیقی نیاز به این ملاطفت است اینجا هم این روند را انجام داده‌اند. روند کاهش نیرو نیز در هر دو گروه با توجه به خطوط برازش شده دیده می‌شود، این روند برای گروه دوم بسیار نزولی‌تر از گروه اول است که به این معناست که گروه اول سعی در کاهش نیرو و رساندن نیروی خود به سطح پایین‌تر و متناسب با بافت مغزی است.

#### ۴-۳-۲-۴-۴ انحراف معیار نیروی وارد شده

برای این معیار، انحراف معیار نیرویی هر شخص به صورت جداگانه محاسبه شد و سپس این انحراف معیار برای هر آزمایش میانگین گیری شد. نمودار زیر میانگین انحراف معیار نیرو برای دو گروه را نمایش می‌دهد.

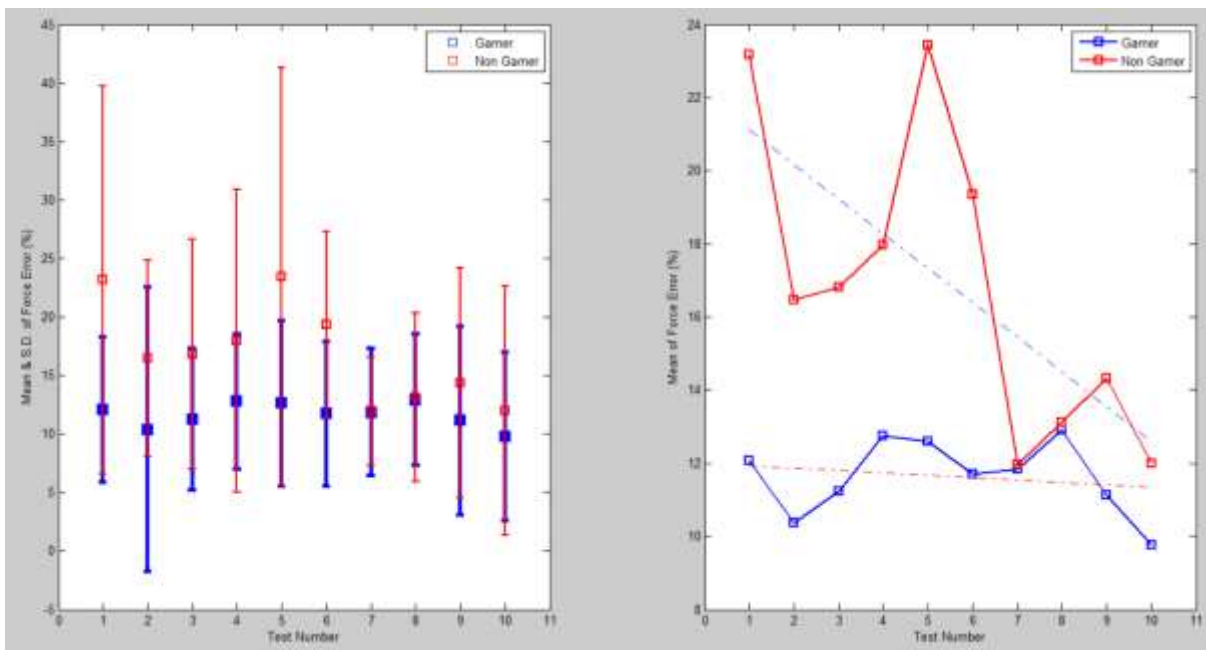


شکل ۴-۲۲ میانگین انحراف معیار نیرو برای دو گروه

ملاحظه می‌شود که گروه اول با انحراف معیار تقریباً یکنواخت و کم‌تر از گروه دوم پیش رفته است ولی گروه دوم با انحراف معیار بالاتری شروع کرده است و در آزمایش‌های آخر با پیشرفتی که داشته‌است سعی در رساندن خود به گروه اول دارد. مقدار  $P$  بدست آمده برای دو گروه در این معیار برابر  $0.0007$  است که به این ترتیب با اطمینان می‌توان تفاوت معناداری بین دو گروه قایل شد و نتیجه گرفت که گروه اول با یکنواختی بیشتری نسبت به گروه دوم نیرو به رگ وارد کرده‌اند.

#### ۴-۲-۳-۵ درصد خطای نیروی اعمالی

میانگین و انحراف معیار درصد خطای نیروی اعمالی که به جای رگ به بافت مغزی غیر رگ اعمال شده است برای دو گروه در نمودار زیر به تصویر کشیده شده‌است.



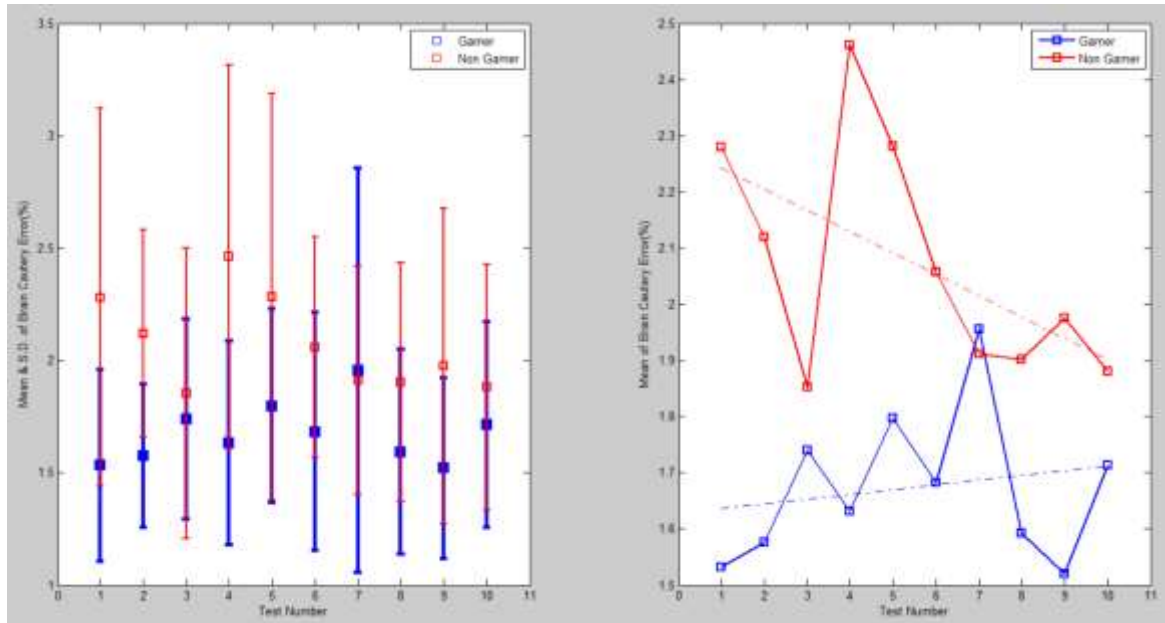
شکل ۴-۲۳ میانگین و انحراف معیار درصد خطای نیروی اعمالی

مشاهده می‌شود که گروه اول نسبت به گروه دوم دارای میانگین و انحراف معیار به نسبت کمتری است. گروه اول با میانگین و انحراف معیار کمتر نسبت به گروه دوم آزمایش را شروع کرده و تقریباً با همان مقادیر اولیه، آزمایش را به اتمام رسانیده است اما گروه دوم با مقدار خطا و انحراف معیار بیشتر شروع کرده و با پیشرفتی که در یادگیری داشته است در آزمایش‌های آخر این مقدار را به گروه اول نزدیک کرده است. بنابراین با مشاهده‌ی خطوط برازش شده می‌توان دریافت روند کاهش این خطا در گروه دوم بسیار بیشتر از گروه اول است که در نهایت به یک مقدار همگرا می‌شوند. مقدار  $P$  برای این دو گروه در این معیار برابر  $0.0001$  است که می‌توان گفت این دو گروه در این معیار اختلاف معنا داری دارند و بر این اساس قابل تفکیک‌اند.

#### ۴-۲-۳-۶ درصد خطای منعقدسازی بافت مغزی

نمودار زیر درصد خطای منعقدسازی بافت مغزی توسط دو گروه را نشان می‌دهد. مشاهده می‌شود که میانگین و انحراف معیار گروه اول کمتر از گروه دوم است اما در طول انجام آزمایشات گروه اول خطایشان بیشتر و گروه دوم خطایشان کمتر شده است و شاید بتوان گفت به سمت خطای ثابتی در

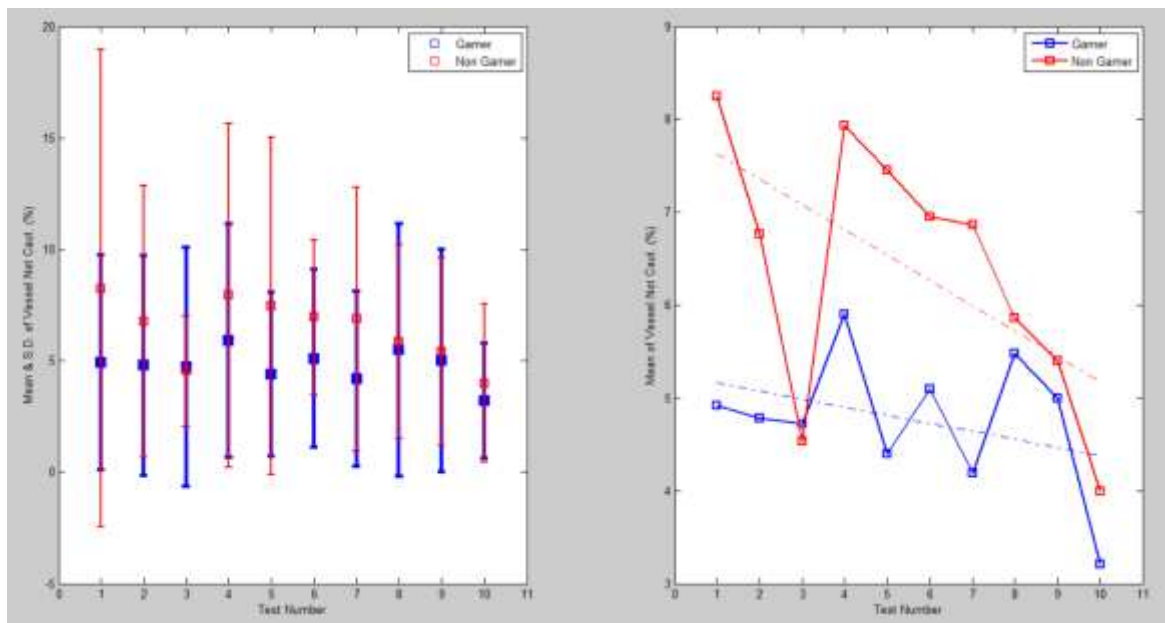
حال همگرا شدن هستند. مقدار  $P$  بدست آمده تقریباً برابر صفر است که نشان دهنده‌ی تفاوت قابل ملاحظه برای دو گروه در این معیار است.



شکل ۴-۲۴ درصد خطای منعقد سازی بافت مغزی توسط دو گروه

#### ۴-۲-۳-۷- درصد خطای رگ منعقد نشده

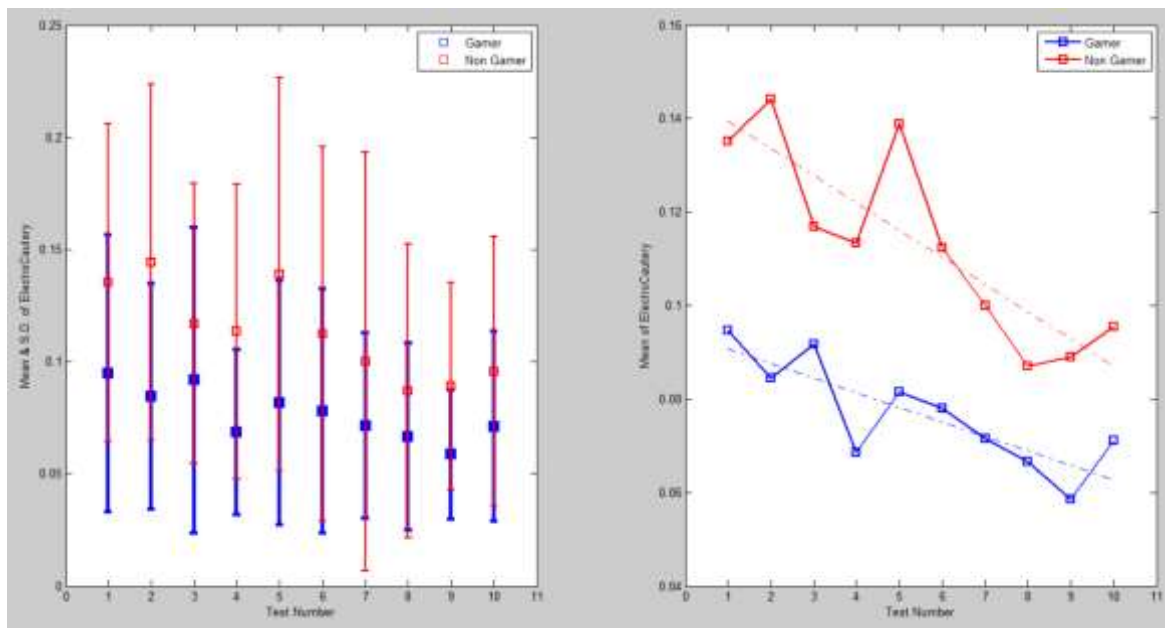
در این معیار نیز گروه اول میانگین و انحراف معیار کمتری نسبت به گروه دوم دارد و نیز هر دو گروه سعی در کم کردن این خطا دارند. مقدار  $P$  محاسبه شده برای این معیار برابر ۰,۰۳۶۷ است. همانطور که انتظار می‌رفت با توجه به مقدار  $P$ ، تفاوت چشمگیری بین دو گروه در این معیار وجود دارد و گروه اول خطای کمتری در منعقد کردن رگ داشته است.



شکل ۴-۲۵ انحراف معیار و میانگین درصد خطای رگ منعقد نشده برای دو گروه

#### ۴-۳-۸- میانگین میزان منعقد سازی

میانگین و انحراف معیار میانگین میزان منعقدسازی در نمودار زیر برای دو گروه مقایسه شده است. پیداست که میانگین و انحراف معیار میانگین برای گروه اول کمتر از گروه دوم است و هر دو گروه با توجه به خطوط برازش شده سعی در کم کردن این میزان دارند. مقدار  $P$  برای این معیار در حدود صفر است که نشانگر اختلاف معناداری برای دو گروه است. همانطور که گفته شد بازخورد میزان منعقدسازی به صورت بصری به داوطلب منتقل می‌شود که با توجه به این گروه اول سعی در منعقد سازی کمتر رگ را داشته‌اند.

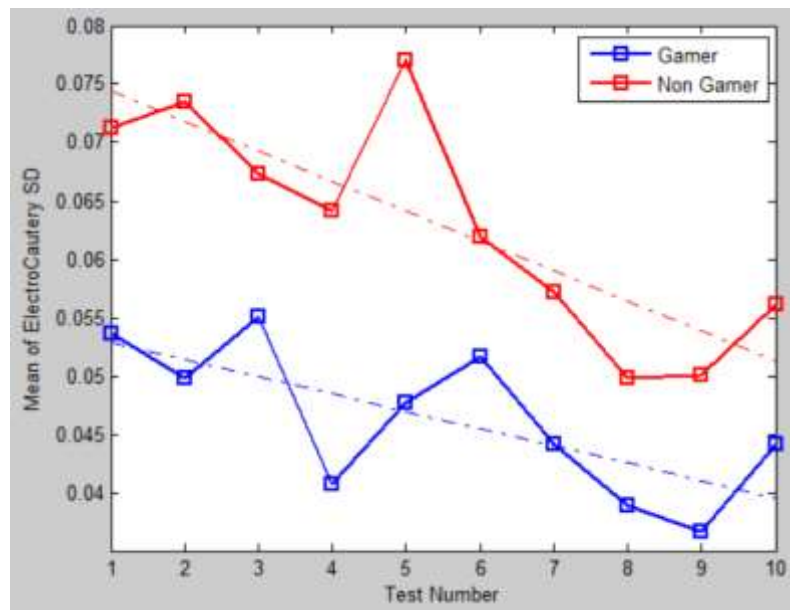


شکل ۴-۲۶ میانگین و انحراف معیار میانگین میزان منعقد سازی

#### ۴-۲-۳-۹- انحراف معیار میزان منعقد سازی

انحراف معیار میزان منعقد سازی برای هر آزمایش و برای هر داوطلب محاسبه شد و سپس در طول ۱۰ آزمایش برای دو گروه میانگین‌گیری شد. نتیجه‌ی بدست آمده در نمودار زیر گزارش شده‌است. مقدار P بدست آمده برای این معیار در حدود صفر است که نشانگر تفاوت چشمگیر بین انحراف معیارهای دو گروه است. با توجه به نمودار، میانگین این معیار برای گروه اول کمتر از گروه دوم است و هر دو گروه با توجه به خطوط برازش شده در حال کم کردن این معیار هستند. بنابراین در کل می‌توان این نتیجه را گرفت که داوطلبان این هدف را دنبال می‌کنند که منعقدسازی یکنواختی داشته باشند اما گروه اول یکنواخت‌تر منعقد کرده‌است.

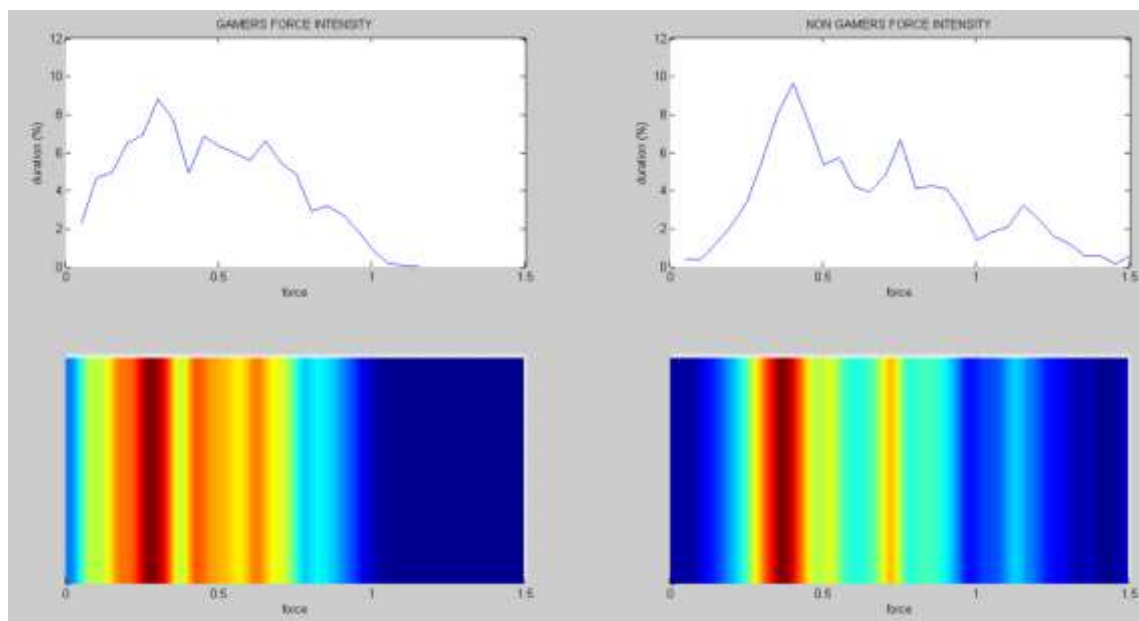




شکل ۴-۲۷ انحراف معیار میزان منعقد سازی

#### ۴-۲-۳-۱۰- سایر تحلیل‌ها

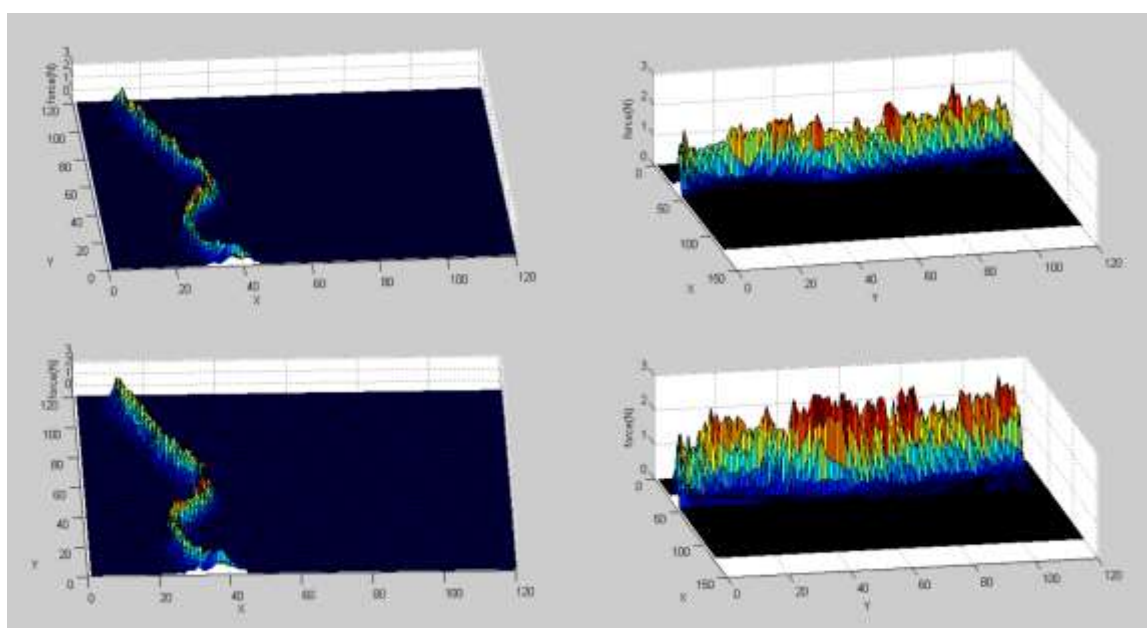
نمودار زیر میزان استفاده از هر نیرو را توسط دو گروه نشان می‌دهد. دو نمودار سمت چپ، گروه اول و دو نمودار سمت راست گروه دوم را مورد بررسی قرار داده‌است. مشاهده می‌شود که افراد گروه اول از نیروی کمتر در مدت زمان بیشتر استفاده کرده‌اند که نشانگر برخورد با دقت و ملاحظت بیشتر با بافت نسبت به گروه دوم است.



شکل ۴-۲۸ نمودار شدت نیرویی دو گروه

دو نمودار سمت چپ گروه اول - دو نمودار سمت راست گروه دوم

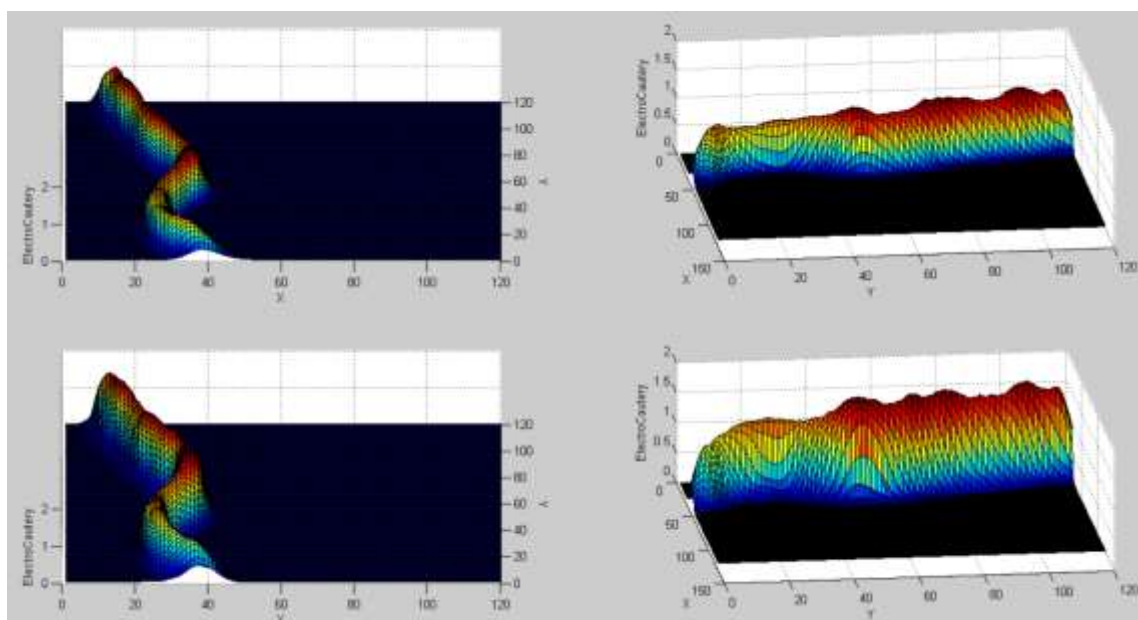
شکل زیر میانگین نیروهای وارد شده توسط داوطلبان به بافت مغزی را نشان می‌دهد. دو نمودار بالا متعلق به گروه اول و دو نمودار پایین متعلق به گروه دوم در دو زاویه دید مختلف است. همانطور که مشاهده می‌شود، هر دو گروه نیروی خود را به رگ وارد کرده‌اند ولی افراد گروه اول به صورت یکنواخت-تر و با میانگین کمتر این نیرو را اعمال کرده‌اند.



شکل ۲۹-۴ میانگین نیروی اعمالی برای دو گروه

دو نمودار بالا گروه اول-دو نمودار پایین گروه دوم

نمودار زیر میزان منعقدسازی میانگین برای داوطلبان را نشان می‌دهد. دو نمودار بالا برای گروه اول و دو نمودار پایین برای گروه دوم رسم شده است. مشاهده می‌شود که میانگین منعقدسازی برای گروه اول کمتر از گروه دوم است و همچنین گروه اول نسبت به گروه دوم یکنواخت‌تر منعقد کرده‌اند.



شکل ۳۰-۴ میانگین منعقد سازی برای دو گروه

دو تصویر بالا گروه اول-دو تصویر پایین گروه دوم

#### ۴-۲-۳-۱۱- نتیجه گیری

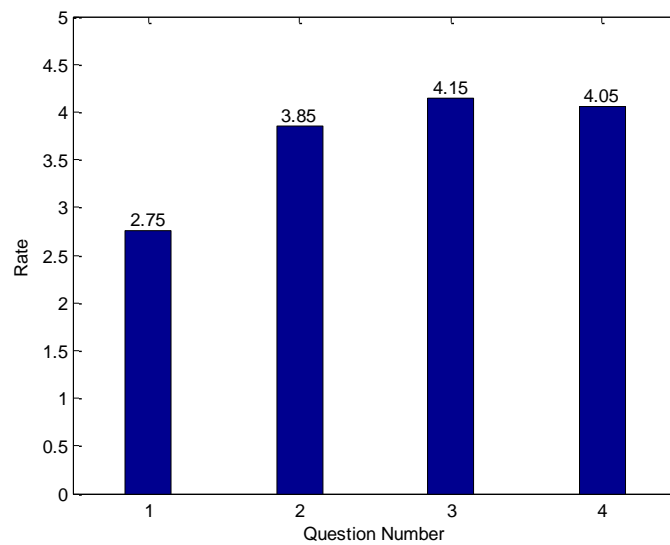
با توجه به تحلیل‌هایی که برای این دو گروه صورت گرفت، گروه اول در تمامی معیارها به جز زمان که نمی‌توان معیار خوبی برای ارزیابی عمل کرد دانست، بسیار بهتر از گروه دوم عمل کردند. با این اوصاف می‌توان گفت، گروه اول چون هر روز با کاری سرگرم می‌شوند که نیاز به تمرکز و دقت عمل دارد، در نتیجه در رفتار با بافت دقیق‌تر و با ملاحظت بیشتری برخورد می‌کنند و در مجموع کارهایی که نیاز به ظرافت دارند مانند جراحی را با روند بهتری یاد می‌گیرند و انجام می‌دهند. بنابراین می‌توان نتیجه گرفت که این افراد جراحان موفق‌تری نسبت به افرادی که بازی کامپیوتری انجام نمی‌دهند و یا ساز موسیقی نمی‌نوازند خواهند بود.

#### ۴-۳- ارزیابی شبیه‌ساز توسط داوطلبان

پس از اتمام کار به هر داوطلب فرمی مبنی بر ارزیابی پروژه داده شد. این فرم همراه با دیگر فرم‌ها در پیوست قرار دارد. ارزیابی داوطلب بر اساس پاسخ به موارد زیر بود.

- (۱) میزان دشواری این آزمایش را از ۱ تا ۵ نمره دهی کنید. (۱-بسیار آسان ۵-بسیار دشوار)
- (۲) میزان واقعی بودن محیط دیداری شبیه سازی شده (۱-کاملاً غیر واقعی ۵-کاملاً واقعی)
- (۳) میزان واقعی بودن نیروهای وارد شده به دست (۱-کاملاً غیر واقعی ۵-کاملاً واقعی)
- (۴) رضایت کلی شما از این شبیه ساز جراحی چیست؟ (۱-کاملاً ناراضی ۵- کاملاً راضی)

با توجه به امتیازی که هر یک از افراد به هر سوال دادند، میانگین کلی برای ۲۰ نفر گرفته شد و نمودار میله‌ای زیر رسم شد.



شکل ۴-۳ امتیاز کسب شده در هر مورد

با توجه به نمودار، داوطلبان میزان دشواری آزمایش را در حد متوسط، میزان واقعی بودن محیط دیداری را در حد خوب، میزان واقعی بودن نیروها و رضایت کلی از شبیه‌ساز را در حد خیلی خوب ارزیابی کردند.

#### ۴-۴- جمع بندی

در این فصل آزمایشی بر اساس استفاده از شبیه‌ساز جراحی، طراحی شد و با استفاده از آن، روند یادگیری داوطلبان مورد مطالعه قرار گرفت. همانطور که مشاهده شد تمامی داوطلبان سعی در کم کردن خطا و افزایش مهارت‌های خود داشتند.

## فصل پنجم

### جمع‌بندی و نتیجه‌گیری و پیشنهادها

## جمع‌بندی و نتیجه‌گیری

در این پروژه ابتدا با یک تکنولوژی جدید با نام هپتیک که رسالتش اضافه کردن حس لامسه به زمینه‌ی واقعیت مجازی است، آشنا شدیم. با استفاده از این دستگاه و توسعه‌ی نرم‌افزاری که صورت گرفت یک شبیه‌ساز عمل جراحی مغز و اعصاب طراحی شد. ویژگی این شبیه‌ساز تغییر شکل بافت در صورت وارد کردن نیرو و امکان منعقدسازی رگ خونی در بافت مغزی است.

در نهایت برای ارزیابی روند یادگیری افراد در عمل جراحی مغز از این شبیه‌ساز استفاده گردید و مشاهده شد که افراد در هر آزمایشی که انجام می‌دادند مهارت کسب می‌کردند و سعی در کم کردن کلیه‌ی خطاهایی که در آزمایش ممکن بود رخ دهد، داشتند.

در قسمت تحلیل بر روی داوطلبان، آن‌ها را به دو گروه افرادی که بازی رایانه‌ای انجام می‌دهند و یا ساز موسیقی می‌نوازند و افرادی که هیچ یک از این دو کار را انجام نمی‌دهند، تقسیم کردیم. با تحلیل‌هایی که بر روی دو گروه انجام شد ملاحظه گردید که گروهی که بازی رایانه‌ای انجام می‌دهند یا ساز می‌نوازند با دقت عمل و خطای کمتری نسبت به گروهی که هیچ‌یک را انجام نمی‌دهند، عمل می‌کنند.

بنابراین به صورت کلی می‌توان گفت نقش شبیه‌سازها در آموزش و ارزیابی جراحان بسیار کارآمدتر و کم هزینه‌تر از روش‌های دیگر است و روند رشد و یادگیری را در انجام عمل‌های جراحی سرعت می‌بخشد.

### ۵-۱- پیشنهادها

برای ارتقای هرچه بیشتر این پروژه می‌توان کارهایی انجام داد که به صورت مختصر به آن اشاره می‌کنیم:

در بخش شبیه‌ساز عمل جراحی، در این پروژه امکان برداشتن قسمتی از مغز و دستکاری آن وجود ندارد. پیشنهاد می‌شود در کارهای آتی از یک مدل سه بعدی مغز استفاده شود که لایه‌های مختلف آن را در بر می‌گیرد و امکان برداشتن بافت تومور و مغزی به آن اضافه شود تا شبیه‌ساز به حداکثر کارایی خود برسد.

در این پروژه، تنها ابزار منعقدساز دو قطبی شبیه‌سازی شد. در کارهای آینده می‌توان ابزارهای دیگر مانند مکنده‌ها و قیچی‌های جراحی را اضافه کرد.

در این پروژه، برای ارزیابی، از دانشجویان آزمایش به عمل آمد حال آن‌که شبیه‌سازهای عمل جراحی برای جراحان و دستیاران جراحی طراحی می‌شود. پیشنهاد می‌شود در کارهای آینده آزمایش‌ها بر روی جراحان و دستیاران جراحی صورت گیرد تا نتایج قابل اعتمادتری به دست آید.



## منابع و مراجع

١. Coles, T.R., D. Meglan, and N.W. John, *The role of haptics in medical training simulators: A survey of the state of the art*. IEEE Transactions on haptics, ٢٠١٠. ٤(١): p. ٥١-٦٦.
٢. WebMd. Available from: <https://www.webmd.com/a-to-z-guides/news/٢٠١٢١٢٢٠/thousands-mistakes-surgery#١>.
٣. Solyar, A., et al., *Endoscopic Sinus Surgery Simulator as a teaching tool for anatomy education*. The American journal of surgery, ٢٠٠٨. ١٩٦(١): p. ١٢٠-١٢٤.
٤. Fried, M.P., et al., *The use of surgical simulators to reduce errors*. ٢٠٠٥, AGENCY FOR HEALTHCARE RESEARCH AND QUALITY ROCKVILLE MD.
٥. Vince, J., *Introduction to virtual reality*. ٢٠٠٤: Springer Science & Business Media.
٦. Iwata, H., *History of haptic interface*, in *Human haptic perception: Basics and applications*. ٢٠٠٨, Springer. p. ٣٥٥-٣٦١.
٧. Salisbury, K., F. Conti, and F. Barbagli, *Haptic rendering: introductory concepts*. IEEE computer graphics and applications, ٢٠٠٤. ٢٤(٢): p. ٢٤-٣٢.
٨. Jyothi, B.D. and R. Krishnaiah, *Haptic Technology-A Sense of Touch*. International Journal of Science and Research (IJSR), India Online ISSN, ٢٠١٣: p. ٢٣١٩-٧٠٦٤.
٩. Meshram, M.S.N. and A.M. Sahu, *Haptic science and technology in surgical simulation, medical training and military application*. International Journal of Computer Science and Mobile Computing, ٢٠١٤. ٣(٤): p. ١٥٦-١٦٥.
١٠. ٣dsystems. Available from: <https://www.٣dsystems.com/haptics-devices/touch>.

۱۱. GOYAL, M., et al., *Haptics: Technology Based on Touch*. International Journal of Scientific Research Engineering & Technology (IJSRET), ۲۰۱۳. ۲(۸): p. ۴۶۸-۴۷۱.
۱۲. Mullins, J., C. Mawson, and S. Nahavandi. *Haptic handwriting aid for training and rehabilitation*. in ۲۰۰۵ *IEEE International Conference on Systems, Man and Cybernetics*. ۲۰۰۵. IEEE.
۱۳. Song, G., S. Guo, and Q. Wang. *A Tele-operation system based on haptic feedback*. in ۲۰۰۶ *IEEE International Conference on Information Acquisition*. ۲۰۰۶. IEEE.
۱۴. *neuro Sim*.
۱۵. Delorme, S., et al., *NeuroTouch: a physics-based virtual simulator for cranial microneurosurgery training*. Operative Neurosurgery, ۲۰۱۲. ۷۱(suppl\_۱): p. ons۳۲-ons۴۲.
۱۶. Azarnoush, H., et al., *Neurosurgical virtual reality simulation metrics to assess psychomotor skills during brain tumor resection*. International journal of computer assisted radiology and surgery, ۲۰۱۵. ۱۰(۵): p. ۶۰۳-۶۱۸.
۱۷. CHAI<sup>۳</sup>D. Available from: <https://www.chai۳d.org/download/doc/html/chapter۱-introduction.html>.
۱۸. Wasserman, L., *All of statistics: a concise course in statistical inference*. ۲۰۱۳: Springer Science & Business Media.

## پیوست ها

پ- ۱- فرم معرفی پروژه برای داوطلبان

### معرفی پروژه

**عنوان:** شبیه سازی واقعیت مجازی تعامل ابزار با بافت جهت مطالعه روند یادگیری اعمال نیروی دلخواه

**محققان:** رضا کریمزاده، دکتر حامد آذرنوش

- تقاضا داریم قبل از شروع آزمایش اطلاعات راجع به پروژه و دستورالعمل در این مطالعه را که در این چند صفحه به آن پرداخته ایم را به دقت مطالعه فرمایید. پیشاپیش برای این همکاری کمال تشکر را داریم.

#### معرفی:

در این پروژه با استفاده از یکی از تکنولوژی های واقعیت مجازی به نام Haptic که حس لامسه را به محیط مجازی اضافه می کند، یک شبیه ساز جراحی برای تعامل ابزار جراحی و بافت مغز ارائه شده است. در این مطالعه هدف بررسی یادگیری و ارزیابی اعمال نیروی مناسب به بافت مغزی شبیه سازی شده است.

دستگاه مورد استفاده Geomagic Touch نام دارد، این دستگاه ساخت شرکت 3D Systems است. تعامل این دستگاه با محیط مجازی به صورت بلادرنگ<sup>۱</sup> است و با وصل شدن به رایانه و گرفتن اطلاعات محیط مجازی نیروی حاصل از برخورد نشانگر با اجسام و خواص فیزیکی آنها را به دست کاربر منتقل می کند (شکل ۱).

<sup>۱</sup> Realtime



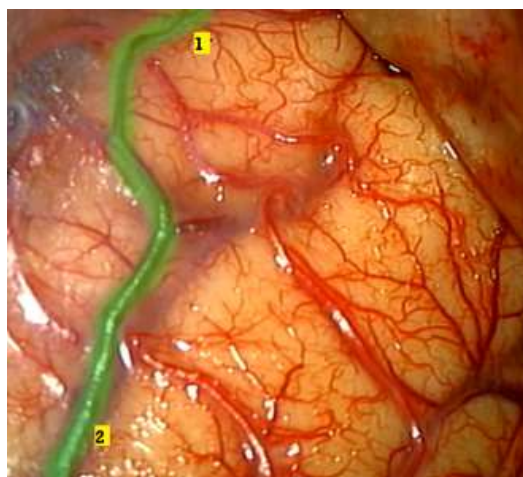
شکل ۳۲ دستگاه Geomagic touch

### دستورالعمل:

قصد داریم با استفاده از ابزار منعقد سازی خونریزی دوقطبی<sup>۱</sup> که با استفاده از جریان الکتریسیته بین دو الکترود خود، رگ خونی را منعقد می سازد، یکی از رگ های خونی بافت مغزی مدل شده که در شکل ۲ با رنگ سبز مشخص شده است را به صورت مجازی منعقد کنیم.

- این کار باید از بالا به پایین، مطابق عدد گذاری در تصویر صورت گیرد.
- باید نیروی اعمالی متناسب با بافت باشد و سرعت منعقد کردن مناسب باشد تا در اثر منعقد کردن بافت نسوزد، در صورت ایجاد سوختگی، بافت به رنگ مشکی در می آید که نشانه اعمال نیروی بیش از حد به بافت یا نگه داشتن بیش از حد منعقد ساز دوقطبی در آن نقطه است.

<sup>۱</sup> Bipolar Electrocautery



شکل ۳۳ ناحیه‌ی مورد آزمایش

### نکات مهم در انجام تست:

- این آزمایش ۱۱ مرتبه برای شما تکرار می‌گردد.
- برای شروع هر آزمایش گوی آبی رنگ در پنجره‌ی مجازی (شکل ۳) را با کلید تیره رنگ که روی عملگر دستگاه است بزنید (شکل ۴)؛ در صورت شروع گوی روشن می‌شود.



شکل ۳۴ پنجره‌ی مجازی

- به متن‌های که در گوشه‌ی بالا-چپ تصویر نشان داده می‌شود توجه نمایید، این متن‌ها نشانگر شماره‌ی آزمایش و شروع یا پایان آن است.

- پس از اتمام هر آزمایش گوی قرمز رنگ را با کلید تیره رنگ که روی دستگیره‌ی دستگاه است بزنید که در صورت فشردن، روشن خواهد شد و متن پایان آزمایش در گوشه‌ی بالا-چپ نشان داده خواهد شد.
- دقت کنید در طول آزمایش بر روی گوی قرمز کلید نزنید.
- بعد از زدن گوی شروع، امکان منعقد کردن رگ با فشردن همان کلید تیره رنگ که روی دستگیره‌ی دستگاه است، فراهم می‌شود.
- در صورتیکه سوالی برایتان پیش آمده، قبل از شروع آزمایش از ناظر آزمایش سوال فرمایید.



شکل ۳۵ دکمه‌ی عملگر

پ-۲- فرم رضایت‌نامه حضور داوطلب در آزمایش

### فرم رضایت‌نامه:

این فرم دلیل حضور و نقش شما در این مطالعه را روشن می‌سازد لذا خواهشمندیم آن را به دقت مطالعه کنید، و در صورت تمایل به شرکت در مطالعه در پایان این فرم رضایت مندی خود را تایید کنید.

### **هدف پروژه:**

در این پروژه هدف بررسی روند یادگیری انجام یک بخش از عمل جراحی مغز است. این بررسی با استفاده از یک دستگاه واقعیت مجازی و تعامل مجازی با بافت انجام می‌گیرد.

### **نقش شما:**

با انجام این آزمایش اعمالی که شما بر روی بافت انجام می‌دهید ثبت و ذخیره می‌شود و پس از آن به تحلیل داده‌های بدست آمده از آزمایشی که شما انجام داده‌اید می‌پردازیم. حضور شما و جدیت در انجام این کار کمک شایانی برای پیش‌برد این پروژه است.

## مزایای شرکت در مطالعه:

با شرکت در این مطالعه شما با تکنولوژی جدید در حوزه‌ی واقعیت مجازی آشنا می‌شوید. همچنین با تعامل داشتن با محیط مجازی که به صورت شبیه‌ساز عمل جراحی مغز طراحی شده‌است، یک تجربه‌ی علمی جدید کسب خواهید کرد.

## اطلاعت شخصی:

اطلاعات وارد شده‌ی شخصی محرمانه خواهد ماند و هیچگاه در معرض دید عموم قرار نخواهد گرفت، لذا خواهم‌شمن‌دیم اطلاعات واقعی و صحیح خود را وارد کنید تا در تحلیل داده‌های بدست آمده نتیجه‌ی واقعی حاصل شود.

## پذیرایی و جایزه‌ی فرد برتر:

پس از اتمام آزمایش برای تشکر از مشارکت شما و وقتی که صرف کردید ناظر آزمایش، یک بسته پذیرایی در اختیار شما می‌گذارد. همچنین پس از تحلیل تمامی نتایج، به شخصی که بهترین نتایج را کسب کرده باشد جایزه‌ای ناقابل به مبلغ یک میلیون ریال تعلق خواهد گرفت.

## ارتباط با ما:

در صورتی که سوال یا نگرانی در خصوص این مطالعه برایتان پیش آمد، در هر زمانی می‌توانید با استفاده از راه‌های ارتباطی زیر با مسوول آزمایش (رضا کریم‌زاده) تماس برقرار کنید. پاسخ دادن به سوال شما در خصوص این مطالعه باعث مسرت ماست.

ایمیل: RezaKarimzadeh۱۹۹۶@gmail.com شماره همراه: ۰۹۱۳۷۰۳۲۰۶۶

## رضایت‌نامه:

اینجانب ..... از حقوق خود در این مطالعه آگاهی کامل دارم و موافق با شرکت در این آزمایش هستم.

نام و نام خانوادگی: ..... تاریخ: ..... امضا

توسط ناظر آزمایش تکمیل شود:

اینجانب ..... ناظر آزمایش، گواهی می‌کنم که شرکت کننده تمامی فرم‌ها را دریافت کرده و پس از مطالعه، با رضایت خود در این آزمایش شرکت کرده‌است.

نام و نام خانوادگی: ..... تاریخ: ..... امضا

پ- ۳ فرم اطلاعات شخصی داوطلب

### فرم اطلاعات شخصی :

- نام و نام خانوادگی: .....
  - شماره تلفن همراه: .....
  - ایمیل: .....
  - جنسیت: ☐ مرد ☐ زن
  - سن: .....
  - شغل: (اگر دانشجو هستید رشته‌ی تحصیلی را بنویسید). .....
  - آیا حرفه‌ای که انجام می‌دهید نیاز به استفاده دقیق حرکت دست دارد؟  
 پیوسته و دایم ☐ گاه به گاه ☐ هیچگاه ☐
  - حرفه‌ی شما چیست؟ .....
  - به طور میانگین هر هفته چند ساعت یک ساز موسیقی می‌نوازید؟ (در صورتی که نمی‌نوازید \*  
 وارد کنید). .....
  - به طور میانگین در هر هفته چند ساعت بازی‌های کامپیوتری، بازی می‌کنید؟ (در صورت بازی  
 نکردن، \* وارد کنید). .....
- مشخص کنید چه نوع بازی کامپیوتری را بازی می‌کنید:



- ☐ شخص اول تیراندازی ( مانند Call of Duty )
- ☐ ورزشی (مانند PES)
- ☐ استراتژیک (مانند Starcraft )
- ☐ کینکت XBOX یا عینک واقعیت مجازی
- ☐ سایر (مشخص کنید) .....

پ- ۴ فرم ارزیابی داوطلب از آزمایش

### فرم ارزیابی و نظرخواهی:

- این فرم را پس از اتمام آزمایش برای یاری هرچه بهتر ما در رفع ایرادات محیط طراحی شده و پیشرفت شبیه ساز جراحی پر کنید.
- همانطور که گفته شد هدف ما شبیه سازی بافت مغزی و ابزار منعقد کننده‌ی دو قطبی است و در این آزمایش قسمتی از یک رگ خونی مغز را منعقد کردید.

(۵) میزان دشواری این آزمایش را از ۱ تا ۵ نمره دهی کنید. (۱-بسیار آسان ۵-بسیار دشوار)

۱ ۲ ۳ ۴ ۵

(۶) میزان واقعی بودن محیط دیداری شبیه سازی شده (۱-کاملاً غیر واقعی ۵-کاملاً واقعی)

۱ ۲ ۳ ۴ ۵

(۷) میزان واقعی بودن نیروهای وارد شده به دست (۱-کاملاً غیر واقعی ۵-کاملاً واقعی)

۱ ۲ ۳ ۴ ۵

(۸) رضایت کلی شما از این شبیه ساز جراحی چیست؟ (۱-کاملاً ناراضی ۵-کاملاً راضی)

۱ ۲ ۳ ۴ ۵

۹) نظرتان را در مورد چگونگی بهبود محیط شبیه سازی شده و توصیه‌هایی برای بهتر شدن آزمایش بنویسید.

.....

.....

پ-۵ کد کامل شبیه‌ساز عمل جراحی

```
=====//
=====

*/
Software License Agreement (BSD License)
.Copyright (c) ۲۰۰۳-۲۰۱۶, CHAI۳D
(www.chai۳d.org)

.All rights reserved

Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions
:are met

Redistributions of source code must retain the above copyright *
.notice, this list of conditions and the following disclaimer

Redistributions in binary form must reproduce the above *
copyright notice, this list of conditions and the following
disclaimer in the documentation and/or other materials provided
.with the distribution
```

Neither the name of CHAIrD nor the names of its contributors may \*  
be used to endorse or promote products derived from this software  
.without specific prior written permission

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND  
CONTRIBUTORS

AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT"  
LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS  
FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE  
COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT,  
,INDIRECT

INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES  
, (INCLUDING

;BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES  
LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER  
CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT  
LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN  
ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE  
.POSSIBILITY OF SUCH DAMAGE

<author <<http://www.chaird.org>\

author Francois Conti\

\$ version ۳,۲,۰ \$Rev: ۲۰۴۹\

/\*

=====//  
=====

-----//  
-----

```
"include "chaird.h#

----- //
-----

#include <GLFW/glfw.h#

----- //
-----

#include <fstream#
#include <iostream#
#include <direct.h#

;using namespace chaird
;using namespace std

----- //
-----

----- //
-----

GENERAL SETTINGS //

----- //
-----

stereo Mode //

*/

C_STEREO_DISABLED:          Stereo is disabled
C_STEREO_ACTIVE:           Active stereo for OpenGL NVIDIA
QUADRO cards

C_STEREO_PASSIVE_LEFT_RIGHT: Passive stereo where L/R images are
rendered next to each other

C_STEREO_PASSIVE_TOP_BOTTOM: Passive stereo where L/R images are
rendered above each other
```

```
/*  
  
;cStereoMode stereoMode = C_STEREO_DISABLED  
  
fullscreen mode //  
;bool fullscreen = false  
  
mirrored display //  
;bool mirroredDisplay = false  
  
state machine states //  
;const int STATE_IDLE = ۱  
;const int STATE_MODIFY_MAP = ۲  
;const int STATE_MOVE_CAMERA = ۳  
  
-----//  
-----  
DECLARED VARIABLES //  
-----//  
-----  
  
;bool updateRender = false, firstTime = false  
for setting position of curser//  
;double maxpx = ۰, maxpy = ۰  
  
;float prev_time = ۰  
  
;string CaseName = "test
```

```
a line of text //

;CLabel* text

;CLabel* start_text

;float prev_t

for start electrocautery//

;bool electrocautery = false


;int test_number = \


for reinitialize map //

;bool reinit = true

;bool stop_working = false

;CAudioSource* electrocauterysource


;CAudioBuffer* electrocauterysound


;CVector<double> posTool


a virtual voxel like object //

;CVoxelObject* tumorobject


D texture object<double> //

;CTexture<double>Ptr texture


;int voxelModelResolution = ٦٤

D image data<double> //

;CMultiImagePtr tumorimage
```

```
mutex to voxel //
;CMutex mutexVoxel

a virtual voxel like object //
;cVoxelObject* tumor

---saving data flag//
;bool islogging = false

;cShapeSphere* start_obj
;cShapeSphere* end_obj

;vector<double> timeVector
;vector<double> forceVector
;vector<double> forcefrictionX
;vector<double> forcefrictionY
;vector<double> PositionX_vec
;vector<double> PositionY_vec
;vector<double> PositionZ_vec
;vector<double> AngleZ_vec
;vector<double> AngleY_vec

;const int sizeX = ۱۲۰, sizeY = ۱۲۰
;{ \ } = float array_color_coef[sizeX*sizeY]
;float set_array_color_coef[sizeX*sizeY]
```

```

;float color_coef = \
;double maxForce = .

;cPrecisionClock *timer
for saving map position//
;double mapposition[r][sizeX*sizeY]

;float vertexcolorArray[r][sizeX*sizeY]

;cVectorrd forceFieldfriction
;cVectorrd forceField

;cLevel* levelForce

d drillr //
;cMultiMesh* drill
;cMultiMesh* metal

a world that contains all objects of the virtual environment //
;cWorld* world

a camera to render the world in the window display //
;cCamera* camera

a light source to illuminate the objects in the world //
;cDirectionalLight *light

a haptic device handler //
;cHapticDeviceHandler* handler

```



```
a pointer to the current haptic device //
```

```
;cGenericHapticDevicePtr hapticDevice
```

```
a virtual tool representing the haptic device in the scene //
```

```
;cToolCursor* tool
```

```
radius of tool for graphic and haptic representation //
```

```
;double hapticRadius
```

```
;double displayRadius
```

```
a virtual mesh like object //
```

```
;cMesh* object
```

```
a small magnetic line used to constrain the tool along the  //
```

```
vertical axis
```

```
;cShapeLine* magneticLine
```

```
two sphere placed at both end of the magnetic line //
```

```
;cShapeSphere* sphereA
```

```
;cShapeSphere* sphereB
```

```
a colored background //
```

```
;cBackground* background
```

```
a font for rendering text //
```

```
;cFontPtr font
```

```
a label to display the rate [Hz] at which the simulation is  //
```

```
running
```

```
;cLabel* labelRates

a label to explain what is happening //

;cLabel* labelMessage

a flag that indicates if the haptic simulation is currently //
running

;bool simulationRunning = false

a flag that indicates if the haptic simulation has terminated //
;bool simulationFinished = true

a frequency counter to measure the simulation graphic rate //
;cFrequencyCounter freqCounterGraphics

a frequency counter to measure the simulation haptic rate //
;cFrequencyCounter freqCounterHaptics

state machine //

;int state = STATE_IDLE

camera status //

;bool flagCameraInMotion = false

informs the graphic display callback to update the display list //
.when the topology of the 2D height map is changed //

;bool flagMarkForUpdate = false

;bool flagcomputeN = false

haptic thread //
```

```

; cThread* hapticsThread

a handle to window display context //
; GLFWwindow* window = NULL

current width of window //
; int width = .

current height of window //
; int height = .

swap interval for the display context (vertical //
synchronization)
; int swapInterval = \

root resource path //
; string resourceRoot

----- //
-----

DECLARED MACROS //

----- //
-----

convert to resource path //
define RESOURCE_PATH(p)      #
(char*) ((resourceRoot+string(p)).c_str())

```

```
----- //
-----

DECLARED FUNCTIONS //

----- //
-----

for saving data//
;void dataSaving(void)

for reinitialize map//
;void reinitialize(void)

callback when the window display is resized //
void windowSizeCallback(GLFWwindow* a_window, int a_width, int
;a_height)

callback when a key is pressed //
void keyCallback(GLFWwindow* a_window, int a_key, int a_scancode,
;int a_action, int a_mods)

callback when an error GLFW occurs //
;void errorCallback(int error, const char* a_description)

this function renders the scene //
;void updateGraphics(void)

this function contains the main haptics simulation loop //
;void updateHaptics(void)

this function closes the application //
;void close(void)

loads a bitmap file and create 2D height map based on pixel //
```

```

color

;()int loadHeightMap

;()void applyforce

;void buildVoxelShape(double)

=====//
=====

*/

DEMO:    interaction forces are then computed and sent to the
.device

/*

=====//
=====

int main(int argc, char* argv[])
}

-----//
-----

INITIALIZATION //

-----//
-----

;cout << endl

;cout << "-----" << endl

;cout << "CHAIrD" << endl

;cout << "Surgical Simulator" << endl

cout << "-----" << endl <<
;endl << endl

;cout << "Instructions:" << endl << endl

;cout << "Keyboard Options:" << endl << endl

;cout << "[\] - Haptic Shading (ON/OFF)" << endl

;cout << "[\] - Wireframe (ON/OFF)" << endl

```

```

;cout << "[f] - Enable/Disable full screen mode" << endl
;cout << "[m] - Enable/Disable vertical mirroring" << endl
;cout << "[\۳] - Exit application" << endl
;cout << endl << endl

parse first arg to try and locate resources //
resourceRoot = string(argv[۰]).substr(۰,
;string(argv[۰]).find_last_of("/\\") + ۱)

-----create folder for save data//

;":cout << "Enter Your Name*/
/*;cin >> CaseName
;string(dir) = "D:\\term۱\\final project\\Data\\" + CaseName
;()auto folder = dir.c_str
;mkdir(folder)_

-----timer//

;()timer = new cPrecisionClock

for (int i = ۰; i < sizeX*sizeY; i++) array_color_coef[i] //
;= \

-----//
-----

OPEN GL - WINDOW DISPLAY //

-----//
-----

initialize GLFW library //

```

```
if (!glfwInit())
{
;cout << "failed initialization" << endl
;(\...\...)cSleepMs
;return \
{

set error callback //
;glfwSetErrorCallback(errorCallback)

compute desired size of window //
const GLFWvidmode* mode =
;glfwGetVideoMode(glfwGetPrimaryMonitor())
;int w = \,\^ * mode->height
;int h = \,\^ * mode->height
;int x = \,\^ * (mode->width - w)
;int y = \,\^ * (mode->height - h)

set OpenGL version //
;glfwWindowHint(GLFW_CONTEXT_VERSION_MAJOR, \)
;glfwWindowHint(GLFW_CONTEXT_VERSION_MINOR, \)

set active stereo mode //
if (stereoMode == C_STEREO_ACTIVE)
{
;glfwWindowHint(GLFW_STEREO, GL_TRUE)
{
else
}
```

```
;glfwWindowHint(GLFW_STEREO, GL_FALSE)

{

create display context //
;window = glfwCreateWindow(w, h, "CHAIrD", NULL, NULL)
if (!window)
}
;cout << "failed to create window" << endl
;(1000)cSleepMs
;()glfwTerminate
;return \
{

get width and height of window //
;glfwGetWindowSize(window, &width, &height)

set position of window //
;glfwSetWindowPos(window, x, y)

set key callback //
;glfwSetKeyCallback(window, keyCallback)

set resize callback //
;glfwSetWindowSizeCallback(window, windowSizeCallback)

set current display context //
;glfwMakeContextCurrent(window)
```



```
sets the swap interval for the current display context //
```

```
;glfwSwapInterval(swapInterval)
```

```
ifdef GLEW_VERSION#
```

```
initialize GLEW library //
```

```
if (glewInit() != GLEW_OK)
```

```
}
```

```
;cout << "failed to initialize GLEW library" << endl
```

```
;()glfwTerminate
```

```
;return \
```

```
{
```

```
endif#
```

```
----- //
```

```
-----
```

```
WORLD - CAMERA - LIGHTING //
```

```
----- //
```

```
-----
```

```
.create a new world //
```

```
;()world = new cWorld
```

```
set the background color of the environment //
```

```
;()world->m_backgroundColor.setBlack
```

```
create a camera and insert it into the virtual world //
```

```
;camera = new cCamera(world)
```

```
;world->addChild(camera)
```

```

define a basis in spherical coordinates for the camera //
camera->setSphericalReferences(cVector3d(0, 0, 0), //
origin
cVector3d(0, 0, 1), // zenith direction
cVector3d(1, 0, 0)); // azimuth direction

camera->setSphericalDeg(2, 70, // spherical coordinate
radius
spherical coordinate polar angle // 0, 180
spherical coordinate azimuth angle // 0, 360);

set position and orientation /**/
camera->set(cVector3d(0, 0, 0, 0, 0, 0), // camera position
(eye
cVector3d(0, 0, 0, 0, 0, 0), // lookat position (target)
cVector3d(1, 0, 0, 0, 0, 0)); // direction of the (up)
/*vector

set the near and far clipping planes of the camera //
anything in front or behind these clipping planes will //
not be rendered

;(0, 0, 0, 0)camera->setClippingPlanes

set stereo mode //
;camera->setStereoMode(stereoMode)

set stereo eye separation and focal length (applies only //
if stereo is enabled)

;(0, 0.2)camera->setStereoEyeSeparation

```

```
; (۲,۰) camera->setStereoFocalLength

set vertical mirrored display mode //
; camera->setMirrorVertical(mirroredDisplay)

create a light source //
; light = new cDirectionalLight(world)

attach light to camera //
; world->addChild(light)

enable light source //
; light->setEnabled(true)

position the light source //
; (۲,۰ , ۰,۰ , ۲,۰) light->setLocalPos//

define the direction of the light beam //
; (۰,۰- , ۰,۰ , ۲,۰-) light->setDir

enable this light source to generate shadows //
; light->setShadowMapEnabled(false)*/

set the resolution of the shadow map //
; () light->m_shadowMap->setQualityLow
; () light->m_shadowMap->setQualityMedium//

set light cone half angle //
```

```
/*;(۲۰)light->setCutOffAngleDeg

-----//
-----

HAPTIC DEVICES / TOOLS //
-----//
-----

create a haptic device handler //
;()handler = new cHapticDeviceHandler

get access to the first available haptic device //
;handler->getDevice(hapticDevice, ۰)

retrieve information about the current haptic device //
cHapticDeviceInfo hapticDeviceInfo = hapticDevice-
;()>getSpecifications

if the device has a gripper, then enable it to behave //
like a user switch

;hapticDevice->setEnableGripperUserSwitch(true)

create a ۲D tool and add it to the camera //
;tool = new cToolCursor(world)
;world->addChild(tool)

connect the haptic device to the tool //
;tool->setHapticDevice(hapticDevice)
```

```
.hide the device sphere. only show proxy //
```

```
;tool->setShowContactPoints(true, false)
```

```
set color of tool //
```

```
tool->m_hapticPoint->m_sphereProxy->m_material-
```

```
;()->setWhiteAliceBlue
```

```
.set the physical radius of the proxy //
```

```
;hapticRadius = .,.
```

```
;displayRadius = .,....۲
```

```
;tool->setRadius(displayRadius, hapticRadius)
```

```
map the physical workspace of the haptic device to a //
```

```
.larger virtual workspace
```

```
; (۲,۰)tool->setWorkspaceRadius
```

```
oriente tool with camera //
```

```
;tool->setLocalRot(camera->getLocalRot())//
```

```
haptic forces are enabled only if small forces are first //
```

```
;sent to the device
```

```
this mode avoids the force spike that occurs when the //
```

```
application starts when
```

```
.the tool is located inside an object for instance //
```

```
;tool->setWaitForSmallForce(true)
```

```
initialize tool by connecting to haptic device //
```

```
;()->start
```

```
////////////////////////////////////  
////////////////////////////////////  
  
MAP //  
  
////////////////////////////////////  
////////////////////////////////////  
  
create a virtual mesh //  
;()object = new cMesh  
  
add object to world //  
;world->addChild(object)  
  
set the position of the object at the center of the world //  
;(0,0,0,0,0,0)object->setLocalPos  
  
Since we want to see our polygons from both sides, we //  
.disable culling  
;object->setUseCulling(false)//  
  
load default map //  
;()loadHeightMap  
  
  
////////////////////////////////////  
////////////////////////////////////  
  
"OBJECT "DRILL //  
  
////////////////////////////////////  
////////////////////////////////////
```

```

.create a new mesh //

;()drill = new cMultiMesh

load a drill like mesh and attach it to the tool //

bool fileload = drill-
>loadFromFile(RESOURCE_PATH("../resources/models/drill/bipmamad.۳d
;s"))

if (!fileload)

}

if defined(_MSVC)#

fileload = drill-
>loadFromFile("../../bin/resources/models/drill/drill.۳ds")

endif#

{

if (!fileload)

}

printf("Error - ۳D Model failed to load
;correctly.\n")

;()close

;()- return

{

resize tool mesh model //

;(,)\)drill->scale

;(,۰۰۰- ,۰,۰۰- ,۲,\)drill->translate

;(\۸۰ ,۰ ,۰)drill->rotateAboutGlobalAxisDeg//

;(\۸۰ ,۲ ,۲ ,۰)drill->rotateAboutLocalAxisDeg

```

```

attach drill to tool //

;tool->m_image->addChild(drill)

----- //
-----

CREATING OBJECTS //

----- //
-----

read the scale factor between the physical workspace of //
the haptic

device and the virtual workspace defined for the tool //

double workspaceScaleFactor = tool-
;()>getWorkspaceScaleFactor

get properties of haptic device //

;double maxLinearForce = hapticDeviceInfo.m_maxLinearForce

double maxLinearDamping =
;hapticDeviceInfo.m_maxLinearDamping

double maxStiffness = hapticDeviceInfo.m_maxLinearStiffness
;/ workspaceScaleFactor

-----around metal//
-----

.create a new mesh //

;()metal = new cMultiMesh

;world->addChild(metal)

;(0,1,2,1,18)metal->translate//

;(0,70,1,5,1,5)metal->setLocalPos

```



```

; (180, 2, 2, 0) metal->rotateAboutLocalAxisDeg

load a metal like mesh //

fileload = metal-
>loadFromFile(RESOURCE_PATH("../resources/models/drill/aroundmetal
; ۲ds"))

if (!fileload)

}

if defined(_MSVC)#

fileload = metal-
>loadFromFile("../..../bin/resources/models/drill/aroundmetal.۲ds
; ")

endif#

{

if (!fileload)

}

printf("Error - ۲D Model failed to load
; correctly.\n")

; () close

; ()- return

{

; (0, 0.۲۰۸) metal->scale

-----start obj//

; (0, 1) start_obj = new cShapeSphere
; world->addChild(start_obj)

; (0, ۲۰- , 1, ۴ , 0, ۲-) start_obj->setLocalPos

; () start_obj->m_material->setBlueDark

create haptic effect and set properties //

```

```

;()start_obj->createEffectSurface
;start_obj->setStiffness(0,7*maxStiffness)
create haptic effect and set haptic properties //
;()start_obj->createEffectMagnetic
;(0,0.8)start_obj->m_material->setMagnetMaxDistance
start_obj->m_material->setMagnetMaxForce(0,2 *
;maxLinearForce)
-----end_obj//

;(0,1)end_obj = new cShapeSphere
;world->addChild(end_obj)
;(0,200,1,1,0,2)end_obj->setLocalPos
;()end_obj->m_material->setRedDark
create haptic effect and set properties //
;()end_obj->createEffectSurface
;end_obj->setStiffness(0,7*maxStiffness)
create haptic effect and set haptic properties //
;()end_obj->createEffectMagnetic
;(0,0.8)end_obj->m_material->setMagnetMaxDistance
end_obj->m_material->setMagnetMaxForce(0,2 *
;maxLinearForce)

-----//

use display list to increase graphical rendering //
performance
;object->setUseDisplayList(true)

```

```
----- //
-----

WIDGETS //
----- //
-----

create a font //
;()font = NEW_CFONTCALIBRI۲۰

create a label to display the haptic and graphic rate of //
the simulation

;labelRates = new cLabel(font)
;camera->m_frontLayer->addChild(labelRates)

set font color //
;()labelRates->m_fontColor.setBlack

create a label with a small message //
;labelMessage = new cLabel(font)
;camera->m_frontLayer->addChild(labelMessage)

set font color //
;()labelMessage->m_fontColor.setBlack
```

```

set text message //

;labelMessage->setText("touch surface - press user switch ")

create a background //

;()background = new cBackground
;camera->m_backLayer->addChild(background)

set background properties //
,(background->setCornerColors(cColorf(0,0f, 0,20f, 0,0f
,cColorf(0,0f, 0,20f, 0,0f)
,cColorf(0,0f, 0,20f, 0,0f)
;(cColorf(0,0f, 0,20f, 0,0f)
-----notifications//

;()font = NEW_CFONTCALIBRI;
create a small text //
;text = new cLabel(font)
;camera->m_frontLayer->addChild(text)
;()text->m_fontColor.setRedCrimson
text->setText("For Starting Test Use UserSwitch \nAnd Press
;Blue Bottom.")
;(100 , 200)text->setLocalPos

start buttom text//
;start_text = new cLabel(font)
;camera->m_frontLayer->addChild(start_text)
;()start_text->m_fontColor.setRedCrimson
;start_text->setText("START")
;(100 , 100)start_text->setLocalPos

```

```
finish button text//
;start_text = new cLabel(font)
;camera->m_frontLayer->addChild(start_text)
;()start_text->m_fontColor.setRedCrimson
;start_text->setText("STOP")
;(۴۳ , ۱۰۱۰)start_text->setLocalPos

-----//
-----

START SIMULATION //
-----//
-----

create a thread which starts the main haptics rendering //
loop
;()hapticsThread = new cThread
hapticsThread->start(updateHaptics,
;CTHREAD_PRIORITY_HAPTICS)

-----//
-----

MAIN GRAPHIC LOOP //
-----//
-----

call window size callback at initialization //
;windowSizeCallback(window, width, height)

main graphic loop //
```

```
while (!glfwWindowShouldClose(window))
{
    get width and height of window //
    ;glfwGetWindowSize(window, &width, &height)

    render graphics //
    ;()updateGraphics

    swap buffers //
    ;glfwSwapBuffers(window)

    process events //
    ;()glfwPollEvents

    signal frequency counter //
    ;()\freqCounterGraphics.signal
    {

        close window //
        ;glfwDestroyWindow(window)

        terminate GLFW library //
        ;()glfwTerminate

        exit //
        ;return .
    }

    ----- //
```

```
-----

void windowSizeCallback(GLFWwindow* a_window, int a_width, int
a_height)
{
update window size //
;width = a_width
;height = a_height
{
-----//
-----

void errorCallback(int a_error, const char* a_description)
{
;cout << "Error: " << a_description << endl
{
-----//
-----

void keyCallback(GLFWwindow* a_window, int a_key, int a_scancode,
int a_action, int a_mods)
{
filter calls that only include a key press //
if ((a_action != GLFW_PRESS) && (a_action != GLFW_REPEAT))
{
;return
{
```

```
option - exit //

else if ((a_key == GLFW_KEY_ESCAPE) || (a_key ==
GLFW_KEY_Q))

}

;glfwSetWindowShouldClose(a_window, GLFW_TRUE)

{

:option - haptic shading //

else if (a_key == GLFW_KEY_)

}

bool useHapticShading = !object->m_material-
;()>getUseHapticShading

object->m_material-
;>setUseHapticShading(useHapticShading)

if (useHapticShading)

;"cout << "> Haptic shading enabled      \r

else

;"cout << "> Haptic shading disabled      \r

{

option - wire mode //

else if (a_key == GLFW_KEY_)

}

;()bool useWireMode = !object->getWireMode

;object->setWireMode(useWireMode)

if (useWireMode)

;"cout << "> Wire mode enabled            \r

else

;"cout << "> Wire mode disabled            \r
```



```
{

option - save to file //

option - toggle fullscreen //
else if (a_key == GLFW_KEY_F)
}
toggle state variable //
;fullscreen = !fullscreen

get handle to monitor //
;()GLFWmonitor* monitor = glfwGetPrimaryMonitor

get information about monitor //
;const GLFWvidmode* mode = glfwGetVideoMode(monitor)

set fullscreen or window mode //
if (fullscreen)
}
glfwSetWindowMonitor(window, monitor, 0, 0,
;mode->width, mode->height, mode->refreshRate)
;glfwSwapInterval(swapInterval)
{
else
}
;int w = 0,8 * mode->height
;int h = 0,8 * mode->height
;int x = 0,8 * (mode->width - w)
```

```

;int y = ٠,٥ * (mode->height - h)

glfwSetWindowMonitor(window, NULL, x, y, w, h,
;mode->refreshRate)

;glfwSwapInterval(swapInterval)

{
{

option - toggle vertical mirroring //
else if (a_key == GLFW_KEY_M)
}

;mirroredDisplay = !mirroredDisplay
;camera->setMirrorVertical(mirroredDisplay)

{

{

----- //
-----

void close(void)
}

stop the simulation //

;simulationRunning = false

wait for graphics and haptics loops to terminate //
} ;(١٠٠)while (!simulationFinished) { cSleepMs

close haptic device //

;()tool->stop

```

```
delete resources //
;delete hapticsThread
;delete world
;delete handler
;delete timer
;delete object
;delete metal
;delete start_obj
;delete end_obj
;delete camera
;delete light
;delete tool
;delete drill

{

----- //
-----

void updateGraphics(void)
}

////////////////////////////////////
////////

UPDATE WIDGETS //

////////////////////////////////////
////////

update haptic and graphic rate data //

labelRates->setText(cStr(freqCounterGraphics.getFrequency()),
```

```

+ " / .) + " Hz

;("cStr(freqCounterHaptics.getFrequency(), .) + " Hz

update position of label //
labelRates->setLocalPos((int)(.,. * (width - labelRates-
;getWidth()))), ١٥)

update position of message label //
labelMessage->setLocalPos((int)(.,. * (width - //
;labelMessage->getWidth()))), ٥٠)

update position of scroll text //
;text->setLocalPos(٢٠,., height - ٨٠)

//////////////////////////////////////
//////////

UPDATE MODEL //

//////////////////////////////////////
//////////

update object normals //
if (flagcomputeN)
}

;()object->computeAllNormals
;flagcomputeN = false

{

if the mesh has been modified we update the display list //
if (flagMarkForUpdate)

```

```
}

;cout << "y" << endl//

;object->markForUpdate(false)

;flagMarkForUpdate = false


{

read timer//

;()double time=timer->start//

;cout << time << endl//

////////////////////////////////////
////////////////////////////////

RENDER SCENE //

////////////////////////////////////
////////////////////////////////

update shadow maps (if any) //

;world->updateShadowMaps(false, mirroredDisplay)

render world //

;camera->renderView(width, height)

wait until all GL commands are completed //

;()glFinish
```

```
check for any OpenGL errors //

;()GLenum err = glGetError

if (err != GL_NO_ERROR) cout << "Error: " <<
;gluErrorString(err) << endl

{

----- //
-----

void updateHaptics(void)
}

initialize state to idle //
;state = STATE_IDLE
;cVector<rd> prevposTool
current tool position //
cVector<rd> toolGlobalPos;          // global world coordinates
cVector<rd> toolLocalPos;          // local coordinates

previous tool position //
cVector<rd> prevToolGlobalPos;      // global world coordinates
cVector<rd> prevToolLocalPos;      // local coordinates

simulation in now running //
;simulationRunning = true
;simulationFinished = false

main haptic simulation loop //
while (simulationRunning)
}
```

```

-----//

compute global reference frames for each object //
;world->computeGlobalPositions(true)

update position and orientation of tool //
;()tool->updateFromDevice

compute interaction forces //
;()tool->computeInteractionForces
}if (tool->isInContact(metal))*/
send forces to haptic device //
;()tool->applyToDevice
;cout << "iaysdh" << endl

/*{

read user switch //
;(.)bool userSwitch = tool->getUserSwitch
;(\)bool userSwitch? = tool->getUserSwitch//

if (tool->isInContact(start_obj)){ tool-
} ;()>applyToDevice

if (tool->isInContact(start_obj) && userSwitch &&
reinit)

}

;()start_obj->m_material->setBlue

```

```
;()end_obj->m_material->setRedDark

;islogging = true

;electrocautery = true

;()reinitialize

;"cout << "> start working ... \r

;reinit = false

;stop_working = true

string starting_text = "Test " +
to_string(test_number) + " Started" + "\nIf You Finish The Test
;\"Press Red Button

;text->setText(starting_text)

;++test_number

;()timer->start

;()timer->on

{

if (tool->isInContact(end_obj)){ tool-
} ;()>applyToDevice

if (tool->isInContact(end_obj) && userSwitch &&
stop_working)

}

;()dataSaving

;electrocautery = false

;()start_obj->m_material->setBlueDark

;()end_obj->m_material->setRed
```



```
;islogging = false

;reinit = true

;stop_working = false

string finishing_text = "Test " +
to_string(test_number - 1) + " Finished" + "\nFor Start New Test
;\"Press Blue Button

;text->setText(finishing_text)

;(0,0)timer->reset

;()timer->stop

;prev_time = 0

if (test_number > 10)
}

glfwSetWindowShouldClose(window,
;GLFW_TRUE)

{
{

update tool position //

;()toolGlobalPos = tool->getDeviceGlobalPos

;()toolLocalPos = tool->getDeviceLocalPos

;()posTool = tool->m_hapticPoint->getGlobalPosProxy

;cout << "x: " << posTool.x() << endl*/

;cout << "y: " << posTool.y() << endl

/*;cout << "z: " << posTool.z() << endl

user clicks with the mouse //

if ((state == STATE_IDLE))

}
```

```

start deforming object //
if (tool->isInContact(object))
{
}

else if (state == STATE_MODIFY_MAP)
{

;()double y = posTool.y
;()double x = -posTool.x
;cout <<"x: " << x << endl//
;cout <<"y: " << y << endl//
;double max = sizeX*sizeY
;int nv, nv۲, nv۳, nv۴
;int column, row, index
;((row = floor((sizeX / ۲) * (x + ۰,۹۵))); // maxpx
column = floor((sizeY / ۲) * (y + ۱,۰۹)); //
;((maxpy
;cout <<"row: " << row << endl//
;cout <<"column: " << column << endl//
;index = (column - ۱) * (sizeX) + sizeY - row - ۱
;cout << index << endl//
cout << mapposition[۰][(row - //
;۱) * (sizeX) + ((column) - ۱)] << endl
;cout << posTool.x() << endl//

if ((posTool.z() <= mapposition[۲][index]) &&
abs(x) <= ۱,۰۹ && abs(y) <= ۱,۰۹ && row <= sizeY && column <=
sizeX && row >= ۰ && column >= ۰) // && y >= -۱ && y <= ۱ && x >= -۱ &&
(x <= ۱
}

```

```

;updateRender = true

cVector<double> offset = toolGlobalPos -
;prevToolGlobalPos

;()double offsetHeight = offset.z//

-----apply force//
-----

if (index < sizeX*sizeY && index>0 &&
posTool.z() <= mapposition[2][index])
}

;cVector<double> desiredPosition

;(0,0,0)cVector<double> force

desiredPosition.set(posTool.x(),
posTool.y(), mapposition[2][index]); // mapposition[1][index],
;(mapposition[2][index]

double Kp = 10; // [N/m]

compute linear force //

forceField = Kp * (desiredPosition -
;posTool)

;force.add(forceField)

-----compute friction//
-----

calculate angle //

cMatrix<double> AngleTool = tool-
;()>getDeviceGlobalRot

cVector<double> Angle =
;()AngleTool.getCol0

;double friction_const = 0.1

forceFieldfriction.set(-
friction_const*sin(Angle.z()*C_PI_DIV_2)*forceField.z(), -

```

```

;friction_const*sin(Angle.z()*C_PI_DIV_۲)*forceField.z(), ۰)

;force.add(forceFieldfriction)

;hapticDevice->setForce(force)

{
else
}

;(۰, ۰, ۰) cVector۳d zeroforce
;hapticDevice->setForce(zeroforce)

{

;double max = sizeX*sizeY

;double desired

double deformation_radius =
;ceil(abs((posTool.z() - mapposition[۲][index]) * ۱۰۰))

for (int i = ۰; i <= deformation_radius;
i++)

}

for (int j = ۰; j <=
ceil(sqrt(deformation_radius*deformation_radius - i*i)); j++)

}

nv = (column + i -
;۱)*(sizeX)+sizeY - (row + j) - ۱

nv۲ = (column - i -
;۲)*(sizeX)+sizeY - (row + j) - ۱

nv۳ = (column + i -
;۱)*(sizeX)+sizeY - (row - j)

nv۴ = (column - i -
;۲)*(sizeX)+sizeY - (row - j)

;cVector۳d posVertex

double relativeDistance =

```

```

;sqrt(i*i + j*j) / (deformation_radius)

float clampedRelativeDistance
:= cClamp(0,1)(relativeDistance)

double w = 0,25*
posTool.z()*exp(-clampedRelativeDistance * 4)* (0,0 +
;0,0*cos(relativeDistance * C_PI))

if (w > 0){ w = 0; }

if (nv >= 0 && nv < max &&
nv<((column + i)*(sizeX)) && nv>((column + i - 1)*(sizeX)))
}

desired =
;mapposition[2][(int)nv]

posVertex = object-
;>m_vertices->getLocalPos(nv)

posVertex.z(w +
;desired)

object->m_vertices-
;>setLocalPos(nv, posVertex)

{

if (nv2 >= 0 && nv2 < max &&
nv2<((column - i - 1)*(sizeX)) && nv2>((column - i - 2)*(sizeX)))

}

desired =
;mapposition[2][(int)nv2]

posVertex = object-
;>m_vertices->getLocalPos(nv2)

posVertex.z(w +
;desired)

object->m_vertices-
;>setLocalPos(nv2, posVertex)

```

```

{

if (nvξ ≥ ۰ && nvξ < max &&
nvξ ≤ ((column - i - ۱)*(sizeX)) && nvξ ≥ ((column - i -
۲)*(sizeX)))

}

desired =
;mapposition[۲][(int)nvξ]

posVertex = object-
;>m_vertices->getLocalPos(nvξ)

posVertex.z(w +
;desired)

object->m_vertices-
;>setLocalPos(nvξ, posVertex)

{

----- cauterization//
-----

if (electrocautery && sqrt(i*i
+ j*j) <= userSwitch && nv ≥ ۰ && nv < max && nv۲ ≥ ۰ && nv۲
< max && nv۳ ≥ ۰ && nv۳ < max && nvξ ≥ ۰ && nvξ < max)

}

;cVector۳d velocity

velocity = tool-
;()>getDeviceLocalLinVel

;double coef_ther = ۰,۱

if (forceField.z() > ۰)

}

color_coef =
;(forceField.z() / ۳۰۰۰)

if
(array_color_coef[(int)nv] > coef_ther && nv != nv۳)
array_color_coef[(int)nv] = array_color_coef[(int)nv] -
;color_coef

```

```

if
(array_color_coef[(int)nv۲] > coef_ther && nv۲ != nv۴)
array_color_coef[(int)nv۲] = array_color_coef[(int)nv۲] -
;color_coef

if
(array_color_coef[(int)nv۳] > coef_ther)
array_color_coef[(int)nv۳] = array_color_coef[(int)nv۳] -
;color_coef

if
(array_color_coef[(int)nv۴] > coef_ther)
array_color_coef[(int)nv۴] = array_color_coef[(int)nv۴] -
;color_coef

{

;color          cColorb

color.set(vertexcolorArray[۰][nv] * array_color_coef[nv],
vertexcolorArray[۱][nv] * array_color_coef[nv],
;vertexcolorArray[۲][nv] * array_color_coef[nv])

object->m_vertices-
;>setColor(nv, color)

;color۲          cColorb

color۲.set(vertexcolorArray[۰][nv۲] * array_color_coef[nv۲],
vertexcolorArray[۱][nv۲] * array_color_coef[nv۲],
;vertexcolorArray[۲][nv۲] * array_color_coef[nv۲])

object->m_vertices-
;>setColor(nv۲, color۲)

;color۳          cColorb

color۳.set(vertexcolorArray[۰][nv۳] * array_color_coef[nv۳],
vertexcolorArray[۱][nv۳] * array_color_coef[nv۳],
;vertexcolorArray[۲][nv۳] * array_color_coef[nv۳])

object->m_vertices-

```

```

;>setColor(nv۳, color۳)

;color{      cColorb

color{.set(vertexcolorArray[.][nv{] * array_color_coef[nv{],
vertexcolorArray[\][nv{] * array_color_coef[nv{],
;vertexcolorArray[۲][nv{] * array_color_coef[nv{])

object->m_vertices-
;>setColor(nv{, color{)

{

;flagcomputeN = true
;flagMarkForUpdate = true

{

{

{

else//if (updateRender && (tool-
>isInContact(object)))

}

;cout << "ksdb" << endl//

if (updateRender)

}

for (int j = ۰; j < sizeX*sizeY;
j++)

}

object->m_vertices-
>setLocalPos(j, mapposition[.][j], mapposition[\][j],
;mapposition[۲][j])

```



```

{

;updateRender = false

;(· ,· ,·)cVectorrd force·
;hapticDevice->setForce(force·)

;(· ,· ,·)forceField.set
;(· ,· ,·)forceFieldfriction.set

{

{

if (posTool.z()>mapposition[۲][index])*/
}

;(· ,· ,·)cVectorrd force·
;hapticDevice->setForce(force·)

;(· ,· ,·)forceField.set
;(· ,· ,·)forceFieldfriction.set

/*{

if (((posTool.x() > ۱ && posTool.x() < ۱,۲) || */
(posTool.x() < -۱ && posTool.x() > -۱,۲) || (posTool.y() > ۱ &&
posTool.y() < ۱,۲) || (posTool.y() < -۱ && posTool.y() > -۱,۲))&&
posTool.z()<۰)

}

;cVectorrd desiredPosition

;(· ,· ,·)cVectorrd forcemetal

;cVectorrd forcem

desiredPosition.set(posTool.x(),

```

```

; (postTool.y(), 0); // mapposition[۱][index], mapposition[۲][index]

double Kp = ۳۰; // [N/m]

compute linear force //
;forcemetal = Kp * (desiredPosition - postTool)
;forcem.add(forcemetal)
;hapticDevice->setForce(forcem)

/*{
{

-----obtain data//

;()float time = timer->getCurrentTimeSeconds
cout << time << endl; // int(fmod(time * ۱۰۰۰, ۲۰)) //
;<< endl

if (islogging && (int)((time - prev_time) * ۱۰۰۰) >=
(۲۰) // && time != ۰
}

;prev_time = time
;cout << time << endl//

force//

;forceVector.push_back(static_cast<double>(forceField.z()))

friction//

forcefrictionX.push_back(static_cast<double>(forceFieldfriction.x(
;)))

```

```
forcefrictionY.push_back(static_cast<double>(forceFieldfriction.y(
;)))

read timer//
;timeVector.push_back(time)

read position//

;PositionX_vec.push_back(static_cast<double>(posTool.x()))

;PositionY_vec.push_back(static_cast<double>(posTool.y()))

;PositionZ_vec.push_back(static_cast<double>(posTool.z()))

read Angle//
cMatrix<rd> AngleTool = tool-
;()>getDeviceGlobalRot
;()cVector<rd> Angle = AngleTool.getCol.

;AngleZ_vec.push_back(static_cast<double>(Angle.z()))

;AngleY_vec.push_back(static_cast<double>(Angle.y()))

{
store tool position //
;prevToolLocalPos = toolLocalPos
;prevToolGlobalPos = toolGlobalPos
;prevposTool = posTool
```

```
update frequency counter //
;(\)freqCounterHaptics.signal
{

exit haptics thread //
;simulationFinished = true
{

----- //
-----

()int loadHeightMap
}
create an image //
;CImage image

load a file //
bool fileload =
image.loadFromFile(RESOURCE_PATH("../resources/images/b_surg۲.jpg"
;))
if (!fileload)
}
if defined(_MSVC)#
fileload =
;image.loadFromFile("../.../bin/resources/images/map.jpg")
endif#
{
if (!fileload)
}
```

```
cout << "Error - Texture image failed to load
;correctly." << endl

;()close

;()\- return

{

;CImage segmented

segmented.loadFromFile(RESOURCE_PATH("../resources/images/segmente
;d.png"))

check size of image //
if ((sizeX < 1) || (sizeY < 1)) { return (false); }

we look for the largest side //
;int largestSide = cMax(sizeX, sizeY)

scale the image to fit the world //
;double scale = 1,0 / (double)largestSide//

;double scalex = 1,0 / ((double)sizeX)
;double scaley = 1,0 / ((double)sizeY)

we will create an triangle based object. For centering //
puposes we
<sizeY; y++)
}

for (x = 0; x<sizeX; x++)
}

;()index = object->newVertex
```

```

get color of image pixel //
;color          cColorb
;image.getPixelColor(x, y, color)

compute vertex height by averaging the color //
.components RGB and scaling the value

;double vessel_h = .
;colorseg        cColorb
;segmented.getPixelColor(x, y, colorseg)
if ((double)colorseg.getLuminance() != .)
}
vessel_h =
;().,.....*(double)colorseg.getLuminance
{

;double px = scalex * (double)x - offsetX
;double py = -scaley * (double)y + offsetY

set vertex position //
object->m_vertices->setLocalPos(index, px, -py,
;-.,r + vessel_h)

if (index <= ۲۰)*//
}
;cout << "x: " << px << endl
;cout << "y: " << -py << endl
/*{
set vertex color//
;object->m_vertices->setColor(index, color)

```

```

-----save colors//

vertexcolorArray[0][index] =
;() (float)color.getR

vertexcolorArray[1][index] =
;() (float)color.getG

vertexcolorArray[2][index] =
;() (float)color.getB

;float max_color = vertexcolorArray[0][index]

if (vertexcolorArray[1][index]>max_color)
;max_color = vertexcolorArray[1][index]

if (vertexcolorArray[2][index]>max_color)
;max_color = vertexcolorArray[2][index]

;array_color_coef[index] = ۲۵۵ / max_color

set_array_color_coef[index] =
;array_color_coef[index]

; ++index

{

{

save initial color coef//

;string co = "InitialcolorCoef.txt

ofstream file\\("D:\\term^\\final project\\Data\\" +
;CaseName + "\\\" + co)

for (int i = ۰; i < sizeX*sizeY; i++)

}

;file\\ << set_array_color_coef[i] << "\\t

{

```

```
;()file\close

;cout << maxp<<endl//

Create a triangle based map using the above pixels //
for (x = 0; x<(sizeX - 1); x++)
{
for (y = 0; y<(sizeY - 1); y++)
{

get the indexing numbers of the next four //
vertices

unsigned int index00 = ((y + 0) * sizeX) + (x +
;0)

unsigned int index01 = ((y + 0) * sizeX) + (x +
;1)

unsigned int index10 = ((y + 1) * sizeX) + (x +
;0)

unsigned int index11 = ((y + 1) * sizeX) + (x +
;1)

create two new triangles //
;object->newTriangle(index00, index01, index10)
;object->newTriangle(index10, index01, index11)
{
{

we indicate that we were rendering triangles by using //
```



```
specific colors for each of them (see above)

;object->setUseVertexColors(true)

compute normals //

;()object->computeAllNormals


compute boundary box //

;object->computeBoundingBox(true)

;()cVector3d min = object->getBoundaryMin
;()cVector3d max = object->getBoundaryMax


compute size of object (largest side) //
;cVector3d span = cSub(max, min)
;double size = cMax(span.x(), cMax(span.y(), span.z()))


scale object //
;const double DESIRED_MESH_SIZE = ۲,۰
;double scaleFactor = DESIRED_MESH_SIZE / size
;object->scale(scaleFactor)


compute boundary box again //
;object->computeBoundingBox(true)


create collision detector for haptics interaction //
;object->createAABBCollisionDetector(۰,۰ * hapticRadius)


-----//
```

```

for (int i = 0; i < sizeX*sizeY; i++)
{
    cVector<rd> posVertex = object->m_vertices-
    ;>getLocalPos(i)

    ;()mapposition[0][i] = posVertex.x
    ;()mapposition[1][i] = posVertex.y
    ;()mapposition[2][i] = posVertex.z

    if (i <= ۲۳۹)*/
    {
        ;cout << "x: " << posVertex.x() << endl
        ;cout << "y: " << posVertex.y() << endl

        /*{

        if ((mapposition[0][i]) < maxpx) maxpx =
        ;()posVertex.x

        if (abs(mapposition[0][i]) > maxpy) maxpy =
        ;()posVertex.y

        {
            ;cout << maxpx << endl//
            ;cout << maxpy << endl//

            success //

            ;() return

            {

            void reinitialize(void)

            }

            for (int i = 0; i < sizeX*sizeY; i++)

            }

```

```

;array_color_coef[i] = set_array_color_coef[i]

;CColorb color
color.set(vertexcolorArray[0][i],
;vertexcolorArray[1][i], vertexcolorArray[2][i])

set vertex color//
;object->m_vertices->setColor(i, color)
{
cleaning vectors//
;()forceVector.clear
;()timeVector.clear
;()PositionX_vec.clear
;()PositionY_vec.clear
;()PositionZ_vec.clear
;()forcefrictionX.clear
;()forcefrictionY.clear

;flagcomputeN = true
;flagMarkForUpdate = true
{

void dataSaving(void)
}

;cout << "> Saving Data ... \r
;string f = "force" + to_string(test_number - 1) + ".txt
ofstream file("D:\\term1\\final project\\Data\\" + CaseName
;+ "\\\" + f)

;("file.open(file);// "force.txt//

```

```
for (int i = 0; i < forceVector.size(); i++)
{
; "file << forceVector[i] << "\t
{
; ()file.close

; "string t = "time" + to_string(test_number - 1) + ".txt
ofstream file2("D:\\term\\final project\\Data\\" + CaseName
;+ "\\\" + t)
;file2.open(t)//
for (int j = 0; j < timeVector.size(); j++)
{
; "file2 << timeVector[j] << "\t
{
; ()file2.close

string x = "positionX" + to_string(test_number - 1) +
; ".txt
ofstream file3("D:\\term\\final project\\Data\\" + CaseName
;+ "\\\" + x)
;file3.open(x)//
for (int i = 0; i < PositionX_vec.size(); i++)
{
; "file3 << PositionX_vec[i] << "\t
{
; ()file3.close
```

```
string y = "positionY" + to_string(test_number - 1) +  
;"".txt  
  
ofstream file{"D:\\term\\final project\\Data\\" + CaseName  
;+ "\\" + y}  
  
;file.open(y)//  
  
for (int i = 0; i < forceVector.size(); i++)  
{  
;file << PositionY_vec[i] << "\t"  
  
}  
  
;()file.close  
  
  
  
string z = "positionZ" + to_string(test_number - 1) +  
;"".txt  
  
ofstream file{"D:\\term\\final project\\Data\\" + CaseName  
;+ "\\" + z}  
  
;file.open(z)//  
  
for (int i = 0; i < forceVector.size(); i++)  
{  
;file << PositionZ_vec[i] << "\t"  
  
}  
  
;()file.close  
  
  
  
string co = "colorCoef" + to_string(test_number - 1) +  
;"".txt  
  
ofstream file{"D:\\term\\final project\\Data\\" + CaseName  
;+ "\\" + co}  
  
for (int i = 0; i < sizeX*sizeY; i++)  
  
}
```

```

; "file\ << array_color_coef[i] << "\t
{
; ()file\ .close

string frx = "forcefrictionX" + to_string(test_number - \) +
; ".txt

ofstream filev("D:\\term\\final project\\Data\\" + CaseName
;+ "\\\" + frx)

for (int i = 0; i < forcefrictionX.size(); i++)
{
; "filev << forcefrictionX[i] << "\t
{
; ()filev .close

string fry = "forcefrictionY" + to_string(test_number - \) +
; ".txt

ofstream file^("D:\\term\\final project\\Data\\" + CaseName
;+ "\\\" + fry)

for (int i = 0; i < forcefrictionX.size(); i++)
{
; "file^ << forcefrictionY[i] << "\t
{
; ()file^ .close

; "string az = "AngleZ" + to_string(test_number - \) + ".txt
ofstream file^("D:\\term\\final project\\Data\\" + CaseName
;+ "\\\" + az)

for (int i = 0; i < AngleZ_vec.size(); i++)
{
; "file^ << AngleZ_vec[i] << "\t

```

```

{
;()file\close

;string ay = "AngleY" + to_string(test_number - \) + ".txt
ofstream file\("D:\\term\\final project\\Data\\" +
;CaseName + "\\\" + ay)

for (int i = 0; i < AngleY_vec.size(); i++)
}

;file\ << AngleY_vec[i] << "\t
{
;()file\close

;cout << "> Data Saved... \r" << endl
}

```

پ- ۶ کد متلب مربوط به قطعه بندی تصویر و جداسازی رگ ها

```

clc;clear;close all
pic=imread('capture.jpg');
resiz_pic=imresize(pic,[۱۲۰ ۱۲۰]);
resiz_pic=imrotate(resiz_pic,۹۰);
% imwrite(resiz_pic,'b_surg.png');
% gray_pic=rgb2gray(resiz_pic);
gray_pic=resiz_pic(:,:,۲);
se = strel('disk',۱۵);
tophat=imtophat(gray_pic,se);
% figure;imshow(tophat)
bin_pic=imcomplement(im2bw(tophat,0.05));
% figure;imshow(bin_pic)
se2=strel('disk',۲);
dilated=imdilate(bin_pic,se2);

```

```

% figure;imshow(dilated)
segmented=bwlabel(dilated);
figure;imshow(segmented,[])

[m n]=size(segmented);
vessel=zeros(m,n);
for i=1:m
    for j=1:n
        if(segmented(i,j)==1 || segmented(i,j)==8 ||
segmented(i,j)==6 || segmented(i,j)==2 || ...
        segmented(i,j)==10 || segmented(i,j)==16 ||
segmented(i,j)==18) %2 6 8 10 16 18
            vessel(i,j)=1;
        end
    end
end
figure;imshow(vessel);
h = fspecial('average', 5);
vessel1=filter2(h, vessel);
vessel2=zeros(m,n,3);
vessel2(:,:,1)=vessel1;
vessel2(:,:,2)=vessel1;
vessel2(:,:,3)=vessel1;
figure;imshow(vessel2);
imwrite(vessel2,'segmented.png');

```

پ- ۷ کد متلب استخراج معیارها برای مقایسه

```

clc;clear;close all;
close all
gamers={'yasaman_safaiyan'
,'mohammadAli_dordi','amirhossein_rahmati'...
,'maryam_shakori','ali_towhidi','pardis_khalili','armin_frozanfar'...

```



```

        , 'fateme_bagheri', 'mojtaba_kazemi', 'nima_zakeri'};
ordinary={'hamzeh_abedi', 'asef', 'delaram_sadatamin'...
        , 'reihaneh_Daneshmand', 'kianosh_azizi', 'amin_khayat'...
        , 'amirhossein_nezamdest', 'ali_azizi', 'abolfazl', 'mamd_firouz'};

% i=۱۰;

im=rgb2gray(imread('exactseg.png'));
imref=im2bw(imresize(im, [۱۲۰ ۱۲۰]), ۰, ۰);
se1=strel('disk', ۰);
forceReference=imdilate(imref, se1);
se2=strel('disk', ۲);
cauteryReference=imerode(imref, se2);
% figure; imshow(forceReference);
load('../dataSet\InitilacolorCoef.mat');
person=۱;
for person=۱:۱۰
    %      clearvars -except gamerAnalyzed ordinalAnalyzed person
gamers ordinal forceReference cauteryReference InitilacolorCoef initialco

    cumoElectroCautery=۰;
    cumoForceMap=۰;
    forcemap=zeros(۱۲۰, ۱۲۰, ۱۰);
    forceError=zeros(۱, ۱۰);
    timer=۰;
    forcebar=۰;

    filename=char(ordinary(person));
    load(['../dataSet\' filename '.mat']);

    mkdir(['../DataAnalyzed\' filename]);
for i=۱:۱۰

```

```

%% read data from struct
AngleY=data(i).AngleY;
AngleZ=data(i).AngleZ;
colorCoef=data(i).colorCoef;
force=data(i).force;
forcefrictionX=data(i).forcefrictionX;
forcefrictionY=data(i).forcefrictionY;
positionX=data(i).positionX;
positionY=data(i).positionY;
positionZ=data(i).positionZ;
time=data(i).time;

%% plot trajectory
% close all
% figure;plot3(positionX,positionY,positionZ,'LineWidth',\,0);
% view(79,60);%zlim([-\ \]);xlim([-\ \,2]);ylim([-\ \,8]);
% grid on;xlabel('X');ylabel('Y');zlabel('Z');
% title(['3D trajectory of ' filename ' ,test number '
num\str(i)])
%
%
% saveas(gcf,['..\DataAnalyzed\' filename '\3D_trajectory'
num\str(i) '.png'])
%
% figure;subplot(\,2,\);plot(positionX,positionY)
% grid on;xlabel('X');ylabel('Y');
% title(['XY trajectory of ' filename ' ,test number '
num\str(i)])
%
% subplot(\,2,2);plot(positionY,positionZ)
% grid on;xlabel('Y');ylabel('Z');
% title(['YZ trajectory of ' filename ' ,test number '
num\str(i)])
% saveas(gcf,['..\DataAnalyzed\' filename '\3D_trajectory'

```

```

num=str(i) '.png'])

%% Test time
timer(i)=time(end);

%% path length

pathL=0;
for j=2:numel(time)
    dx=positionX(j)-positionX(j-1);
    dy=positionY(j)-positionY(j-1);
    dz=positionZ(j)-positionZ(j-1);
    pathL=pathL+sqrt(dx^2+dy^2+dz^2);
end

pathLength(i)=pathL*2; %convert to cm

%% force map
% hold on
for j=1:numel(positionX)
    if (positionX(j)<1 && positionX(j)>-1 && positionY(j)<1 &&
positionY(j)>-1)
        row=round((119/2)*(positionX(j)+1)+1);
        column=round((119/2)*(positionY(j)+1)+1);
        if column>120
            column=120;
        end
        if(force(j)<1,0)
            forcemap(row,column,i)=force(j);
        end
    end
end

forcemap(1:end,10:end,i)=0;
cumoForceMap=forcemap(:, :, i)+cumoForceMap;

```

```

%force intensity percent-----
resul=0,05;
maxForce=1,0;%max(max(forcemap(:,:,i)));
force_num=zeros(2,ceil(maxForce/resul));
[rr cc]=size(force_num);
for l=1:cc
    force_num(1,l)=l*resul;
end

counter=0;
for l=1:120
    for w=1:120
        if(forcemap(l,w,i) ~= 0)
force_num(2,ceil(forcemap(l,w,i)/resul))=force_num(2,ceil(forcemap(l,w,i)
)/resul))+1;
            counter=counter+1;
        end
    end
end
force_num(2,:)=(force_num(2,:)/sum(force_num(2,:)))*100;
%mean-----
summ=0;
temp=0;
for l=1:120
    for w=1:120
        summ=summ+forcemap(l,w,i);
        if (forcemap(l,w,i) ~=0)
            temp=temp+1;
        end
    end
end

```

```

        end

    end

    force_mean(i)=summ/temp;

    %var-----

    summ=0;
    for l=1:120
        for w=1:120
            if(forcemap(l,w,i) ~=0)
                summ=summ+(forcemap(l,w,i)-force_mean(i))^2;
            end
        end
    end

    force_stdev(i)=sqrt(summ/(sum(force_num(2,:))-1));

    %sum of applied forces-----
    SumAppliedForceET(i)=sum(sum(forcemap(:,:,i)));

    %Force Error for each test-----
    forceErr=0;
    for p=1:length(cumoForceMap)
        for q=1:length(cumoForceMap)
            if(forceRefrence(p,q)==0)
                forceErr=forceErr+forcemap(p,q,i);
            end
        end
    end

    forceError_ETper(i)=(forceErr/sum(sum(forcemap(:,:,i))))*100;

    %% electrocautery
    electroca(:,:,i)=initialco-colorCoef;

```

```

electroca(1:7,5:21,i)=0;
electroca(115:120,27:52,i)=0;
% figure;imshow(electroca(:,:,i),[])

%error of cauterizing brain tissue-----

cauteryErr=0;
temp=0;
for p=1:length(forceRefrence)
    for q=1:length(forceRefrence)
        if (forceRefrence(p,q)==0 )
            if (electroca(p,q,i) ~=0)
                cauteryErr=cauteryErr+1;
            end
            temp=temp+1;
        end
    end
end
BrainCauteryError_ETper(i)=(cauteryErr/temp)*100;

%error of cauterizing vessel-----
-----

cauteryErr=0;
temp=0;
for p=1:120
    for q=1:120
        if (cauteryRefrence(p,q)~=0 )
            if (electroca(p,q,i) ==0)
                cauteryErr=cauteryErr+1;
            end
            temp=temp+1;
        end
    end
end
end
end

```

```

VesselNotCaut_ETper(i)=(cauteryErr/temp)*۱۰۰

%mean of cautery-----

summ=۰;
temp=۰;
for p=۱:۱۲۰
    for q=۱:۱۲۰
        if (electroca(p,q,i) ~=۰)
            summ=summ+electroca(p,q,i);
            temp=temp+۱;
        end
    end
end
electroCautery_Mean(i)=summ/temp;

%standard Devision cautery-----

summ=۰;
temp=۰;
for p=۱:۱۲۰
    for q=۱:۱۲۰
        if (electroca(p,q,i) ~=۰)
            summ=summ+(electroCautery_Mean(i)-electroca(p,q,i))^۲;
            temp=temp+۱;
        end
    end
end
electroCautery_stDev(i)=sqrt(summ/(temp-۱));
%-----
cumoElectroCautery=electroca(:, :, i)+cumoElectroCautery;

end

%% saving data

```

```

analyzed(person).Time=timer;
analyzed(person).path_Length=pathLength;
analyzed(person).cumoForce_Map=cumoForceMap;
analyzed(person).Force_intensity=force_num;
analyzed(person).Force_mean=force_mean;
analyzed(person).Force_stdev=force_stdev;
analyzed(person).SumAppliedForce_ET=SumAppliedForceET;
analyzed(person).ForceError_ETper=forceError_ETper;
analyzed(person).BrainCauteryError_ET_per=BrainCauteryError_ETper;
analyzed(person).VesselNotCaut_ET_per=VesselNotCaut_ETper;
analyzed(person).ElectroCautery_Mean=electroCautery_Mean;
analyzed(person).ElectroCautery_stDev=electroCautery_stDev;
analyzed(person).CumoElectroCautery=cumoElectroCautery;
end
for tem=1:10
    if(strcmp(filename,gamers(tem)))
        save(['..\DataAnalyzed\gamerAnalyzed.mat'],'analyzed');
    end
    if(strcmp(filename,ordinary(tem)))
        save(['..\DataAnalyzed\ordinaryAnalyzed.mat'],'analyzed')
    ;
    end
end
%      title('Time      for      each      test');xlabel('Test
Number');ylabel('time(s)');
%
% imshow(cumoElectroCautery)
% ratio=sum(force);
% force_per=sort(force/ratio)*100;
% figure;plot(force_num(2,:))

```



## پ- ۸ کد متلب مقایسه ی معیارها برای دو گروه

```

clc;clear;close all;

g=load('..\DataAnalyzed\gamerAnalyzed.mat');
o=load('..\DataAnalyzed\ordinaryAnalyzed.mat');

gamerAnalyzed=g.analyzed;
ordinaryAnalyzed=o.analyzed;

%% path length
meanPathLengthG=0;meanPathLengthO=0;
PathLengthG=0;PathLengthO=0;
for test=1:10
    for person=1:10
        pathLengthG(person,test)=gamerAnalyzed(person).path_Length(test);
        pathLengthO(person,test)=ordinaryAnalyzed(person).path_Length(test);
    };
end
meanPathLengthG(test)=mean(pathLengthG(:,test));
meanPathLengthO(test)=mean(pathLengthO(:,test));
varpathLG(test)=sqrt(var(pathLengthG(:,test)));
varpathLO(test)=sqrt(var(pathLengthO(:,test)));
end

figure;subplot(1,2,1);errorbar(meanPathLengthG,varpathLG,'sb','linewidth',2);hold
on;errorbar(meanPathLengthO,varpathLO,'sr','linewidth',2);
    legend('Gamer','Non Gamer');xlim([0 10]);
    xlabel('Test Number');ylabel('Mean & S.D. of Path Length')

subplot(1,2,2);plot(meanPathLengthG,'sb-','LineWidth',2);hold
on;plot(meanPathLengthO,'sr-','LineWidth',2);hold on
    legend('Gamer','Non Gamer');
    xlabel('Test Number');ylabel('Mean of Path Length')
    xlim([0 10]);

x=1:10;

```

```

P = polyfit(x,meanPathLengthG,1);
yfit = P(1)*x+P(2);
plot(x,yfit,'b-.');

x=1:10;
P = polyfit(x,meanPathLengthO,1);
yfit = P(1)*x+P(2);
plot(x,yfit,'r-.');

[~,p] = ttest(pathLengthG(:),pathLengthO(:));
disp('Path Length P-value:');disp(p);

%% time

meanTimeG=0;meanTimeO=0;
TimeG=0;TimeO=0;
for test=1:10
    for person=1:10
        TimeG(person,test)=gamerAnalyzed(person).Time(test);
        TimeO(person,test)=ordinaryAnalyzed(person).Time(test);
    end
    meanTimeG(test)=mean(TimeG(:,test));
    meanTimeO(test)=mean(TimeO(:,test));
    varTimeG(test)=sqrt(var(TimeG(:,test)));
    varTimeO(test)=sqrt(var(TimeO(:,test)));
end

%           figure;plot(meanTimeG,'LineWidth',2);hold on;plot(
meanTimeO,'r','LineWidth',2);

figure;subplot(1,2,1);errorbar(meanTimeG,varTimeG,'sb','linewidth'
,2);hold on;errorbar(meanTimeO,varTimeO,'sr','linewidth',2);
legend('Gamer','Non Gamer');xlim([0 10]);
xlabel('Test Number');ylabel('Mean & S.D. of Time')

subplot(1,2,2);plot(meanTimeG,'sb-','LineWidth',2);hold on;plot(

```

```

meanTimeO,'sr-','LineWidth',۲);hold on

legend('Gamer','Non Gamer');

xlabel('Test Number');ylabel('Mean of Time')

xlim([۰ ۱۱]);

x=۱:۱۰;

P = polyfit(x,meanTimeG,۱);

yfit = P(۱)*x+P(۲);

plot(x,yfit,'b-.');

x=۱:۱۰;

P = polyfit(x,meanTimeO,۱);

yfit = P(۱)*x+P(۲);

plot(x,yfit,'r-.');

[~,p] = ttest(TimeG(:),TimeO(:));

disp('Time P-value:');disp(p);

%% Force Mean Applied

meanForceMeanG=۰;meanForceMeanO=۰;

ForceMeanG=۰;ForceMeanO=۰;

for test=۱:۱۰

for person=۱:۱۰

ForceMeanG(person,test)=gamerAnalyzed(person).Force_mean(test);

ForceMeanO(person,test)=ordinaryAnalyzed(person).Force_mean(test);

end

meanForceMeanG(test)=mean(ForceMeanG(:,test));

meanForceMeanO(test)=mean(ForceMeanO(:,test));

varFMG(test)=sqrt(var(ForceMeanG(:,test)));

varFMO(test)=sqrt(var(ForceMeanO(:,test)));

end

% figure;plot(meanForceMeanG,'LineWidth',۲);hold on;plot(
meanForceMeanO,'r','LineWidth',۲);

figure;subplot(۱,۲,۱);errorbar(meanForceMeanG,varFMG,'sb','linewidth

```

```

th',2);hold on;errorbar(meanForceMeanO,varFMO,'sr','linewidth',2);

    legend('Gamer','Non Gamer');xlim([0 11]);

    xlabel('Test Number');ylabel('Mean & S.D. of Force')

    subplot(1,2,2);plot(meanForceMeanG,'sb-','LineWidth',2);hold
on;plot(meanForceMeanO,'sr-','LineWidth',2);hold on

    legend('Gamer','Non Gamer');

    xlabel('Test Number');ylabel('Mean of Force')

    xlim([0 11]);

    x=1:10;

    P = polyfit(x,meanForceMeanG,1);

    yfit = P(1)*x+P(2);

    plot(x,yfit,'b-.');

    x=1:10;

    P = polyfit(x,meanForceMeanO,1);

    yfit = P(1)*x+P(2);

    plot(x,yfit,'r-.');

    [~,p] = ttest(ForceMeanG(:),ForceMeanO(:));

    disp('Force Mean P-value:');disp(p);

    %% Force Standard Deviation
    meanForceSDG=0;meanForceSDO=0;

    ForceSDG=0;ForceSDO=0;

    for test=1:10

        for person=1:10

            ForceSDG(person)=gamerAnalyzed(person).Force_stdev(test);

            ForceSDO(person)=ordinaryAnalyzed(person).Force_stdev(test);

        end

        meanForceSDG(test)=mean(ForceSDG);

        meanForceSDO(test)=mean(ForceSDO);

    end

```

```

figure;plot(meanForceSDG,'sb-','LineWidth',۲);hold on;plot(
meanForceSDO,'sr-','LineWidth',۲);legend('Gamer','Non Gamer');

xlabel('Test Number');ylabel('Force Standard Deviation')

x=۱:۱۰;

P = polyfit(x,meanForceSDG,۱);
yfit = P(۱)*x+P(۲);
plot(x,yfit,'b-.');

x=۱:۱۰;

P = polyfit(x,meanForceSDO,۱);
yfit = P(۱)*x+P(۲);
plot(x,yfit,'r-.');

[~,p] = ttest(meanForceSDG,meanForceSDO);%%
disp('Force SD P-value:');disp(p);

%% Force Error Percent

meanForceErrorG=0;meanForceErrorO=0;
ForceErrorG=0;ForceErrorO=0;
for test=۱:۱۰
for person=۱:۱۰
ForceErrorG(person,test)=gamerAnalyzed(person).ForceError_ETper(
test);
ForceErrorO(person,test)=ordinaryAnalyzed(person).ForceError_ETper
(test);
end
meanForceErrorG(test)=mean(ForceErrorG(:,test));
meanForceErrorO(test)=mean(ForceErrorO(:,test));
varFEPG(test)=sqrt(var(ForceErrorG(:,test)));
varFEPO(test)=sqrt(var(ForceErrorO(:,test)));
end

% figure;plot(meanForceErrorG,'LineWidth',۲);hold on;plot(
meanForceErrorO,'r','LineWidth',۲);

figure;subplot(۱,۲,۱);errorbar(meanForceErrorG,varFEPG,'sb','linew

```

```

idth',۲);hold on;errorbar(meanForceErrorO,varFEPO,'sr','linewidth',۲);
    legend('Gamer','Non Gamer');xlim([۰ ۱۱]);
    xlabel('Test Number');ylabel('Mean & S.D. of Force Error (%)')

    subplot(۱,۲,۲);plot(meanForceErrorG,'sb-','LineWidth',۲);hold
on;plot(meanForceErrorO,'sr-','LineWidth',۲);hold on
    legend('Gamer','Non Gamer');
    xlabel('Test Number');ylabel('Mean of Force Error (%)')
    xlim([۰ ۱۱]);

    x=۱:۱۰;
    P = polyfit(x,meanForceErrorO,۱);
    yfit = P(۱)*x+P(۲);
    plot(x,yfit,'b-.');

    x=۱:۱۰;
    P = polyfit(x,meanForceErrorG,۱);
    yfit = P(۱)*x+P(۲);
    plot(x,yfit,'r-.');

    [~,p] = ttest(ForceErrorG(:),ForceErrorO(:));
    disp('Force Error P-value:');disp(p);
    %% Brain Cautery Error

    meanBrainCauteryErrorG=۰;meanBrainCauteryErrorO=۰;
    BrainCauteryErrorG=۰;BrainCauteryErrorO=۰;
    for test=۱:۱۰
        for person=۱:۱۰
            BrainCauteryErrorG(person,test)=gamerAnalyzed(person).BrainCautery
Error_ET_per(test);
            BrainCauteryErrorO(person,test)=ordinaryAnalyzed(person).BrainCaut
eryError_ET_per(test);
        end
        meanBrainCauteryErrorG(test)=mean(BrainCauteryErrorG(:,test));

```

```

meanBrainCauteryErrorO(test)=mean(BrainCauteryErrorO(:,test));
varMBCG(test)=sqrt(var(BrainCauteryErrorG(:,test)));
varMBCO(test)=sqrt(var(BrainCauteryErrorO(:,test)));

end

% figure;plot(meanBrainCauteryErrorG,'LineWidth',۲);hold on;plot(
meanBrainCauteryErrorO,'r','LineWidth',۲);

figure;subplot(۱,۲,۱);errorbar(meanBrainCauteryErrorG,varMBCG,'sb'
,'linewidth',۲);hold
on;errorbar(meanBrainCauteryErrorO,varMBCO,'sr','linewidth',۲);

legend('Gamer','Non Gamer');xlim([۰ ۱۱]);

xlabel('Test Number');ylabel('Mean & S.D. of Brain Cautery
Error(%)')

subplot(۱,۲,۲);plot(meanBrainCauteryErrorG,'sb-
','LineWidth',۲);hold on;plot(meanBrainCauteryErrorO,'sr-
','LineWidth',۲);hold on

legend('Gamer','Non Gamer');

xlabel('Test Number');ylabel('Mean of Brain Cautery Error(%)')
xlim([۰ ۱۱]);

x=۱:۱۰;

P = polyfit(x,meanBrainCauteryErrorG,۱);
yfit = P(۱)*x+P(۲);
plot(x,yfit,'b-.');

x=۱:۱۰;

P = polyfit(x,meanBrainCauteryErrorO,۱);
yfit = P(۱)*x+P(۲);
plot(x,yfit,'r-.');

[~,p] = ttest(BrainCauteryErrorG(:),BrainCauteryErrorO(:));
disp('Brain Cautery Error P-value:');disp(p);

%% vessel Not Caut

```

```

    meanVesselNotCautedG=0;meanVesselNotCautedO=0;

    VesselNotCautedG=0;VesselNotCautedO=0;

    for test=1:10
        for person=1:10
            VesselNotCautedG(person,test)=gamerAnalyzed(person).VesselNotCaut_
ET_per(test);
            VesselNotCautedO(person,test)=ordinaryAnalyzed(person).VesselNotCa
ut_ET_per(test);
        end
        meanVesselNotCautedG(test)=mean(VesselNotCautedG(:,test));
        meanVesselNotCautedO(test)=mean(VesselNotCautedO(:,test));
        varVNCG(test)=sqrt(var(VesselNotCautedG(:,test)));
        varVNCO(test)=sqrt(var(VesselNotCautedO(:,test)));
    end

    % figure;plot(meanVesselNotCautedG,'LineWidth',2);hold on;plot(
meanVesselNotCautedO,'r','LineWidth',2);

    figure;subplot(1,2,1);errorbar(meanVesselNotCautedG,varVNCG,'sb','
linewidth',2);hold
on;errorbar(meanVesselNotCautedO,varVNCO,'sr','linewidth',2);

    legend('Gamer','Non Gamer');xlim([0 11]);

    xlabel('Test Number');ylabel('Mean & S.D. of Vessel Not Caut.
(%)');

    subplot(1,2,2);plot(meanVesselNotCautedG,'sb-','LineWidth',2);hold
on;plot(meanVesselNotCautedO,'sr-','LineWidth',2);hold on

    legend('Gamer','Non Gamer');

    xlabel('Test Number');ylabel('Mean of Vessel Not Caut. (%)');

    xlim([0 11]);

    x=1:10;

    P = polyfit(x,meanVesselNotCautedG,1);

    yfit = P(1)*x+P(2);

    plot(x,yfit,'b-.');

```



```

x=1:10;
P = polyfit(x,meanVesselNotCautedO,1);
yfit = P(1)*x+P(2);
plot(x,yfit,'r-.');

[~,p] = ttest(VesselNotCautedG(:),VesselNotCautedO(:));
disp('Vessel NotC P-value:');disp(p);

%% Electro Caутery Mean
meanElectroCautMeanG=0;meanElectroCautMeanO=0;
ElectroCautMeanG=0;ElectroCautMeanO=0;
for test=1:10
for person=1:10
ElectroCautMeanG(person,test)=gamerAnalyzed(person).ElectroCaутery
_Mean(test);
ElectroCautMeanO(person,test)=ordinaryAnalyzed(person).ElectroCaut
ery_Mean(test);
end
meanElectroCautMeanG(test)=mean(ElectroCautMeanG(:,test));
meanElectroCautMeanO(test)=mean(ElectroCautMeanO(:,test));
varECMG(test)=sqrt(var(ElectroCautMeanG(:,test)));
varECMO(test)=sqrt(var(ElectroCautMeanO(:,test)));
end
% figure;plot(meanElectroCautMeanG,'LineWidth',2);hold on;plot(
meanElectroCautMeanO,'r','LineWidth',2);
figure;subplot(1,2,1);errorbar(meanElectroCautMeanG,varECMG,'sb','
linewidth',2);hold
on;errorbar(meanElectroCautMeanO,varECMO,'sr','linewidth',2);
legend('Gamer','Non Gamer');xlim([0 10]);
xlabel('Test Number');ylabel('Mean & S.D. of ElectroCaутery')

subplot(1,2,2);plot(meanElectroCautMeanG,'sb-','LineWidth',2);hold
on;plot( meanElectroCautMeanO,'sr-','LineWidth',2);hold on
legend('Gamer','Non Gamer');

```

```

xlabel('Test Number');ylabel('Mean of ElectroCautery')
xlim([0 10]);

x=1:10;
P = polyfit(x,meanElectroCautMeanG,1);
yfit = P(1)*x+P(2);
plot(x,yfit,'b-.');

x=1:10;
P = polyfit(x,meanElectroCautMeanO,1);
yfit = P(1)*x+P(2);
plot(x,yfit,'r-.');

[~,p] = ttest(ElectroCautMeanG(:),ElectroCautMeanO(:));%%
disp('Electrocautery Mean P-value:');disp(p);

%% Electro Cautery SD
meanElectroCautSDG=0;meanElectroCautSDO=0;
ElectroCautSDG=0;ElectroCautSDO=0;
for test=1:10
    for person=1:10
        ElectroCautSDG(person)=gamerAnalyzed(person).ElectroCautery_stdDev(test);
        ElectroCautSDO(person)=ordinaryAnalyzed(person).ElectroCautery_stdDev(test);
    end
    meanElectroCautSDG(test)=mean(ElectroCautSDG);
    meanElectroCautSDO(test)=mean(ElectroCautSDO);
end

figure;plot(meanElectroCautSDG,'sb-','LineWidth',2);hold on;plot(meanElectroCautSDO,'sr-','LineWidth',2);legend('Gamer','Non Gamer');
xlabel('Test Number');ylabel('Mean of ElectroCautery SD')

```

```
x=1:10;  
P = polyfit(x,meanElectroCautSDG,1);  
yfit = P(1)*x+P(2);  
plot(x,yfit,'b-.');  
  
x=1:10;  
P = polyfit(x,meanElectroCautSDO,1);  
yfit = P(1)*x+P(2);  
plot(x,yfit,'r-.');  
  
[~,p] = ttest(meanElectroCautSDG,meanElectroCautSDO);%%  
disp('Electrocautery SD P-value:');disp(p);
```

### **Abstract**

One of the most widely used tools in the training and evaluation of physicians and surgeons in recent years is surgical simulators. Introducing virtual reality talk and adding haptic technology to the field made it possible to develop even better surgical simulators. In this project, we have attempted to design a simulator of brain surgery using haptic technology and software development. Finally, an experiment was designed using the features provided in the simulator. Then, the experiment was performed on the volunteers and a statistical analysis was performed on the learning process of error reduction during surgery. Finally, by analyzing the obtained results, the positive effect of surgery simulator on enhancing user skills was investigated.

#### **Key Words:**

Virtual Reality, Haptic, Brain surgery simulator, CHAI<sup>2</sup>D, Bipolar Electrocautery, Learning process



**Amirkabir University of Technology  
(Tehran Polytechnic)**

**Biomedical Engineering**

**BSc**

**Simulation of the virtual reality for interaction of the instrument  
with the texture to studying  
process of learning and applying desired force**

**By  
Reza Karimzadeh**

**Supervisor  
Dr. Azarnoush**

**Oct ۲۰۱۹**