

基于虚拟现实的社交网络三维可视化系统

孙灿燃, 崔晨, 徐静乐, 骆岩林*

(北京师范大学信息科学与技术学院, 北京市 100875)

摘要: 随着信息爆炸式发展, 社交网络产生海量数据, 对这些数据三维可视化能够辅助人们从不同视角观察网络空间结构及聚类关系, 进行数据探索, 挖掘其潜在规律。针对社交网络三维可视化系统较少且体验效果不太理想的现状, 提出将 Gephi 二维布局图转换为三维并提高趣味性、交互性。采用双目立体视觉、虚拟现实技术及 Leap Motion, 得到一个帮助用户更加有趣直观理解社交网络的三维可视化系统。最后实验结论是具有良好的沉浸感和交互性, 且计算出的网络属性值准确。

关键词: 社交网络; 三维可视化; 双目立体视觉; 虚拟现实; Leap Motion

中国法分类号: TP391.9

文献标识码: A

Social Network 3D Visualization System Based on Virtual Reality

SUN Can-ran, CUI Chen, XU Jing-le, LUO Yan-lin*

(College of Information Science and Technology, Beijing Normal University, Beijing 100875, China)

ABSTRACT: Social network has generated massive data along with the explosive development of information. The three-dimensional visualization of these data can help researchers observe the spatial structure and the clustering relationship of network from different perspectives, explore the potential regulation as well. There are scarce three-dimension visualization systems for the social networks, so it is meaningful that to convert Gephi's two-dimension layout to a three-dimensional one and increase its interest and interactivity. Get a 3D visualization system by binocular stereo vision, virtual reality technology, and Leap Motion, which helps users understand social networks more interestingly and intuitively. The final experimental conclusion is that it has good immersion and interaction, the calculated network attribute values are accurate as well.

KEYWORDS: Social Network; three-dimension visualization layout; binocular stereo vision; virtual reality; Leap Motion interaction

1 引言

人与人之间的关系网共同组成了社交网络。借助于互联网平台, 社交网络的发展是跨越式的。据中国互联网络发布, 截至 2016 年, 中国网民达到了 7.1 亿次, 人均上网时长达到每天 3.7 小时[1]。社交网络中的数据呈爆炸式发展, 将抽象的源数据转化为人们容易理解且认知习惯的图像信息, 可大大降低因数据冗杂而带来的研究难度。因此, 社交网络可视化技术在近年来备受瞩目。

对于一些网络, 传统的二维可视化在展现性能上仍有不足, 其能够展示的维度十分有限。而将节点和边进行三维可视化, 使数据和关系以三维形式

“真实”地显示出来, 可辅助人们从不同视角观察网络空间结构及聚类关系等。

国内外二维社交网络可视化比较成熟, 较突出的有利用谱聚类大规模社交网络可视化方法对用户聚类、分布和关注倾向等特征进行形象化展示[2]。在国外, Nathalie Henry[3]等提出了基于“NodeTrix”混合可视化的方法, 用户能看到整体的可视化结果, 网络中节点和边的详细信息; 在国内, 基于新浪微博的社交网络数据分析与可视化系统[4], 基于 Gephi 的社交网络消息可视化分析系统[5]等应运而生。

本文设计开发将 Gephi 生成的二维社交网络拓展到三维空间, 提升立体性和直观性。采用双目立

体视觉、虚拟现实技术，佩戴立体 3D 眼镜观察立体渲染结果方法，提升体验者的直观感受和研究趣味性。采用 Leap Motion 手势交互，简化可视化模型操作，帮助用户理解节点和边的关系。

通过 Grapi 得到的网络特征属性个数有限且精度较小，而我们优化其计算方法，使网络特征值准确且更加精确。现有的社交网络可视化系统多为二维显示，我们对其进行改进，突破二维限制。此外采用 LeapMption 增强趣味性、交互性。

论文分为四部分，第一部分介绍系统概况，第二部分介绍系统实现的关键技术，第三部分进行实验与分析，第四部分给出结论。

1 系统介绍

1.1 开发环境

Web 应用框架是一种开发框架，用来支持动态网站、网络应用程序及网络服务的开发。对于 Web 应用来说，当客户端想要获取动态资源时，就会发起一个 HTTP 请求，Web 应用程序会在后台进行相应计算，取出对应的数据并生成 HTTP 响应。本系统采用的 Flask 是一个微型的 Python 开发的 Web 框架，该框架采用 Werkzeug 进行路由分发，使用 Jinja2 模板引擎通过 HTML Templating 动态生成 HTML。由于系统需要通用于各组不同的输入数据，所以在构建的时候所有的信息严格按照导入 Gephi 的原始数据进行演化和计算，使得代码与数据独立，具有较好的泛化能力。

系统使用 Three.js 实现社交网络的三维图形绘制。Three.js 是一个 3D Javascript 库，封装了底层图形接口，如虚拟照相、几何形状、材质、渲染器等。我们利用 three.js 中的 raycaster 类，设计用于获取社交网络模型中被焦点选中的节点，然后 Web 应用程序在后台进行相应计算，取出数据并生成 Http 响应，实时显示被选中节点的各种信息。

为加快三维可视化速度，采用 GPU 加速绘制。将三维可视化中计算部分转移到 GPU 进行，高度并行化地处理大规模数据。由于渲染过程中图像适配器的固定渲染管线被用来处理普遍所有渲染状态设置，因此它不适于执行更具体细致的要求。所以

我们使用自定义的颜色着色器代替传统固定功能渲染管线，以达到渲染过程中过滤掉绿色通道效果。此外，为进行直观三维交互，引入 LeapMotion 来运用手势进行直接控制，满足手势简易、操作直接、交互自然特点。

整体概括，我们的 3D 社交网络三维可视化系统的工作流程图如图 1 所示。

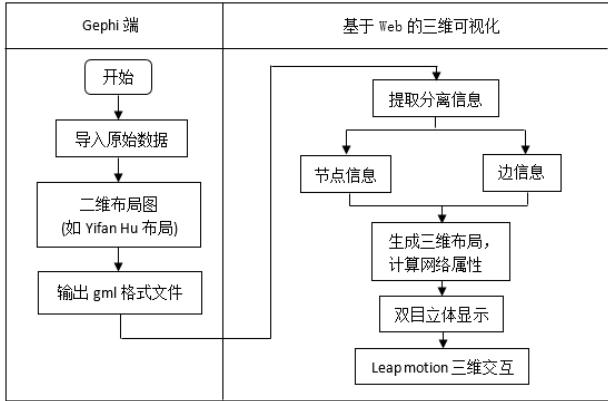


图 1 三维可视化系统工作流程

1.2 系统界面

系统界面如图 2 所示，左侧显示三维社交网络模型，右侧显示当前节点属性信息。当选择网络模型中的某一节点时，系统会实时计算并更新当前节点的各种属性信息，将其显示在界面右侧。佩戴立体 3D 眼镜观看立体渲染效果，更具有沉浸感；使用交互设备对模型进行平移、旋转等操作时，方便体验者对模型进行观察。

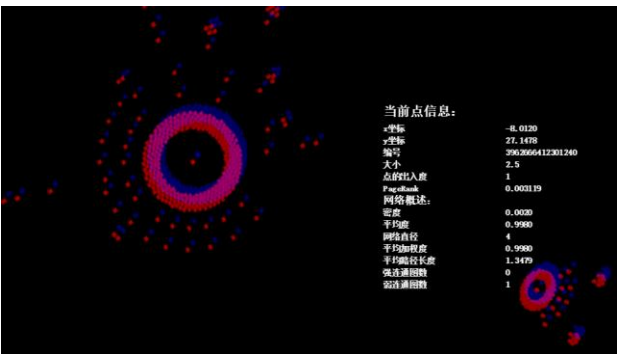


图 2 三维可视化系统的界面设计

2 关键技术

2.1 三维布局生成

起始节点和目标节点之间的连线表示节点与节点之间的连接关系。通过 Gephi 且经过 Yifan Hu

布局处理后的网络图如图 3 所示, 该图清晰展示了节点之间的连接关系。

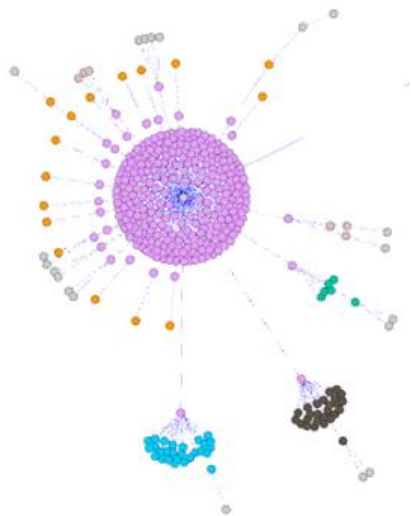


图 3 经过 Yifan Hu 布局处理后的网络图

通过 Gephi 生成的 gml 文件, 将其分离成 csv 格式的节点和边信息文件。Gephi 生成的可视化图是二维的, 节点信息文件中节点位置坐标中的 Z 为 0。根据这些信息, 调用 Three.Mesh 方法为每个位置的节点绘制网格, 渲染后使其显示在客户端。通过双目立体视觉实现, 增强三维立体感。

社交网络布局中节点之间联系复杂且其节点数繁多, 引入 Three.js 中的粒子系统 Points。进一步, 使用 GPU 加快绘制速度, 实现所有粒子(节点)的特征变化。为创建粒子系统, 采用 Three.js 中提供的 BufferGeometry 类和 BufferAttribute 创建几何对象 Geometry, 该类将所有数据包括顶点位置、法线、位置和其他自定义属性以数组形式存储在缓冲区中, 以此来减少数据传递到 GPU 的负荷。然后, 使用 three.js 中的 ShaderMaterial 方法来创建粒子材质。通过它, 使用自定义的着色器, 在 WebGL 环境中运行并显示。最后将有关联的节点通过绘制直线将其连接起来。

2.2 网络属性描述

系统对网络属性进行了描述, 包括密度、平均度、网络直径、平均加权度、平均路径、强联通图数、弱联通图数、网络节点 PageRank 值[6]。

呈现每个节点属性时, 我们引入 Three.js 提供的 Raycaster 类, 设计用于获取选中的节点, 然后

Web 应用程序在后台进行相应计算, 取出数据并生成 Http 响应。Raycaster 运用的原理与坐标系有关, 在进行投影时, 通过算法将模拟的三维空间内的物体映射到屏幕上生成二维图像。无论正交投影和透视投影, 投影可视空间都有近裁剪面和远裁剪面。因此可视的场景物体深度坐标必然在远近裁剪面之间。在 WebGL 坐标系中, 坐标被限制为(1,1,1)和(-1, -1, -1), 其中 z 轴为深度坐标。鼠标(或其他交互设备)在屏幕上进行点击时, 可得到二维坐标(X, Y), 再加上深度坐标的范围(-1, 1), 形成两个三维坐标 A(x1, y1, -1)和 B(x2, y, 1)。由于 Z 轴坐标不同, 转变到投影坐标系时, 分别是前剪切平面上的点和后剪切平面上的点, 因此 AB 连线穿过的物体就是被点击的物体。

利用 Javascript 获得鼠标(或其他交互设备)点击屏幕的坐标为 event.clientX 和 event.clientY。首先将其转变为标准化设备坐标: 将 X、Y 坐标映射到(-1,1)。接着, 利用 Raycaster 在坐标系中形成一条射线, 该射线的起点向量是照相机, 射线的方向向量从照相机出发到所经过的点。将射线与 WebGL 模型求交, 选取第一个相交的节点, 确定为点击的目标节点。从而实现实时显示特定节点的所属信息。

2.3 双目立体视觉呈现

双目立体视觉能直接模仿人类双眼处理客观场景, 使观察者获得明显的深度感, 使场景呈现更具立体感。相比较于其他类别的体视方法, 如全息照相、透镜板三维成像等, 双目立体视觉技术处理立体场景的方式更加可靠、简便。

将内容显示到 Web 端后, 为增强三维立体感, 需增加其双目立体视觉的效果, 采用 WebGL 中的后期处理通道和特效合成器技术。根据需要, 我们可以自定义若干个渲染通道, 每个通道可实现特定功能的特效, 如发光、褐色、旧电视机效果等。Three.js 是一个强大的资源库, 为我们提供了多个后期处理通道, 其本质都是从 OpenGL 发展而来, 可直接使用。ShaderPass 通道可传入一个自定义的着色器, 以生成一个满足用户特定需求的高级的后期处理通道。

特效合成器的本质相当于一个渲染器。在特效合成器中, 通过调用 addPass() 将多个着色器通道链

接在一起,以实现将所有叠加在当前 Web 端显示画面中的所有通道结果计算并渲染出来,从而得到理想的效果。在这一链接过程中,每个渲染通道都会对场景产生不同的渲染效果,且会影响到后续通道对图像的渲染,所以其调用顺序是重要的。

实现整体双目立体视觉,关键是设置后期处理和添加后期处理通道。首先,创建一个特效合成器对象,同时在这个对象的构造函数里出入 WebGL-Renderer,拟在该合成器中添加各种组合通道以合成渲染出理想的效果。第一个加入的通道是 RenderPass,这个通道会初步渲染场景,并保留最原始渲染的效果,但不会将渲染结果输出到屏幕。然后添加一个可以将结果输出到屏幕上的通道 copyPass,并将其 renderToScreen 属性设置为 True。接着将需要的效果通道添加到特效合成器中。根据特定需求,我们自定义了一个渲染通道,并且在该通道中传入自定义着色器,以生成高级的后期处理通道。我们选择了经修改后的 RGBShader 着色器,以实现纹理中将红色、蓝色分开。设置完所有的处理通道后,在每次循环渲染时分别调用特效合成器的 Render 函数进行重新渲染,如此体验者可通过佩戴红蓝眼镜观察 3D 效果良好的三维社交网络布局。

2.4 Leap Motion 三维交互

为结合虚拟现实,采用 Leap Motion 传感器。它可通过实时跟踪并采集人手势的运动信息与系统进行自然的人机交互,具有操作简单、方便快捷,并且直观的高效交互的特点。在传统的鼠标、键盘交互方式中,需要用户对系统有一定的了解,掌握基本操作技术。而 Leap Motion 运用的是直观的手势交互方式,通过设计合理简单的交互手势,便可以实现对系统的操纵,是一种人性化简易化的交互方式;同时 Leap Motion 具有识别效率、识别精度高,成本低廉特点,是良好的体交互实现途径。

2.4.1 手势设计

用 Leap Motion 来实现模型的 6 自由度运动。为触发不同的操作,设计了 3 个不同的操作步骤。交互式响应不断在屏幕上实时显示,为用户提供有效的视觉反馈。定义的手势如下:程序刚运行时,为防止误操作,使用手势 G1 开启交互操作;之后使用手势 G0 切换到停止状态;对模型的平移使用手

势 G2;使用手势 G3 对模型进行旋转。

G0 : 右手张开;

G1 : 当手指的顶端向下转向手掌,然后回弹到原来的位置,如同点击一样,按键点击手势识别

G2 : 右手单个手指,在三维空间移动,图 4(a);

G3 : 右手张开,以大拇指为轴旋转,图 4(b)-(d);

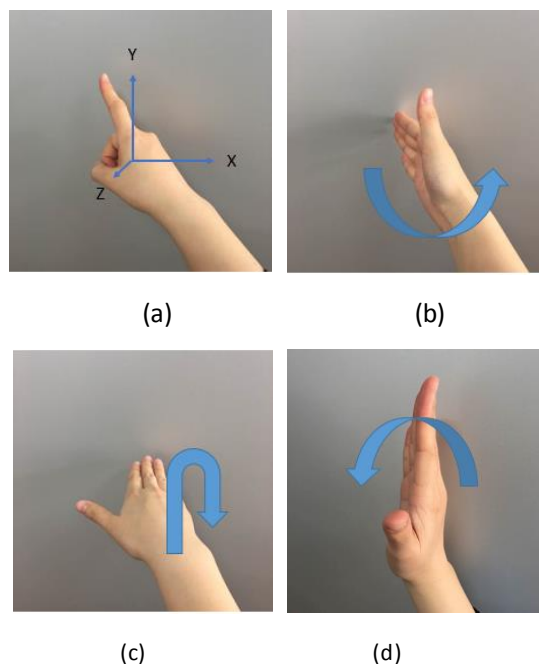


图 4 手势: a 单手指, b-d 张开

右手位于 leapMotion 正上方进行交互。右手使用图 4(a)手势沿 Z 轴移动,实现模型沿 Z 轴方向移动,呈现出缩放的交互效果;沿 X、Y 轴移动实现模型沿相同方向的平移效果。右手张开,沿图 4(b)方向旋转实现模型绕 Y 轴旋转;沿图 4(c)方向旋转实现模型绕 X 轴旋转;沿图 4(d)方向旋转实现模型绕 Z 轴旋转。如此,实现右手对模型六个自由度的平移、旋转操作。

2.4.2 具体实现

在 Web 应用程序中引入 Leap Motion,首先引入 leap.js,它是一个开源的 JavaScript API,为运行 Leap motion 做前提准备。由于 Leap Motion 控制器连续提供数据,因此需使用大跃进运动 API 提供的 Leap.loop 来建立事件循环以处理各个数据帧。接着,为实现跟踪访问数据,我们利用 Leap Motion 探测到的手的相关数据与 Js 中的变量进行交互。其中,对手势的识别和实现如下:

G0 的识别:在获取的帧中,检察手掌数和各手

掌的手指数，由于识别误差的存在，我们设定，当识别到一个手掌有 4 根以上的手指时，视为手掌张开，识别为 G0；

G1/G2/G3 的识别：在获取的帧中，检察手势和手指个数，当手势为 `Gesture::TYPE_KEY_TAP` (手指有点击动作) 时，判断手指数，为 1 时，识别为 G1；无点击动作则识别为 G2。当手指数大于 4，其手掌旋转识别为 G3；

G2/G3 功能的实现：首先，获取前一帧和当前帧，计算手掌的位移和前后手掌法向量与坐标轴的夹角差值，通过一定的比例计算模型的位移和在对应坐标轴的旋转角；然后，通过 `OpenGL` 函数计算出对应的模型视图矩阵，并将其逆矩阵 `invViewMatrix` 传入设备端；最后，根据逆矩阵，计算出视点，进行光线投射算法得出最终的渲染效果。由于 `OpenGL` 旋转操作的相关性，使得用户在进行多次旋转时，得到的是多次旋转的复合结果，而用户需要的是多次独立的旋转的结果。因此，引入四元数[7]来解决这个问题。

3 实验与分析

本文以新浪微博为平台，结合微博数据实例，对其可视化模型进行三维呈现和交互。选取“李开复”的单条微博

(<http://weibo.com/1197161814/DjYlqdeVn>)，其具体内容如图 5 所示。采取网络爬虫的方法获取转发过程中的各微博 MID，用户 UID，用户名称，转发内容等信息，并保存为 `csv` 文件。该文件中有 `Source`、`Target`、`Type`、`Id`、`Lable`、`Weight` 等信息，分别对应数据的起始节点，目标节点，边类型，编号，边信息，权重等信息，以便 `Gephi` 读取。其中，`Source` 为当前微博的直接上一级微博的 MID，`Target` 为当前微博的 MID，边的类型均为有向型。



图 5 李开复微博内容

在 `Gephi` 中以按边方式导入 `csv` 数据文件，生成带有节点和边数据的 `gml` 格式。接着将生成的 `gml` 文件分离成 `csv` 格式的节点信息文件和边信息

文件。其中节点信息文件包括节点的编号、X、Y、Z 坐标、权重值以及 `Gephi` 生成的 `fill` 编号；边信息包括边的编号，源节点编号，目标节点编号以及边的权重。根据得到的节点和边信息进行图计算，然后在 `HTML` 文件里进行绘制，最终在 `Chrome` 浏览器上得到图像的显示。

图 6 为志愿者佩戴红蓝立体 3D 眼镜，通过右手在三个维度进行移动以控制模型平移的场景。可清楚看出显示界面是一个由中心点向四周辐射形成类似漩涡的圆形，越靠近核心用户的地方点分布越密集。表明该用户发布微博后，有大量粉丝进行了先后转发，但传播范围主要集中在第一层转发。因此所有参与转发的用户点主要分布在以核心用户为中心的圈内，体现出广播式扩散[8]的特点。该结果中发布微博的用户即为微博传播过程中唯一的意见领袖，因此不存在大规模的二次或多次转发，整体可视化传播的形态呈现向四周自然均匀扩散的特点。

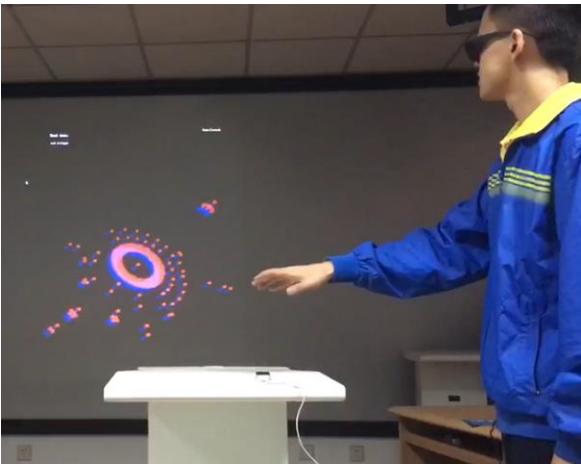


图 6 用户体验系统场景

图 7 左侧是网络属性计算结果的一个举例，可以看出，其运行结果与 `Gaphi` 计算的结果（图 7 右侧）基本一致。此外，`Gaphi` 中计算的结果值保留到了小数点后 3 位，而在我们的系统中所计算出的结果保留到小数点后 4 位，因此我们的计算精度更高。

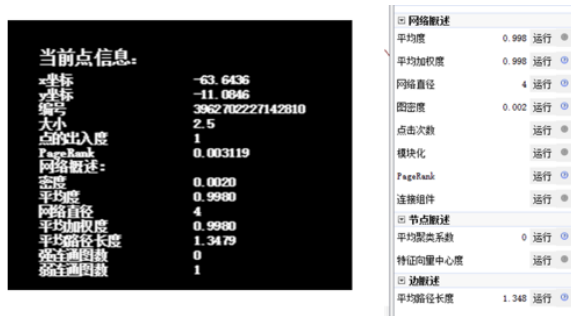


图 7 网络属性描述及测试

4 结束语

基于虚拟现实技术，以 Gephi 布局为基础，设计并实现了一款具备双目立体视觉的三维社交网络可视化交互系统。该系统具有以下几个优点：一、对从 Gephi 中得到的社交网络中各节点和边的信息进行相应计算，并优化其方法，获得准确且更加详细的网络特征，包括 pagerank 值、强连通图数、弱连通图数、密度、平均度、网络直径、平均加权度、平均路径长度；二、利用 WebGL 技术突破了二维的限制，将社交网络模型呈现在三维空间中，并且具有双目立体视觉效果，增强了沉浸感；三、允许用户用特定手势与其进行交互，增强了交互性，提高了使用的便捷度、简化了操作。

通过实验分析，最终的结论是相较于现有的社交网络可视化系统，有更加良好的沉浸感和交互性，且计算出的网络属性值准确，在社交网络领域拥有较好的应用前景。

参考文献

- [1] 王德禄. 社交化:颠覆性创新的新源泉[J]. 科技传播, 2014(11):32-37.
- [2] 陈召群. 车向前, 谷源涛. 基于谱聚类的大规模社交网络可视化方法研究[J]. 计算机仿真, 2015, 32(09):318-322.
- [3] Nathalie Henry, Jean-Daniel Fekete, Michael J. McGuffin. NodeTriX: A Hybrid Visualization of Social Networks[J]. IEEE Transactions on Visualization and Computer Graphic, 2007(6): 1302-1309.
- [4] 陈胜楠. 基于新浪微博的社交网络数据分析与可视化系统设计与实现[D]. 上海交通大学, 2015.
- [5] 宁跃飞. 李艳萍. Gephi 的社交网络消息可视化分析系统的设计与实现[J]. 现代电子技术, 2017, 40(17):183-186.
- [6] 翟因虎. 复杂网络确定性模型研究[D]. 广东工业大学, 2016.
- [7] Clark J. Hierarchical geometric models for visible surface algorithms, Communications of the ACM, 1976, 19(10): 547-554.
- [8] 谢瀛. 微博的“即时广播式”传播方式[D]. 中国传媒大学, 2013.