

分类号： TM863

单位代码： 10335

申 请 号： 127436

浙江大学

学士学位论文



中文论文题目： Hatu

英文论文题目： Hatu

申请人姓名： 海杰文

指导教师： 郑能干

合作导师： 赵挺

专业名称： 计算机科学与技术专业

研究方向： 计算机科学

所在学院： 计算机科学与技术学院

论文提交日期 2013 年 9 月 20 日

Hatu



论文作者签名:

指导教师签名:

论文评阅人 1: Name Title Organization

评阅人 2: Name Title Organization

评阅人 3: Name Title Organization

评阅人 4: Name Title Organization

评阅人 5: Name Title Organization

答辩委员会主席: Name Title Organization

委员 1: Name Title Organization

委员 2: Name Title Organization

委员 3: Name Title Organization

委员 4: Name Title Organization

委员 5: Name Title Organization

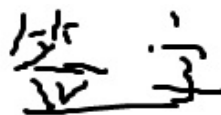
答辩日期: 2013 年 9 月 20 日

HVlab L^AT_EX Fast Guide

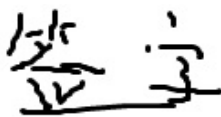
The Second Edition



Author's signature: _____



Supervisor's signature: _____



External Reviewers:	Name	Professional Title	Organization
	Name	Professional Title	Organization
	Name	Professional Title	Organization
	Name	Professional Title	Organization
	Name	Professional Title	Organization

Examining Committee Chairperson:

Name	Professional Title	Organization
------	--------------------	--------------


Examining Committee Members:

Name	Professional Title	Organization
Name	Professional Title	Organization
Name	Professional Title	Organization
Name	Professional Title	Organization
Name	Professional Title	Organization

Date of oral defence: 2013 年 9 月 20 日

浙江大学研究生学位论文独创性声明

本人声明所呈交的学位论文是本人在导师指导下进行的研究工作及取得的研究成果。除了文中特别加以标注和致谢的地方外，论文中不包含其他人已经发表或撰写过的研究成果，也不包含为获得 浙江大学 或其他教育机构的学位或证书而使用过的材料。与我一同工作的同志对本研究所做的任何贡献均已在论文中作了明确的说明并表示谢意。


学位论文作者签名： 


签字日期： 2013 年 10 月 11 日

学位论文版权使用授权书

本学位论文作者完全了解 浙江大学 有权保留并向国家有关部门或机构送交本论文的复印件和磁盘，允许论文被查阅和借阅。本人授权浙江大学可以将学位论文的全部或部分内容编入有关数据库进行检索和传播，可以采用影印、缩印或扫描等复制手段保存、汇编学位论文。

（保密的学位论文在解密后适用本授权书）

学位论文作者签名： 

导师签名： 

签字日期： 2013 年 10 月 11 日

签字日期： 2013 年 10 月 11 日

致 谢

在我开发 **Hatu** 系统的过程当中，得到了赵挺、郑能干老师的指导以及曲衡的同学的支持。在此一并感谢。

摘 要

Hatu 是基于 Web 的神经图像自动化重构以及编辑软件，支持大数据图像重构以及可视化。

关键词： 神经元重构，SWC，大数据

Abstract

Hatu is a neuronal image reconstruction and editing software based on Web, which could support big data image reconstruction and visualization.

Keywords: neuron reconstruction, SWC, big data

缩写、符号清单、术语表

缩写、符号清单、术语表

目 次

致谢	I
摘要	II
Abstract.....	III
缩写、符号清单、术语表	IV
目次	
1 简介	1
2 软件架构	2
2.1 可视化模块	2
2.2 用户模块	3
3 编辑与操作	4
3.1 编辑节点	4
3.2 添加节点	4
3.3 添加分支	4
3.4 删除节点	4
4 多分辨率切片更新	5
4.1 更新策略	6
5 总结	7
参考文献	8
附录 A.....	9
附录 B.....	10

1 简介

由于神经元重构是非常重的计算，而且要处理的神经元图往往非常巨大，并不是所有用户都能够提供足够的计算以及存储能力。这也是我们将 **neuTube**^[1] 所无法很好解决的地方，因此，**Hatu** 将编辑和计算逻辑区分开来，把 **SWC** 文件的可视化、编辑放在前端，而神经元图像的重构计算以及存储则放在后端。

2 软件架构

整个系统采用 B/S 架构设计，前后端完全分离。该系统设计的主要目的是能够承受数十 GB 的图像数据的重构与可视化，因此将计算承载分为两个部分。切片图像,SWC 文件的可视化与编辑都放在浏览器上，服务器负责切片内容的提供以及调用 DVID 中间件提供的计算与存储服务。在 DVID 之后部署有 Spark 集群，所有图像重构的计算会利用这个集群进行处理。



图 2.1 软件架构

2.1 可视化模块

系统的可视化模块是基于 WebGL 并且采用 Three.js^[2] 进行开发。前端采取事件驱动，将数据可视化与编辑区之间解耦合。该模块主要用四个主要的子模块组成，分别为渲染器、

控制器、切片处理器以及操作处理器。

2.1.1 渲染器

渲染器负责将神经元图像以及 **SWC** 可视化。该模块将神经元和突触抽象成树的节点和边，并且使用切片处理器获得当前窗口中的切片图像进行显示。

2.1.2 控制器

控制器负责处理各种用户的输入行为，主要是鼠标的点击、拖动以及滚轮的前后滚动。处理完毕后以标准的接口发出事件，交由操作处理器处理。控制器以下七个事件：拖动开始、拖动、拖动结束、悬浮上、悬浮出、空点击以及窗口变化。

2.1.3 操作处理器

操作处理器使用工厂模式设计，会根据当前的选择而创建操作。每一个单独的操作都使用状态机模式设计，而控制器发出的事件则是一个个动作，状态机则对这些动作做出反应进入下一个状态。目前支持的操作有编辑节点、添加节点、添加分支和删除节点，具体会在编辑操作一章中进行说明。

2.2 用户模块

与用户交互的网页部分使用 **React** 进行编写。

3 编辑与操作

神经元编辑是网页前端的主要部分之一，由于自动化重构的并不能完美匹配每一个神经元，因此需要人工手动进行校验。因此这个部分所提供的操作主要是面对神经元的调整编辑，而不是人工手动的全部重建。

3.1 编辑节点

通过点击选定节点，然后通过处理鼠标的拖动事件来改变节点位置，每一次节点位置改变，都需要调整与其相邻的几条边的位置、形态。在此模式下选定完节点后，会把一旁的图形界面设置激活。用户可以使用图形界面进行更精确的形态修改，包括三维坐标以及节点大小。

3.2 添加节点

在此模式下，前方的控制器将只发出点击边的事件。当一条边被点击时，会在此边的中点生成一个新的节点。原来的树结构会被更新，移除一条边，加入一个节点及两条新边。

3.3 添加分支

首先选中一个节点，然后在点击空白。一个新节点会在空白处生成，并且将选中的节点作为父节点。

3.4 删除节点

点击一个节点，此节点被删除。

4 多分辨率切片更新

每当用户改变窗口时，控制器会发出时间，此时切片处理器就会根据当前摄像头的数据来计算需要显示的切片大小以及位置，然后根据相应策略来调用多分辨率切片接口。

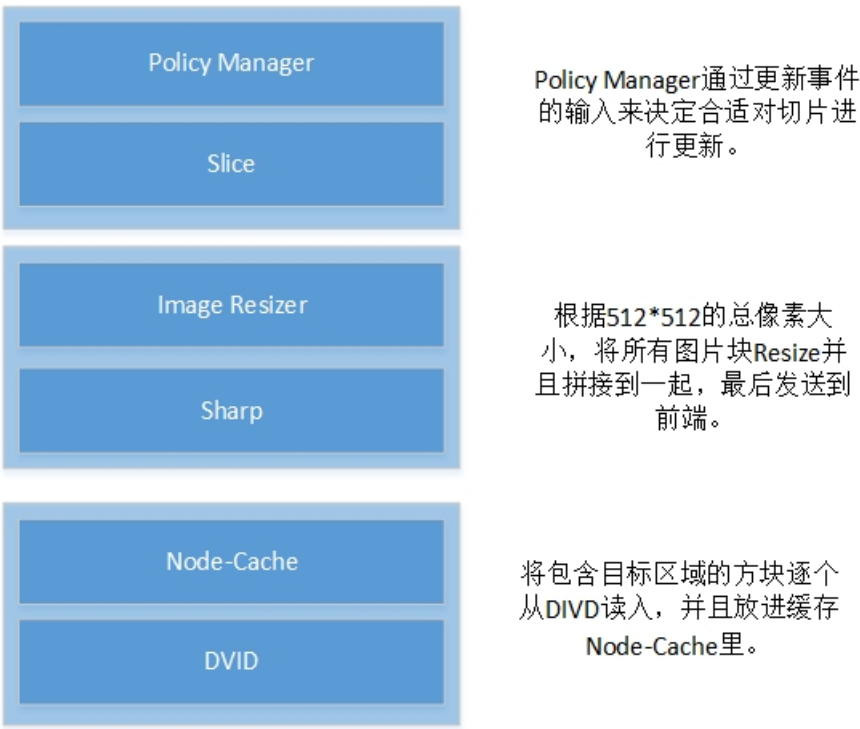


图 4.1 多分辨率更新流程

整个多分辨率更新分为三步，首先将切片从 **DVID** 当中读出来。由于图片大小可能很大，这里使用 **DVID** 中的 **multi-tile** 数据类型进行存储。在读入的时候每读入一片方块就处理一片，并且将其放进缓存之中来减少外部存储访问以加快速度。每一个方块在缓存中的寿命为 100 秒，每 120 秒清理一遍缓存中的过期数据。

在这之后，每一个图片都会被缩放成相应大小以保证最后的切片大小是 **512*512**。为了达到这一点，这里加入了一个 **ImageResizer** 的组件。他将根据每一个的方块的位置来计算其需要缩放的大小，然后将所有的方块拼接在一起。至此，在服务器上的所有处理到此结束。

在浏览器端，当我做任何一个改变摄像头位置或者缩放操作的时候，会有切片更新的事件发送到 PolicyManager。而策略管理器则会根据设定好的策略来更新当前切片的贴图。

4.1 更新策略

目前的更新策略简单有效，即若在一秒钟之内没有任何更新事件的话就根据最后一个事件对切片贴图进行更新。由于目前每次传输的贴图被限定在 256KB 一下，所以更新的速度很快。在用户操作完成最慢 2S 之内，当前的切片会被更新完成，用户体验不会比在本机上的小规模数据差。

5 总结

我们提出了一种解决了大规模图像切片数据的重构与可视化的方法。使得普通用户能够使用一台普通的电脑就可以对相当规模的神经元图片进行分析以及校正。

参考文献

- [1] Linqing Feng, Ting Zhao, Jinhyun Kim. neutube 1.0: A new design for efficient neuron reconstruction software based on the swc format[J]. eneuro, 2015. 2(1). doi:10.1523/ENEURO.0049-14.2014.
- [2] Three.js. <https://threejs.org>.

附录 A

这是附录 A 的内容。

附录 B

这是附录 B 的内容。