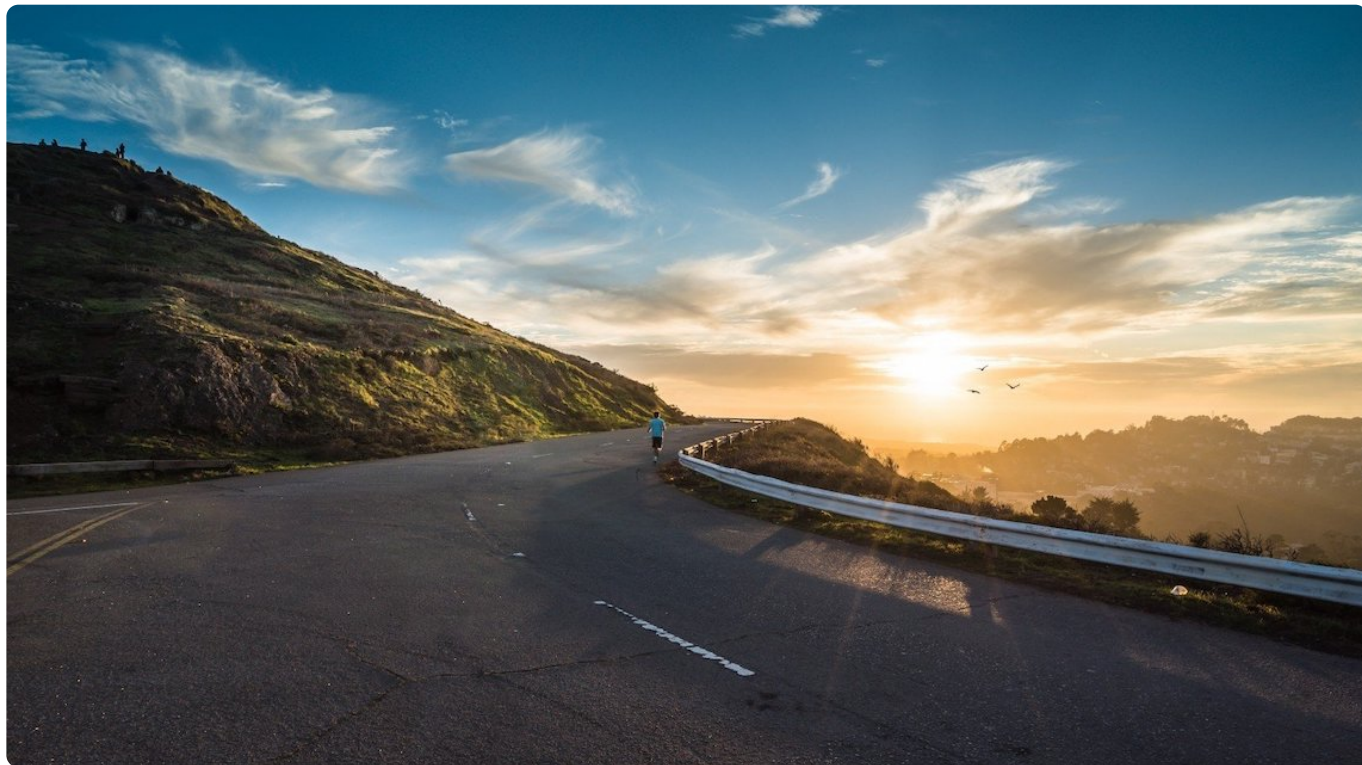


开篇词 | 不写网页的前端工程师，还能干什么？

2020-06-22 月影

跟月影学可视化

[进入课程 >](#)



讲述：月影

时长 14:23 大小 13.19M



你好，我是月影。

可能你知道我，是因为 360 奇舞团，而我是奇虎 360 奇舞团的团长。我们奇舞团是 360 的大前端团队，主要工作是支持公司的一部分业务，包括可视化大屏、技术中台的开发。

把团长这层身份抛开，我还是一个有着 15 年工作经验的前端工程师，其中有很多年在做 JavaScript 开发。如果你也做过 JavaScript 开发，我的这本《JavaScript 王者归来》，你很可能读过。



作为前端工程师，很多人的主要工作就是和网页打交道。那扪心自问一下，写了这么多网页之后，你是不是也遇到过前端职业发展的天花板，想要做些尝试或者突破呢？如果是的话，我建议你试试可视化。

可视化是前端领域中，一个几乎不用写网页的特殊分支，也是最近这几年我的主要研究方向。通过对图形学和渲染引擎底层的深入研究，我作为核心开发者，开发出了跨平台的开源图形系统 [SpriteJS](#)。同时，这个过程，也让我对可视化领域有了更多自己的想法。

所以今天我想和你聊一聊可视化，说说前端工程师不写网页，到底还能干什么。

前端工程师为什么要学可视化？

首先，我们来说说可视化的两个重要优势。借助它们，你就能明白可视化到底能干什么。

第一，现在很多 C 端或者 B 端的互联网产品都离不开可视化。

C 端最著名的例子就是淘宝“双十一”购物节的可视化大屏。B 端产品的例子那就更多了，许多中后台的管理系统，都会用统计图表来显示系统运行的各种状态。另外，很多企业级应用中的态势感知和指挥调度大屏也是可视化非常常见的应用，以及国家大力推动的智慧城市、智慧社区、智慧生活等等。如果你有关关注过，就会发现，可视化在其中都有着非常广泛的应用。

也正因为如此，行业内对可视化工程师的需求也越来越大，许多一线的互联网公司都有可视化相关的招聘需求，但是相关的专业人才却寥寥无几。

第二，可视化能实现很多传统 Web 网页无法实现的效果。

在很多人看来，可视化就是绘制各种图表，学习可视化就是学习如何画饼图、折线图、柱状图、玫瑰图这类图表。如果你也这么认为，那你可能还没有真正理解可视化的本质。实际上，真正的可视化领域，和你想得还真不太一样。

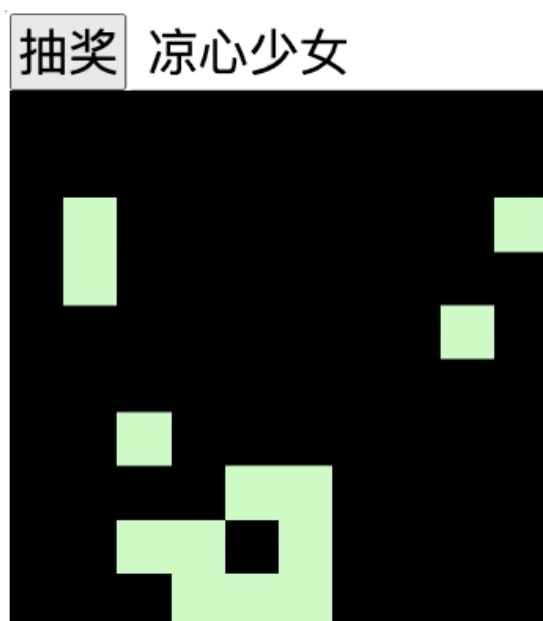
那可视化的本质到底是什么呢？我们先来看一个真实的例子。

我们奇舞团有一个传统，那就是每年年会时，我会给大家现场写一个抽奖程序，所有在场的人共同审查代码，确认没有问题后，开启这一年愉快的年会抽奖活动。

在最近的一次年会上，我写了一个比较另类的抽奖程序，怎么另类呢？

我们知道，一般的抽奖程序，如果用 JavaScript 代码来实现，其核心无非就是产生一个随机数，然后用它来判断究竟谁该获奖。这个程序当然最终要通过 CPU 来执行。但是，我写的这个抽奖程序却是使用 [GPU](#) 来抽奖。我们知道，CPU 是运算单元，GPU 是图形处理单元，那用 GPU 怎么来做抽奖这件事儿呢？

我把这个抽奖程序最终实现的效果放在了下面。你会看到，这个小小的画布上，一共分了 100 块，每一块代表一个员工。第一次抽奖后，代表中奖者的方块会变色，之后再抽奖颜色就会不断叠加。这个程序不仅能保证每个人中奖的概率是公平的，还能保证一个已经中奖的人，不会重复中奖。



你可能会觉得，这个抽奖好像没有什么用处啊。其实并不是，我们要知道，这个程序在理论上可以同时对上百万人（我们可以用一个 1024×1024 的画布，用一个像素点表示一个区块）进行抽奖！虽然，很少有对上百万人同时抽奖的场景，但是，很多程序，比如一些 AI 的程序，其实核心原理类似于这样的抽奖。

在这里，我是用了 GPU 的能力，通过渲染图形来完成的抽奖，这实际上就用到了可视化相关的核心技术。所以你看，可视化可不只是在绘制图表，图表只是可视化工程师日常工作产出的一部分。而就像上面这个例子一样，**可视化更重要的是利用 WebGL 这样的图形系统，来更高效地实现更多有趣的图形**。尤其是，当你在写一个可视化的项目，要在大屏上用浏览器渲染数以万计的元素时，也许 WebGL 或者 GPU 就是目前唯一可行的方案。

总的来说，通过研究可视化，你能够比较深入地掌握图形学和 WebGL 技术。而这些技术的应用呢，并不局限于可视化领域，它们在游戏、AI 等其他技术领域也有广泛的应用场景。

所以，就算你并不想成为一名可视化工程师，但学习相关的图形、视觉呈现的原理和方法，也能最大化地丰富你的知识面，拓宽你的技术成长路线，让你的技术天花板变得更高。更详细点来说，像视觉呈现技术中也有涉及高级 CSS 原理的部分，所以如果你学会了可视化，也会对你的 CSS 技能有很大的启发和提升。

如何成为一名可视化工程师？

既然可视化这么有用，那想要成为一名优秀的可视化工程师，我们该怎么做呢？

我觉得，一名优秀的可视化工程师，最应当具备的能力就是要根据产品经理和设计师的想法，实现真正贴合用户需求的、灵活多变的、具有视觉震撼力，并且充满创意的优秀可视化作品。

这句话说起来简单，但是要做到，绝不是会使用某个库、会调用某个 API 就能实现的。

比如说，现在我们要把一个图表的线条样式，从实线段改成虚线段，那我们就可能不能使用某个现有的图表库来完成。这是因为，有可能这个图表库中并没有提供参数支持线条的虚线。这时候，我们要么放弃这个需求，要么就得换图表库或者自己去修改图表库的源代码。

如果我们不放弃，想要自己去修改图表库的源代码的话，其实会遇到更多的问题。因为，这些图表应用不仅业务代码相当复杂，图表库本身的源代码也很复杂，所以，不管是修改代码还是后期维护，我们都会面临很多复杂的问题。

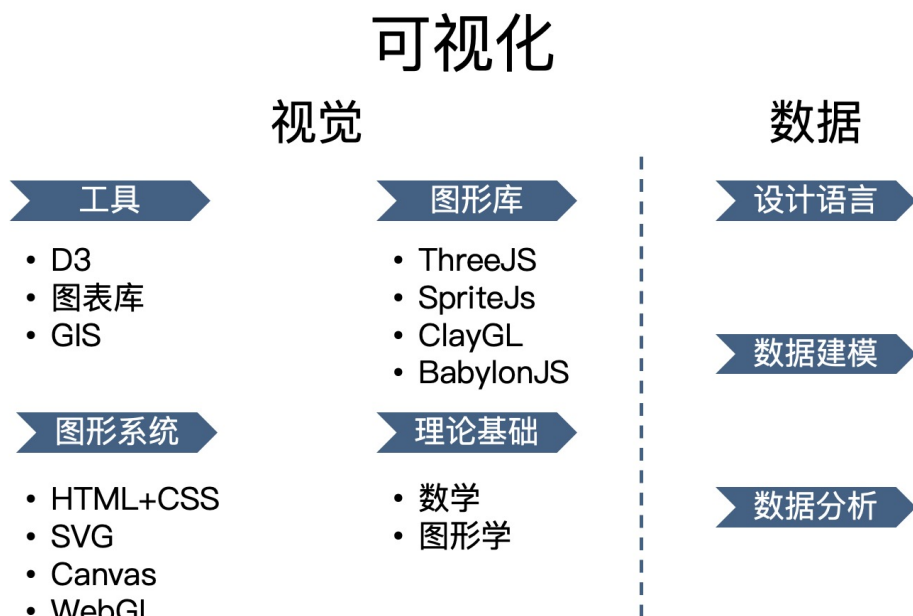
再比如说，我团队的一名同学最近在实现一个 3D 可视化大屏，需要渲染大约 70 万个元素，处理各种视觉和交互细节。这可不是调用某某库的某个组件就能轻松实现的，在技术实现的背后有着非常深的图形学、物理学和数学知识。

因此，想要成为一名可视化工程师，**我们不能停留在可视化现有的库、框架和工具的使用层面上，必须深入技术栈的底层，充分理解数据，掌握视觉呈现相关的技术和基本原理。打好基础之后，再系统地学习可视化领域的方法论，才能提高解决问题的能力。**

也正因为如此，我专门打造了这门课，目的就是帮你打好底层基础，建立可视化知识体系，助力你成为合格的可视化工程师。

如何快速掌握可视化？

首先，想要学好可视化，我们就必须要了解可视化的体系都有什么。这里我画了一张知识结构图，你可以看看。



从图中你可以看到，可视化涉及的内容非常多。光是数据就包括数据分析、数据建模和设计语言，它们每一个都能单独开一门课程了。可想而知，可视化涉及的知识有多广。既然如此，我们该如何从这么多的内容中，快速掌握可视化呢？**答案是要有重点地学习。**

那如何有重点地学习呢？我们下面分别来分析。

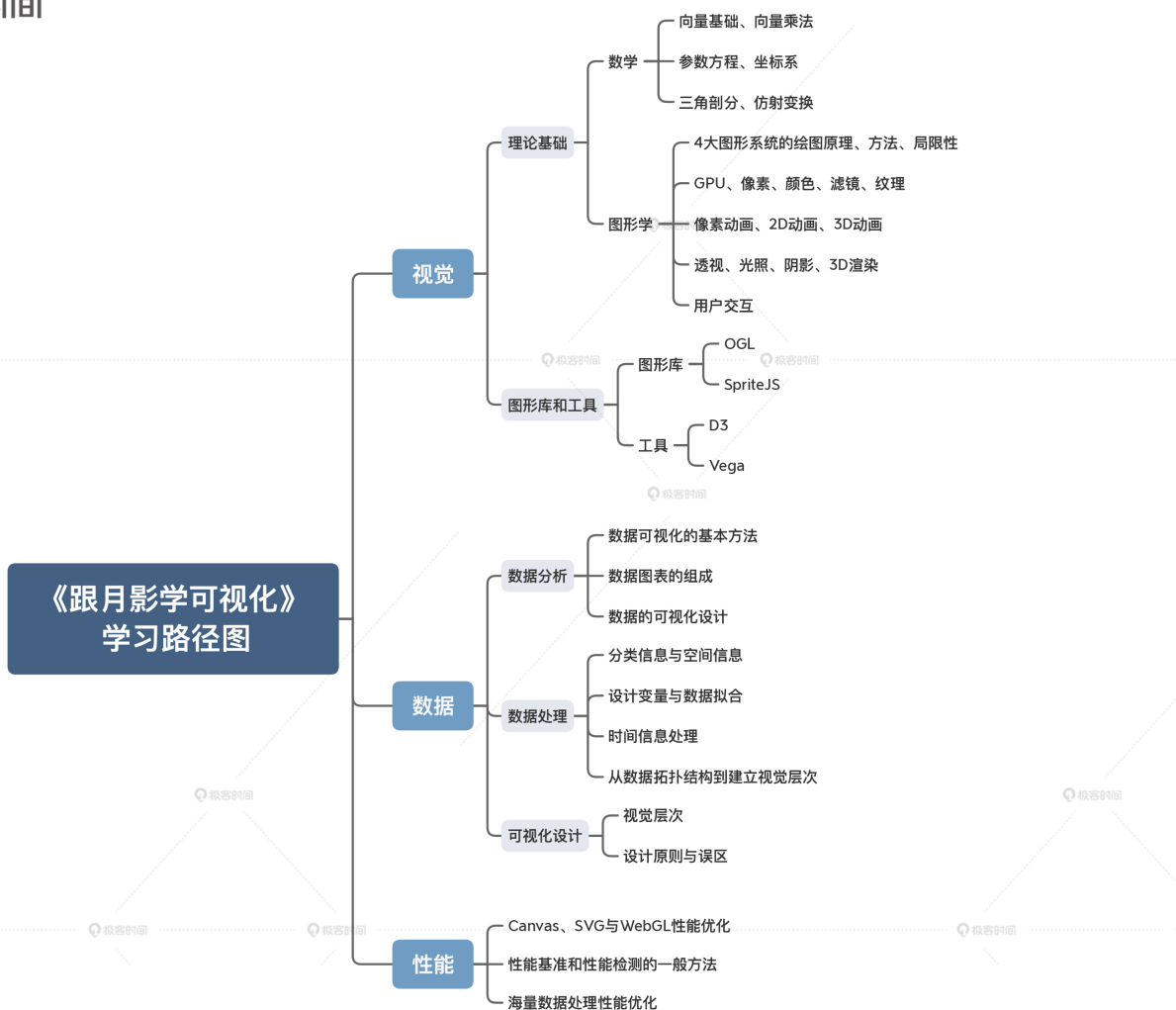
首先，可视化的视觉主要涉及四个部分，分别是**理论基础**、**图形系统**、**图形库**和**工具**。看起来内容很多，但我们没有必要全都深入学习。而是应该把重点放在理论上，**因为随着技术的发展，工具会改变，而基础不会**。视觉部分的理论基础就是数学和图形学，也就是说，我们必须要学会如何运用数学和图形学知识，来绘制各种各样的几何图形，这样才能实现更加酷炫的视觉效果。

数据这一块呢，虽然相对比较简单，但是内容也不少。如果我们要深入学习细节的话，压力太大，也很难坚持下去。好在，可视化工程师对数据的理解，会随着业务的深入而逐步加深。所以，在数据部分，我们只要理解解决问题的整体构思和方法论，知道如何从一个可视化工程师的角度去理解和运用数据，就已经足够为我们未来的深入研究，打开一扇大门了。

最后，谈可视化离不开性能，性能往往决定了可视化项目最终所能呈现的整体感官效果。怎么理解呢？举个例子，当你想要通过一段代码，来实现非常具有视觉冲击力的效果时，却发现即使 GPU 和 CPU 都以最大功率运转了，还是没能把你构思的画面，在屏幕上流畅地显示出来。这是为什么呢？

这很大可能是因为性能没有达到最优。也许是你的技术选型和代码实现在渲染方面遇到了瓶颈；也许是你过于关注渲染本身，而忽视了计算；也有可能是核心算法有改进空间。有时候，我们花大力气去调优渲染机制，其实不如换一种更加合理的算法取得的收益更大。上面这些呢，都有可能是导致性能不佳的元凶。因此，在性能方面，我们应该学习的，是如何快速找到合理的性能优化点。

好了，学习可视化的重点我们说完了，那到底该怎么学呢？我结合这些重点，梳理了一张可视化的学习路径图。接下来，我就结合这个路径图来说说，我们的课程到底是怎么设计的。



我们的课程是如何设计的？

我把整个课程分为了五个部分，分别是图形基础、数学基础、视觉呈现、性能优化和数据驱动。下面，我们来详细说说。

第一部分，图形基础。首先，我会带你熟悉 HTML/CSS、SVG、Canvas2D 和 WebGL 这四种图形系统，并会结合一些简单的实际案例，帮助你全面掌握它们的基本用法、原则、优点和局限性。通过这部分的内容，你能对图形系统有一个大体的认知，并且掌握一些必须的绘图技巧，从而能够更好地学习后续的课程。

第二部分，数学基础。在这一部分里，我希望通过对向量和矩阵运算、参数方程、三角剖分和仿射变换等内容的深入讲解以及综合运用，来帮助你建立一套通用的数学知识体系，能适用于所有图形系统，以此来解决可视化图形呈现中的大部分问题。

第三部分，视觉呈现。视觉呈现效果是衡量可视化效果的重要指标，因此，实现各种视觉效果的能力，可以说是可视化工程师的核心竞争力。视觉呈现也是可视化中非常重要的内容，那这一部分我会分两个阶段来讲：在基础阶段，我们主要讨论如何以像素化的方式处理图案，来实现各种细节效果；在高级阶段，我们讨论动画、3D 和交互等更深入的话题。并且，我还会借助美颜、图片处理和视觉特效等实际例子，来教你怎样更好地应用数学和图形学知识，帮你全面提升视觉呈现的能力。

第四部分，性能优化。这一部分，我会讲解 WebGL 渲染复杂的 2D、3D 模型的方法，让你了解可视化高性能渲染的技术思路，比较轻松地找到可视化应用中的性能瓶颈并着手优化。再结合我总结出的一些成熟的方法论，你就能在实现可视化项目的时候，解决大规模数据批量渲染的性能瓶颈问题，来实现更酷炫的视觉效果了。

最后一部分，我们将重点讨论数据驱动。通常来说，数据的格式和量级决定了图形系统的选择和渲染方式。一个成熟的可视化系统，应该包括数据引擎和渲染引擎两部分。所以，对于可视化而言，数据和渲染同样重要。那么在这一部分中，我会结合 3D 柱状图、3D 层级结构图、3D 音乐可视化等案例，讲解数据处理的技巧，真正将数据和视觉呈现结合起来，实现具有科技感的 3D 可视化大屏效果，并最终形成完整的可视化解决方案。

以上，就是这门课我想分享给你的全部内容了。总的来说，我希望通过这门课的学习和讨论，你能真正系统地掌握可视化开发的原理和方法，理解如何处理数据，学会各种视觉呈现方式，最终创建出自己的可视化工具集，并且用它来更高效地解决可视化开发中的问题。

其实到这儿，课程的核心内容就说完了。不过，在学习上，我还想再啰嗦几句，要掌握这些内容，肯定需要你付出非常多的耐心和毅力，并且保持积极的思考。

课程中难免有部分内容比较抽象和枯燥，这也正是我们最容易放弃的地方。所以，在课程中，我也会尽量用一些比较浅显而有趣的例子把内容串起来。同时，为了帮助你理解，我会添加大量的代码。我希望你能真正在浏览器中运行这些代码，如果遇到问题，也非常欢迎你在留言区和我一起讨论。

最后，我是一名工程师，喜欢写代码，我认为很多道理通过代码来讲是最直接的。如果你也和我一样，那就快点加入可视化的学习，和我一起享受“创造世界”的乐趣吧！

源代码

推荐阅读

[1] 🔗 Canvas API 的官方文档

[2] 🔗 SVG 的官方文档

[3] 🔗 WebGL API 的官方文档

跟月影学可视化

系统掌握图形学与可视化核心原理

月影

奇虎 360 奇舞团团长

可视化 UI 框架 SpriteJS 核心开发者



新版升级：点击「👤 请朋友读」，20位好友免费读，邀请订阅更有**现金**奖励。

© 版权归极客邦科技所有，未经许可不得传播售卖。页面已增加防盗追踪，如有侵权极客邦将依法追究其法律责任。

下一篇 预习 | Web前端与可视化到底有什么区别？

精选留言 (8)

💬 写留言



陈强

2020-06-22

非常期待！

之前有用过各种JavaScript图形库写过一些可视化的图，但也仅仅是调用API而已，虽已经

满足的大部分需求，但还是不得原理（比如拿antv或者echarts举例，默认的图不一定符合需求，当需要太强的个性化需求，而它的API没有开放对应的属性接口时，几乎是无法的，还有各种版本之间，各种手机系统，浏览器之间的差异导致的不兼容问题等等也很让人...
展开



5



Geek_c46dfc

2020-06-22

期待前端可视化+智能化

展开



2



ZT.QIN

2020-06-22

我觉得双十一的购物大屏属于b端，而不是c端。这个主要不是为了跟客户看的，更多的是企业内部用来分析数据的。

展开

作者回复: 也可以这么认为吧，本身属于B端，不过数据来源于C端。



1



快乐小球球

2020-06-22

```
float random(vec2 co) {  
    highp float a = 12.9898;  
    highp float b = 78.233;  
    highp float c = 43758.5453;  
    highp float dt= dot(co.xy ,vec2(a,b));...
```

展开

作者回复: 哈哈，这个是用sin函数的小数部分来模拟随机



1

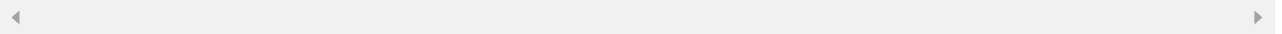


jokar

2020-06-23

老师，我之前理解的 d3 和 threejs 都算是js的库，为什么您这里把它们分类为工具和图形库呢？

作者回复: 都是js库, 但d3是数据驱动文档, 而threejs是3D图形库, 它们的作用是不一样的。

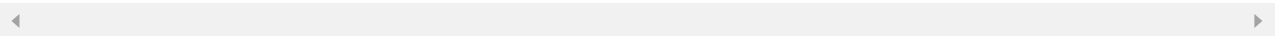


twelve

2020-06-23

老师, 前提要有什么基础以及想问问个人设备没有GPU, 如何解决?

作者回复: 学习图形学, 需要有一定的数学基础。没有GPU的是什么设备, 是嵌入式系统吗? 现在连树莓派都有板载的图形处理器, 所以都有GPU。不过如果没有GPU, 也是可以用Canvas2D渲染的。



宁康

2020-06-22

确实如老师所说, 前端开发几年就会有瓶颈, 而可视化这部分刚好是一个突破口, 期待接下来的内容。



haha123

2020-06-22

老师好, 我是一个前端新手, 最近在学习前端过程中, 遇到最大一个问题就是前端框架中 (比如bootstrap) 标签 class属性内容特别多, 一个一个去查总是查不到, 效率特别低, 特别苦恼, 老师能不能给像我一样的前端新手在学习路径和学习方法方面一些建议?

展开 ∨

