

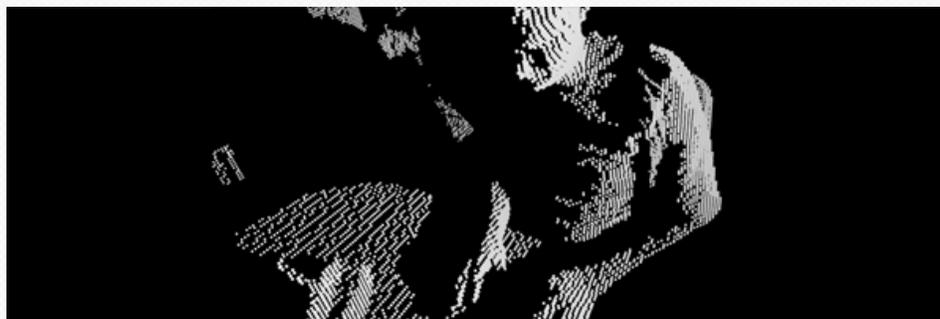
# creative JS

[Home](#)
[Tutorials](#)
[Resources](#)
[About CreativeJS](#)
[CreativeJS Training](#)

## DepthCam

[Tweet](#)
[Like](#)

9



What do you get when you mix together Python, Node.js, Three.js, CoffeeScript, WebGL, Websockets and a Kinect? If your answer was “A live-streaming, rotatable, zoomable 3D camera that wouldn’t be out of place on an episode of CSI”, you’d be right.

George MacKerron from the [Centre for Advanced Spacial Analysis](#) at University College London combined these things into [DepthCam](#) to allow us to look over his shoulder or snoop around his desk. It’s best viewed during UK office hours while George is at his desk but there is also a [YouTube video](#) showing it in action.

### How it works

First, George records the depth image of his desk from the Kinect. This is a high-resolution 3D image which is then downsampled in Python and compressed using LZMA. The data stream is also compressed using an algorithm similar to MPEG-2 where the majority of frames only contain the differences between frames instead of the whole scene. Finally, this is streamed via a binary WebSocket to a node.js server

George programmed the client-side in CoffeeScript. The browser connects to the node.js server – again using binary WebSockets – to receive the compressed depth data in real time. This is then rendered in WebGL using three.js. The use of binary WebSockets means that this only works in an up-to-date version of Chrome or Firefox 11 but that’s the price of being cutting-edge.

Best of all, the whole project is [available in GitHub](#). You can find out more technical details in on [George MacKerron’s blog](#).

[Follow @creativjs](#)
[Subscribe \(RSS\)](#)

### Recent Posts

[Google Cube](#)
[Score Rush](#)
[requestAnimationFrame – The secret to silky smooth animation](#)
[CSS Tilt](#)
[Retro rippling reflections with lake.js](#)
[CreativeJS](#)
[About CreativeJS](#)
[The CreativeJS Team](#)
[Suggest a site](#)

## More

This isn't the first time this kind of single-camera, 3D point-cloud representation has been used, indeed, Joel Gethin Lewes, Arturo Cassuro and perennial favourite Mr. doob used something similar to stream the [2011 Art&&Code conference](#) in 3D. In that case, the heavy-lifting on the server-side was built using [openFrameworks](#) – a set of C++ tools for creative projects – and used cleverly encoded ogg/vorbis video to store the depth data. The source for [this project is also available on GitHub](#). Unfortunately, there aren't any videos of the conference but Mr. doob does have a similar [looping interactive demo](#) on his site.

For an AMA (Ask Me Anything) interview for Reddit, James George and Jonathan Minard filmed [Golan Levin using a similar technique](#) with the added complexity of a second camera to capture colour information, a technique James George has used in [several projects](#). This required post-processing but, given that DepthCam is open-source, it might not be long before this can be done in real time as well.

This kind of 3D depth sensing technology has been available for a few years inside university research projects and complex art installations but so far it has always required expensive technology, hardcore server programming or a combination of both. It's very exciting to see this is now completely possible using (soon-to-be) standard web technologies. In-browser 3D video chat would be an impressive direction to take it but once people start experimenting with the possibilities, it's difficult to imagine where it could end up. CSI: 3D, anyone?

[DepthCam](#)

[GitHub project for DepthCam](#)

[DepthCam on Youtube](#)

[George MacKerron's blogpost about DepthCam](#)

[Mr. doob's Kinect demo](#)

[Interview with Golan Levin](#)

This entry was posted in [Video](#), [WebGL](#) by [Simon Madine](#) on [February 28, 2012](#).



About Simon Madine

[Simon \(@thingsinjars\)](#) is a recovering game designer who spends his time making [educational interactives](#), [handy tools](#) and [digital toys](#). As a Senior Frontend Developer on [Nokia Maps](#), he is as likely to be found organizing Tech Talks as he is to be writing code. A specialist in rapid prototyping, Simon is always on the lookout for a way to do things faster. In his spare time, he moonlights as the tech side of [Museum140](#).

[View all posts by Simon Madine](#) □

□ [Previous](#)

[Next](#) □

## Leave a Reply

Your email address will not be published. Required fields are marked \*

Name

\*