

TABLE OF CONTENTS

Why?	2
How do I use it?	3
Any requirements?	4
Available methods?	5
Inspector fields?	6
Any bugs, issues?	7
Anything else?	8

Why?

Did you ever make a mobile WebGL app, and your client said "The scene looks distant on landscape". Well, this plug-in attempts to fix that, by adjusting the field of view. See the before and after on the screenshot. You might be thinking, why don't I lock the phone's orientation. I'd do that, except it doesn't work on Safari. Hence, this plug-in.

How do I use it?

Attach the script under the scripts folder to your main camera. Leave the autostart selected if you want the field adjustment to begin since the start of the experience. That's it.

Any requirements?

Well, your browser needs to support the [orientationchange](#) api. I tested on android chrome and iOS Safari. So it **must** work on those. Hopefully it works on other browsers as well, but Safari and Chrome are the only ones that concern me.

Available methods?

There is *Stop* and *Play* . To disable and enable field of view adjustment, respectively. Although I don't see why you'd want to disable it.

Inspector fields?

There is just one. *Autostart* . If true, the field of view adjustment will begin as soon as the app starts. If false, it won't, and you'll need to call *Play* yourself.

Any bugs, issues?

Yes, concerning iOS Safari. If you start the experience on landscape, the field of view might not adjust correctly at first. You might need to switch to portrait, and back to landscape again to work correctly. Or simply start at portrait, then switch to landscape. I attempted to fix the issue, but I am unsure if my fix works on every device. Check the *.jslib* file for more details. If you know a better fix, don't hesitate to contact me. I appreciate any help.

Anything else?

Nope, it's a simple plug-in. It just adjusts the field of view. Attach it to your main camera and forget about it.