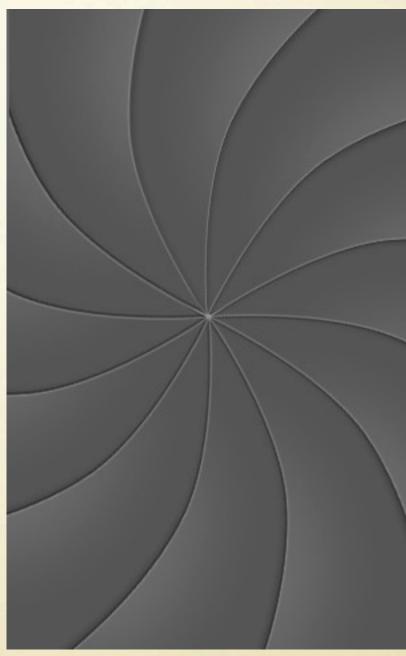
getUserMediaメソッドを使って HTML5で作ったカメラアプリ

チームAKB

• 写真を撮ります



• カシャ!



• カシャ!



• ブラシを選択します



• ブラシを選択します



女子も大好き ドロイドくん!



ブラシを再度選んで 何度でも!



ブラシを再度選んで 何度でも!



デザインのポイント

- 説明がないので、パッと見てだれでもわかるUI デザイン
- 写真がメインなので、黒基調のデザインでおさ えめに
- すぐにアプリが始まらないように、スタートアップ画面から開始



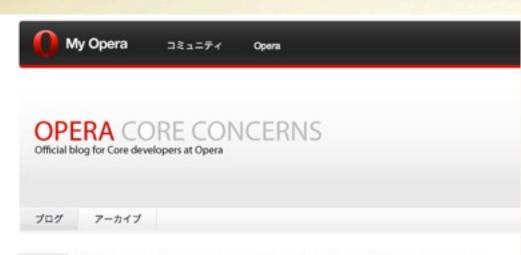


デザインで苦労したところ

ブラウザ用にCSSを適用させること (URL、スクロールバー、メニューが邪魔)









Native webcam support and orientation events technology preview

By Rich Tibbett (richtr), Wednesday, 23, March, 10:01 getUserMedia, mobile, html, whatwg, W3C, prientation, device

Last week we wrote a blog post discussing our internal prototyping of <u>web camera streaming in the browser</u>. On the very same day, the proposed standard interface on which that was built <u>changed</u> considerably.

This week we are pleased to announce the release an updated preview build of Opera Mobile for Android enabling web developers to access and interact natively with a device's webcam via JavaScript. This build has been entirely re-based on the new standards proposal introduced last week so it's goodbye to the <device> element and a big warm helio to the getUserNedia JavaScript API.

In addition to web camera streaming, Opera is also previewing support for the draft <u>W3C Orientation</u> <u>Events</u> specification. These capabilities open up some exciting possibilities for the future of the web itself.

getUserMedia()

今のところ Android用Operaの 中でも特殊なOperaだけ 対応。

http://my.opera.com/core/blog/2011/03/23/ webcam-orientation-preview

getUserMedia()で得たstreamオブジェクトをvideo要素のsrcに当て込むだけ!

```
<video autoplay>
JavaScript
 var video = $("video").get(0);
 if(navigator_getUserMedia) {
     navigator.getUserMedia('video', 成功時, エラー時);
      function successCallback( stream ) {
        video.src = stream;
```



• 仕様が不確かなところや、バグ、不具合



開発版なので、奇妙な問題が...

Canvas.renderingContext2DオブジェクトのgetImageData()というメソッドがimagedataを返さない。(ピクセル操作ができない)

注意

getUserMedia()で得たstreamサイズは 多分240px!

videoをcanvasにdrawImage()したら 小さく表示された、ここに数時間ハマる。 <video width=300でも video.videoWidthプロパティで調べたら 240pxを返した!

注意

getUserMedia()で得たstreamサイズは 多分240px!

 $300 \div 240 = 1.25$ なので… ctx.setTransform(1.25, 0, 0, 1.25, 0, 0); で、今回は無理矢理リサイズしました。

