Using Webcams For Product Presentations in HTML5

M. Borg[1] and M. Kraus[2]

[1]School of Information and Communication Technology, Aalborg University, Denmark
[2]Department of Architecture, Design and Media Technology, Aalborg University, Denmark

Abstract

The Media Capture and Streams API ("getUserMedia API") enables plug-in free webcam support in HTML5. This opens new ways of interacting with visualisations using open web standards. In this work, two techniques are proposed that employ the user's webcam for interactive product presentations on websites. The first technique tracks the user's face and uses its position to create the impression of a user-controlled rotation of a product by displaying a sequence of product images for different view points. The second technique uses photographs of a product with reflections of coloured patterns to render the user's webcam stream as a reflection in the product presentation. Thus, users in front of a webcam are able to see their virtual reflection in a picture of a real product in real time. Our implementation of two prototypes shows that it is technically possible to implement these features with HTML5 and JavaScript only.

Categories and Subject Descriptors (according to ACM CCS): H.5.1 [Information Interfaces and Presentation]: Multimedia Information Systems—Artificial, augmented, and virtual realities I.3.7 [Computer Graphics]: Three-Dimensional Graphics and Realism—

1. Introduction

In the footsteps of HTML5, the WebRTC API (Web Real-Time Communication) has been proposed to create plug-in free video conferencing in web browsers. With the development of WebRTC, the "getUserMedia API" has been defined, which makes it possible to enable the user's webcam without the use of plug-in based technologies. This API is currently supported by several browsers, e.g. Opera (v. 12+), Chrome (v. 21+) and Firefox Nightly (v. 18+).

Even though the API has been developed for video chatting, there are many more uses of the user's webcam stream on websites. This work proposes two such possibilities for product visualisation: product rotation by head movements, and reflections of the user in surfaces of products. Tracking the user's face and displaying a series of product images correspondingly to the position of the face can achieve the impression of a user-controlled rotation of the product. Furthermore, the webcam stream can be mapped to product photos to create the impression of interactive reflections in the product. Our implementation of these features is based on a simple image capturing setup. A silver watch was chosen to illustrate the proposed techniques as it features reflecting surfaces and consumers might be interested to rotate this kind of product to view it from all sides.

2. Product Rotation

A setup using a turntable was used to capture images of a watch from 360 degrees horizontally. Thus, the watch will appear to rotate when displaying the images as a sequence. An implementation of face detection and tracking in JavaScript [Hea13] was used to detect and track the frontal face of the user in a webcam stream. The position of the face is then mapped to a specific image in the sequence of the captured images (see Figure 1 (a)–(d)).

An online user study determined the trade-off between image quality (i.e. loading time) and angular resolution (i.e. the necessary number of images to create the rotation), where test subjects rated the two variables on a 7-point scale. The intersection of the response curves was at 90 images. Furthermore, the maximum rotation angle was determined by averaging to be 272 degrees (e.g. 68 out of 90 images covering 360 degrees) and the sensitivity was determined to be 6 degrees of product rotation per 1% head movement relative to the field-of-view of the webcam.

© The Eurographics Association 2013.

DOI: 10.2312/conf/EG2013/posters/021-022

delivered by EUROGRAPHICS DIGITAL LIBRARY www.eg.org diglib.eg.org

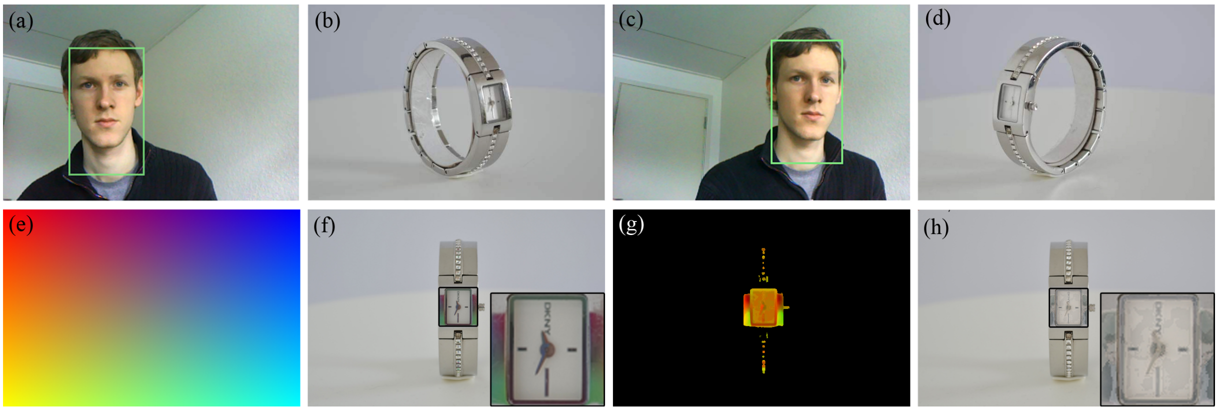

Figure 1: *Top row: The positions of the user's face in (a) and (c) are used to display corresponding images (b) and (d) to achieve an impression of a user-controlled rotation. Bottom row: A gradient image (e) is reflected in a product photo (f), from which a lookup image (g) of the mapping can be created. This lookup image can be used to map the user's webcam stream to the product photo (h).*

3. Product Reflections

To create reflections of the user in product photos, a setup based on work by Chuang et al. [CZH*00] for capturing pictures of reflections of different images is employed (uniformly coloured images in red, green, blue, white, and black, and a gradient pattern (see Figure 1 (e)). The pictures are captured through small holes in the middle of the images, such that the reflections of the images in the product are captured (see Figure 1 (f)). By comparing the photos with the reflected primary colours, a reflectivity map is created, which is used to find the pixels that show specular reflection. From these pixels, the three colours closest to the three primary colours are determined. The resulting three colours span a triangle in the RGB-space, which is used to map the reflected colours of the gradient pattern to the original colours and, therefore, to the coordinates in the image of the gradient pattern. A simple box-filter blur is applied to limit the effect of noise and aliasing in the picture of the reflected gradient pattern. The computed coordinates are then saved as a lookup image, where the red component represents the x-coordinate in the gradient pattern and the green component represents the y-coordinate (see Figure 1 (g)). A new representation of the RGB-space based on the mentioned triangle of colours is also saved.

When rendering the product picture, a lookup in the precomputed image is used for each pixel to determine the coordinates of the virtually reflected pixel of the webcam stream. The pixel colour of the webcam at this position is then converted to the new representation of the RGB-space to simulate absorption of light. Furthermore, a white balance variable is multiplied such that a dark photo creates dark reflections and a bright photo creates bright reflections. The final product of these mapped colours from the webcam stream is added to a product picture without reflection (i.e. a photo reflecting a black image). Because the reflectance value from the product is unknown, a weighting is then used between a product picture reflecting a white background and the extracted reflection. This way, a plausible simulation of reflection in a product can be achieved (see Figure 1 (h)).

4. Results and Discussion

The two prototypical implementations of the features were assessed in an online user study to evaluate the mediation of enjoyment, usefulness and risk on the consumer responses of such features for interactive product presentation. The results showed no significant difference compared to more traditional product presentations. One reason is supposedly that not all test subjects noticed the features. Another limitation is that most users do not trust a website to turn on their webcam. To sum up, the prototypes show that it is technically possible to implement the proposed techniques with HTML5 and JavaScript only. However, user studies indicated that there are still many challenges with respect to the usability of these techniques.

References

[CZH*00] Chuang Y.-Y., Zongker D. E., Hindorff J., Curless B., Salesin D. H., Szeliski R.: Environment matting extensions: towards higher accuracy and real-time capture. In *Proceedings of the 27th annual conference on Computer graphics and interactive techniques* (2000), SIGGRAPH '00, pp. 121–130. 2

[Hea13] Headtrackr Development Team: headtrackr.js. http://auduno.github.com/headtrackr/documentation/reference.html, 2013. Last accessed: 07/02 2013, published under the MIT License. 1

© The Eurographics Association 2013.