YouTube4Two: Socializing Video Platform for two Co-present People

Alessio Bellino, Giorgio De Michelis, Flavio De Paoli
University of Milan - Bicocca
Department of Computer Science, Systems and Communication (DISCo)
Viale Sarca 336/14, 20126, Milan, Italy
{bellino, depaoli, gdemich}@disco.unimib.it

ABSTRACT

YouTube4Two is an application that exploits the YouTube media library (through its API) to demonstrate a new style of social interaction. Two co-present people can share a video and act autonomously to navigate the related-video and comment lists, and search for videos. The novelty is that they can use their own smartphones connected via Internet to control the shared application. The application has been designed according to the responsive-web-design (RWD) principle to smoothly pass from desktop interface (controlled by mouse and keyboard) to smartphone interface (with touch control). YouTube4Two introduces the multi-device responsive Web design (MD-RWD) style that extends the RWD style by introducing the separation between displayed content (on a shared screen) and displayed control commands (on personal smartphones) to support shared control over an application.

Categories and Subject Descriptors

D.2.2 [**Design Tools and Techniques**]: User Interface; H.5.2 [**User Interfaces**]: Input devices and strategies (e.g., mouse, touchscreen)

General Terms

Design, Experimentation, Human Factors.

Keywords

Mobile Web interfaces; Multi-Device Web Applications; Responsive Web Design;

1. INTRODUCTION

Up to some years ago, most, if not all, devices for accessing and interacting with digital content have been designed for individual use. New systems and devices that was introduced to support social interaction has created innovative usage situations, and consequently new issues due to the physical settings. Large (multi-)touch screens, for example, restrict to few (up to 5) users their use to avoid that people who are away from the screen will not be able to even see what is displayed [5]; moreover, users need to move to interact with a specific part of the screen, hence creating possible conflicts to access the displayed content. The limit on the number of users that can collaborate in front of a touch screen can be overcome by systems composed of a large screen and hand-held devices (e.g., smartphones) that allows more users to interact with the digital content from a distance of some meters. Even in this case there are interesting challenges for interaction designers, who has to define (i) the correspondence of what appears on the private smartphone to what appears on the shared screen to facilitate the command and control by users, and (ii) how simultaneous interaction of multiple users can be managed to avoid inconsistencies and conflicts.

In this paper, we present the YouTube4Two demo prototype, which is a Web application that exploits the YouTube media library (through its API) to let two people watch a video and, at the same time, navigate related comments and related videos. The novelty is that users can act on the shared application (and hence on the shared screen) independently and concurrently from each another. Coordination is achieved by exploiting the co-presence of users that can talk each other and see the effects of individual acting on the shared screen.

YouTube4Two exploits HTML5/JS/CSS3 features to adapt its interface according to the situation, since it can be uses as a traditional application or multi-device application that exploits two smartphones as remote controllers. Moreover, WebSocket connections are exploited to ensure real-time communication among involved devices.

In the next section we discuss a motivating scenario to state the goals of YouTube4Two. Section 3 describes the interfaces and the interactions in the different settings. Section 4 illustrates some technical details. Section 5 draws some conclusions and illustrates future developments.

2. A MOTIVATING SCENARIO

Let's consider two musicians (Alice and Bob) that use YouTube to discover video of classical music by looking for different versions of a piece or similar pieces. In traditional interaction paradigms, only one user at a time can operate a system leading to a scenario like the following (Figure 1a): while Alice (A) is free to interact and navigate, Bob (B) can

only see the Alice's behavior and the effects on the shared screen. Bob can interfere with the Alice interaction only by pointing at the screen to show the content of his interest, or asking Alice to perform a certain action. At any rate, the role of Alice and Bob is not equal.

While looking at a video, Alice and Bob may be willing to autonomously browse and read comments about the played video to find and discuss the different points of view expressed by the other users, or to browse related videos to identify similar performances. Such a behavior is quite common dealing with classical music, but also with many other subjects; therefore we can assume it as typical in many situations.

We address this behavior by letting users to use smartphones as control devices in order to support multi-user interaction (Figure 1b): Alice and Bob can stay away from the screen, e.g., sitting on a sofa in a living room, and be able to focus on different elements (related videos, comments, search list). YouTube4Two allows them to navigate the system independently, while keeping the opportunity to share the content and coordinate their actions to increase the social behavior and improve the collective experience.
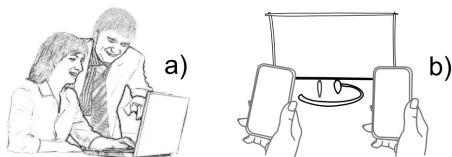

Figure 1: Direct vs. remote control.

YouTube4Two is based on YouTube API [4] to search for videos, find related videos and comments. YouTube4Two is designed to self-adapting according to the different ways of use: i) single user interacting with mouse and keyboard control (i.e., desktop version); ii) single user interacting with a smartphone (i.e., mobile version with touch screen); iii) single user interacting with a smartphone used as a remote control (i.e., single-user multi-device version); iv) two users interacting with two smartphones to control the application (i.e., multi-user multi-device version). YouTube4Two is able to smoothly pass from a version to another automatically according to the devices on use at any time.

3. ADAPTIVE WEB APPLICATION

YouTube4Two[1] is a Web application that exploits the HTML5/JS/CSS3 features to adapt its interface according to the size of the windows and the kind of control (local or remote). In this section, we outline interface and interaction of the three main versions (desktop, smartphone, and multi-device).

3.1 Desktop Version

The interface of the desktop version is shown in Figure 2. Since the control is local and the interaction occurs via mouse and keyboard, the window includes the displayed content along with commands. Users interacts with YouTube4Two like with any Web application. Figure 2 shows the interface split in different areas: the central area B shows the current video surrounded by the list of related videos on the

---

[1]Demo screencast available at bellinux.com/youtube4two

Figure 2: YouTube4Two interface. Content windows: related videos list (A); the shared video (B); comment list (C). Commands: search (D); connect (smartphone pairing) (E).

left (section A) and the list of comments uploaded by users (section C). The user can scroll the two lists to select the next video and read comments. Moreover, the hidden menu on the right (D in the figure) allows the user to search for videos. The control bar on the top of the interface includes a button on the right (E in the figure) that allows for enabling the multi-device configuration of YouTube4Two (see Section 3.3).
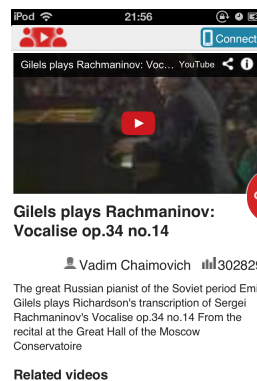

Figure 3: The YouTube4Two interface adapted to a smartphone display.

3.2 Smartphone Version

According to the Responsive Web Design principles [6, 7], the interface of Figure 2 changes its layout to fit smartphone displays as shown in Figure 3. The layout becomes a vertical list with the current video on the top, then some comments followed by a menu entry that enables the display of related video. To be consistent with the desktop interface, the connect and search commands are left in the same positions: top right corner and right side, respectively. Users can interact with the application with the usual gestures on touch-screens.

3.3 Multi-Device Version

The desktop and smartphone versions resemble typical interfaces and typical interaction paradigms, which do not requires further explanations. In this section we describe the innovative aspects of YouTube4Two: interaction by using two smartphones (multi-device, two-user configuration) to remotely control a shared interface. YouTube4Two is an ex-

ample of decoupling of content presentation from application control so that we can put them on two devices according to the multi-device responsive Web design (MD-RWD) that we proposed to support multi-user interaction by sharing the control through personal devices [2]. The basic principles enforced by MD-RWD are that (i) users should be able to operate the smartphone with only one hand (using the thumb to drag/swipe and tap) [8] without looking at it too often (the gaze should always be at the shared interface); and (ii) the effects of control operations are displayed on the shared interface to constantly create social awareness through explicit feedback.

Pressing the "connect" button on both the smartphone and desktop interfaces activates the multi-device configuration of YouTube4Two. After pressing the button on the desktop interfaces, a pairing number is displayed: it should be inserted into the smartphone to couple the two devices (see Figure 4). The same pairing number has to be used by the second user to connect the second smartphone.
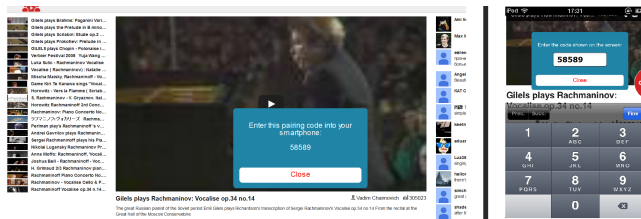

Figure 4: Registration of a smartphone as a remote control for shared application.

After pairing, the smartphones play the role of controllers and the controlled interface keeps the original layout with a small change in the control bar. The shared interface should look like the original one to create social awareness in the users: co-present users can act independently on the interface and must be able to easily follow their own moves and the moves of the others.

Figure 5: Control Bar with two users. The first user (blue) controls the "related videos" and the second user (green) controls the shown video.

To let each user freely interact with the application (scroll comments and related videos; look for videos) and create social awareness through visual feedback, we use two different colors to highlight the interactions performed by the two users. In the control bar of the shared interface (Figure 5), every user has a colored marker that can be moved horizontally to identify one of the three interface areas which is the target of the control gestures on the smartphone. In such a way both users are aware of the actions performed on the shared interface.

An issue is when both users decide to navigate the same interface area, e.g. the related comments. We introduce an enhanced version of popular fisheye menu [1]: a double-focus fisheye menu to provide visual feedback and give freedom of acting to users (Figure 6). Comments, related videos, and search results are considered as unlimited lists, therefore, if

users decide to focus on elements too far from each other, the size of the elements in between is shirked (HE in the figure) to keep the elements of interests at their original size (UF in the figure).
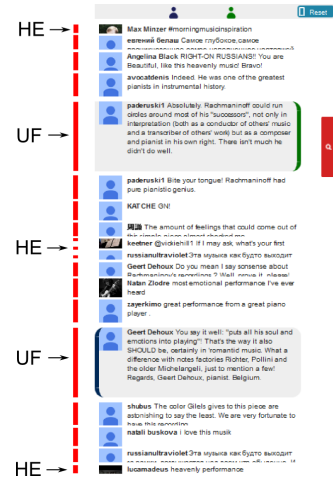

Figure 6: Fisheye effect on the comment list with focuses (UF) and hidden elements (HE).

In the multi-device version of YouTube4Two, the smartphone interface has been completely redesigned to accommodate the control commands to operate the application. The main interface is split in three areas: top, center and bottom (Figure 7). The bottom area is used to navigate the browser history by swiping horizontally (leftward/back, rightward/-forward). The top area is used to move the user marker on the control bar by dragging horizontally. To select a content area, the marker should be moved on top of it. In each area, the control interface in the middle area changes according to the supported action for the current content area. A user can (i) navigate the related video list, after positioning the marker on the left, by dragging vertically and possibly play one of them with a double tap (Figure 7, A1); (ii) control the current video, after positioning the marker in the middle, by tapping to play/pause the video and dragging horizontally to navigate the video back and forward (Figure 7, A2); (iii) browse the comments, after positioning the marker on the right, by moving vertically a finger in the central area of the control interface (Figure 7, A3); and (iv) search for a video by swiping from the right edge of the screen to open the search interface (Figure 7, B). Through the search interface, a user can perform searching by keyword, navigate the results by swiping vertically, possibly select a result by double tapping, and eventually close the search interface by swiping leftward.

Opening the search interface makes the application interface change: a new panel is opened to display the results (Figure 8). YouTube4Two forces users to share the same window to display the searching results. If both users perform a search at the same time, a straightforward last-come-last served policy has been enforced to display the last performed search that hides the previous one. Anyway, the system keeps the two actions and their effects independent from each other, which means that any of the two users can recall his/her own results by simply tapping on his/her smartphone.

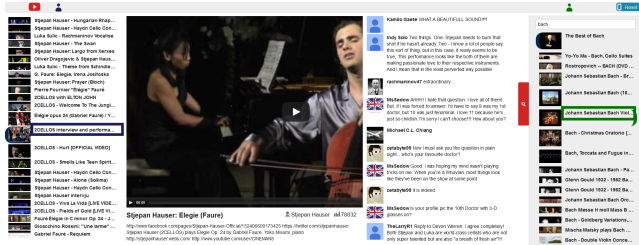**Figure 7: Control interfaces on a smartphone.**

Figure 8: Alice searches for a video (on the right search window), while Bob navigates the related video list.

Sharing windows carries a high risk of conflict. YouTube4Two does not impose any policy for managing conflicts to allow users to operate as desired. The interaction was designed with the idea that users could take advantage of co-presence to coordinate outside of the system by observing each other's behavior and by talking. So for example, Alice and Bob could agree to search in different areas as in Figure 8, and Alice should ask Bob before playing the found video.

4. THE APPLICATION ARCHITECTURE

A key choice for YouTube4Two is to rely only on standard technologies to be executable on virtually any device (with an Internet connection) equipped with the last version of any browser. It has been tested on the most popular desktop browsers (Chrome, Explorer, Firefox, Safari) and mobile operating systems (iOS, Android).

YouTube4Two is a browser-based JavaScript application that takes advantage of the HTML5/CSS3 features with media queries [9] to implement responsiveness, and WebSockets [3] technology to implement real-time full-duplex communication between the smartphones (acting as remote controllers) and the shared screen (the main application interface).
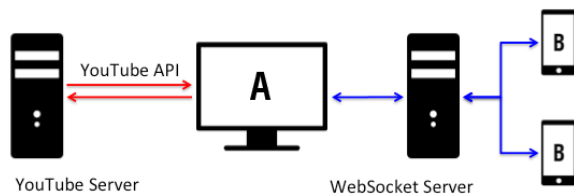

Figure 9: The architecture of YouTube4Two.

Figure 9 illustrates the system architecture after pairing the three devices. The main application running on the shared screen A interacts with the standard YouTube server

to retrieve the displayed contents, while receiving command events from the two controlling smartphones B through a dedicated real-time channel. Command events are generated by touch events [10] captured on the smartphones and transformed into application events according to the MD-RWD principles [2].

5. CONCLUSIONS

YouTube4Two is a demo prototype to show and experiment the new MD-RWD interaction paradigm that separates the content interface from the control interface to enable for multi-device Web applications. YouTube4Two shows that it is not a disruptive paradigm since classic RWD applications can be also accommodated. Moreover, the decoupling between the target application (YouTube in our case) and the interface and control shows that the proposed solution can be generalized to support a wide class of applications.

We are currently testing YouTube4Two and other prototypes to identify usability strengths and weaknesses of MD-RWD interaction paradigm. We think that freedom of acting (any user can take the initiative) and social awareness (every user can understand what is going on) are two important factors to foster the social involvement of users.

The goal is to consolidate user requirements and best practices in terms of interaction with the system and collaboration among users. In the meanwhile we are developing a framework to provide designers with a set of design patterns to support the most common applications.

6. REFERENCES

[1] B. B. Bederson. Fisheye menus. In *Proceedings of the 13th annual ACM symposium on User interface software and technology*, pages 217–225. ACM, 2000.

[2] A. Bellino, F. De Paoli, and G. De Michelis. Responsive web design for a marriage between large screens and smartphones. Submitted for publication, 2014.

[3] I. Fette and A. Melnikov. The websocket protocol. http://tools.ietf.org/html/rfc6455, 2011.

[4] Google. Youtube data api. https://developers.google.com/youtube/v3/, Nov 2013.

[5] E. Kurdyukova, M. Obaid, and E. André. Direct, bodily or mobile interaction?: comparing interaction techniques for personalized public displays. In *Proc. of the 11th International Conference on Mobile and Ubiquitous Multimedia*, page 44. ACM, 2012.

[6] E. Marcotte. *Responsive web design*. Editions Eyrolles, 2011.

[7] M. Nebeling and M. C. Norrie. Responsive design and development: methods, technologies and current issues. In *Web Engineering*, pages 510–513. Springer, 2013.

[8] K. Seipp and K. Devlin. Enhancing one-handed website operation on touchscreen mobile phones. In *CHI'13 Extended Abstracts on Human Factors in Computing Systems*, pages 3123–3126. ACM, 2013.

[9] W3C. Media queries. http://www.w3.org/TR/css3-mediaqueries/, Nov 2013.

[10] W3C. Touch events version 1. http://www.w3.org/TR/touch-events/, Nov 2013.