A visualization technique for quality control of massive digitization programs

Rodrigo Andrade de Almeida, Pedro Alessio, Alexandre Topol, Pierre Cubaud

Centre d'études et de recherches en informatique (CEDRIC)
Conservatoire National des Arts et Métiers (CNAM)
292 rue Saint-Martin, 75003 Paris, France
{rodrigo.almeida, pedro.alessio, topol, cubaud}@cnam.fr

Abstract. Massive digitization programs need massive visualization techniques for quality control. We describe the functional prototype of a 3D interactive environment enabling a rapid inspection of pages conformity for large batches of digitalized book.

Keywords: visualization techniques, massive digitization, image control quality.

1 Introduction

Massive digitization programs for books collections have been proposed by a few major organizations in the past decade. In Feb. 2008, the millionth volume has been digitized by the partnership between *Google* and the University of Michigan Library [1]. In Europe, similar projects are undertook under the umbrella of the European Commission, which funds the *Europeana* portal. To this day, about 70 european institutions collaborate to *Europeana*. The French national library, for instance, digitizes approximately 100K vol. per year.

These demands of public institutions have fostered the creation of specialized digitization centers that act as external contractors. The digitization process is in fact a pipeline, divided in the following tasks: (1) batch preparation, (2) scanning, (3) image processing, (4) OCR, (5) metadata production, and (6) ebook delivery. Statistical quality control is used for each step of the process. A random sample is drawn at each step. If rejected, the entire batch returns to the step before, requiring more human supervision [2]. A scanning operator can reach a speed of 700 pages/h. So, for one single digitization center with 5 to 10 scanning equipment, the productivity is typically around 1K vol./week. Under such workload, it is clear that the digitization process must be carefully designed: a good example is given in [1]. Unfortunately, this process is currently applied with difficulty to antiquarian collections, because of their inherent heterogeneity (mix of pages and plates, fragile bindings, erratic pagination and tables, etc.).

We report in this paper some preliminary results of the DEMAT-FACTORY project, a consortium of french industrial digitization companies and research
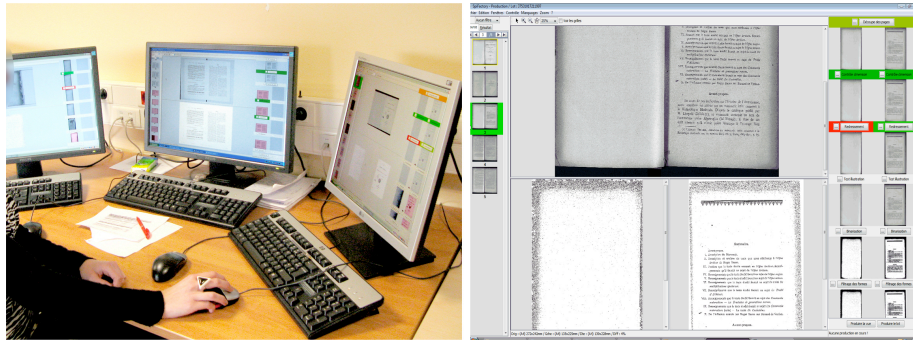
Fig. 1. Current imaging quality control in Demat-Factory : working space and interface

laboratories, partly funded by the State Dept. of Industry and the Paris region. The goal of this 3-years project is to study how to significantly upgrade the overall quality of digitization for large, heterogeneous, antiquarian book collections, while keeping the high workload already achieved for homogenous, contemporary printed works. Among the tasks under study are the visualization methods for quality control. We focus here on the steps (2) - (3): scanning and image processing.

Step (3) is divided into the following subtasks: (a) pages separation, (b) crop and dimension control, (c) skewness removal, (d) figures identification, (e) binarization, and (f) marks removal. It is conducted automatically, usually at night. During the day, an operator validates this step. Fig 1 shows a typical working space for the operator and a screenshot of the main interface for image control. The operator processes one volume at a time on each of the three screens. The upper central part of the interface shows the scanned picture and the lower part, what is obtained after (f). The user can navigate in these views simultaneously. Thumbnails of all the scanned pictures of the volume are shown on the left side, labeled with results from automatic analysis. On the right side, the operator can restart each step from (a) to (f). All the operations are time stamped and logged.

Only a dozen pages can be checked at a time on each screen, so knowledge about the digitized document is reduced. Thumbnails ease and accelerate checking many aspects of digitized images. In some situations, however, one must inspect the images using different zoom scales or in a different zoom area different than that that is proposed. In those cases, one has to load the full view image and that, besides being a more time-consuming operation, hinders the fluidity of the interaction (the visual context is abruptly changed).

We claim that massive digitizing pipelines require not only a sample browsing, but also an exhaustive browsing approach, where all images can be rapidly observed and inspected, using recent results in information visualization. In [3], authors argue that visualizations commonly only use one of the two following approaches:

- The *time based strategy* in which discovering data depends upon time dependant actions is commonly implemented in standard time-scrolling interfaces. Their main known issue is the abrupt change when switching from a context-browsing task to a detailed document reading.

- The *space strategy* in which information is packed in a static view is exploited by the *Space-filling thumbnail* technique that tries to solve the scrolling issue [4]. This kind of display technique reveals relations between documents with little
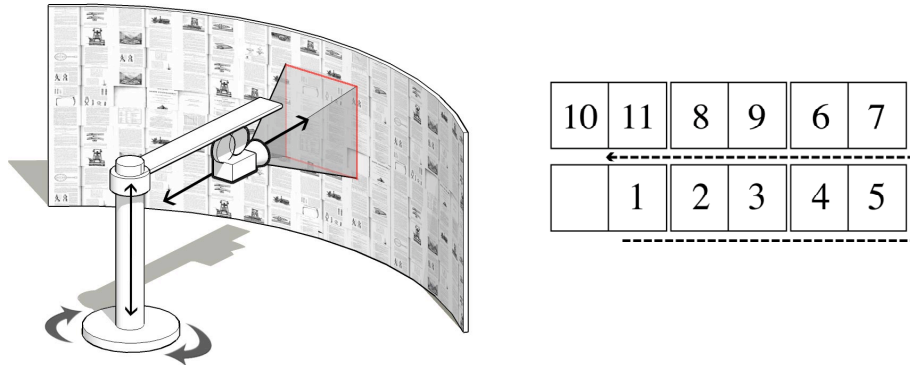
Fig. 2. On the left, 3D scene and camera movements metaphor. On the right, the layout of the displayed pages on the wall and the trajectory of a sequential navigation.

cognitive effort. Another advantage is the use of spatial memory for retrieving already visited resources thank to fixed positions.

The *Perspective Wall* uses both space and time based strategies to improve the task of going back and forth from detail to context in textual documents, with effortless and user-friendly techniques. With the same concern, the *Document Lens* offers dynamic zooming on page thumbnails while maintaining context awareness [5]. In [6], the same effect is achieved with a simple 3D perspective projection that offers simultaneously zooming on details and seeing near upcoming context. The *Photomesa* project also addresses the focus&context issue by combining the thumbnail approach with treemap zooming [7]. However, *Photomesa* was developed for pictures with no obligation to have relations with neighborhood elements whereas digitization-reviewing tasks need these relations.

3   A Pan and Zoom Paper Half Pipe

The Pan&Zoom Paper Half Pipe (*PZPHP*) is a simplified 3D environment in which scene and navigation are reduced to an essential set of elements and operations. Pages are arranged on a wall, which has a partial pipe shape. The user controls the camera (i.e., his or her point of view in the scene), going up and down (Y), rotating clockwise (CW) and counter-clockwise (CCW) (R_Y), and zooming in or out any page (Z). The camera spatial parameters are thus mapped to a subset of three degrees-of-freedom (DOF) sensed by the input device (figure 2, left). The goal of this interface is to provide rapid and high-resolution browsing of digitized book pages in order to detect problems in a given set of images. There are two important browsing strategies in the quality control task: **overview and detail** and **sequential browsing**. First, one must be able to see the whole set at a glance in order to visual anomalies or disparities (lighter or darker pages, larger or smaller margins). Then, displaying the details of "suspicious" pages or regions of pages must be done through a rapid operation. Second, one must be able to navigate through the volume following the order of the
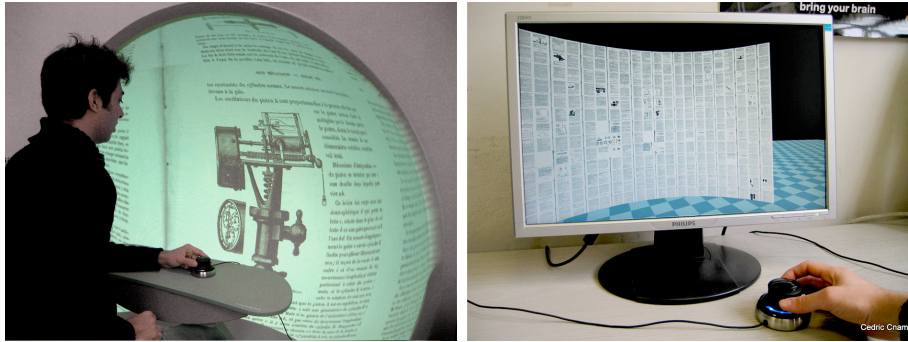
Fig. 3. The *PZPHP* interface displayed on an immersive display (left) and on a 22" screen (right).

pages. One may thus identify missing pages through visual disparities between one page and its neighboring pages (e.g., a paragraph that finishes on a given page and no new paragraph starts on the succeeding one).

3.1 Scene Design and Camera Control

The pages are mapped on the inner wall of a semi-cylindrical surface. This layout affords equal length trajectories from the center of the cylinder to any of the displayed pages on a given row. On a plane wall layout, one would navigate longer to attain the pages located on the boundaries of the wall. When the camera is at the center of the scene, all the pages are visible (the radius of the cylinder varies according to the number of pages being displayed). The longitude is the main navigation axis and it is dedicated to sequential browsing. Pages are nevertheless also tiled vertically in order to take advantage of free screen area when one zooms out. Moreover, panning vertically may still serve as a "random navigation axis", along which, one examines a random sample of pages. Scroll-based interfaces like *Adobe Acrobat Reader* support mainly the sequential browsing strategy, displaying the pages in a column layout.

Unlike other visualization interfaces in which vanishing points distort document boundaries [3, 6], in the *PZPHP*, every page is always orthogonal to the camera (when zoomed out, boundary images present a light distortion). Thus, pages are neither displayed tilted nor deformed by the camera perspective. That ensures that geometrical distortions will neither confound the user nor hide hints about potential problems in the digitized pages (whether the problem comes from the physical volume or from the digitizing process). This effect is achieved thanks to the cylindrical layout and to the fact that camera rotates not around its own center but around the scene's center.

As a user may want to uninterruptedly visualize all pages of the document, from the first to the last page, we have considered other scene topologies that would be better adapted to such operation. For instance, a complete cylinder with a helical distribution of pages would permit the user to, through a same continuous movement, browse all the displayed pages in a given order. Such configuration would nevertheless make the complete overview impossible and would provide very few landmarks (retrieving an already visited page would be harder). Figure 2 (right) illustrates the order in which the pages are tiled on the wall and the ideal path for one who wants to browse pages in the sequential order. We are implementing a "pan lift" mechanism in order to compensate the discontinuity of the pan movement in the semi-cylindrical navigation. This behavior would transport the camera to the "next" line when it reaches the end of a row during the pan (i.e., a CW pan would transport the camera to the above row when panning through an even row). A condition to trigger

the "pan lift" is the viewport being mainly filled by one row. This behavior would permit the user to concentrate on controlling important variable of the task, namely the pan speed.

3.2 Implementation and Users' Feedbacks

We used a *SpaceNavigator*[1] as the input device that controls the movements of the camera in the scene. The *SpaceNavigator* is an isometric joystick providing a tiny elastic feedback to movements along some axes. It consists in a small cylinder mounted on a heavy base. The fingertips can grasp the cylinder and it can be pushed, pulled, twisted, tilted, and rolled. Isometric devices are considered to be superior to position devices (e.g., mouse) in controlling, with higher precision and lower fatigue, trajectories of cameras in 3D scenes (and in typical document scrolling tasks as well)[9].

The prototype application is built using *C* and the *OpenGL* library. The master raw image of each page is cropped so as to present a 2:1 ratio and it is then sampled to seven smaller power-of-two dimensions (2048x1024 until 32x16 pixels). Power-of-two images take advantage of graphic memory space when they are loaded as textures. Moreover, they permit clear and fluid zooming thanks to the *Mipmap* functionality, which employs pre-calculated intermediary size images in order to reduce flickering and aliasing when the texture is scaled up or down in the viewport. During initialization of the application, the GIFs of the pages are all loaded in the graphic card's memory using texture compression functionality. As textures are all resident, there is no paging. Animation is thus completely smooth, even if the camera is rapidly moved from a 500 pages view to a high-resolution single-page detail view in less than two seconds. The final tiled wall is almost a gigapixel image – actually, 500 Mpixels. When zoomed out, pages become a visual texture; illustration zones, obscure pages, distinct zones, and other critical visual information may be rapidly spotted and inspected.

We have presented our prototype under two visual display device configurations: on a regular 22" screen and on a *Elumens VisionStation*, an immersive visual display device. The *VisionStation* is a hemispherical display that uses a projection-based hardware and software, for displaying images in 160 by 160 degrees field-of-view (FOV). It consists of a hemispherical surface (1.5 m diameter) attached to a small table. The user sits in front of this table where he may place the interaction input. A normal projector, coupled to a fisheye lens, is installed into the table. Such a lens provides a 180-degree horizontal and vertical projection distributing equally the same amount of pixels all over the projection surface.

A number of users (20 persons among librarians, specialists in the digitizing process, expert and novice users) have already tried our prototype; it was also presented at a digitizing hardware and services professional exhibition (FAN'2008, Paris). Many users, mainly those that are not used to 3D input devices, had a hard time in controlling the camera for zooming in and out the pages. The device's form factors and its poor elastic feedback seem to trouble most of them in performing the wanted manipulations. Moving the camera up and down, for example, is an uncomfortable manipulation - the device's top part barely moves when its pushed or

---

[1] A six degrees-of-freedom input device produced by *3D Connexion*.

pulled along that axis. Igarashi [8] describes the problem of the visual overflow generated by and accelerated scrolling display. Such problem is similar to the disorientation experimented by the users that tried the *PZPHP* when the camera is too close to the pages. A device-camera mapping that would reduce speed of yawing and vertical movements as the camera approaches the pages is right now being studied. Anyway, all users that were sensible to quality control issues were very interested in this interface and a major part of them found that it offers a value that they could not find in any other application that they had previously tried.

4 CONCLUSION AND FUTURE WORK

We described a first prototype of a visualization interface to support high-resolution browsing of large digitized page sets. Its response time has proved to be satisfactory. At the present stage, however, the software supports neither edition nor annotation. Thus, no real world test, in which a control quality user browses a set and tags defective images in a limited time, has yet been proposed. Besides implementing such functions, we would like to conduct a usability study that helps us to find a comfortable and efficient camera mapping. Moreover, we plan to make the prototype more scalable. We would like to design a resident memory manager that would load and unload high-resolution images according to their distance from the camera viewport.

References

1. http://www.lib.umich.edu/news/millionth.html

2. Riley, J., Whitsel, K., Practical quality control procedures for digital imaging projects. *OCLC Systems & Services* 2005. 21, 1. 40-48.

3. Mackinlay, J.D., Robertson, G.G., & Card, S.K. 1991. The perspective wall: detail and context smoothly integrated. CHI'91. ACM, 173-176.

4. Cockburn, A., Karlson, A., and Bederson, B. B. 2008. A review of overview+detail, zooming, and focus+context interfaces. *ACM Comput. Surv,* 2008, 41 (1), 1-31.

5. Robertson, G.G. and Mackinlay, J.D. 1993. The document lens. UIST '93. ACM, 101-108.

6. Guiard, Y., Chapuis, O., Du, Y., and Beaudouin-Lafon, M. 2006. Allowing camera tilts for document navigation in the standard GUI: a discussion and an experiment. AVI '06. ACM, New York, NY, 241-244.

7. Bederson, B. 2001. PhotoMesa: A zoomable image browser using quantum treemaps and bubblemaps. UIST '01, ACM Press,71–80.

8. Igarashi, T., Hinckley, K. 2000. Speed-dependent automatic zooming for browsing large documents, UIST'00, ACM, 139-148.