

# Panoramic Viewfinder: Shooting Panoramic Pictures With the Help of a Real-Time Preview

Patrick Baudisch, Desney Tan, Drew Steedly, Eric Rudolph,  
Matt Uyttendaele, Chris Pal, and Richard Szeliski

Microsoft Research, One Microsoft Way, Redmond, WA 98052, USA

{baudisch, desney, steadily, erudolph, mattu, szeliski}@microsoft.com, pal@cs.umass.edu

## ABSTRACT

Image stitching allows users to combine multiple regular-sized photographs into a single wide-angle picture, often referred to as a panoramic picture. During stitching users often discover that the produced panorama contains artifacts or is incomplete. Fixing these flaws requires retaking individual images, which is often difficult by this time. In this demo, we present Panoramic Viewfinder, an interactive system for panorama construction that offers a real-time preview of the panorama while shooting.

**Keywords:** Panorama, stitching, real-time, preview.

## INTRODUCTION

To allow digital camera users to take pictures with a viewing angle wider than the one supported by the built-in lens, researchers have proposed techniques for *stitching* multiple photographs or a stream of video into a single contiguous panoramic picture, or *panorama* [5]. Recent full-view stitching methods even allow automatic stitching of photos taken in arbitrary order and spatial arrangement [3]. Stitching is generally performed as a post-hoc step to picture taking.

Making flawless panoramas using a post-hoc stitcher can be challenging. We conducted an informal survey on an internal company mailing list for stitching users and received 26 responses out of 163 mailing list members. All except one participant had discovered at least one of the following flaws when stitching their pictures: (1) *Ghosting* (88% respondents): Objects that move between frames end up appearing translucent when the stitcher blends frames together. (2) *Missing content* (65% respondents): Relevant areas may disappear when cropping the panorama to its final rectangular shape (Figure 1). (3) *Stitching failed* (38% respondents): The stitcher did not have enough overlap between frames to connect them to the rest of the panorama. Fixing these flaws requires retaking images, but when users discover the errors during stitching, it is often too late.

While other researchers have looked at real-time previews generated using real-time stitching from a technical perspective [6], we address the above issues by providing users with a novel user interface that allows users to utilize these previews in order to verify which areas of the scene have been successfully covered. *Panoramic View-*

*finder* is designed to match the interaction model of existing digital cameras. In fact, we have constrained our designs to depend only on a mode selector and a shutter button so that it may be ported to utilize existing camera interaction mechanisms (Figure 2).



Figure 1: The problem: (a) The user trying to cover the Seattle Space Needle does not cover enough of the sky. During cropping, the user can either (b) include white fringes or (c) crop relevant content.

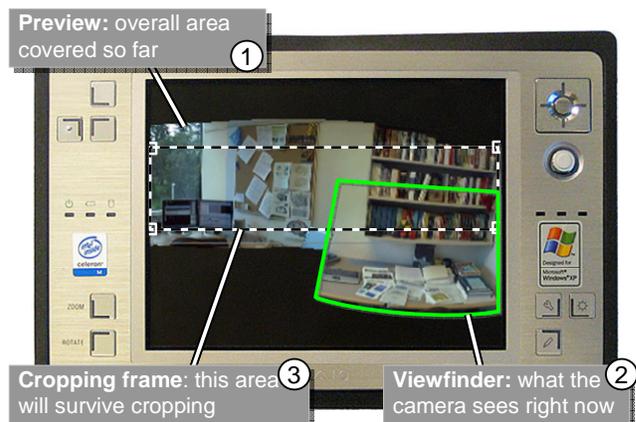


Figure 2: The Panoramic Viewfinder user interface on a Sony U50 ultra portable PC & web cam

## PANORAMIC VIEWFINDER WALKTHROUGH

Figure 3 illustrates the user's interaction with the Panoramic Viewfinder [2] using an example scenario. The user's task is to take a panoramic picture of the front of a building complex. (a) The LCD shows what the camera sees, just as in any other camera. (b) The user has pressed the shutter button and the camera has taken the first picture. From now on, the viewfinder is displayed in its relative location with respect to the panorama. (c) The user

pans the camera and takes additional pictures. Since these are automatically added to the panorama, the panorama grows. The panorama preview is automatically scaled to fit the screen. The cropping frame is also updated continuously. It always represents the largest rectangle that can be inscribed into the current panorama. Note how the cropping frame indicates that stopping now would cause part of the building to be removed during cropping! (d) The user takes an additional shot of the top left corner. This completes the bounding box and the cropping frame now contains the entire building complex. (e) When uploading the photo to the PC, the panorama is restitched using our high-quality offline stitcher [3] to obtain maximum image quality. When auto-cropping the panorama to rectangular shape now (users may manually overwrite that), all desired content is preserved.

During operation, the system matches every new image with the preview. If no match is found, the system changes the viewfinder frame from green to red and plays a continuous error sound. Users may then pan the camera back towards the last recognized shot.

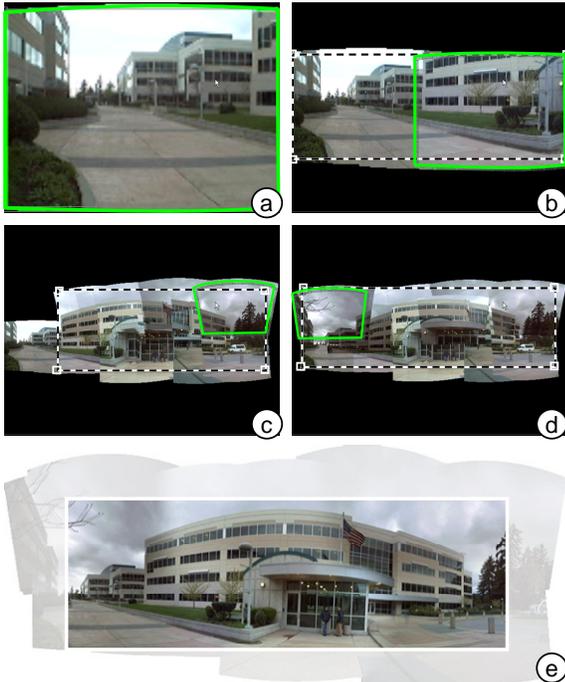


Figure 3: Panoramic Viewfinder walkthrough (actual screen captures enhanced to improve readability on b/w printouts)

In summary, Panoramic Viewfinder allows users to take pictures by first shooting the desired scene elements, then filling the bounding box around these elements. The key element in this process is the real-time cropping frame, which serves as a visual reference. The real-time cropping frame inverts the traditional cropping model: Traditional post-hoc cropping requires users to *narrow* the image down to what is desired; using the real-time cropping frame users keep adding content until the cropping frame has *grown* to cover the desired area.

## RELATED WORK

The concept of real-time stitching was first explored by the *VideoBrush* system [6]. While *VideoBrush* did not offer any user interface beyond displaying images on a computer screen, we propose a user interface concept to be added *on top of* real-time stitchers, such as *VideoBrush*. Some digital consumer cameras, such as the Canon Powershot series, help users take panoramas by showing the viewfinder next to part of the last frame taken (“stitch assist”). In contrast, Panoramic Viewfinder allows users to take content in random order. The HP Photosmart R707 camera allows users to review panoramas on the camera’s LCD screen in playback. This allows users to see flaws while still on-site, yet still requires retaking the entire panorama in order to fix flaws.

The interaction types embodied by Panoramic Viewfinder have been peripherally explored in small-screen device interaction research. For example, the Chameleon system [4] allows users to explore digital and physical spaces by moving a display around. Panoramic Viewfinder allows users to “brush” over the desired areas; similar interactions have been explored with paintable interfaces [1].

## IMPLEMENTATION

Our Panoramic Viewfinder prototype is written in C and runs under Windows XP with two front ends using GDI+ and DirectX9. Our prototype is loosely based on the Microsoft Research Image Stitcher [3] and shares some libraries with it. By allowing the addition of images without reprocessing the images already stitched, our prototype reaches frame rates of up to 4 frames/sec on a 2GHz tablet PC with 1GB of RAM [2]. We obtain slightly slower performance on an ultra portable 1 GHz Sony U50 (Figure 2).

## CONCLUSIONS

In this paper, we have presented Panoramic Viewfinder. While mainly designed to help users avoid flaws, showing users the result of their work right away offers “instant gratification”, which we believe is one reason why many users today prefer digital to traditional photography. As future work, we are planning to optimize our algorithms to increase the frame rate and to reduce accumulated error. We also plan on running a user study evaluating user’s performance and experience with Panoramic Viewfinder.

## REFERENCES

1. Baudisch, P. Don't Click, Paint! Using Toggle Maps to Manipulate Sets of Toggle Switches. *Proc. UIST '98*, pp. 65-66.
2. Baudisch, P., Tan, D., Steedly, D., Rudolph, E., Uyten-daele, M., Pal, C., and Szeliski, R. Panoramic Viewfinder: providing a real-time preview to help users avoid flaws in panoramic pictures. To appear in *Proc. OZCHI'05*.
3. Brown, M., Szeliski, R., and Winder, S. Multi-Image Matching using Multi-Scale Oriented Patches. *CVPR2005*.
4. Fitzmaurice, G. (1993). Situated Information Spaces and Spatially Aware Palmtop Computers. *CACM 36(7)*:38-49.
5. Mann, S. and Picard, R. (1994). Virtual bellows: Constructing high-quality images from video. *Proc. ICIP-94*, 363-367.
6. Sawhney, H.S., Hsu, S., and Kumar, R. Robust Video Mosaicing through Topology Inference and Local to Global Alignment. (1998). *Proc. ECCV'98*, 103-119.