Chapter 19
Video Streaming with Interactive Pan/Tilt/Zoom

Aditya Mavlankar and Bernd Girod

Abstract. High-spatial-resolution videos offer the possibility of viewing an arbitrary region-of-interest (RoI) interactively. The user can pan/tilt/zoom while watching the video. This chapter presents spatial-random-access-enabled video compression that encodes the content such that arbitrary RoIs corresponding to different zoom factors can be extracted from the compressed bit-stream. The chapter also covers RoI trajectory prediction, which allows pre-fetching relevant content in a streaming scenario. The more accurate the prediction the lower is the percentage of missing pixels. RoI prediction techniques can perform better by adapting according to the video content in addition to simply extrapolating previous moves of the input device. Finally, the chapter presents a streaming system that employs application-layer peer-to-peer (P2P) multicast while still allowing the users to freely choose individual RoIs. The P2P overlay adapts on-the-fly for exploiting the commonalities in the peers’ RoIs. This enables peers to relay data to each other in real-time, thus drastically reducing the bandwidth required from dedicated servers.

1 Introduction

High-spatial-resolution digital video will be widely available at low cost in the near future. This development is driven by increasing spatial resolution offered by digital imaging sensors and increasing capacities of storage devices. Furthermore, there exist algorithms for stitching a comprehensive high-resolution view from multiple cameras [1, 2]. Some currently available video-conferencing systems stitch a large panoramic view in real-time [3]. Also, image acquisition on spherical, cylindrical or hyperbolic image planes via multiple cameras can record scenes with a wide

---

Aditya Mavlankar · Bernd Girod
Stanford University, 350 Serra Mall, Stanford CA 94305, USA
e-mail: maditya@stanford.edu, e-mail: bgirod@stanford.edu
field-of-view while the recorded data can be warped later to the desired viewing format [4]. An example of such an acquisition device is [5].

Imagine that a user wants to watch a high-spatial-resolution video that exceeds the resolution of his/her display screen. If the user were to watch a downsampled version of the video that fits the display screen then he/she might not be able to view local regions with the recorded high resolution. A possible solution to this problem is a video player that supports interactive pan/tilt/zoom. The user can thus choose to watch an arbitrary region-of-interest (RoI). We refer to this functionality as interactive region-of-interest (IRoI). Figure [1] shows screen-shots of a video player supporting IRoI. Such a video player could also offer to track certain objects, whereby the user is not required to control pan and tilt, but could still control the zoom factor.

Some practical scenarios where this kind of interactivity is well-suited are: interactive playback of high-resolution video from locally stored media, interactive TV for watching content captured with very high detail (e.g., interactive viewing of sports events), providing virtual pan/tilt/zoom within a wide-angle and high-resolution scene from a surveillance camera, and streaming instructional videos captured with high spatial resolution (e.g., lectures, panel discussions). A video clip that showcases interactive viewing of soccer in a TV-like setting can be seen here [6].

Consider the first example mentioned above, i.e., playback from locally stored media. In this case, the video content is encoded offline before storing it on the relevant media, for example, a high-capacity portable disk. Note that the RoI trajectory is not known while encoding the content. An RoI trajectory is determined each time a user watches the video with interactive pan/tilt/zoom. This leads us to two design choices; 1) the video player can be designed to decode the entire high spatial resolution while displaying only the RoI or 2) the adopted compression format could allow decoding only relevant regions, possibly with some overhead. Depending on the resolution of the video and the hardware capability of the player, the first design choice might be prohibitive. Other application scenarios mentioned above entail streaming from a remote source. In most cases, streaming the full spatial extent of the video to a user can be ruled out due to prohibitive bandwidth requirement. If RoI-specific portions can be streamed to the remote user, the RoI dimensions could be adapted to suit the available data rate for communication apart from the user’s display screen as noted above.

Now let us consider the difficulty of employing a standard video encoder in the streaming scenario. A standard video encoder generally does not provide efficient spatial random access, i.e., the ability to extract regions from the compressed bitstream. The video streaming can be for live content or for pre-stored content. For live content, the server can crop out an RoI sequence on-the-fly considering the user’s pan/tilt/zoom commands and compress it as a video sequence using standard video encoding. The load of encoding might get prohibitively large with increasing numbers of users. Pre-stored content might not be stored in raw format implying that the server has to decode the high-spatial-resolution video prior to cropping the RoI sequence. Not only does the load of encoding increase, but if multiple users watch the content asynchronously then even the decoding load at the server increases. On the