Display Stream Compression for VR Media

Neil Movva*
Stanford University
Department of Electrical Engineering
nmovva@stanford.edu

Abstract

VR head-mounted displays (HMDs) currently depend on high-bandwidth physical datalinks to transmit high-fidelity images at fast frame rates. These display streams are currently too bandwidth intensive to be served by wireless interfaces, so we propose an adaptive compression scheme that more intelligently compresses the display stream to maximimize perceptual quality, subject to a range of bandwidth constraints. We find that perceptual display stream error can be meaningfully reduced versus traditional uniform compression systems, such as H.264, especially at higher bandwidth availability. Finally, we conclude with discussion on the potential for near-term wireless HMDs.

1. Introduction

We present novel techniques in the use of lossy video compression, targeted for use in VR display devices. We aim to achieve significant display stream bandwidth reductions while maintining high perceptual fidelity, evaluated with respect to the unique demands of VR applications. In particular, we significantly reduce bandwidth requirements between the head-mounted display (HMD) and the rendering host.

1.1. Motivation

Modern VR headsets demand extremely high resolution displays and fast refresh rates to deliver a perceptually fluid experience. As such, current headsets must be tethered to powerful computer systems via expensive high-bandwidth interfaces, sacrificing mobility in order to achieve acceptable visual fidelity. Implementing visually lossless display stream compression is important to enable wireless and potentially even network-based connections to these rendering systems, decoupling VR experiences from necessary support infrastructure.

*WIM Note - Primary author of this report. Collaborated with Stephen Lopez [slopez27@stanford.edu] for technical implementation of work.

Furthermore, the VR HMD provides significantly more information about viewer behavior than the traditional desktop monitor interface. Compression systems are generally unable to guess on the 'importance' of data, but we exploit a variety of prior assumptions to selectively weight some information with higher importance, i.e., perceptual significance to the viewer.

1.2. Related Work

General Video Compression

Video compression is a common task that is generally essential to modern media consumption. Industry leaders have largely converged on a few popular compression schemes for general distribution of video, such as H.264 and HEVC (H.265) [2]. While these codecs are rigorously designed to be extremely efficient in terms of reproduction quality and computational execution (especially when considering ubiquitous availability of hardware accelerated implementations), they are generally unable to infer relative importance of entities in the frame and must compress uniformly. Instead, recent work suggests dynamically prioritizing scene elements according to estimates of the viewer's preceptual focus [7]. Furthermore, work from the computer vision domain suggests applying additional inference on scene content to better decide on relative importance [10] [8] [5].

Panoramic Video

Panoramic video is a special use case for VR that consumes even more bandwidth, yet it is immediately obvious that the user cannot capture all the information presented (i.e., the human field of view is significantly smaller than the 360° panorama). Furthermore, humans are known to exhibit fairly predictable (or at least, high temporal locality, i.e. low tendency to jump between perspectives quickly) exploration patterns in VR spaces [9]. These observations suggest especially attractive compression opportunities in the delivery pipeline of panoramic VR content. However, we intend

to focus on display stream compression, which lies downstream of these optimizations - in particular, view frustrum culling removes most extraneous information shortly before the final framebuffers are painted.

VESA Display Stream Compression

Display stream compression has become a recent industry objective as display resolutions continue to increase. The VESA display standards body has thus published an official specification for 'visually lossless' Display Stream Compression (DSC), first implemented in embedded DisplayPort 1.4. While details on the final DSC algorithm are available only to VESA members, high level whitepapers available on the topic suggest that DSC is designed to minimize cost and complexity of hardware codec implementations, potentially at the expense of compression efficiency (DSC targets a 3X bandwidth reduction). Additionally, the DSC algorithm attempts to maintain fidelity uniformly over the image, lacking saliency estimation.

Adaptive Compression

'Salsify' is a recent work that aims to optimize video streaming in unpredictable network conditions by communicating bandwidth availability to the compression engine [6]. This principle of dynamically optimzing for the endpoint user experience aligns well with our VR-specific goals, and we foresee similar issues with bandwidth variance in wireless VR environments. In our implementation, we stress rapid codec adaptation to meet stringent latency requirements.

1.3. Our Contributions

We present a solution that integrates many of the techniques mentioned, which have been heretofore unlinked or unspecialized for the VR use case. We achieve a significant reduction in bandwidth HMD-host bandwidth, demonstrating a viable path to untethered VR HMDs. In addition to the techniques well-explored in prior works, we carefully optimize for the specific constraints of the EE267 HMD platform.

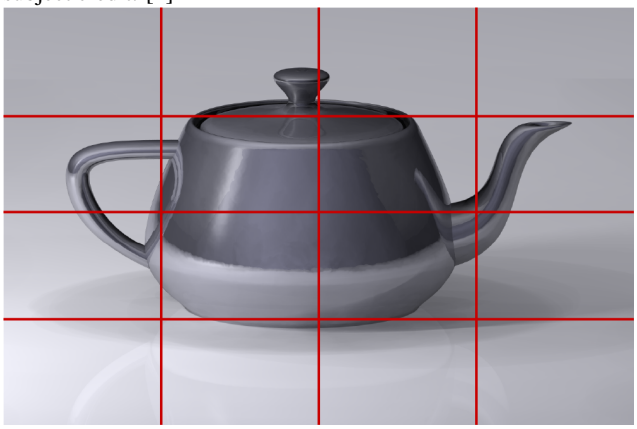
2. Methods

Our technique can be described in two distinct stages. First, we aim to generate a weighted map of perceptual significance over the display frame, scoring each region corresponding roughly to the viewer's immediate attentiveness it. Intuitively, high-scoring regions represent high viewer scrutiny and thus little room for error in image reproduction. Second, we use this significance map to apply compression of varying 'strength' (i.e., trading off bandwidth reduction for image accuracy) across the display frame. In this compression stage, we attempt to minimize perceived error (defined using objective error metrics such as PSNR, then weighted by the perceptual significance map) while satisfying the hard constraint of available bandwidth.

2.1. Stage 1: Perceptual Significance Map

In order to apply differential compression effectively, we impose certain prior assumptions to estimate which areas of the display frame are most important to the viewer's experience. Together, these assumptions yield a map of visual importance scores, which we use in the next stage to penalize image errors incurred by lossy compression differently throughout the frame.

Figure 1. Example of subview map. For simplicity of implementation, the number and arrangement of subview zones is static (and in this case, regular). Future work may consider dynamically retiling these zones to achieve a better fit to scene content. Teapot subject credit: [4]



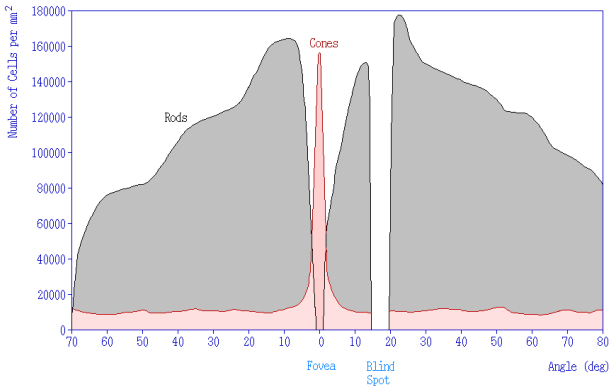
2.1.1 Foveal Prioritization

Given some estimate on the viewer's gaze direction, we assign high importance scores to the regions directly in the gaze path, and decay importance radially outward from this center. The human visual system exhibits peak acuity in this central cone, known as the 'foveal zone' after the retinal fovea (a high-density region of cone cells in the center of the retina).

Outside the foveal zone, we can be much more aggressive in pushing the limits of tolerance to compression artifacts. However, we must take great care not to expose any artifacts to the viewer's foveal gaze; even brief glimpses of strong compression artifacts can significantly degrade the percieved image.

While we do not present any novel gaze tracking solution in this work, the present authors are broadly aware of the current limitations in viewer gaze tracking. In particular, we cannot rely on especially high precision or low

Figure 2. Plot of cone density across the retina [3]. Note the narrow foveal zone, containing a majority of the total cone cells.



latency from current commercially available eye trackers, and must therefore be more liberal in assigning importance scores (i.e., assign high scores to a wider area and decay importance more slowly from the gaze center). This allows for minor misses in gaze tracker accuracy, but somewhat reduces our opportunities for compression. Future advances in gaze tracking accuracy will allow us to assign a tighter distribution of importance scores.

2.1.2 Lens-Matched Compression

VR HMDs rely on near-eye lenses to extend the display's apparent field of view. However, these lenses introduce distortion to the optical path, which we correct for during rendering by pre-warping the output image.

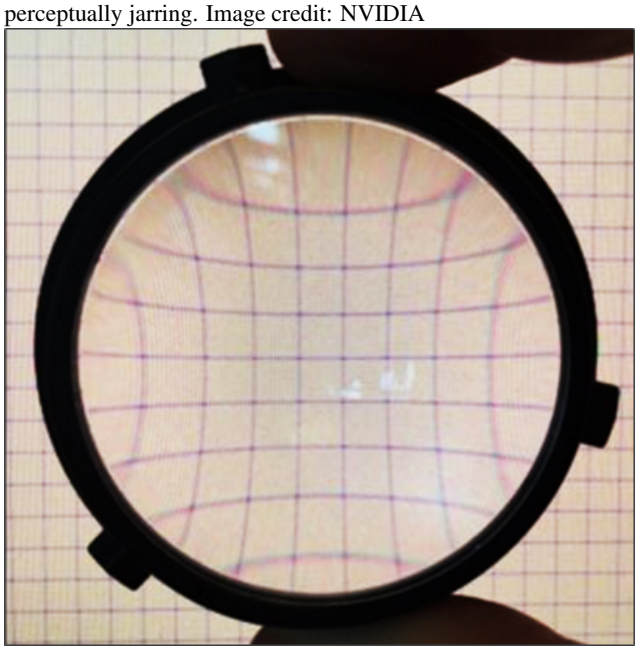
As a result, the 'effective' resolution of peripheral screen areas is reduced, as virtual fragment elements are rasterized to a smaller number of physical pixels. Other lens non-idealities can further degrade peripheral acuity. Thus, we assume higher tolerance to image errors in these peripheral zones.

2.2. Stage 2: Differential Compression

After generating a map of perceptual importance scores, we are left with a constrained optimization problem - to find the set of compression algorithm parameters that consumes no more bandwidth than can be carried over the current display link, while also minimizing perceptual error.

An ideal solution to this problem might leverage our known error tolerance map to compress the entire scene image in a single pass. However, presenting an entirely novel and competitive compression algorithm incorporating this feature is well outside the scope of this work, and so we rely on existing lossy compression schemes that have wide modern adoption. In particular, we use the H.264 algorithm as our core compression engine, and vary parameters to this

Figure 3. Representative image of HMD lens distortion. Note the peripheral distortion, where otherwise sharp lines are blurred and expanded - compression artifacts in these areas will likely be less perceptually jarring. Image credit: NVIDIA



algorithm in order to explore the bandwidth-error tradeoff space.

To establish a baseline proof-of-concept for our differential compression proposal, we make some further simplifying decisions. Instead of iteratively exploring the continuous space of possible compression parameters, we choose a discrete set of 4 parameters that map to predictable and consistent bandwidth demands, spanning a 20x range of bitrates. Thus, our original problem is much simplified to choosing one of these points for each subview, and we can estimate bandwidth savings before completing the compression step.

2.3. Testing Environment

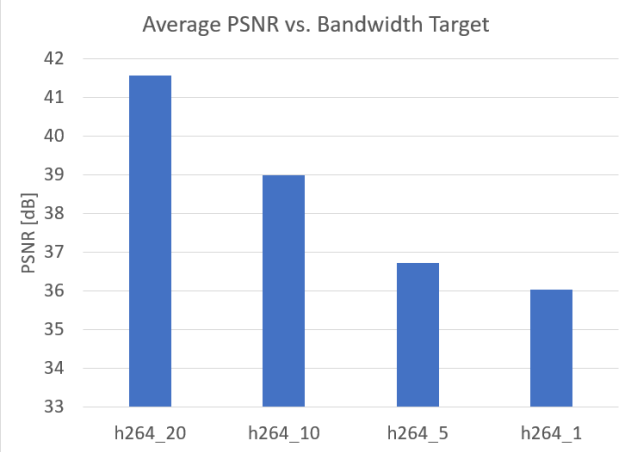
2.3.1 EE267 VR Rendering Engine

We use the JavaScript and WebGL-based rendering framework developed by the EE267 course staff [1] to render a representative VR display stream.

2.3.2 The VRDuino Platform

Our work targets the VRDuino HMD platform, which follows the specifications given in the Stanford EE267 course materials [1]. In particular, this HMD offers a 960x1080 viewport resolution for each eye, 9-DoF absolute orientation tracking, and positional tracking updates at 60Hz provided by an implementation of HTC/Valve's 'Lighthouse'

Figure 4. Graph of average PSNR at each compression optimization point. On the horizontal axis, the number after the underscore represents the target bitrate in megabits per second.



system.

3. Results

3.1. Uniform Compression Baseline

To establish a baseline target, we first capture the display stream using a set of H.264 parameters designed to capture most of the 'obvious' compressibility in the image, without introducing perceptible error anywhere in the frame. We accomplish this by using the default parameters for 'visually lossless' quality in the Handbrake encoding suite, resulting in a 'golden standard' display stream with an average bitrate of 50 megabits per second.

3.2. Perceptually Lossless Compression

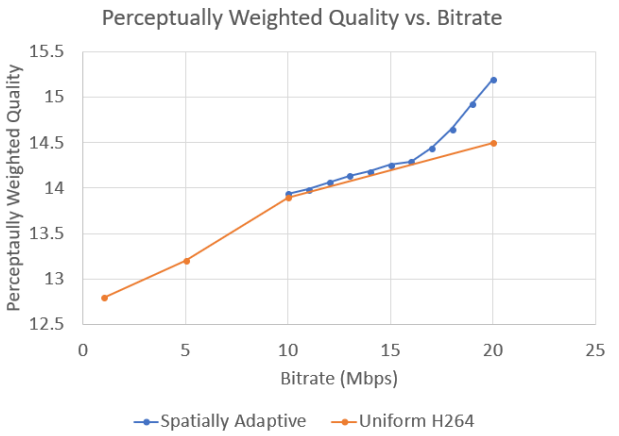
Our heuristic significance map produces relative weights that do not correspond to any globally consistent quality metric. As a result, we assess the semantic meaning of our arbitrary 'weighted quality' through informal user consensus. Generally, users complain about excessively distracting artifacts when the perceptually weighted quality score drops below 14.

3.3. Comparison to Uniform H.264

To evaluate our methods, we sweep across a series of bandwidth targets and measure the performance of our two stage algorithm against uniform H.264 compression.

We observe that our algorithm is able to achieve a significant perceptual quality advantage over uniform H.264 when less bandwidth constrained, largely by applying strong compression to peripheral / non-foveal zones and reinvesting the bandwidth savings into the highly weighted regions.

Figure 5. Graph of perceptual quality subject to various bandwidth constraints.



However, as bandwidth requirements tighten, our methods struggle to significantly improve on uniform H.264.

4. Discussion and Future Work

Our work demonstrates a novel system of applying known compression techniques in a spatially adaptive manner. Our primary contribution is the proposal of a two-stage architecture, cleanly dividing our goal into two tasks: first, to determine importance of scene elements; and second, to use the importance map to tune compression strength throughout the image. Performance on these two tasks can be improved independently in future work.

4.1. Pathway to Untethered VR

The base EE267 HMD platform demands a total display bandwidth of roughly 3 gigabits per second (1920x1080 resolution, 24 bits-per-pixel, 60Hz refresh rate). Modern handheld wireless interfaces, such as 802.11ac in a single antenna configuration with 80MHz channel width, can achieve theoretical data link rates of up to 433 megabits per second, but can experience significant variability in actual throughput between frames. As a result, the most important feature needed to deploy this work in a wireless HMD is the low-latency adaptation to bandwidth constraints, which we believe can be done more effectively using a relatively temporally stable importance map.

References

[1] EE267: Virtual reality, 2018.

[2] A. Banitalebi-Dehkordi, M. Azimi, M. T. Pourazad, and P. Nasiopoulos. Compression of high dynamic range video using the HEVC and h.264/AVC standards. 2018.

[3] J. Burghardt. Cone density plot, 2013.

[4] Dhatfield. The utah teapot, 2008.

[5] T. Dumas, A. Roumy, and C. Guillemot. Autoencoder based image compression: can the learning be quantization independent? 2018.

[6] S. Fouladi, J. Emmons, E. Orbay, C. Wu, R. S. Wahby, and K. Winstein. Salsify: Low-latency network video through tighter integration between a video codec and a transport protocol. In *15th USENIX Symposium on Networked Systems Design and Implementation (NSDI 18)*, pages 267–282. USENIX Association, 2018.

[7] G. Illahi, M. Siekkinen, and E. Masala. Foveated video streaming for cloud gaming. 2017.

[8] R. Sindhu. A feature based approach for video compression. 2016.

[9] V. Sitzmann, A. Serrano, A. Pavel, M. Agrawala, D. Gutierrez, B. Masia, and G. Wetzstein. How do people explore virtual environments? 2016.

[10] L. Zhao, H. Bai, A. Wang, and Y. Zhao. Learning a virtual codec based on deep convolutional neural network to compress image. 2017.