

Scalable Video Coding Using Wavelets

Sai Karthik Vuppalapati (09d07050) Satya Naren (09d07051)
Swrangsar Basumatary (09d07040)
Group 21

April 2013

Abstract

This is a report for the Scalable Video Coding Using Wavelets application assignment in the course EE 678 Wavelets. Our goal in this assignment is to demonstrate the spatial aspect of video scalability by encoding different spatial video resolutions into a single bitstream in a scalable manner.

1 Introduction

Spatial Scalability within the H.264/AVC Scalable Video Coding Extension has already been demonstrated[2] by well-known organizations like Motion Pictures Experts Group(MPEG). But here we are trying to demonstrate spatial scalability using wavelets. Wavelets have the beautiful property of being inherently scalable. Many researchers are trying to harness the scalable property of wavelets to improve the efficiency of scalable video coding techniques.

In this experiment, we decided to use *three* different spatial resolutions for demonstration. We encoded the three different resolutions at the transmitter side into a single bitstream and transmitted it. At the receiver, we decoded the bitstream to get the resolution required.

The YUV format is known to be more efficient than RGB. We used YUV format for our video frames. Every frame has a Y, U and V component. The size of U and V components were chosen to be half the size of the Y component. This is because the U and V

components carry less information compared to the Y component meaning they can be sampled at a relatively lower rate.

2 Encoding the video

At first, we reshaped every frame of the video into a $2^N * 2^N$ square frame where,

$$N = \text{floor}(\log_2(\max(\text{size}(\text{frame}))))$$

Then we did the following to each frame:

1. We did a 2-level 2-D wavelet decomposition to get the subbands $A2$, $H2$, $V2$, $D2$, $H1$, $V1$ and $D1$.
2. $A2$, the approximation subband, is our base layer. $H2$, $V2$ and $D2$ forms our first enhancement layer. The second enhancement layer is made up of $H1$, $V1$ and $D1$.
3. Then we compressed each layer using SPIHT[1] algorithm and quantized them.

Quantizing the subbands

We quantized the compressed layers to achieve a peak Signal-to-Noise ratio (PSNR) that is above a user-specified minimum value. The number of quantization levels is chosen so that the PSNR requirement is satisfied. We start from two quantization levels and keep doubling the number of levels as long as PSNR

is below user-specified value. This loop is finite because PSNR increases with the number of quantization levels. Of course, the user-specified value should be below the maximum possible PSNR value, or else we would run into an infinite loop.

3 Transmission

We encoded all the quantized layers of each frame using Run Length Encoding.

We then arranged every encoded frame such that for every frame the data was arranged in the following order: the base layer, the first enhancement layer and then the second enhancement layer. Then we combined all the frames of the video into a single bitstream.

For demonstration purposes, we wrote the encoded frames into binary files instead and read them back at the receiver.

4 At the receiving end

At the receiving end, if we want the smallest of the three pre-decided resolutions then we retain only the base layer data of each frame. And if we want the intermediate resolution then we retain the first enhancement layers also. For the maximum resolution, we retain all the three layers.

In the experiment, we reconstructed all the three video resolutions.

Reconstructing the video

First, we run length decoded the retained data. Then we uncompressed them using SPIHT[1] algorithm.

- For the lowest resolution video, we just uncompressed the compressed base layers.
- For the medium resolution video, we added the uncompressed first enhancement layers to the base layers through 2-D wavelet decomposition.
- For the highest resolution video, we added the uncompressed second enhancement layers to the

medium resolution frames of the medium resolution video, again, using 2-D wavelet reconstruction.

5 Comparing scalable video to single-layer bitstream and simulcast video

In single layer approach only one layer ,the highest resolution layer, is transmitted without any scaling. And in simulcast all the three different resolutions are transmitted together in a single bitstream without any scalability among the different resolutions. In single layer approach and simulcast, we don't have any sort of enhancement layers and all the layers are self-contained.

Therefore single layer approach is supposed to be faster than the scalable approach whereas the simulcast is supposed to be slowest of them all.

To show that the speed of the scalable approach lies in between that of single layer approach and simulcasting, we demonstrated the single layer approach and the simulcast approach also.

Single layer approach

We compressed the highest resolution layer of each frame using SPIHT algorithm and quantized them. Then we did a run length encoding of the frames and transmitted them.

At the receiving end, the incoming bitstream was run length decoded and uncompressed to get back the output video frames.

Simulcasting

Here we compressed all the three different resolution frames using SPIHT algorithm, and then quantized, run length encoded and transmitted them.

At the receiving end, we run length decoded and uncompressed the frames using SPIHT. The output was the video with the required resolution.

Observation

It turned out that the scalable approach required a bitrate higher than that of single-layer approach and lower than that of simulcast approach.

6 Summary

Though we demonstrated the spatial scalability of video for only three different resolutions. This concept can be expanded to any number of resolutions. It is easier to implement dyadic scaling because 2D wavelet decomposition directly gives you a dyadically scaled down version of the original frame. Though there is a change in the brightness level of the down-scaled frame, it can be rectified by multiplying the frame with the appropriate scaling factor.

Further, the bitrates can be improved by using better compression and quantization techniques.

References

- [1] Amir Said and William A. Pearlman. A new, fast, and efficient image codec based on set partitioning in hierarchical trees. *IEEE Transactions on Circuits and Systems for Video Technology*, 6(3), 1996.
- [2] C. Andrew Segall and Gary J. Sullivan. Spatial scalability within the h.264/avc scalable video coding extension. *IEEE Transactions on Circuits and Systems for Video Technology*, 17(9), September 2007.