# A Backward Compatible HDR Encoding Scheme

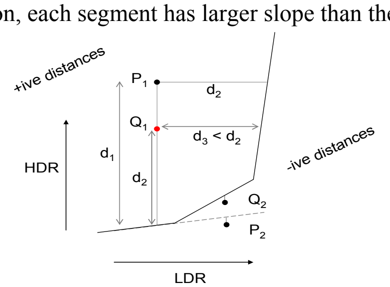

Ishtiaq Rasool Khan
A*STAR Institute for Infocomm Research†, Singapore


Two-layer encoding schemes for HDR images (and video) can not only reduce the storage requirements, but more importantly they can also ensure backward compatibility during transition from LDR to HDR age. The first layer is a tone-mapped LDR image, which can be shown on existing displays. The second layer is another LDR image and contains the residual information lost in tone-mapping, which can be used by HDR applications.

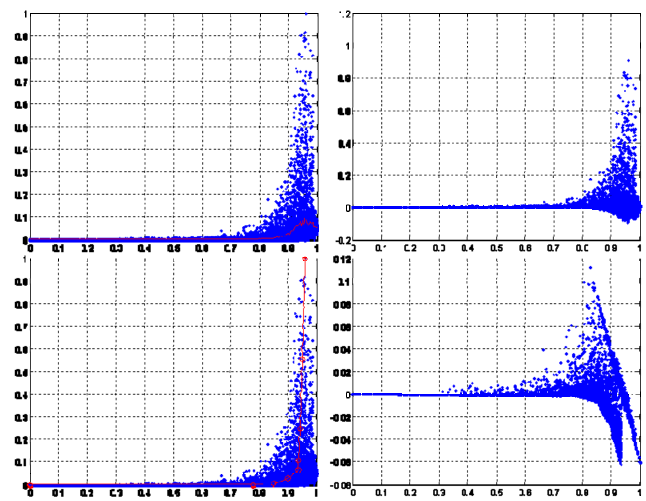
In a scheme proposed by Ward et al [Ward05], the second layer is ratio of the original HDR image to the first layer LDR image. Mantiuk et al [Mantiuk06] used a reconstruction function that approximates the HDR image as function of the first layer image, and the residual information is stored in the second layer. Mantiuk et al noted that using the ratio image by Ward et al was equivalent to approximating the HDR image as a linear function (which did not fit the data very well) of the front layer LDR image. They showed that their reconstruction function which consists of 256 x-y points in which y is the average of all HDR values that map to a single LDR value x, better fits the data. At expense of a slight overhead of the reconstruction function, details of the HDR image can be better preserved at the same bit rate. Okuda et al [Okuda07] achieved further improvement by using a more compact reconstruction using Hill function that fits the data optimally in least square sense.

Considering each column of the HDR vs LDR plot as 1-D sequence, the effect of using reconstruction functions is simply shifting the origin of the sequence from 0 to the value of the reconstruction function $R_0$ in that column. This does not actually reduce the dynamic range of the data. The data in $[R_{min}, R_{max}]$ range is simply shifted to $[R_{min} - R_0, R_{max} - R_0]$ range. However, if the reconstruction function fits the data well, most of the values in the second layer image will be smaller after the shift, and therefore can be represented more accurately in the same number of bits. The better the reconstruction function fits the data, better would be details preserved in the second layer.

Our method is inspired by the methods mentioned above. Starting with the reconstruction function used by Mantiuk et al, but instead of keeping average values for all 256 bins, we find their piecewise linear (PWL) approximation. A few key-points cleverly selected and joined by linear segments can represent the function quite accurately. This step is meant not just to reduce the storage overhead, but more importantly, PWL approximation helps in reducing the dynamic range of the second layer image. In our approximation, each segment has larger slope than the previous.




Now consider a point $P_1$ shown in figure above. It is at distance $d_1$ from the segment right below it. Mantiuk et al and Okuda et al store this distance in the second layer. It can however be noted that $d_2$, the distance of the point from the segment on its right, is smaller than $d_1$. We store this distance $d_2$ in the second layer. While reversing the process to retrieve actual distance, the first guess would be $Q_1$, which is above the segment below by distance $d_2$. However $Q_1$ has a distance $d_3$ from the segment at right which is smaller than $d_2$, and that disqualifies $Q_1$. Therefore, the point will be moved to the next option $P_1$. This procedure works for all points lying above the curve, because the slopes of the linear segments are in increasing order. For points lying below the curve, this does not work. For example, both $P_2$ and $Q_2$ will have same value stored in the second layer. This is because the segments can extend in the space below the curve and intersect each other. As a result, a point can be at the same vertical and/or horizontal distance from multiple lines. Therefore, for the points below the curve, we store the actual vertical distances. The only exception is the last segment. The points below it can be identified uniquely and restored irrespective of which distance (vertical or horizontal) is stored. We assign positive sign to the distances above or left of the curve and negative to those below or on right side of it.



The figure shown above gives a comparison of our proposed scheme (bottom) with that of Mantiuk (top). HDR luminance as function of LDR luminance for memorial church image and the reconstruction functions (red curves) are shown at left. Note that our function consists of very few points (8 in this example) compared to 256 points of Mantiuk. The residual data that need to be stored in the second layers are shown at right. Note that the dynamic range of our second layer is several times lower than that obtained by Mantiuk's method. Therefore, the information can be stored more accurately (smaller rounding error) in our scheme.

## References


[Ward05] WARD G., AND SIMMONS M. 2005. JPEG-HDR: A Backwards-Compatible, High Dynamic Range Extension to JPEG. *Proceedings of the Thirteenth Color Imaging Conference.*

[Mantiuk06] MANTIUK R., EFREMOV A., AND MYSZKOWSKI K. 2006. Backward Compatible High Dynamic Range MPEG Video Compression. *SIGGRAPH.*

[OKUDA07] OKUDA M., AND ADAMAI N. 2007. Two-Layer Coding Algorithm for High Dynamic Range Images based on Luminance Compensation. *Journal of Visual Communication and Image Representation 18*, 5, 377-386.



†e-mail: irkhan@i2r.a-star.edu.sg



SIGGRAPH 2010, Los Angeles, California, July 25 – 29, 2010.
ISBN 978-1-4503-0210-4/10/0007