Expanding Low Dynamic Range Videos for High Dynamic Range Applications

Francesco Banterle*
University of Warwick

Patrick Ledda
University of Warwick

Kurt Debattista
University of Warwick

Alan Chalmers
University of Warwick

Abstract

In this paper we introduce an algorithm and related methods that expand the contrast range of Low Dynamic Range (LDR) videos in order to regenerate missing High Dynamic Range (HDR) data. For content generated from single exposure LDR sequences, this is clearly an under constrained problem. We achieved the expansion by inverting established tone mapping operator, a process we term inverse tone mapping. This approach is augmented by a number of methods which help expand the luminance for the required pixels while avoiding artifacts. These methods may be used to convert the large libraries of available legacy LDR content for use, for instance, on new content-starved HDR devices. Moreover, these same methods may be used to provide animated emissive surfaces for image based lighting (IBL). We demonstrate results for all the above applications and validate the resultant HDR videos with original HDR references using the HDR Visual Difference Predictor (HDR-VDP) image metric.

Keywords: High Dynamic Range Imaging, Inverse Tone Mapping

1 Introduction

High dynamic range imaging has enhanced the use of a significant number of applications that use digital content, by preserving data with larger pixel depth than is used with low dynamic range imaging. This now makes it possible to account for more than all the dynamic range that can be seen by the human visual system which can deal with about $1 : 10^4$ at any given eye adaptation level. The HDR imaging community has been very active in the last decade and several key new application areas have arisen; due to the original shortage of HDR capture devices researches have generated HDR content from sequences of more readily available LDR images, see for example [Debevec and Malik 1997]. Since currently widely available viewing displays can only handle LDR imagery, techniques, known as tone mapping, which compress the luminance range of HDR images and attempt to recreate them for such displays have been developed. The need to display HDR data on LDR devices has led to numerous tone mapping algorithms. A description of some of the most popular methods can be found in [Reinhard et al. 2005]. HDR has also been used to enhance rendering methods by enabling the ability to capture the physical properties of emissive materials and use them within a rendering context, known as IBL [Debevec 1998].

While the use of HDR imagery has become increasingly widespread, the vast majority of digital content still has a limited dynamic range. HDR photography is growing in popularity with the ability to capture multiple bracketed exposures now a standard feature in medium to high range cameras. Yet, the same cannot be said for HDR video. Video cameras capable of capturing a large luminance range are not readily available commercially. Even for high-end commercial applications, such as movies, music videos, or advertising, traditional 35mm motion-picture cameras are still

---

*e-mail: f.banterle@warwick.ac.uk
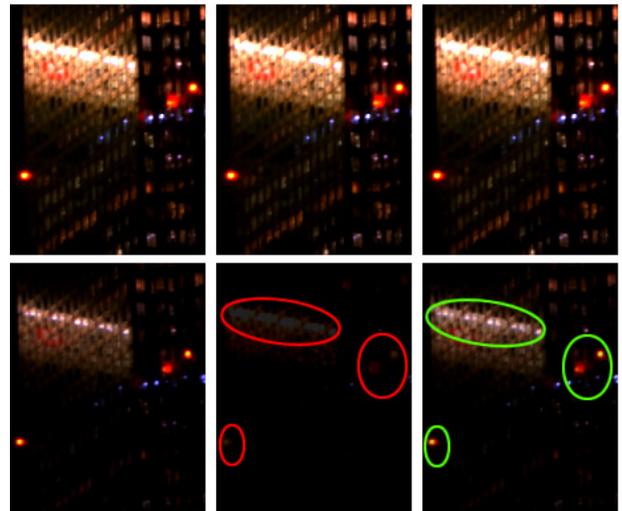
Figure 1: *This image illustrates how our method is able to retrieve the luminance range from an LDR image (middle) and generates a new high dynamic range image (right) which matches the reference HDR image (left). At high exposure levels (top row) the three images appear similar. The importance of our iTMO is clear when the images are shown at a lower exposure level (bottom row). Red circles represent clipped regions in the LDR. Green circles highlight the regions retrieved with our algorithm [see the video].*

the option of choice. With the advent of HDR displays, for example [Seetzen et al. 2004], there is a growing desire to recreate the luminance range of the original content which is missing in the multitude of film and video captured prior to the popularity of HDR. Furthermore, even now, capturing high dynamic range information of a scene is not always possible due to the nature of the scene or for practical and financial reasons. A typical example is the visual effects industry. While most post-production studios have an HDR pipeline these days, quite often, due to the limited timing given to visual effects on set, there is only time for single exposure captures.

In this paper we show how to regenerate missing HDR content from legacy LDR data by conceptually inverting the tone mapping process, which we term inverse tone mapping, see Figure 1. Since most of the LDR content is captured at a single exposure, the problem is considerably more difficult than creating HDR content from multiple exposure LDR images. In this case our input is a single low contrast ratio image or video that needs to be expanded to generate HDR data from. We tackle this problem by extending the algorithm in [Banterle et al. 2006]. We provide a fully automated temporal expansion map that is suitable for video streams. Furthermore we enhanced contrast around edges transferring them using cross/joint bilateral filter. We provide a new set of TMOs, that can be used for the expansion process, and we identify the better one using an HDR-VDP for a set of images. We introduce a new method for the reconstruction of lost colors in over exposed areas using Poisson equation. We also extend their approach to videos, solving flickering problems, exploiting temporal coherence in light samples and finding a heuristic for the automatic calculation of parameters in the density estimation. Finally we present applications of our algorithm
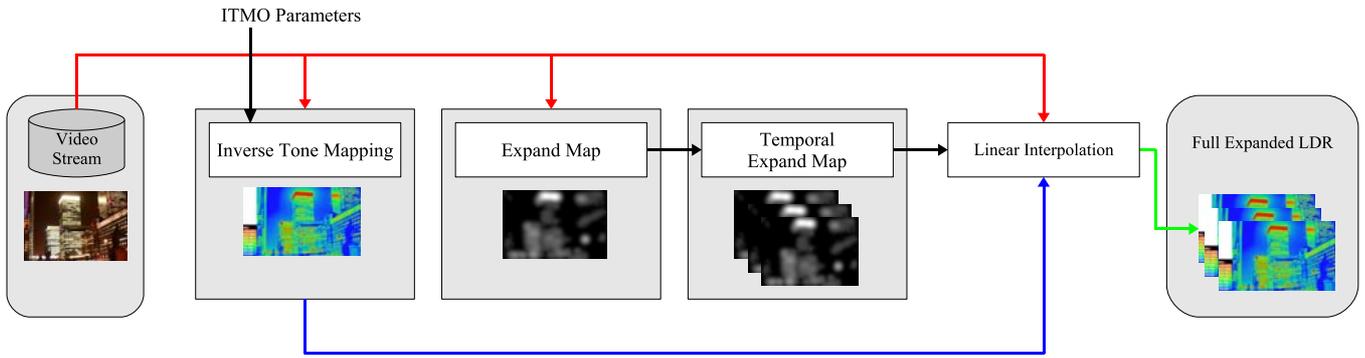
Figure 2: *The inverse tone mapping pipeline.*

such as expansion of video sequences and IBL.

2 Previous Work

One of the first publications on inverse tone mapping was [Landis 2002], in which an LDR image was expanded using an exponential function for pixel values above a certain threshold. This algorithm works for IBL, but not well for visualizing HDR images because it does deal effectively well quantization and noise during the expansion. A straightforward method to expand single exposure LDR images for creating specular highlights was presented in [Meylan et al. 2006]. Two linear tone scales with thresholding were applied to LDR images. Psychophysical experiments with a Dolby DR-37P HDR Monitor [Dolby 2005] were presented in order to validate the images generated by the algorithm. As in [Landis 2002], this method was designed for a very specific task rather than the general case. In [Banterle et al. 2006] a more general method for expanding the range of single exposure LDR images, an iTMO based on the Global Photographic Tone Reproduction Operator (GPTR) [Reinhard et al. 2002] was proposed. In their model, what they called an *expand map*, representing the density of light sources, was used to expand the luminance range for those pixels which belong to light sources. The expand map was derived from sampling the image using the median cut algorithm [Debevec 2005]. The main limitation of their method is that it can not be applied to videos; this is because manual selection of some parameters is needed and furthermore flickering would occur due to the nature of the median cut sampling algorithm. A similar, but computationally more efficient, method for performing LDR expansion in real-time for high definition content was presented in [Rempel et al. 2007]. Instead of the density estimation of the previous approach, a large Gaussian convolution of pixels over a certain threshold is adopted. An edge-stop function is applied to the expansion map to enhance contrast around edges. Finally, the image is expanded linearly according to the expansion map.

Recently in [Akyüz et al. 2007] two psychophysical experiments using the Dolby DR-37P HDR Monitor were presented. From a perceptual point of view the conclusion of these experiments was that a linear scaling of LDR images gives an HDR experience that can be equal or even surpass the appearance of a true HDR image. However their solution was designed for an HDR Monitor application only and not IBL, which needs expansion in the high luminance areas instead of the whole image. Finally [Wang et al. 2007] proposed a system for adding HDR details to the over-exposed and under-exposed regions of an LDR image. This system transfers texture details from a correct exposed patch to an over-exposed and under-exposed patch, followed by an expansion. The main disadvantages of this system were the possibility to not find a patch for transferring detail and the manual interaction that restricts this method only to static images and not video sequences.

Furthermore inverse tone mapping has been used for HDR compression for still images and videos [Mantiuk et al. 2006; Hateren 2006; Li et al. 2005; Okuda and Adami 2007]. In all these approaches the content is firstly tone mapped, then the inverse tone mapping function is calculated using an optimization process, and finally the LDR content is coded using classic algorithms for LDR content such as JPEG and MPEG. Additional information for the inverse tone mapping function are usually embedded in the metadata tag of LDR coding standards.

3 Inverse Tone Mapping Framework

We begin this section by presenting an overview of our inverse tone mapping framework, illustrated in Figure 2, based on the work of [Banterle et al. 2006]. We show the framework in the context of inverse tone mapping for videos. Inverse tone mapping for single images would be just a special and simpler case of this framework.

The design of our framework, and the processes within this framework, are motivated by the expansion using an iTMO to regenerate HDR from LDR content. Naive application of the iTMO may result in incorrect expansion of certain pixels causing areas of the image to have quantization problems. During expansion the difference between a single integer value can be dramatically increased causing banding. This problem can be further compounded in video sequences resulting in temporal flicker. Our framework uses several methods to solve these problems.

Our framework is composed of two sequences. Firstly, an iTMO is applied to an LDR image or a frame of a video to expand its luminance range. In the second one, areas of high luminance are identified, what we term the *expand map* is built from these regions using density estimation. The expand map solves problems related to incorrect expansion from the inverse tone mapped images. For video sequences, the expand map is extended into a temporal expand map to account for temporal coherence and reduce flickering. Finally, the temporal expand map is used as weighting for the interpolation of the LDR and the inverse tone mapped image.

The above framework description outlines the general solution for our inverse tone mapping, the individual methods used for some of the processes warrant further discussion. Below, we discuss the theory used to generate our iTMOs. Furthermore, the discussion on

the expand map will take into account the automatic generation of parameters and optimizations to improve the quality for expanding video sequences.

4 An Inverse Tone Mapping Operator

The first step when an image needs to be inverse tone mapped is to linearize the signal. In the general case, a camera response function (CRF) was applied to the input image when it was captured using a camera or a videocamera. The linearization is achieved by inverting the CRF, and applying it to the signal. This function can be calculated using several exposures [Debevec and Malik 1997; Mitsunaga and Nayar 1999]. However in our case it is more suitable to estimate it from a single image as in [Lin et al. 2004]. More specifically assuming that a video was captured by the same videocamera, and it is not an edited video (with different video sources), we can calculate the CRF only once.

After this step we need to expand the range, this can be achieved using a function that boosts values. Unfortunately, not every TMO can be easily inverted. Typically, global TMOs have an easy inverse, since they are invertible functions, while local TMOs do not. The problem is difficult due to the convoluted nature of local TMOs. In our work, we follow the approach presented in [Banterle et al. 2006]. We inverted the GPTR [Reinhard et al. 2002] because it presents a good trade-off between simplicity and flexibility for the expansion curve, indeed only two parameters are needed. The GPTR is defined as:

$$
\begin{cases}
f(L_{i,j}) = L'_{i,j} = \dfrac{\alpha L_{i,j}(\alpha L_{i,j} + L_{\mathrm{H}} L^2_{\mathrm{white}})}{L_{\mathrm{H}} L^2_{\mathrm{white}}(\alpha L_{i,j} + L_{\mathrm{H}})} \\[2ex]
[R'_{i,j}, G'_{i,j}, B'_{i,j}]^\top = \dfrac{L'_{i,j}}{L_{i,j}} [R_{i,j}, G_{i,j}, B_{i,j}]^\top
\end{cases}
\tag{1}
$$

where L is the HDR luminance calculated as $L = 0.213R + 0.715G + 0.072B$ where R, G, and B are respectively red, green and blue channel. L' is the compressed luminance and L_{H} the harmonic mean is defined as:

$$
L_{\mathrm{H}} = \exp\left(\frac{1}{N} \sum_{i,j} \log\left(L_{i,j} + \delta \right) \right)
$$

where N is the number of pixels in the image and δ a small positive value. Banterle et al. [Banterle et al. 2006] showed that L_{H} can be approximated using L' instead of L. If $g = f^{-1}$ exists, and this is the case when the TMO is an invertible function, we can exploit it to expand LDR images. Equation 1 can be inverted to obtain:

$$
\begin{cases}
g(L'_{i,j}) = L_{i,j} = \dfrac{L^2_{\mathrm{white}} L_{\mathrm{H}}}{2\alpha}\left(L'_{i,j} - 1 + \sqrt{(1 - L'_{i,j})^2 + \dfrac{4}{L^2_{\mathrm{white}}} L'_{i,j}} \right) \\[2ex]
[R_{i,j}, G_{i,j}, B_{i,j}]^\top = \dfrac{L_{i,j}}{L'_{i,j}} [R'_{i,j}, G'_{i,j}, B'_{i,j}]^\top
\end{cases}
\tag{2}
$$

The iTMO presented in Equation 2 has two parameters that can be set: L_{white} and α. L_{white} the white point is quite intuitive as it stretches the boosting curve, see Figure 3. On the other hand, a user can find it difficult to predict the shape of the iTMO as a function of α. To solve this problem, we introduce an interface parameter L_{Max}, representing the maximum luminance that will be present in the inverse tone mapped image. α is calculated by solving Equation

1 for α using L_{white} and setting $L = L_{\mathrm{Max}}$ and $L' = L'_{\mathrm{Max}}$, where L'_{Max} the maximum luminance of the LDR image:

$$
\alpha = \frac{L_{\mathrm{H}} L_{\mathrm{white}}}{2L_{\mathrm{Max}}}\left((-1 + L'_{\mathrm{Max}})L_{\mathrm{white}} + \right. \tag{3}
$$

$$
\left. \sqrt{4L'_{\mathrm{Max}} + L^2_{\mathrm{white}} - 2L'_{\mathrm{Max}} L^2_{\mathrm{white}} + L^{2\prime}_{\mathrm{Max}} L^2_{\mathrm{white}}} \right) \tag{4}
$$
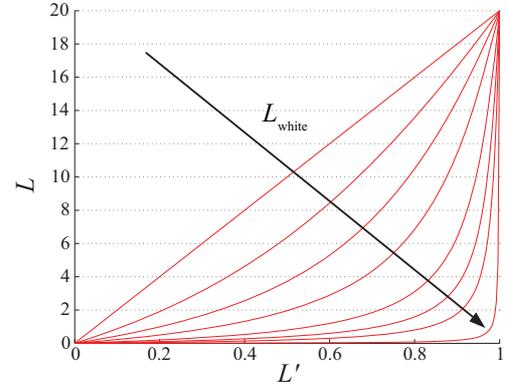
Figure 3: *iTMO curves, showing how L_{white} is related to the shape of the iTMO. In this example different curves are plotted with an increasing value for L_{white} 1, 2.5, 5, 10, 25, 50, 100, and 1000 (the arrow shows the increasing direction).*

5 The Expand Map

The expansion of the range using merely a global function for the whole image can introduce quantization problems such as noticeable differences between two luminance level. These problems can be attenuated using an expand map, that expands the range only in the areas of high luminance.

In our framework the expand map is generated in two steps: the first identifies the light samples present in an image, the second generates a smooth field using these samples.

Light samples are identified using importance sampling. Several algorithms could be used to do this for example structured importance sampling [Agarwal et al. 2003], blue noise sampling [Ostromoukhov et al. 2004], quadrature sampling [Kollig and Keller 2003], and Median Cut [Debevec 2005]. In our approach we employed the median cut algorithm which identifies area of high luminance by splitting the image in 2^n regions of similar light energy. The image is split along the longest dimension. A sample representing clustered luminance is placed in the centroid of each region.

The expand map details a weight associated with each pixel such that when expanded it would achieve a smooth transition between pixels. A quick solution for generating an expand map, could be to use Gaussian filtering on the areas of high luminance identified using a threshold. However this method does not take into account the spatial distribution of light. Another solution, which we employ to obtain better results, is density estimation (see [Duda et al. 2001] for more details), a statistical technique that constructs an estimate, based on observed data, in our case light samples. This technique has already been successfully used in the field of computer graphics, for example in photon mapping [Jensen 2001].

The density estimation is calculated for each pixel (i, j) to determine the density of light samples inside an area of influence which

is a circle defined by the radius r_t. The density estimation is given by the following:

$$\hat{\gamma}_{i,j} = \frac{1}{\pi r_{\max}^2} \sum_{k=1}^{n} L_k \qquad (5)$$

where $\hat{\gamma}_{i,j}$ is the estimate for pixel (i,j), L_k is the luminance value for the k-th light sample in the set of all light sources inside the area of influence given by radius r_t and centered in (i,j), and $r_{\max}$ is the distance of the farthest light sample $L_{\max}$ in the set inside the area of influence. The application of smoothing filters in Equation 5 can improve the estimate. To speed-up the search of samples in the radius r_t, light sources obtained from the Median Cut algorithm are stored in a 2d-tree.

The density estimation can be applied to each color of light samples. In this way a sort of expansion for colors is generated avoiding non flickering free methods, such as interpolation of colors around edges of saturated regions which are defined using thresholding.

5.1 Edge Enhancement

Once the expand map is generated we would like to enhance contrast around the edge and avoid potential halos. This problem was solved in [Rempel et al. 2007] using an edge stopping function, however this function needs a threshold for defining edges and it can potentially produce flickering. In order to transfer edges, which are present in the original frame, to the expand map and to avoid thresholding we opted for using cross/joint bilateral filtering [Eisemann and Durand 2004; Petschnigg et al. 2004]:

$$\gamma_{i,j} = \frac{1}{k_{i,j}} \sum_{(l,m)\in W} \gamma_{l,m} G_S(\|(i,j)-(l,m)\|) G_R(\|E_{i,j}-E_{l,m}\|) \quad (6)$$

$$k_{i,j} = \sum_{(l,m)\in} G_R(\|(i,j)-(l,m)\|) G_S(\|E_{i,j}-E_{l,m}\|) \quad (7)$$

where $\gamma_{i,j}$ is the final expand map, W is the filtering window, G_R and G_S are two Gaussian functions, E is the guidance map, in our case the luminance of the image. The variance for the first function is equal to $\sigma_S = 1$ while for the former $\sigma_R = 0.2$.

Figure 4 compares the basic expand map $\hat{\gamma}$, the basic using joint bilateral filtering γ, and simple method using the combination of thresholding, Guassian filtering and edge stopping function.

6 The Temporal Expand Map

The method for generating an expand map in Section 5 will cause flickering for video sequences. There are two main reasons for this, median cut coherence and parameters for the density estimation.

Applying median cut to each frame does not create coherent samples. For example, in two adjacent frames a sample could appear and disappear in the same area. This occurs because the sampling algorithm is based on a splitting axis mechanism, and energy balancing can change frame by frame with only small changes within the frame. One solution that conforms to our simple method of just using a Gaussian filter to create the expand map, is to use a temporal Gaussian. However for maintaining the same quality obtained using the density estimation, we introduce a new method we term temporal median cut.

The second issue that arises is how to determine the parameters for density estimation, the radius and the minimum number of samples in the estimate. Since the computational cost for long video sequences may be prohibitive, in order to produce a good density map, we must compute the parameters automatically for each frame.

6.1 Temporal Median Cut and Density Estimation

In order to solve the problem of incoherent samples we introduce the temporal median cut, which is a 3D version of median cut, where the third dimension is time. In our case, the density estimation is evaluated by cutting a slice of samples for the evaluating frame. However, due to its nature, median cut needs to have the uncompressed image or its summed-area table [Crow 1984] in memory. For an 8 second sequence in high definition of 1920×1080 at 24 frames per second this would require more than 1GB of memory. Clearly, this solution is not very efficient.

We solve this problem using an efficient memory-wise approach. Firstly, we calculate median cut at each frame, then we store samples of a frame onto a 3D-tree, where the third coordinate is set to the frame number. When we need to calculate the density estimation for a given frame we calculate the volume density estimation, using a similar approach to volume photon mapping [Jensen 2001]. This algorithm is performed for every frame belonging to the same shot. In a sequence, this process is applied individually to each shot. To determine what constitutes a shot we apply a video partition technique based on [Hanjalic 2002].

The temporal filtering ensures that we remove flickering while maintaining temporal coherence. Since the spatial estimation usually requires a larger estimation radius than the temporal estimate, we opt for an ellipsoidal volume rather than a spherical one. The density estimation is filtered using a cone kernel that removes unwanted noise:

$$w_k = 1 - \frac{d_k}{\alpha r} \qquad (8)$$

where d_k is the distance from the center of ellipsoid to the k-sample, r is the radius of the ellipsoid along the direction of the k-th sample from the center of ellipsoid and $\alpha \geq 1$ is the parameter of the filter. The final equation for density estimation is given by:

$$\gamma(i,j) = \frac{1}{(1-\frac{2}{3\alpha})\frac{4}{3}\pi a^2 b} \sum_{k=1}^{n} L_k w_k \qquad (9)$$

where n is the number of samples found in the ellipsoid E_i, L_k is the luminance value of the k-th sample, and a are b are the semi axes of E_i.

The calculation of the temporal density estimation is the main cost during the calculation of the expand map. To speed-up our implementation of the expand map, median-cut and density estimation are generated using a downsampled version of luminance channel, $1/8$ the area. For the final step, the application of the cross bilateral filter, cross bilateral upsampling [Kopf et al. 2007] is employed. In this way we perform upsampling and cross bilateral filtering at the cost of one filtering operation. This operation does not degrade the quality of the expand map.
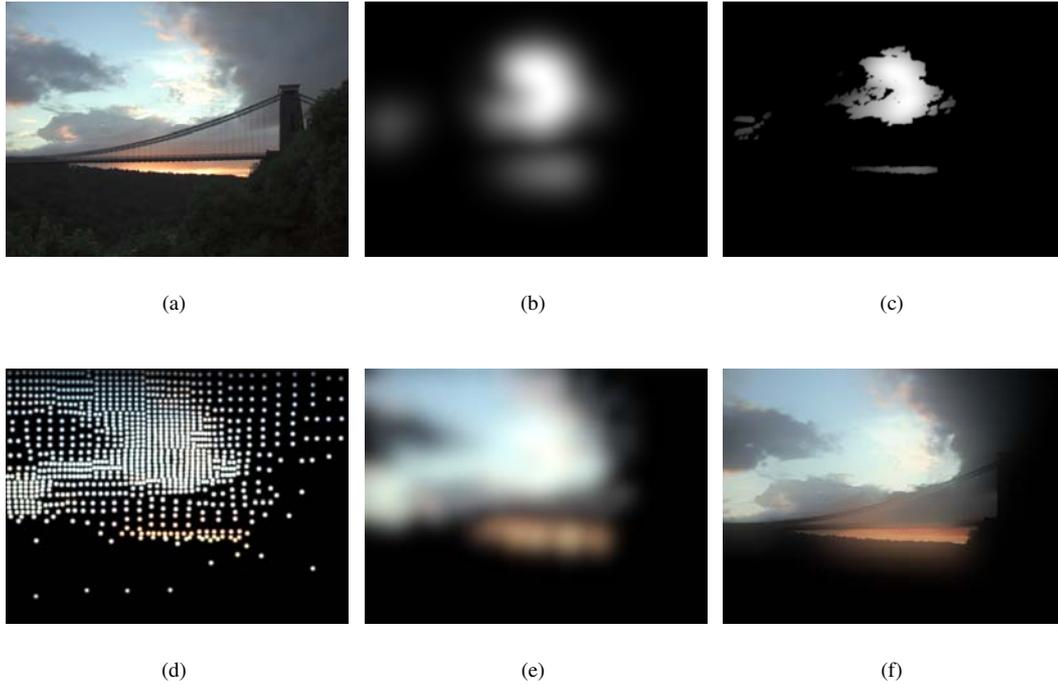
Figure 4: *A comparison between methods for generating the expand map. In the top row the method presented in [Rempel et al. 2007]: a) A single exposure LDR image. b) Gaussian filtered of thresholded luminance for a). c) The final expand map with the application of an edge stopping function to b). In the bottom row our method: d) Light source samples for a) generated using median cut algorithm. e) The expand map, for each color channel, which is the result of the density estimation applied to d). f) Final expand map for a) applying cross/joint bilateral filter to e) using luminance of a) as guidance.*

Figure 5: *A comparison between automatic and non automatic density estimation for the 20-th frame of the sky sequence: a) The LDR frame. b) The light sources generated by median cut. c) The result of the density estimation using automatic parameters estimation. d) The result of the density estimation using parameters which were estimated by a user for the first frame. Note inside red circles the expand map is estimating wrongly the density of light sources generating areas to expand.*

6.2 Automatic Estimation of Density Estimation Parameters

The main problem in the density estimation is how to determine the parameters for the ellipsoid for the search of k-nearest neighbors, and tr_{min}. The parameter b can be calculated in a progressive way, for example the ellipsoid is expanded in the temporal direction until we reach a fixed point. However, we found empirically, analyzing various videos, that $b \in [3, 5]$ works well. This setting can be changed by the user.

For the parameter a, the situation is more difficult as it needs to be expanded until enough samples are reached. On the other hand we have to limit a to some upper bound in areas without lighting. If we expand a indefinitely it will use samples that will give a wrong estimate, since in a non-lit area $\gamma(i, j)$ should be zero or a value close to it. To solve this problem we introduce an heuristic formula to calculate $a_{\max}$, the maximum value a allowed for expansion and $ns_{\min}$ the minimum number of samples needed to terminate the expansion of a. To do that for each frame being processed, we extract the samples of median cut of the current frame. Then, we divide the space XY in a regular square grid with length equal to $l_{square} = max(w, h)/k$, where w, h is the width and height of image, and $k = \log_2 n_{samples}$. This means that we divide the 2D space of the image using the total number of cuts that we can obtain using median cut into non-uniform regions based on the density of samples. Subsequently, we count the number of samples for each region, and we sort these values in an array, T. We then calculate $ns_{\min}$ as the value in T with the biggest slope between its predecessor and its successor, see Figure 6. However, it may happen that there is a constant slope, in this case this metric will not work. To solve this problem we decided to set $ns_{\min}$ equal to the median value. Finally,
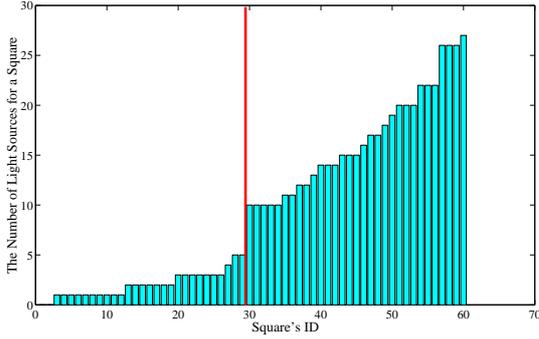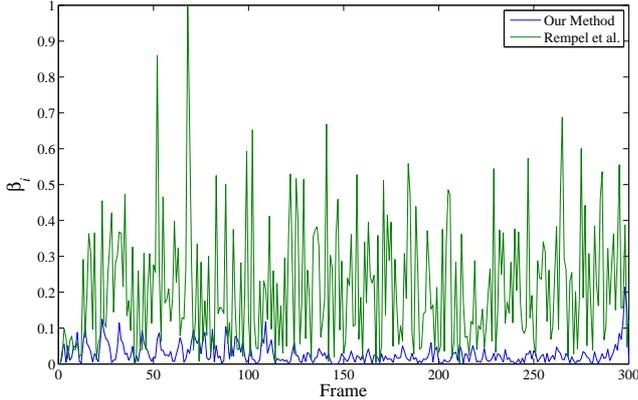
we calculate a_{max} as the distance from the center of the median square to the furthest sample in it.

Figure 5 shows the advantages of automatic parameter estimation using the sky sequence. As can be seen, artifacts appear in the expanded HDR frame where parameters were not chosen correctly.

7 Applications and Results

There are two direct applications for inverse tone mapping of single exposure LDR content. Firstly, we can enhance LDR content in order to display it on HDR displays, for HDR video editing, for example performing common operations, adding filters, *etc...* and for use in HDR pipelines of visual effects and games companies. Secondly, we can use the content for IBL.

In order to demonstrate how close our expanded HDR images are to actual HDR images we compare our results with reference HDR material using the Visual Difference Predictor (VDP) [Daly 1993]. The VDP is a perceptual image metric for comparing two images which takes into account limitations in the human visual system rather than just physical values when comparing images. We use an extension to VDP, HDR-VDP which is specialized in comparing HDR images [Mantiuk et al. 2004]. HDR-VDP outputs a third image showing per pixel false coloring of the percentage of perceptual difference between the two images. In the resulting color coded images grey symbolizes no perceptual difference, green a low perceptual difference and red a high probability of noting a perceptual difference. It also summarizes the results with two values. The first is the percentage of different pixels detected with the probability of 75% ($P(X) \geq 0.75$). The second is the percentage of different pixels detected with the probability of 95% ($P(X) \geq 0.95$).

We compared our results against an HDR reference and Rempel et al. [Rempel et al. 2007] operator **R**, which is the closest method to our technique; in particular it is the only operator presented for videos that handles quantization artifacts during the expansion.

7.1 Enhanced LDR Videos for Visualization

The enhancement of LDR videos for visualization was tested with five HDR sequences. The first is a sky captured from 4.30pm to 5.30pm and sampled at each minute, for a total of 60 frames. The other sequences are lightprobes in which the camera is moved at 360 degrees for a total of 96 frames. In particular: the second and the third are outdoor captured during daylight, the fourth is an indoor captured at night, and the last is an outdoor in a city captured at night. For the first frame of each sequence see Figure 11.

The results are summarized in Table 1. As can be seen in the table our operator performed better than **R** in very difficult cases, such as the reconstruction of Scene 1 and 5 where the error is quite high for both methods. In Scene 2 our operator performed slightly better, while in Scene 3 and 4 **R** is clearly a winner. However in Scene 3 and 4 the error is low in the absolute scale, 1.82% and 0.6% for $P(X > 0.75)$, and 0.4% and 1.05 $P(X > 0.95)$

7.2 Video Temporal Coherence

The main problem with iTMOs so far is that the are not designed directly for videos, and so with current methods flickering occurs. This is particularly evident in the case of **R**, because the expand map is generated using thresholding. To test the quality of our new

(a) (b)

(c)

Figure 6: *Automatic Parameter Estimation: a) The image used for the estimation. b) The result of Median Cut and image divided in squares. c) The graph of number of samples for each square, with division line for ns_{min} (in red).*

Figure 7: *The graph for values of β_i for each frame for the sequence fireball.avi. Our method compared to **R**, present less fluctuations and of lower intensity.*

	Our Operator		Rempel et al. [Rempel et al. 2007]	
Name	$P(X > 0.75)$	$P(X > 0.95)$	$P(X > 0.75)$	$P(X > 0.95)$
Scene 1	42.02%	36.72%	56.46%	48.31%
Scene 2	9.65%	6.46%	10.61%	8.24%
Scene 3	3.18%	1.96%	1.36%	0.91%
Scene 4	0.70%	0.45%	0.10%	0.05%
Scene 5	9.49%	8.25%	25.42%	19.94%

Table 1: *The results for the comparisons for enhanced LDR videos for visualization. For each entry we calculated the average of the VDP results of all frames for $P(X > 0.75)$ and $P(X > 0.95)$.*

Figure 9: *The lightprobes used in the IBL test. In the top from left to right: L_1, L_2, L_3, and L_4. In the bottom from left to right: L_5, L_6, L_7, and L_8.*
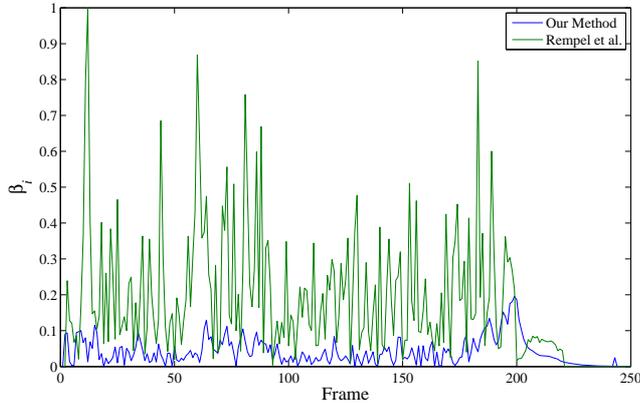
Figure 8: *The graph for values of β_i for each frame for the sequence fireball-smoke.avi. Our method compared to **R**. present less fluctuations and of lower intensity.*

Figure 10: *An example of the rendered scene used in our experiments for IBL using sequence $L_1 + L_2$ for the 10th frame: on the left the HDR reference, on the center the our method, and on the right **R**.*

expand map we decided to expand a computer generated videos of flames and fire in which the animation is smooth but changes rapidly [Fedkiw 2008]. To avoid interferences due to the expansion we used a linear expansion as in Rempel et al. [Rempel et al. 2007]. To determine the amount of flickering we used a simple metric, the Brightness Flicking Metric, which is based on the difference of the mean of luminance between two frames, $\beta_i = \|\bar{L}_i - \bar{L}_{i-1}\|$ where i is the current frame and $\bar{L}_i$ is the average of the luminance for the i-th frame. β_i is calculated for all frames of the sequence, which can be summarized in a graph. High variations in the graph mean that flickering is occouring in the video sequence.

The results are shown in Figure 7, for the sequence fireball.avi and Figure 8 fireball-smoke.avi that can be downloaded from [Fedkiw 2008].

7.3 Image Based Lighting

IBL task was tested using four animated lightprobes, which were generated linearly morphing two lightprobes per sequence, see Figure 9. A simple 3D scene was created with an Happy Buddha and few spheres with different materials: a pure diffuse material, a pure spherical material, and a glossy material using Ward's BRDF ($\alpha_v = 0.3$, $\alpha_u = 0.1$).

The results are summarized in Table 2. As can be seen our method performed slighty better than **R**. This is due to the fact that during integration small details, that can be seen as different in the visualization, are lost.

Figure 11: *The first frame for each sequence used in the experiment for testing enhancing LDR videos for visualization using an HDR reference. In top from left to right: Scene 1, Scene 2, and Scene 3. In the bottom from left to right: Scene 4 and Scene 5.*

	Our Operator		Rempel et al. [Rempel et al. 2007]	
Name	$P(X > 0.75)$	$P(X > 0.95)$	$P(X > 0.75)$	$P(X > 0.95)$
$L_1 + L_2$	1.13%	0.72%	2.33%	1.37%
$L_3 + L_4$	8.02%	5.74%	8.33%	5.50%
$L_5 + L_6$	5.92%	3.87%	9.11%	5.94%
$L_7 + L_8$	0.30%	0.59%	0.77%	0.40%

Table 2: *The results for the comparisons for IBL. The name $L_i + L_{i+1}$ means that lightprobe L_i was morphed linearly with the lightprobe L_{i+1}. For each entry we calculated the average of the VDP results of all frames for $P(X > 0.75)$ and $P(X > 0.95)$.*

8 Conclusion

We have presented a new framework for regenerating missing HDR content using an iTMO for images and videos. A naive implementation of an iTMO can result in a number of issues, we addressed those problems using a number of techniques.

The range was expanded using an iTMO, employing a temporal density estimation with automatic parameters estimation. This helps to expand the dynamic range smoothly and reduce the flickering. The use of the cross/joint bilateral filter improved the previous expand map reducing halos and enhancing contrast around edges.

The introduction of these temporal coherent techniques provides a solid basis for handling HDR content. In this paper we have demonstrated its success for recreating and visualizing HDR content and IBL. We validated our results using HDR-VDP and a combination of averaging techniques.

This paper has shown that inverse tone mapping has significant potential. We have considered recreating missing HDR content from legacy LDR images and videos, for visualization and IBL. In the future we intend to validate our results running psychophysical experiments using an HDR monitor [Seetzen et al. 2004]. Furthermore, we will investigate techniques to speed-up our method using GPUs or FPGAs.

9 Acknowledgment

References

AGARWAL, S., RAMAMOORTHI, R., BELONGIE, S., AND JENSEN, H. W. 2003. Structured importance sampling of environment maps. In *SIGGRAPH '03: ACM SIGGRAPH 2003 Papers*, ACM Press, New York, NY, USA, 605–612.

AKYÜZ, A. O., FLEMING, R., RIECKE, B. E., REINHARD, E., AND BÜLTHOFF, H. H. 2007. Do hdr displays support ldr content?: a psychophysical evaluation. In *SIGGRAPH '07: ACM SIGGRAPH 2007 papers*, ACM, New York, NY, USA, 38.

BANTERLE, F., LEDDA, P., DEBATTISTA, K., AND CHALMERS, A. 2006. Inverse tone mapping. In *GRAPHITE '06: Proceedings of the 4th international conference on Computer graphics and interactive techniques in Australasia and Southeast Asia*, ACM Press, New York, NY, USA, 349–356.

CROW, F. C. 1984. Summed-area tables for texture mapping. In *SIGGRAPH '84: Proceedings of the 11th annual conference on Computer graphics and interactive techniques*, ACM Press, New York, NY, USA, 207–212.

DALY, S. 1993. *The visible differences predictor: an algorithm for the assessment of image fidelity*. MIT Press, Cambridge, MA, USA.

DEBEVEC, P. E., AND MALIK, J. 1997. Recovering high dynamic range radiance maps from photographs. In *SIGGRAPH '97: Proceedings of the 24th annual conference on Computer graphics and interactive techniques*, ACM Press/Addison-Wesley Publishing Co., New York, NY, USA, 369–378.

DEBEVEC, P. 1998. Rendering synthetic objects into real scenes: bridging traditional and image-based graphics with global illumination and high dynamic range photography. In *SIGGRAPH '98: Proceedings of the 25th annual conference on Computer graphics and interactive techniques*, ACM Press, New York, NY, USA, 189–198.

DEBEVEC, P. 2005. A median cut algorithm for light probe sampling. In *ACM Siggraph 2005 Posters*.

DOLBY, 2005. http://www.dolby.com/promo/hdr/technology.html.

DUDA, R. O., HART, P. E., AND STORK, D. G. 2001. *Pattern Classification 2nd Edition*. Wiley Interscience.

EISEMANN, E., AND DURAND, F. 2004. Flash photography enhancement via intrinsic relighting. In *SIGGRAPH '04: ACM SIGGRAPH 2004 Papers*, ACM, New York, NY, USA, 673–678.

FEDKIW, R., 2008. http://physbam.stanford.edu/ fedkiw/.

HANJALIC, A. 2002. Shot-boundary detection: Unraveled and resolved? *IEEE Transaction Circuits System Video Technology, Vol.12, No.2* (February).

HATEREN, J. H. V. 2006. Encoding of high dynamic range video with a model of human cones. *ACM Trans. Graph. 25*, 4, 1380–1399.

JENSEN, H. W. 2001. *Realistic image synthesis using photon mapping*. A. K. Peters, Ltd., Natick, MA, USA.

KOLLIG, T., AND KELLER, A. 2003. Efficient illumination by high dynamic range images. In *EGRW '03: Proceedings of the 14th Eurographics workshop on Rendering*, Eurographics Association, Aire-la-Ville, Switzerland, Switzerland, 45–50.

KOPF, J., COHEN, M. F., LISCHINSKI, D., AND UYTTENDAELE, M. 2007. Joint bilateral upsampling. In *SIGGRAPH '07: ACM SIGGRAPH 2007 papers*, ACM, New York, NY, USA, 96.

LANDIS, H. 2002. Production-ready global illumination. In *Siggraph Course Notes 16*.

LI, Y., SHARAN, L., AND ADELSON, E. H. 2005. Compressing and companding high dynamic range images with subband architectures. In *SIGGRAPH '05: ACM SIGGRAPH 2005 Papers*, ACM Press, New York, NY, USA, 836–844.

LIN, S., JINWEI, G., YAMAZAKI, S., AND SHUM, H.-Y. 2004. Radiometric calibration from a single image. In *Proceedings of the 2004 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 2004*, IEEE, vol. 2, 938–945.

MANTIUK, R., MYSZKOWSKI, K., AND SEIDEL, H.-P. 2004. Visible difference predicator for high dynamic range images. In *Proceedings of IEEE International Conference on Systems, Man and Cybernetics*, 2763–2769.

MANTIUK, R., EFREMOV, A., MYSZKOWSKI, K., AND SEIDEL, H.-P. 2006. Backward compatible high dynamic range mpeg video compression. In *SIGGRAPH '06: ACM SIGGRAPH 2006 Papers*, ACM, New York, NY, USA, 713–723.

MEYLAN, LAURENCE, DALY, SCOTT, AND SÜSSTRUNK, S. 2006. The Reproduction of Specular Highlights on High Dynamic Range Displays. In *IS&T/SID 14th Color Imaging Conference*.

MITSUNAGA, T., AND NAYAR, S. 1999. Radiometric Self Calibration. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, vol. 1, 374–380.

OKUDA, M., AND ADAMI, N. 2007. Raw image encoding based on polynomial approximation. *IEEE International Conference on Acoustics, Speech and Signal Processing, 2007. ICASSP 2007*, 15–31.

OSTROMOUKHOV, V., DONOHUE, C., AND JODOIN, P.-M. 2004. Fast hierarchical importance sampling with blue noise properties. *ACM Trans. Graph. 23*, 3, 488–495.

PETSCHNIGG, G., SZELISKI, R., AGRAWALA, M., COHEN, M., HOPPE, H., AND TOYAMA, K. 2004. Digital photography with flash and no-flash image pairs. In *SIGGRAPH '04: ACM SIGGRAPH 2004 Papers*, ACM, New York, NY, USA, 664–672.

REINHARD, E., STARK, M., SHIRLEY, P., AND FERWERDA, J. 2002. Photographic tone reproduction for digital images. *ACM Trans. Graph. 21*, 3, 267–276.

REINHARD, E., WARD, G., PATTANAIK, S., AND DEBEVEC, P. 2005. *High Dynamic Range Imaging: Acquisition, Display and Image-Based Lighting*. Morgan Kaufmann Publishers, December.

REMPEL, A. G., TRENTACOSTE, M., SEETZEN, H., YOUNG, H. D., HEIDRICH, W., WHITEHEAD, L., AND WARD, G. 2007. Ldr2hdr: on-the-fly reverse tone mapping of legacy video and photographs. In *SIGGRAPH '07: ACM SIGGRAPH 2007 papers*, ACM, New York, NY, USA, 39.

SEETZEN, H., HEIDRICH, W., STUERZLINGER, W., WARD, G., WHITEHEAD, L., TRENTACOSTE, M., GHOSH, A., AND VOROZCOVS, A. 2004. High dynamic range display systems. *ACM Trans. Graph. 23*, 3, 760–768.

WANG, L., WEI, L.-Y., ZHOU, K., GUO, B., AND SHUM, H.-Y. 2007. High dynamic range image hallucination. In *Proceedings of Eurographics Symposium on Rendering*.