Eurographics Symposium on Rendering 2008
Steve Marschner and Michael Wimmer
(Guest Editors)

Volume 27 (2008), Number 4

Enhancement of Bright Video Features for HDR Displays

P. Didyk[1,2], R. Mantiuk[1], M. Hein[3] and H. P. Seidel[1]

[1] MPI Informatik, Saarbrücken, Germany
[2] University of Wrocław, Poland
[3] Saarland University, Saarbrücken, Germany

Abstract

To utilize the full potential of new high dynamic range (HDR) displays, a system for the enhancement of bright luminous objects in video sequences is proposed. The system classifies clipped (saturated) regions as lights, reflections or diffuse surfaces using a semi-automatic classifier and then enhances each class of objects with respect to its relative brightness. The enhancement algorithm can significantly stretch the contrast of clipped regions while avoiding amplification of noise and contouring. We demonstrate that the enhanced video is strongly preferred to non-enhanced video, and it compares favorably to other methods.

Categories and Subject Descriptors (according to ACM CCS): I.4.9 [IMAGE PROCESSING AND COMPUTER VISION]: Applications

1. Introduction

The off-the-shelf LCD or Plasma TV displays available today can show much higher dynamic range (contrast), better brightness, lower black level and more saturated colors than their CRT predecessors. The display back-light modulation techniques (2D dimming) can extend image contrast to the limits of human eye sensitivity [SHS*04]. However, the available video content, such as DVD movies, cannot take full advantage of these new capabilities. The resolution and to some extent the dynamic range of DVD movies can be improved by rescanning film negatives and color grading them for new displays. This, however, cannot restore very bright image features, such as light sources, explosions, or specular highlights. They are over-saturated even for high exposure latitude film stocks and are clipped in the scanned material.

In this paper we propose a semi-automatic system for enhancement of bright luminous objects in video sequences, so that video intended for low contrast (low dynamic range) displays can exploit the full potential of new high contrast displays. The enhanced video shows bright luminous objects, such as lamps, candles, explosions, specular reflections, as much brighter than diffuse surfaces, resulting in a reproduction that is much closer to viewing the actual scenes. Bright objects are important visual cues that guide our depth and shape perception [WBNF06] and let our visual system assess illumination and reflectance in a scene [SDM05]. It has been demonstrated that such enhanced reproductions are usually preferred and regarded to be of better quality [MDS06].

2. Previous work

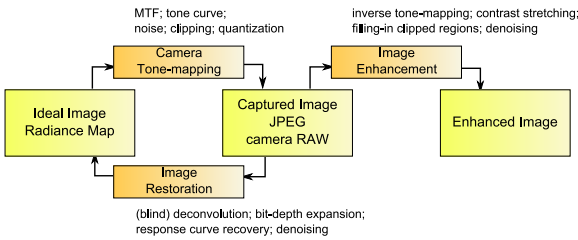

Figure 1: *The distinction between image restoration and image enhancement in the context of the LDR2HDR problem. Most inverse tone-mapping algorithms do not attempt to reverse camera distortions but rather try to produce believable images that look better than the unprocessed originals.*

The problem addressed in this paper is closely related to image restoration, image enhancement and recently proposed "LDR to HDR" algorithms.

Restoration: One of the goals of image restoration techniques is to reverse the distortions introduced by a camera

system (lens + sensor + image processing). Image restoration techniques are depicted on the left side of Figure 1 as an edge going from an actual image captured by a camera to a radiance map, representing an original scene. A method of restoring images affected by blur (MTF) and *sensor non-linearity* (DlogH non-linearity of a negative) has been already proposed in the early publication [Hun77]. Later techniques proposed using multiple images at different exposures for a better estimate of the sensor non-linearity [BKCP93, MN99]. A sensor response curve can also be estimated from a single exposure using higher-order correlations in the frequency domain [Far01] or from the color distribution at color edges [LGYS04]. Probably the most challenging kind of distortion to restore is *clipping* of over- and under-saturated image pixels due to insufficient dynamic range of a sensor. This problem has been solved in the case of 1D signals if the signal is band-pass limited and the number of missing samples is low [ASI91], or if a statistical model of an undistorted signal is known [Olo05]. Neither of these approaches can be easily extended to images because statistics of complex images, and especially clipped image features, are difficult to model and limiting image frequency content will always lead to blurred edges. The pixels that are clipped in one or two color channels can be estimated using correlation across channels [ZB04]. Inpainting techniques [BSCB00, She03, TLQS03], although designed to fill-in missing pixels, are not well suited for restoration of clipped signal since they tend to smooth out (interpolate) missing pixels that should be much brighter than the neighboring pixels used for interpolation.

Enhancement: The original radiance maps are in fact not necessary to produce improved images for new displays. The recently proposed inverse-tone mapping algorithms produce enhanced images of higher contrast that still look believable and visually better than non-processed images on high-contrast displays. Such goals are usually associated with image enhancement as shown on the right part of Figure 1. The resulting images only roughly approximate original radiance maps but the algorithms are simpler, faster and more robust than image restoration techniques. The inverse-tone mapping algorithms typically stretch image contrast, adjust image brightness, enhance color saturation and fill-in clipped regions [BLDC06, MDS07, RTS*07, WWZ*07]. Contrast stretching often results in contouring (banding) artifacts, which can be reduced with spatial and temporal dithering techniques [DF03, BLfZ07]. Filling-in clipped regions can be realized by stretching tone-curve for bright pixels [MDS07], fitting Gaussians [WWZ*07], expanding pixel values using low-pass estimate of the non-clipped image [BLDC06] or a blurred binary mask containing clipped pixels [RTS*07]. Akyüz et al. [AFR*07] investigated the preferred presentation of LDR and HDR images on an HDR display. They found that brighter images are in general preferred, even if their dynamic range is lower.

3. Problem analysis

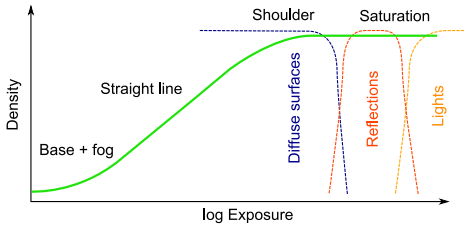

Figure 2: *The typical response of a film negative (characteristic curve or DlogH plot). All exposure magnitudes that fall into the saturated part of the curve have maximum density, regardless of actual luminance levels in a scene.*

The clipping of bright image features is caused by a non-linear response of the negative film, often depicted as so called *DlogH* plot, shown in Figure 2. Well produced movies contain all important image features in the *straight line* part of this curve. Saturation of important image features is usually avoided, but the film dynamic range is not high enough (up to 3.3 $\log_{10}$ units) to register the magnitude of bright reflections and direct sources of light (lamps, candles, explosions, etc.). Some white diffuse surfaces are also saturated, giving them the same density value as much brighter reflections and light sources. Therefore, all image features above the saturation point are flattened to a tiny range of density values (or code values after digital scanning), losing information about the actual luminance levels in a scene.

The goal of this work is to improve the quality of video shown on high contrast (high dynamic range) displays by boosting bright image features, such as light sources and specular highlights. We want to work with professionally produced video content, such as movies intended for DVD distribution, which could be captured with several different cameras, composed with computer generated material and subsequently processed. Assumptions about a camera MTF or color distribution are impractical for such edited content. The method should improve overall subjective video quality but does not aim at restoring the original radiance map. Such restoration is a heavily under-determined problem, requiring many assumptions that can not be made for general video content. Restoration techniques also do not guarantee improved quality and can even result in visually less attractive results, when for example image noise is pronounced and becomes clearly visible.

In this paper we do not consider other enhancements, associated with the "LDR2HDR" task, such as the reversal of tone-curve or contrast stretching. For that purpose our approach can be combined with existing methods [DF04, BLDC06, RTS*07]. We also do not address the problem of restoring details in the under-exposured regions, as they may lead to increased noise visibility and are therefore not desirable.

4. Enhancement of clipped video

Our system strives to achieve the same results as a digital processing expert manually enhancing a movie sequence, but with much less manual labor. Such an expert can interpret a scene and determine that a lamp is brighter than its reflection and the reflection is still brighter than a white diffuse surface. Similarly, as an expert would do, we classify clipped regions into three categories: *diffuse* — bright diffuse objects (usually white) that can become partly saturated, *reflection* — specular reflections on shiny, highly reflective surfaces, *lights* – direct sources of light, such as lamps, candles, explosions, the sun, and sometimes even window panes if they are much brighter than the filmed interior. These three categories differ in absolute luminance levels as illustrated in Figure 2. Having such a classification, we enhance *lights* and *reflections*. We do not modify diffuse surfaces because of two reasons: (a) due to their complex structure they would be very difficult to reconstruct and any failures in reconstruction would lead to visible artifacts; (b) if diffuse surfaces are saturated in the source material, that was most probably the intention of a film maker and they should remain saturated.

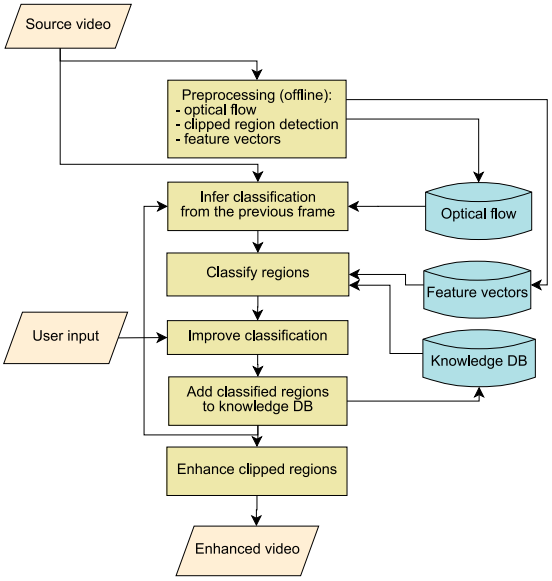

Figure 3: *Data flow diagram of the video enhancement system.*

Figure 3 shows a data flow of our enhancement system. The source video is first preprocessed to avoid delays in the user assisted part of the system. In the preprocessing stage we compute a dense motion flow, detect clipped regions and compute their feature vectors, which will be used for the classification. In the user assisted stage, the clipped regions are first classified automatically. If the previous frame is available, we use motion flow to match regions from the previous frame and reuse their classification. If clipped regions cannot be tracked over time, which is the case for newly appearing objects or scene cuts, an automated classifier attempts to classify them based on their precomputed feature vectors. Then, the result of such classification is shown to the user, who can accept the automatic classification or modify it. The user input is used to update the knowledge database and train the online classifier after each completed frame. In the last step, all regions classified as *lights* or *reflections* are enhanced and user can see a preview of the enhanced frame.

4.1. Detection of clipped regions

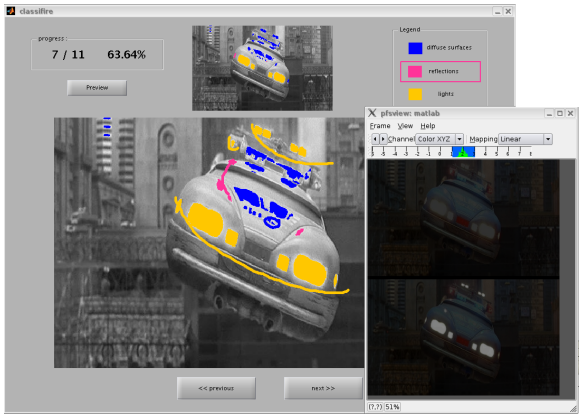

Figure 4: *Screenshot of our sketch based interface for semi-automatic classification.*

Before we can classify clipped regions, we need to find the pixels that belong to them. We consider pixels as part of the clipped region if they exceed a certain threshold along with neighboring pixels that are part of the saturated object. A simple thresholding is not a reliable estimator of clipped regions since the source video is often heavily distorted by noise and image processing. For this reason we mark clipped pixels using a flood fill algorithm, where a single seed (for a single region) constitutes of connected pixels that have at least one channel saturated (value $> t_0$; $t_0 = 230$ for DVD content) and are bright (luma $> t_1$, $t_1 = 222$ for our prototype). To reduce the influence of noise, we take the luma from the image filtered with the bilateral filter [TM98] ($\sigma_s = 2$, $\sigma_r = 25$). The stopping condition for the flood fill is a value of luma smaller than the predefined threshold t_2 ($t_2 = 219$ for our prototype).

4.2. Motion tracking

Motion tracking of clipped regions allows us to reuse the classification from a previous frame. To estimate the position of a clipped region in the previous frame, we average the motion vectors (found using [Bou00]) for the pixels that surround the clipped region. We exclude the motion vectors stemming from the inside of a clipped region as these are

unreliable due to clipping, excessive noise and lack of features that could be matched. Then, we translate the region by the averaged motion vector. We assume that the clipped regions in the current and previous frame belong to the same object if 70% of the pixels overlap (after translation) and the difference in size does not exceed 30% of the area of object.

4.3. Classification

Although the classification of clipped regions into *lights*, *reflections* and *diffuse* surfaces is trivial for a human operator who can interpret the scene content, it is a challenging task for an automated classifier. Unlike other typical classification objects such as faces, clipped regions do not contain a common structure, and it is not possible to create a database of known objects. Based on the statistics of 2,000 manually classified regions (our training set) we designed over 20 features, which included 1st- and 2nd-order image statistics (mean, median, skewness and kurtosis for the luma distribution in clipped regions), geometric features (size, symmetry, shape) and neighborhood characteristics (luma gradients, contrast, smoothness of neighboring pixels). Each feature was tested individually in terms of classification performance and additionally selected pairs of features were tested to check for possible correlations. The features that resulted in the classification performance close to random guess were excluded. The eight remaining and best performing features (Z_i, $i = 1..8$) were:

Mean luma of a frame: Dark scenes are likely to contain *lights* and bright specular *reflections*. Bright scenes on the other hand are usually more likely to include overexposed *textures*.

Similarity to disk and **major axis ratio**: *Lights* usually feature regular shapes and they are round or slightly oval. Specular *highlights* are less regular and can be elongated. The least regular are *textures*, which can be found in any shape. To account for shape, we compute two statistics: **Similarity to disk** is a ratio between the perimeter of a disk having the same field as the clipped region to the perimeter of the clipped region; and **major axis ratio**, which is the ratio of the maximum and the minimum eigenvalue found by computing a PCA of the pixel coordinates for the pixels that belong to a clipped region.

Luma standard deviation: *Reflections* tend to contain higher contrast details than flat *textures* and strongly clipped *lights*. As a measure of this phenomenon we compute the standard deviation of luma in a clipped region.

Median luma of a region and **skewness**: The luma distribution in a clipped region is also discriminative for the three classes. Diffuse surfaces contain more dark pixels than lights and reflections, therefore the **median luma of a clipped region** is shifted towards smaller values for diffuse surfaces and the **skewness** of the distribution is larger.

Slope and **offset** account for the surrounding of the clipped area. The medians of the equidistant pixels to the clipped area form a profile of the surrounding, which we approximate with a straight line. The **slope** and the **offset** of that line are the last two features.

As a preprocessing step we center the individual features and rescale them to have unit variance, so that no feature dominates the other. The chosen features are very good predictors inside a scene (from one scene cut to the next scene cut). Unfortunately, the generalization across scenes is very difficult.

We have divided the classification procedure into a two-stage process. First, we detect scene cuts by computing correlation of luma values with the luma values in the previous frame. If the absolute correlation is less than a threshold (0.3), the frame is regarded as a scene cut and using temporal information for that frame is not advisable. Therefore, in the first stage we train a support vector machine (SVM) using a Gaussian kernel $k(z,z') = e^{-\gamma\|z-z'\|^2}$ on the training set. All parameters are chosen by cross validation. Due to the difficulty of the task, we achieve a relatively high error of 46.0% on an independent test set, which is still better than random guessing (66.6% for evenly distributed classes and a 3-class problem). However, as can be seen in Table 1, the total number of regions in scene cuts is only a tiny portion of the total number of all regions which have to be classified. Since the user is asked to correct all errors of the classifier, the large number of initial errors does not propagate into the motion tracking system or the second stage of the classifier.

In the second stage, after a scene cut frame we classify only regions if they cannot be related to a region of the previous frame by the motion tracking system described in Section 4.2. For this purpose we need a classifier which is adaptive to the scene, works in an online fashion without costly retraining and can make use of the region features as well as spatial and temporal information. A simple yet efficient method which has all these three properties is the nearest neighbor (NN) classifier. In the specific case a region is classified with the category of the nearest neighbor in the set of all regions which have occurred in the scene up to the current frame using a weighted Euclidean metric,

$$d^2((z,x,t),(z',x',t')) = 50\|z-z'\|^2 + \|x-x'\|^2 + 5(t-t')^2,$$

where z are the region features, x are the coordinates in the image and t is the frame number. The weighting factors were adjusted on the training set. Unlike the SVM, the nearest-neighbor classifier can be dynamically extended with newly classified feature vectors each frame at almost no cost. Since the classifier is dynamic and time-dependent, the 8 selected features are much more robust than for the off-line SVM classifier. They are used to group together common objects (for example perfectly round lights in one scene and elongated lights in another scene), so that the user needs to classify a single region from that group to propagate this knowl-

edge to the other regions in the same scene. On our test set of regions which were either new in the scene or could not be identified using the motion tracking system the NN-classifier achieved a good error rate of only 10.0%. Table 1 shows that the overall classifier of the SVM for scene cuts and the NN-classifier inside scene has an error rate of only 12.6% on all regions which had to be classified on the test set.

Classifier	Total Number	#Errors	Percentage
SVM (scene cuts)	526	242	46.0%
NN (within a scene)	6638	661	10.0%
All	7164	903	12.6%

Table 1: *The number of regions that require classification for both classifiers and their classification errors.*

In order to correct potential errors of our two-step classification procedure, the classification results are presented to the user (see the user interface in Figure 4). The current frame is shown in gray-scale and the classified clipped regions are color-coded. Since the regions that are tracked over time are almost always correctly classified, they are marked with desaturated colors and cannot be edited unless editing them is activated by the user. This removes clutter from the screen and requires fewer user corrections. If there are any misclassified regions, a user can click, stroke or encircle them while holding the key combination corresponding to a particular class. Hitting the space key proceeds to the next frame. The frames in which all regions can be tracked over time or that do not contain clipped regions are presented only as a preview and do not need user correction.

4.4. Enhancement

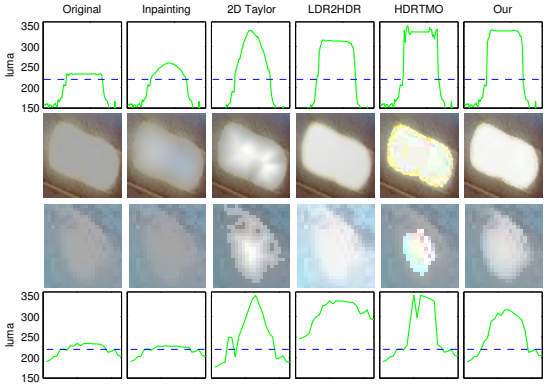

Figure 5: *A comparison of our enhancement method with other approaches, for a light (two upper rows) and a specular highlight (two bottom rows). The plots show a cross-section across the center of each image.*

Although we experimented with several methods for enhancing or reconstructing clipped regions, we found most of them unsuitable because of artifacts, lack of temporal coherence or unconvincing results. The results of these methods for a light source and specular highlight are shown in Figure 5. Fitting smooth functions or **inpainting** [TLQS03] results in flattened profiles, which do not give much brightness boost to the clipped regions. Maintaining temporal coherence is also problematic for these methods. The extrapolation techniques, such as **2D Taylor** series expansion, are not robust because the surrounding pixels used to estimate partial derivatives are often affected by the scene content that is not the part of a clipped region. The resulting reconstruction contains structures in the center of the clipped region, which do not match the appearance of the actual light source or specular highlight. The method of Rempel et al. [RTS*07] (**LDR2HDR**) is strongly affected by the size of clipped region, making larger objects brighter than smaller objects. Linear contrast stretching [MDS07] (**HDRTMO**) is fast and straightforward but it reveals contouring artifacts and strong noise near the saturation point. The last two methods are also discussed in more detail in Section 5.2.

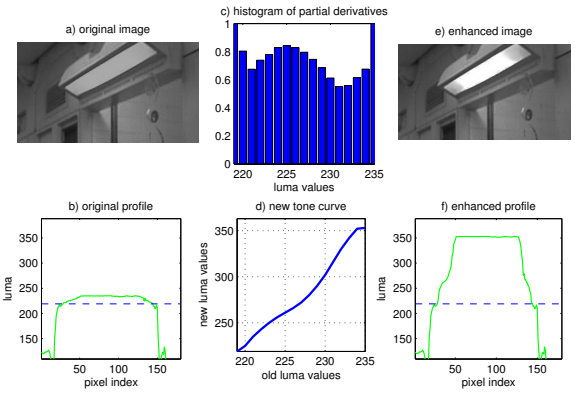

Figure 6: *Enhancement of clipped regions with an adaptive tone curve. a) Initial image and b) its luma profile. c) Histogram of partial derivatives. d) Tone curve derived from the inverted histogram. e) Enhanced region and f) its profile. Dashed lines denote t_2 — the minimum luma level for clipped regions.*

In our approach we use an adaptive non-linear tone-curve to boost clipped areas without amplifying noise. We illustrate our approach on an example of a lamp shown in Figure 6. The fewest artifacts can be expected when large gradients are stretched and small gradients are left intact or moderately enhanced. This is because large gradients are unlikely to represent noise, but also the human visual system is less sensitive to changes of large contrast values (contrast masking) and finally, large gradients often represent object boundaries, where contrast change is the least objectionable. To stretch large gradients, we employ a per-pixel tone-curve that stretches these tone-values, for which the fewest small

gradients are likely to be affected. In order to find such a tone curve, we accumulate within a histogram-like structure the information for all partial derivatives (computed as forward differences) within a clipped region and for all video frames that contain that region (from motion tracking data). For each partial derivative i, whose end-points are a_i and b_i, and $a_i \neq b_i$, we add to our derivative-histogram structure, H, the cost equal $|a_i - b_i|^{-1}$ in the range from $a_i + 1$ to b_i. The histogram is afterwards normalized so that $\max\{H[\cdot]\} = 1$. Figure 7 shows the derivative-histogram after adding three partial derivatives. It indicates that the tone-scale between $a_2 + 1$, b_2 should be stretched the least, and then between $a_3 + 1$, b_1. We can achieve this, if we apply a technique similar to histogram equalization on the inverted derivative-histogram:

$$TC[b] = k \sum_{j=2}^{b} (1 - H[j]) + t_2, \tag{1}$$

where $TC[\cdot]$ is the desired tone-curve, t_2 is the lowest luma value for a clipped region (see Section 4.1) and k is the scaling factor that ensures that the resulting tone-curve does not exceed the maximum boost value m:

$$k = (m - t_2) / \sum_{j=1}^{N} (1 - H[j]), \tag{2}$$

where N is the total number of bins. An example of a tone-curve is shown in Figure 6d. The parameter m is set to 150% of the maximum luma of the original content for *lights* and to 125% for *reflections*. These values were selected to produce visually attractive results, although they do not need to be physically plausible. As a final step we make sure that our tone curve does not compress contrast. This is achieved when $k\,(1 - H[j]) > 1$ for each j. As in [WLRP97], we iteratively update the histogram by lowering bin values and computing a new value of k so that this condition is met up to a certain tolerance threshold. Figure 6 illustrates an example of enhancing a light.

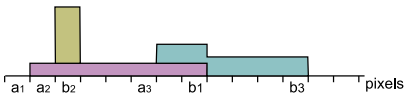

Figure 7: *The histogram of partial derivatives after adding three derivatives.*

Adaptive tone curve reduces noise amplification but it cannot completely prevent it, and it also cannot avoid contouring. To minimize possible artifacts, the tone-curve is applied to the luma values filtered with the bilateral filter ($\sigma_s = 2$, $\sigma_r = 12$), as shown in Figure 8. The bilateral filter not only removes noise, but also interpolates between discrete luma values, thus adding bit-depth precision that prevents contouring. The difference between the filtered region and the original values is added back after contrast stretching to avoid blurring.

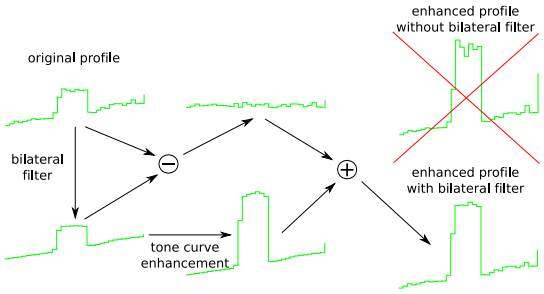

Figure 8: *Enhancement is performed on a filtered image to prevent contouring and noise amplification.*

To restore color, we alter each color channel relative to the luma modifications:

$$C_{new} = C_{old} \cdot \frac{L_{new}}{L_{old}} \tag{3}$$

where C is a color channel (red, green or blue - gamma corrected) and L is luma. Such an approach to color correction prevents color shifts, which are possible when each channel is processed separately. Although the use of linear (not gamma-corrected) color spaces would be preferred in case of HDR restoration, the goal of our method is enhancement and not physical accuracy, therefore it is more convenient to operate in gamma-corrected and approximately perceptually linearized space.

5. Results and validation

Several results of our enhancement compared to the original frames are shown in Figure 9. To validate our method, we first consider the usability of the user-assisted part of our system, then we compare our results with other methods and finally we report the result of a subjective evaluation.

5.1. User interface ergonomics

Work intensive human interaction is a common practice in film enhancement, such as colorization of black and white movies, since the quality of the resulting material cannot be compromised by the failures of an automatic algorithm. The costs of movie post-production and enhancement are often counted in man-months of manual work, which often require artistic skills. Modern stroke-based interfaces can significantly reduce this effort [LLW04]. Our method also offers an efficient stroke-based interface and additionally includes semi-automated classification, which can further reduce the number of required strokes.

We processed 2077 frames of a movie sequence, while gathering statistics from the user interface. The sequence was selected to contain a large number of lights and reflections. There were 11069 clipped regions that had to be labelled. In overall, using motion tracking (Section 4.2) 75%

Figure 9: *Video frames before and after our enhancement. The contrast of the images has been compressed (linear scaling in the logarithmic luminance domain) to match that of a print.*

of the regions could be labelled, leaving 25% of the regions for classification. Due to the low error rate of our classifiers (Table 1 in Section 4.3) only 2.9% of all regions required a manual correction by the user. For this particular video sequence about 20 minutes of work were required to process a minute of a movie. The processing time can be expected to be shorter for sequences with smaller numbers of lights and overall manual effort for a 100-minute movie should not exceed 4-5 man-days of work.

5.2. Comparison with other methods

We compared our method with LDR2HDR [RTS*07], which is the only algorithm that is suitable for video, and HDRTMO [MDS07], which is conceptually the most similar to our approach, although does not guarantee time-coherence necessary for video. We exclude from this comparison the inverse tone mapping method [BLDC06], since it cannot enhance clipped regions in video sequences, and the HDR hallucination [WWZ*07], which is intended for still images only.

In the LDR2HDR method the contrast of a low-dynamic range image is linearly stretched and the brightness of clipped regions is enhanced with a smooth approximation. To compute the smooth brightness enhancement, a binary mask of clipped pixels is blurred with a large Gaussian filter and then rescaled to the fixed range and used as a multiplier of luminance values. To prevent blurring across sharp contrast edges, the brightness enhancement map is limited by a mask that is found using the flood fill algorithm that stops on large image gradients. We implemented the LDR2HDR method as closely as possible in Matlab based on the details given in the original paper [RTS*07] and after correspondence with the authors.

The LDR2HDR algorithm produces very plausible results and is capable of fully automatic, real-time execution on the GPU. Our three points of criticism are the lack of classification, which results in enhancement of objects that should not be enhanced; the inconsistencies in the brightness enhancement map; and temporal flickering. The first problem is visible in Figure 10 center-right, where a white shirt and a part of the wall have been enhanced. The inconsistencies in the brightness enhancement map are caused by a single binary saturation mask that is blurred with a large kernel, so that the actual brightness of enhanced object will depend on the object size, shape and its location with respect to the other objects within a frame. This problem is visible in Figure 10 top-right, where the smaller lamps are darker than the larger explosion and the details far from the center of the explosion are gradually becoming darker. The last major

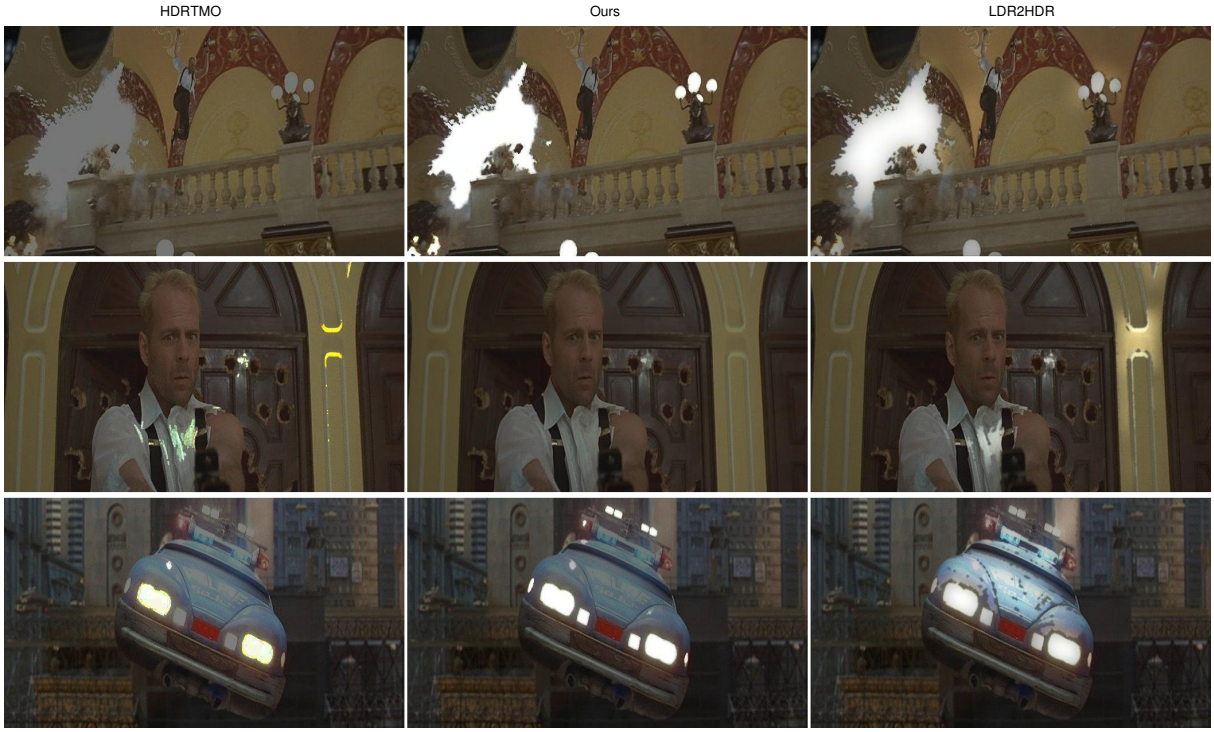

Figure 10: *Our results compared to the HDRTMO and LDR2HDR methods.*

problem is the flickering that can be observed in video sequences (refer to sequences 1, 6 and 9 in the supplementary video). The flickering is due to a) pixels having their value close to 230 (hard threshold used for brightness enhancement), which are once boosted, once left unchanged; and b) growing or shrinking the regions selected by the flood fill algorithm that stops when it encounters large gradients. In our method we avoid these problems by using more conservative conditions on the regions regarded as clipped, and a smooth tone-scale enhancement function, which does not affect much the pixels that are close to the luma threshold (t_2) and at the boundary of clipped regions. Most of the flickering in the LDR2HDR method is visible on diffuse surfaces, which are classified as such by our method and left unchanged.

The HDRTMO method attempts to find specular highlights in an image and then linearly stretches their luma values. The specular highlights are found under the assumption that they are small and bright. The maximum value in the low-pass filtered image estimates the luma of white diffuse surfaces, above which the pixels are classified as specular. Since some specular pixels can have luma values lower than the diffuse white, the original map is expanded using morphological operators until the values reach another threshold, found as the maximum value of a coarsely low-pass filtered image. In our tests we used the original implementation of the algorithm provided by the authors.

The HDRTMO algorithm is fully automatic and computationally inexpensive, although in its original version not suitable for video. We found that it gives excellent results for dark images containing small highlights or light sources. However, the automatic classification fails for images that contain large objects or are bright. This is because large objects do not meet the initial assumptions, and bright scenes often contain diffuse surfaces that are equally bright as lights or specular reflections. Since the distinction between diffuse and specular objects is made solely based on the luma level, the algorithm cannot differentiate between them. This is the reason why no object has been enhanced in Figure 10 top-left, and more objects than desired (shirt, wall, door) have been enhanced in the center-left image. Moreover, the linear luma scaling used in the algorithm leads to amplification of noise, which is high in saturated regions.

We argue that automated classification can never be foolproof and the errors in classification lead to distracting distortions that cannot be tolerated for high quality content. Lack of classification, on the other hand, enhances features that should not be enhanced. The semi-automatic classification proposed in this paper requires a low amount of user interaction, but provides results of highest quality.

5.3. Experimental validation

The enhanced frames shown in Figure 9 may look convincing, but they do not prove that the enhanced video is generally preferred to the original video and to the LDR2HDR method. To validate this claim, we conducted a subjective paired comparison experiment. 14 participants took part in the experiment, all were naive about its purpose. Enhanced and non-enhanced video was shown sequentially in random order to a participant, who had to choose the one that he or she preferred. Each participant ranked this way 9 scenes $\times$ 3 combinations of scene pairs (original video vs. our method; LDR2HDR vs. our method; and original vs. LDR2HDR), shown on a bright LCD display (Barco Coronis Color 3MP Diagnostic Luminance) capable of the the maximum luminance of 650 cd/m^2. We allocated the dynamic range from 1–200 cd/m^2 for the original content and the range from 200-650 cd/m^2 for enhanced features. The viewing conditions were close to the ITU-T Rec. P.910 recommendations.

#	Ranking			Scores			u	χ^2	ζ
1	ours	original	ldr2hdr	1.50	0.86	0.64	0.14	8.29	1.00
2	ours	original	ldr2hdr	1.71	0.71	0.57	0.30	14.86	0.93
3	ours	ldr2hdr	original	1.50	1.29	0.21	0.38	17.71	0.93
4	ours	original	ldr2hdr	1.29	0.93	0.79	-0.01	2.57	0.93
5	ours	ldr2hdr	original	1.43	1.07	0.50	0.17	9.71	0.93
6	ours	original	ldr2hdr	1.50	0.93	0.57	0.17	9.71	0.64
7	ours	ldr2hdr	original	1.71	0.79	0.50	0.32	15.43	0.93
8	ldr2hdr	ours	original	1.21	1.07	0.71	-0.00	2.86	0.71
9	ours	original	ldr2hdr	1.64	1.14	0.21	0.44	20.00	1.00

Table 2: *Quality ranking for the 9 tested scenes. The horizontal lines under the method names indicate no statistically significant difference (multiple comparison test). Scores are given in the same order as the method ranking.*

We analyzed the data using a similar approach as in [LCTS05]. For each scene we computed averaged scores (the number of times the method was preferred), the Kendall coefficient of agreement u (how consistent are the scores between participants), χ^2 test on the coefficient u, and the coefficient of consistency ζ (presence of circular triads in the ranking data). Then, we performed the multiple comparison test to see if the difference in scores is statistically significant ($p = 0.05$). The results of this analysis are summarized in Table 2. The scene 4 and 8 did not pass the χ^2 test, indicating that the participants did not agree on the ranking. For the remaining scenes, our method was either ranked significantly better (3 scenes) or better but with no statistically significant difference to the original scene (3 scenes) or to the LDR2HDR method (1 scene). The surprisingly bad ranking of the LDR2HDR method (in most cases the quality statistically equivalent to the original video) can be attributed to the temporal flickering artifacts, discussed in Section 5.2. The overall ranking for all scenes is:

our method	ldr2hdr	original
1.48	0.79	0.72

which indicates that our method was preferred to both the original scenes and the LDR2HDR method and the difference in preference was statistically significant. All the scenes used in this experiment are included in the supplementary video.

6. Discussion

It is disputable, whether better results can be achieved by enhancing bright image features, or by using the additional dynamic range to stretch overall contrast or brightness. The choice can in fact depend on image content, viewing conditions (bright or dark room), and to some extend on subjective preference. Enhancing bright image features was found advantageous for darker scenes, which would look unrealistic when displayed too bright [MDS06]. It also modifies smaller part of an image, thus reducing the chance of producing artifacts (for example amplifying noise). The other study [AFR*07] found that brighter images are preferred even if they have lower dynamic range, though the study did not report results separately for day-light and night scenes. The thorough study of these considerations is, however, out of scope of this paper.

7. Conclusions

The goal of this study is to improve the subjective quality of video shown on high-contrast displays by enhancing bright video features. We investigated several alternatives methods, from which we choose that which is suitable for video and does not compromise quality of the source content by introducing artifacts.

Unlike other approaches, we do not treat all clipped regions the same way. Instead we distinguish between potential brightness differences in clipped regions. The semi-automatic classifier guarantees almost ideal classification with minimum manual effort. Naive enhancement of clipped regions usually leads to pronounced noise and contouring artifacts. The proposed enhancement algorithm is robust to these problems.

In future work we would like to study the preference for displaying enhanced content on displays of high, but limited dynamic range. For example, it is desirable to know how much dynamic range should be allocated for the enhanced features and how much for the original content. We found the 1:2 division of the dynamic range satisfactory in most cases, but this ratio should be confirmed in a subjective study.

Acknowledgements

We would like to thank Karol Myszkowski, Kaleigh Smith and anonymous reviewers for their helpful comments. The video frames used in the paper are from the movie “The 5th Element”, courtesy of Gaumont Film Company.

References

[AFR*07] AKYÜZ A. O., FLEMING R., RIECKE B. E., REINHARD E., BÜLTHOFF H. H.: Do HDR displays support LDR content? A psychophysical evaluation. *ACM Trans. Graph. 26*, 3 (2007), 38.

[ASI91] ABEL J., SMITH III J.: Restoring a clipped signal. In *Proc of Int. Conf. on Acoustics, Speech, and Signal Processing, 1991* (1991), pp. 1745–1748.

[BKCP93] BURT P., KOLCZYNSKI R., CENTER D., PRINCETON N.: Enhanced image capture through fusion. In *Proc. of 4th Int. Conf. on Computer Vision* (1993), pp. 173–182.

[BLDC06] BANTERLE F., LEDDA P., DEBATTISTA K., CHALMERS A.: Inverse tone mapping. In *Proc. GRAPHITE '06* (2006), pp. 349–356.

[BLfZ07] BHAGAVATHY S., LLACH J., FU ZHAI J.: Mulit-scale probabilisitic dithering for suppressing banding artifacts in digital images. In *IEEE Inter. Conf. on Image Processing (ICIP)* (2007), pp. IV–397–400.

[Bou00] BOUGUET J.: *Pyramidal Implementation of the Lucas Kanade Feature Tracker Description of the algorithm*. OpenCV documentation, Intel Corp., Microprocessor Research Labs, 2000.

[BSCB00] BERTALMIO M., SAPIRO G., CASELLES V., BALLESTER C.: Image inpainting. In *Proc. of the SIGGRAPH'00* (2000), pp. 417–424.

[DF03] DALY S., FENG X.: Bit-depth extension using spatiotemporal microdither based on models of the equivalent input noise of the visual system. In *Color Imaging VIII: Processing, Hardcopy, and Applications* (2003), SPIE, volume 5008, pp. 455–466.

[DF04] DALY S. J., FENG X.: Decontouring: Prevention and removal of false contour artifacts. In *Proc. of Human Vision and Electronic Imaging IX* (2004), SPIE, vol. 5292, pp. 130–149.

[Far01] FARID H.: Blind inverse gamma correction. *IEEE Trans. on Image Processing 10*, 10 (2001), 1428–1433.

[Hun77] HUNT B.: Bayesian methods in nonlinear digital image restoration. *IEEE Transactions on Computers 26* (1977), 219–229.

[LCTS05] LEDDA P., CHALMERS A., TROSCIANKO T., SEETZEN H.: Evaluation of tone mapping operators using a high dynamic range display. *ACM Transactions on Graphics 24*, 3 (2005), 640–648.

[LGYS04] LIN S., GU J., YAMAZAKI S., SHUM H.: Radiometric calibration from a single image. In *Proc. of CVPR'04* (2004), vol. 2, pp. 938–945.

[LLW04] LEVIN A., LISCHINSKI D., WEISS Y.: Colorization using optimization. In *SIGGRAPH '04* (2004), pp. 689–694.

[MDS06] MEYLAN L., DALY S., SUSSTRUNK S.: The reproduction of specular highlights on high dynamic range displays. In *Proc. of the 14th Color Imaging Conference* (2006).

[MDS07] MEYLAN L., DALY S., SUSSTRUNK S.: Tone mapping for high dynamic range displays. In *Human Vision and Electronic Imaging XII* (2007), SPIE 6492.

[MN99] MITSUNAGA T., NAYAR S.: Radiometric self calibration. In *Proc. CVPR* (1999), vol. 1, pp. 374–380.

[Olo05] OLOFSSON T.: Deconvolution and model-based restoration of clipped ultrasonic signals. *IEEE Trans. on Instrument. and Meas. 54*, 3 (2005), 1235–1240.

[RTS*07] REMPEL A. G., TRENTACOSTE M., SEETZEN H., YOUNG H. D., HEIDRICH W., WHITEHEAD L., WARD G.: LDR2HDR: On-the-fly reverse tone mapping of legacy video and photographs. *ACM Transactions on Graphics (Proc. SIGGRAPH) 26*, 3 (2007).

[SDM05] SNYDER J., DOERSCHNER K., MALONEY L.: Illumination estimation in three-dimensional scenes with and without specular cues. *Journal of Vision 5*, 10 (2005), 863–877.

[She03] SHEN J.: Inpainting and the fundamental problem of image processing. *SIAM News 36*, 2 (2003).

[SHS*04] SEETZEN H., HEIDRICH W., STUERZLINGER W., WARD G., WHITEHEAD L., TRENTACOSTE M., GHOSH A., VOROZCOVS A.: High dynamic range display systems. *ACM Transactions on Graphics 23*, 3 (2004), 757–765.

[TLQS03] TAN P., LIN S., QUAN L., SHUM H.: Highlight removal by illumination-constrained inpainting. In *Proc. of IEEE Int. Conf. on Computer Vision (ICCV'03)* (2003), p. 164.

[TM98] TOMASI C., MANDUCHI R.: Bilateral filtering for gray and color images. In *Proc. of ICCV* (1998), pp. 836–846.

[WBNF06] WEIDENBACHER U., BAYERL P., NEUMANN H., FLEMING R.: Sketching shiny surfaces: 3D shape extraction and depiction of specular surfaces. *ACM Trans. on Applied Perc. 3*, 3 (2006), 262–285.

[WLRP97] WARD LARSON G., RUSHMEIER H., PIATKO C.: A visibility matching tone reproduction operator for high dynamic range scenes. *IEEE Trans. on Vis. and Comp. Graph. 3*, 4 (1997), 291–306.

[WWZ*07] WANG L., WEI L., ZHOU K., GUO B., SHUM H.-Y.: High dynamic range image hallucination. In *Rendering Techniques 2007: 18th Eurographics Symposium on Rendering* (2007), pp. 321–326.

[ZB04] ZHANG X., BRAINARD D.: Estimation of saturated pixel values in digital color imaging. *Journal of the Optical Society of America A 21*, 12 (2004), 2301–2310.