Tonemapping approaches for rendering sparse HDR-lightfields

Master thesis

in the field of computer science

Universität des Saarlandes

Presented by

Steffen Altmeier

Supervisor

Private Docent Karol Myszkowski

Saarbrücken September 29, 2018

Abstract

High dynamic range (HDR) imaging has been an important research-area in computer graphics and image processing in the past decades. With the increasing interest in lightfield (LF) capturing and rendering, the need for HDR imaging methods tailored towards LF-data becomes more relevant. To use HDR images on a standard display, the dynamic range of the HDR image is required to be compressed, to match the dynamic range supported by the display in a process known as tonemapping. This thesis is centred around the tonemapping aspect of HDR imaging in the context of LF rendering. Existing tonemapping operators (TMOs) are examined and analysed in regards to their suitability in the context of LF rendering. To accomplish this goal, a LF rendering system capable rendering of sparse HDR-LFs in real-time has been developed. The differing ways in which three existing TMOs may be applied to HDR-data in such a LF rendering system are studied. This study is focussed on which modifications can be considered to better adapt the existing TMOs to the requirements of LF rendering. Lastly, a novel processing method for HDR-LFs has been devised based on results and insights obtained from the above analysis.

Zusammenfassung

Das verarbeiten von Bildern mit hohem Dynamikbereich (HDR) war über die vergangen Jahrzehnte ein Kerngebiet der Forschung im Bereich der Computer-Graphik. Mit wachsendem Interesse in Lichtfeld-Photographie und Reproduktion steigt die Relevanz neuer HDR-Verarbeitungsverfahren, welche speziell auf die Anforderungen von HDR-Lichtfeldern zugeschnitten sind. Um HDR Bilder auf einem Standard display darstellen zu können, muss der dynamikbereich des HDR Bildes auf den unterstützten Dynamikbereich des Display mittels Dynamikkompression reduziert werden. Diese These konzentriert hierbei auf den Aspekt der Dynamikkompression im Kontext von HDR-Lichtfeld Reproduktion. Existierende Dynamikkompressionsverfahren werden in Hinsicht auf deren Eignung zum Einsatz in Zusammenhand mit HDR-Lichtfeld Reproduktion untersucht und analysiert. Um dieses Ziel zu erreichen wurde im Rahmen dieser These ein HDR-Lichtfeld-Reproduktionsverfahren zum visualisieren dünn-abgetasteter Lichtfelder in Echtzeit entworfen. Die Anwendungsmöglichkeiten drei unterschiedlicher Verfahren zur Dynamikkompression im HDR-Lichtfeld-Reproduktionsverfahren sind teil der Studie. Das Ziel dieser Studie ist es herauszufinden, welche Modifikationen dieser Verfahren nötig und möglich sind um die Verfahren auf die Anforderungen von HDR-Lichtfeld-Reproduktion anzupassen. Letztlich wird mit Hilfe von Erkenntnissen und Ergebnissen, welche im Rahmen dieser Studie gewonnen wurden, ein neues Verarbeitungsverfahren für HDR-Lichtfelder vorgestellt.

Table of Contents

1 Introduction

1.1 Overview

In the era before HDR imaging research, image-formats were limited to the colour-ranges supported by displays, typically using an 8-bit integer value-range per colour-channel. A display was assumed to deliver a peak-brightness of approximately 250-300 cd/m^2 for flat-screen displays and below 100 cd/m^2 for CRT displays.
Images provided in such a display-referred format can be directly displayed and thus are considered display-ready images. Display-ready formats make use of colour-spaces, in which the differences between colour-values are perceived in a linear fashion by the human visual system (HVS). This is in contrast to using colour-spaces with linear behaviour in terms of luminance-values. Luminance determines the intensities of red, blue and green colour-channels by taking the sensitivity of the corresponding rods in the human eye to light of differing wavelengths into account. In case an image is given in luminance-values within the dynamic range of the display, it can be converted into display-referred values via gamma-correction. A gamma-correction provides a close approximation of the HVS in terms of luminance-perception.
A High Dynamic Range (HDR) image format allows for the luminance values of the colour-channels to be stored as continuous floating-point values. Due to a use of luminance-values in the scene without accounting for human perception, such formats are considered scene-referred formats.

As the capabilities of current display technology are limited in terms of displayable luminance-ranges and colour-gamut, high dynamic range content cannot be directly reproduced on the screen. The luminance-values stored in HDR-images are required to be mapped into a display-referred colour-space. This process is known as tonemapping and the way this mapping is performed is defined by a tonemapping operator (TMO). The main challenge for TMOs is the compression of luminance to match the dynamic range of the display. Typically, TMOs attempt to preserve properties of the image, such as appearance and contrasts, to enable a resemblance to an observation of the real scene for the HVS.

Tonemapping is still relevant today and for many years to come as recently developed display technology, often referred to as HDR-display technology, is capable of displaying a higher, but still limited, dynamic range. Common HDR standards, such as the HDR 10 standard, allow for an increased colour-gamut, a discretization using 10 bits per colour-channel, as well as 1000 cd/m^2 of peak-brightness. While this increase in dynamic range improves tonemapping results, tonemapping is still an integral part of preparing images for display on HDR-display technology.

The development of **display technology** is not restricted to improvements of dynamic range and colour-gamut, but has developed from single-view displays, to stereo-displays and is now developing towards many-view displays, so-called lightfield-displays. A lightfield (LF) is a collection of many views corresponding to the same scene, commonly generated by capturing the scene with an array of cameras or a specialised lenslet-camera as shown in figure 1. The effects and properties achieved by such LF-displays vary depending on their design and allow for viewing experiences unlike what single-view or stereo-displays can achieve [21, 38, 45, 32].

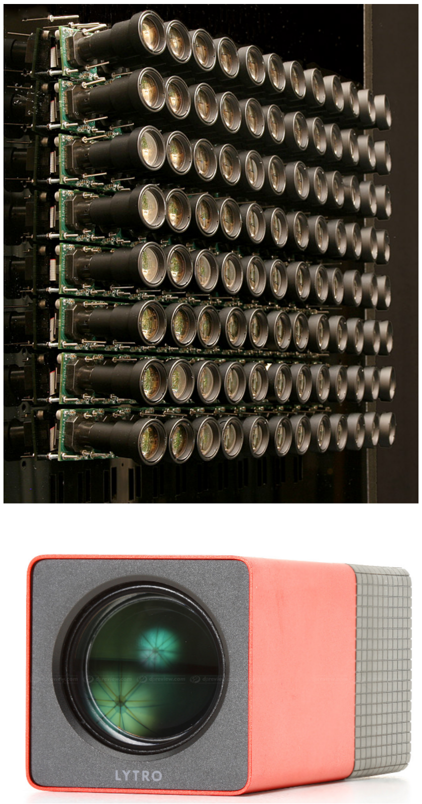


Figure 1: Camera-array (top) [22] and lenslet-camera (bottom) [14]

1.2 HDR-lightfield rendering

Lightfields are, in addition to use with specialized hardware, suitable for image-based rendering applications targeted to standard displays [25, 33]. In the LF rendering system presented in this thesis, new display-ready views, as observed through a virtual camera, are synthesized from data captured in a static HDR-LF. In an HDR-LF, the key-views of the LF are captured in an HDR image format and are not directly ready for display. Using a LF rendering system, a scene captured as an HDR-LF can be explored on a standard display.

The desired result delivered by an HDR-LF rendering system is a series of novel views, for which the geometry of the scene, as well as material appearances, closely match the real scene. Additionally, the tonemapping applied ought to behave consistently in the angular and spatial domains respectively.

In a typical real-time computer graphics application, novel views are generated by rendering a scene consisting of triangle meshes. The appearance of surfaces in the scene is hereby determined according to lighting conditions and material properties during rendering.
For LF rendering, the colour-information of novel views may be gathered directly from the captured key-views of the LF during view-synthesis instead. The key-views contain information about the surface-appearance under the recorded lighting conditions form different viewpoints. This allows for a preprocessing of the LF-data to obtain image statistics or tonemapped key-views before view-synthesis is performed. TMOs, which are commonly considered unsuitable for real-time tonemapping, may still be applied to LF rendering due to the above. In addition, image statistics pre-computed for the key-views can be utilized in the tonemapping process.

The lightfield rendering system presented in this thesis has been applied to synthetic scenes only. Real world scenes were not used due to the lack of captured HDR-LF datasets with established correspondences reliable enough to serve as a ground-truth. Synthetic scenes allow for ground-truth correspondences to be used during view-synthesis and enable an evaluation of results via ground-truth comparisons. The scenes used for evaluation can be designed to be challenging specifically for both, the view synthesis and the tonemapping-aspects of a HDR-LF rendering system.
An effort has been made to allow for a rendering of real world datasets using the lightfield rendering system presented in this thesis to deliver suitable results with only minimal modifications. Modifications are most likely required to account for established correspondences in real world datasets deviating from the ground truth. During view-synthesis, this may lead to the introduction of artefacts when not accounted for.

1.3 Application and relevance

HDR-LF rendering has many possible applications and this thesis focuses on three prominent application-areas.

Virtual reality

In a virtual reality (VR) scenario, when incorporating body-motion into the VR experience as a means of navigation and control, the explorable scene is often limited in space. This is due to the limitations of movement in the real world. LFs capture the scene from multiple views and are poorly suited for rendering large scenes with free movement within the scene. When applied to small scenes, however they can deliver a high fidelity experience, which may be difficult to be matched by conventional 3D-rendering methods. This is due to the high requirements to frame-rate and resolution in a VR-scenario, which are often met at the expense of visual fidelity in the scene in the form of lighting and triangle-budget. This is the case as conventional 3D-rendering methods do not perform independently of scene complexity unlike LFs [33] and as such the quality of scenes is restricted for conventional 3D-rendering methods. The use of real world LF-datasets, as discussed in the next paragraph, is particularly interesting for VR to achieve a photo-realistic rendering experience. Alternatively, a use of key-views obtained via ray-tracing is possible as well and can, depending on the time and effort invested into the scene, be used to render a high-quality HDR-LF.

Scenes from real world datasets

Scenes from the real world, captured as LFs by specialized lenslet-cameras or camera-arrays, can be displayed using novel LF-displays or via LF-rendering targeted at standard displays. A conversion to a 3D-mesh, capable of being rendered by conventional means, is challenging and computationally inefficient. Material-appearances may be lost during the conversion and extensive manual-editing of the geometry may be necessary. The initial triangle-count may be too high to allow for real-time rendering or conversion-artefacts may be visible in the geometry. Alternatively, to faithfully reproduce the scenes using triangle-meshes, the required triangle-count my be prohibitively high.

The creation of high-quality 3D-scenes from scratch, on the other hand, takes considerable time and effort and even with a high investment into a 3D-scene, a photo-realistic appearance on par with LFs captured from the real world is difficult to achieve.
The use of real world datasets is often a requirement in case an exact reproduction of real world scene is required for documentary reasons and a recreation of the scene from hand is not acceptable. While recreations may approach the real scene, a use of LF-capturing can directly deliver a faithful reproduction.

When using HDR-LF rendering for captured content, material appearances are preserved in the HDR-data and synthesized views may reflect the scene-appearance in the real world closely after tonemapping. While the use of captured HDR-LF data is intended for praxis over the use of synthetic data, for the purpose of this thesis no scenes from real world were used due to reasons outlined in section 1.2.

Lightfield-display technology

Lightfield-displays offer significant advantages over current display technology. When applied to TVs for instance, a LF-display does not require glasses for stereo-3D and allows for an unlimited number of simultaneous viewers and may replace current 3D-TV technology in the future [10, 30, 20]. To enable a display of HDR-LFs on such a device, a tonemapping of the key-views of the LF is required and a development of tonemapping for HDR-LFs can be expected to take place alongside the development of LF-displays. A tonemapping of key-views and adaptations of TMOs to LFs are explored in this thesis in a LF-rendering context. As the adaptations made are focussed on reducing tonemapping-inconsistencies between adjacent key-views, these adaptations may also be used to prepare HDR-LFs for novel LF-displays.

2 Related work

2.1 Existing approaches to tonemapping

To display image-content exceeding the dynamic range of the display, the luminance-values contained in the HDR-image must be tonemapped to values within the dynamic range and colour-space supported by the display. A multitude of TMOs with a large variety of properties, characteristics and capabilities have been developed for various application-scenarios [19] [40].

As TMOs are mostly focussed on compressing the overall dynamic range of an HDR-image, brightness measures for both display-referred and scene-referred image-formats are introduced. Luminance as a measure of brightness for scene-referred image-formats is calculated as $Y = 0.2126 \cdot R + 0.7152 \cdot G + 0.0722 \cdot B$ based on the contributions of colour-intensities captured by the rods in the human eye. When applying a gamma-correction to the colour-channels or using display-ready image-formats, generating Y as above yields luma instead, which closely models brightness perception in the HVS.

2.1.1 Categorisation of tonemapping operators

The specific goals and processing methods applied vary strongly between different TMOs. For the purpose of this thesis, a categorisation of TMOs in terms of processing method, tonemapping intent as well as processing-speed is made. This categorisation closely resembles and is inspired by the categorisation of TMOs as proposed in [19].

Processing methods

Commonly a distinction between operators including a processing of the local neighbourhood in the HDR-image, local operators, and operators exclusively relying on global image-statistics, so-called global operators, is made. For global operators, the same mapping-function, commonly referred to as a tonemapping-curve, is used to all pixels regardless of their position in the image. As local operators incorporate information of the local neighbourhood, they may deliver a mapping which is more suitable for the local neighbourhood when compared to global operators.

For tonemapping of LF-data in particular, the robustness of the global image-statistics used in the tonemapping process in regards to outliers is of additional interest. This is the case as adjacent key-views share a large amount of image-content between each other. Only TMOs using exclusively image-statistics, to which many pixels of the image contribute, are considered robust in this context. Common non-robust image-statistics include maxima and minima, such as the peak luminance of the scene. In the case of peak luminance, only one pixel, the pixel with the highest luminance, has a contribution to the image-statistic. In practice, percentiles are commonly used to reduce the impact of outliers and increase the robustness of statistics, such as peak luminance, to which naively only one pixel contributes. By incorporating these characteristics, an additional distinction, in regards to processing methods, between robust and non-robust TMOs is made.

Tonemapping intent

While many different goals can be pursued in the tonemapping process, due to a necessary reduction in dynamic range, only a few different goals can typically be achieved by a TMO at the same time. This is the case as some tonemapping goals are incompatible with each other and cannot be achieved at the same time. The set of goals chosen in a TMO hint at a specific intent inherent to the TMO.

A TMO intended for visual-system-simulation (VSS), is centred around replicating processes in and properties of the human visual system and tonemapping-goals are chosen accordingly. This includes processes, such as loss of colour and contrast in night-vision or glare, which result in a lower subjective quality of the output image. Overall, the aim is to produce output images minimizing the perceptual difference between viewing the real scene and the image shown on the display.

TMOs intended for scene-reproduction (SRP) focus on preserving one or multiple quantifiable attributes of the scene in the output image. Goals such as relative-brightness preservation, preservation of colour-appearance and preservation of contrasts are common for such TMOs. Due to the overall reduction in dynamic range in the output image, the properties preserved by the TMO often appear exaggerated to a user.

In general, a best subjective quality (BSQ) of the output is sometimes preferable to the preservation of scene-attributes or an accurately simulated appearance. Operators in this category tend to produce exaggerated contrasts as well as overly-vibrant colours as humans favour high-contrast scenes up to a certain extent. Artistic goals, such as producing an output in line with a certain colour-palette, can be included in the design of TMOs pursuing subjective quality.

Processing-speed

Processing-speed, i.e. the time required to process a single image, is divided into 3 tiers of speed depending on how the processing-speed restricts the application of a TMO with respect to LF-rendering.

The highest tier of processing-speed, a real-time processing speed, allows for the TMO to be applied to newly-synthesized views produced by a LF rendering system. They can largely avoid a pre-processing of the LF and are the least restrictive when it comes to an adaptation of the tonemapping to changing lighting conditions in the scene.

Operators belonging to the tier of interactive operators are, while not quite fast enough to be applied to newly-synthesized views, capable of being applied to the key-views of a LF in a time-frame of less than a minute. Operators in this category can be used to allow for use of HDR-LFs as inputs and the application of selected tonemapping parameters to the key-views of the LF before viewing the scene. It is notable however, that a pre-computation of image-statistics and data-layers for each key-view of the LF can help create a real-time version of such operators.

Operators are considered offline-operators in case an application to all key-views upon loading a HDR-LF is not feasible for a LF rendering system. A tonemapped LF is generated externally once and supplied as display-ready LF-dataset to the LF rendering system. An adjustment to different viewing conditions is in general not possible for a LF rendering system and a new display-ready LF-dataset has to be generated externally for each desired viewing-condition.

2.1.2 Tonemapping operators used in this thesis

This thesis examines three common TMOs and how they can be applied to and interact with LF rendering. To study and analyse such interactions, the technical specifics of the TMOs applied are crucial. For this reason, the operators examined were chosen to differ in terms of their processing methodology. The chosen operators additionally differ in terms of their tonemapping intent and can be considered to be some of the most common and basic operators used for their respective categorisation. More complex operators often incorporate ideas from these basic operators into their design.

Bilateral-filtering

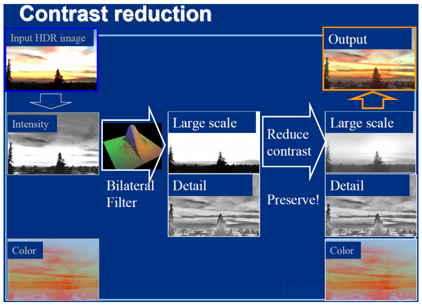


Figure 2: Processing-scheme for bilateral-filtering [18]

The bilateral-filtering TMO [24] is a local TMO aimed at preserving local contrasts in the input image, despite compressing the overall dynamic range of the image. This is achieved by separating the image into two layers of luminance-content, a base-layer and a detail-layer. The base-layer containing the low-frequency luminance content is generated by application of bilateral filtering [42] to the input-luminance in the logarithmic domain. A subtraction of the base-layer from of the input-luminance in the logarithmic domain yields the detail-layer containing high-frequency luminance-content. Compressing only the base-layer by a scale factor in the logarithmic domain achieves the goal of reducing the overall dynamic range, while simultaneously preserving the local luminance-contrast as shown in figure 2. The target-luminance L_{target} is calculated as

$$log(L_{target}) = log(base) \cdot compression + log(detail) - log(brightness),$$

where the parameter *brightness* allows for a global brightness adjustment. The output-luma L_{out} is calculated via gamma-correction of the target-luminance L_{target}.

A division of the colour-intensities of a pixel by its luminance yields the relative colour-intensities of the pixel centred around the value 1. An adjustment to the relative colour-intensities may be performed as a way of controlling colour-contrasts in the output image. A multiplication of the relative colour-intensities with the output-luma yields the output of the TMO.

The goal of local contrast preservation is in accordance with the intent of scene-reproduction, but depending on parameter-settings may generate results with an exaggerated appearance of local contrasts and colours. Due to the large-scale filtering required to generate a sufficiently smooth base-layer, a naive implementation of this TMO allows for interactive processing-speeds, but improvements can be made to achieve real-time processing-speeds [24].

Dynamic Range Reduction inspired by Photoreceptor Physiology (Reinhard 2005) [16]

This TMO, commonly referred to as *Reinhard 2005* throughout this thesis, mimics the photoreceptor response of the human eye towards the colour-stimulus provided by the intensities of each pixel. The adaptation level of photoreceptor I_{a_i} for each colour-channel i must first be determined to estimate the response to the stimulus. To do so, the TMO uses colour-averages $C[i]_{avg}$ for each channel i and the luminance-average L_{avg} of the image to account for the overall lighting conditions in the scene. The colour-values $C[i]$ and luminance L of the pixel to be tonemapped are used to account for local adaptation. The intensity a photoreceptor is adapted to is assumed to follow a linear interpolation of the global intensity of the image I_{global_i} and a local intensity I_{local_i} of the pixel to be tonemapped. I_{local_i}, I_{global_i} and $C[i]_{avg}$ are determined as follows:

$$I_{local_i} = c_a \cdot C[i] + (1 - c_a) \cdot L$$

$$I_{global_i} = c_a \cdot C[i]_{avg} + (1 - c_a) \cdot L_{avg}$$

$$I_{a_i} = l_a \cdot I_{local_i} + (1 - l_a) \cdot I_{global_i},$$

where c_a and l_a are parameters for colour adaptation and brightness adaptation set in the range between zero and one. The response of the photoreceptor to the stimulus is modelled by a modified sigmoid-curve for each colour channel $C[i]$ and its corresponding adaptation level I_{a_i}:

$$C[i] = \frac{C[i]}{C[i] + (e^{-b} \cdot I_{a_i})^m}$$

The shape of the curve is controlled by parameters for desired brightness b and a measure of contrast of the input image m. An automatic choice of this contrast-measure includes image-statistics with a low robustness towards outliers in the form of maximum- and minimum-luminance. The impact of m on the tonemapping is shown in figure 3. Finally, the tonemapped image is normalized by luma. This normalization relies on image-statistics with low robustness towards outliers and ensures the full luma-range supported by the display is utilized for tonemapping.

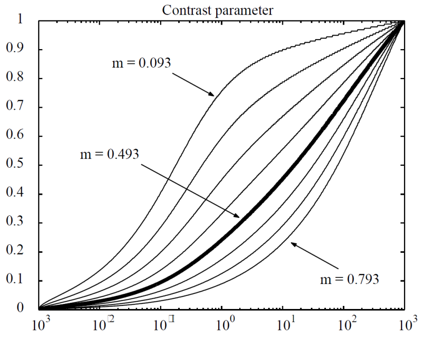


Figure 3: Tonemapping-curves for different contrast-parameters [16]

The intent of the Reinhard 2005 TMO is to simulate parts of the human visual system, specifically the response of a photoreceptor. Due to its lack of local processing, this operator is fast and suitable for real-time applications.

Histogram adjustment

To obtain high-contrast tonemapping results, the histogram adjustment TMO [41] uses a tonemapping-curve inspired by histogram-equalisation. The process of histogram-equalisation is a common method in image-processing with the goal of global contrast-enhancement. For grayscale-images with distinct gray-values, an image histogram maps each gray-value to its occurrence-frequency in the image. The output of histogram-equalisation for a gray-value is given

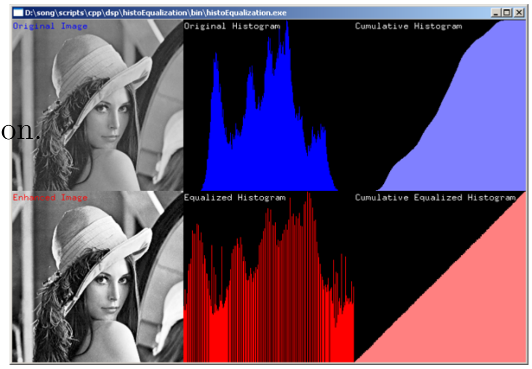

D:\song\scripts\cpp\dsp\histoEqualization\bin\histoEqualization.exe
Original Image
Original Histogram
Cumulative Histogram
Enhanced Image
Equalized Histogram
Cumulative Equalized Histogram

Figure 4: Histogram-equalization and cumulative distribution function [9]

as the sum of all probabilities in the histogram between the lowest gray-value, typically zero, and the given gray-value. This sum is commonly referred to as cumulative histogram or cumulative distribution function. The slope of the cumulative distribution function is by design proportional to the occurrence-frequencies in the histogram. The application of a histogram-equalization is shown in figure 4. For application of histogram-equalisation on floating-point images, distinct gray-values in the histogram are replaced by gray-value-ranges, so-called bins. The amount and distribution of these bins impacts the quality of the mapping.

When naively applying histogram-equalisation as a form of tonemapping, no restrictions on the slope of the cumulative distribution function are made and as such, small luminance-contrasts in the input may be vastly exaggerated in the output.

Instead of modifying the cumulative distribution function to obtain a better result, the input-histogram is adjusted instead by truncating bins with a pixel-count exceeding a set percentage of pixels in the image. This truncation of bins is performed to limit the slope of the cumulative distribution function such that contrasts between pixels after tonemapping do not exceed the respective contrasts in the HDR-image. After truncation of the histogram, the total number of pixels contained in the histogram has changed and as such, occurrence-frequencies are increased. The truncation-process is thus repeated multiple times until only a set percentage of the number of pixels in the image has been truncated in the histogram. A pseudo-code of the truncation-process is shown in figure 5.

```
boolean function histogram_ceiling()
tolerance := 2.5% of histogram total
repeat {
  trimmings := 0
  compute the new histogram total T
  if T < tolerance then
    return FALSE
  foreach histogram bin i do
    compute the ceiling
    if $f(b_i)$ > ceiling then {
      trimmings += $f(b_i)$ - ceiling
      $f(b_i)$ := ceiling
    }
} until trimmings <= tolerance
return TRUE
```

Figure 5: Pseudo-code for histogram truncations [41]

A division of the colour-intensities of a pixel by its luminance yields the relative colour-intensities of the pixel centred around the value 1. An adjustment to the relative colour-intensities may be performed as a way of controlling colour-contrasts in the output image. Multiplying the relative colour-intensities with the target-luma of the pixel, obtained through the use of the cumulative distribution function, yields the output of this TMO.

As only the histogram of the input image is used during the tonemapping-process and the tonemapping goal is to reproduce the visibility of contrast in the output image, this global TMO uses robust image-statistics and the tonemapping intent is to achieve a best subjective quality. The collection of the histogram-data, as well as the repeated trimming of bins restricts this TMO to an interactive processing speed.
Out of the class of histogram-based TMOs, a histogram adjustment is one of the most basic approaches and is used to represent this common class of TMOs for the purpose of this thesis. Other histogram-based TMOs can be expected to show similar tonemapping behaviour when applied to LF-data, as they mostly limit or modify the slope of the mapping function in a similar fashion.

2.2 Existing approaches to lightfield capturing and lightfield rendering

Usage of LF-data is becoming a more popular approach for real-time computer-graphics and rendering as discussed in section 1.3. As new technology, such as VR headsets, give rise to high requirements for frame-rate and resolution, common 3D-rendering approaches struggle to deliver the high fidelity required for an immersive user experience. LF rendering is uniquely well-suited to deliver photo-realistic representations of 3D scenes in real-time largely independent of scene-complexity [33].

Lightfield capturing and structure

Lightfield-based rendering allows for scenes captured from the real world to be used as input-data for LF rendering with comparatively little effort. The methods used for LF capturing vary and deliver datasets with differing properties aimed at different fields of use.

The most common approach for LF acquisition and storage is the capture of the scene using a camera-array or a specialized lenslet-camera.

LFs captured using camera-arrays, as shown in figure 6, generally feature a large distance between neighbouring cameras. LFs with comparatively large disparity values between neighbouring key-views are considered spatially-sparse and for LF rendering correspondence between key-views is required to synthesize new views.

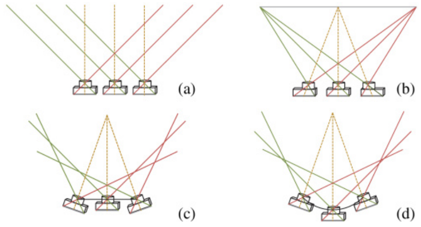


Figure 6: Capture using different camera-array setups [44]

Lenselet-cameras allow for the capture of a LF using a single camera much in the same way as using a conventional camera. Figure 7 shows a schematic of the inner structure of a lenslet-camera. They are a portable and more affordable alternative, however the size of the camera-lense restricts the spatial extent of the captured lightfield. This often makes LFs captured by lenslet-cameras less viable for LF rendering as only a small spatial extent of the scene can be explored. For such spatially dense LFs, the use of correspondences for synthesizing new views is often not required.

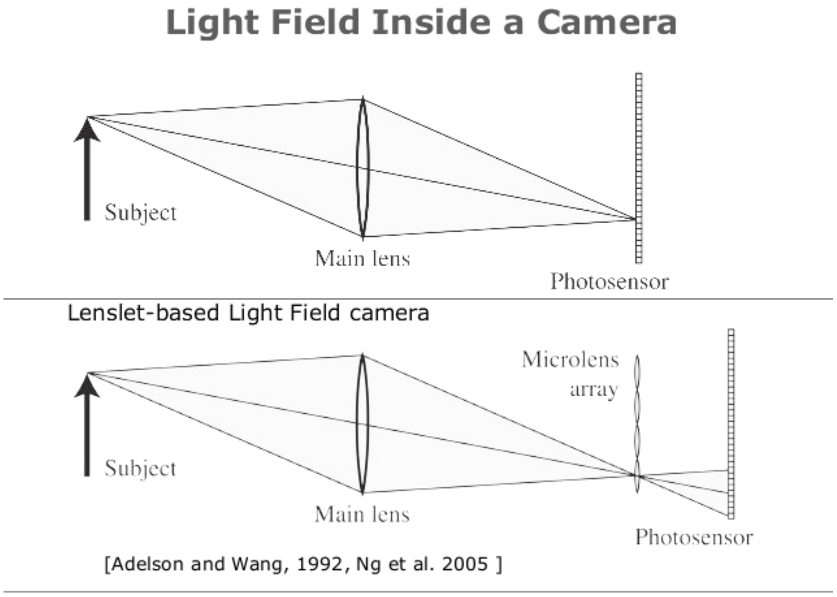

Light Field Inside a Camera
Subject
Main lens
Photosensor
Lenslet-based Light Field camera
Microlens array
Subject
Main lens
Photosensor
[Adelson and Wang, 1992, Ng et al. 2005]

Figure 7: Capture using a lenslet-camera [7]

While LF-data captured via a camera-array or a specialized lenslet-camera contains a very limited field of capture of the scene in the angular domain, the density of key-views in the spatial domain is comparatively high. Such a structure is beneficial for a use on flat LF-displays or for reconstructions using LF rendering targeted at standard-displays as the intended viewing angles of the display are limited in a similar fashion as the field of capture.

For applications in need of a wide field of capture, such as VR-applications, rotating camera-arcs can be used to deliver up to a full 360° capture of the scene. This method has been applied to obtain a capture of the discovery space-shuttle by Google for VR-applications [15], as shown in figure 8. For such methods of capture, the density of key-view in the spatial domain is often limited as a trade-off.

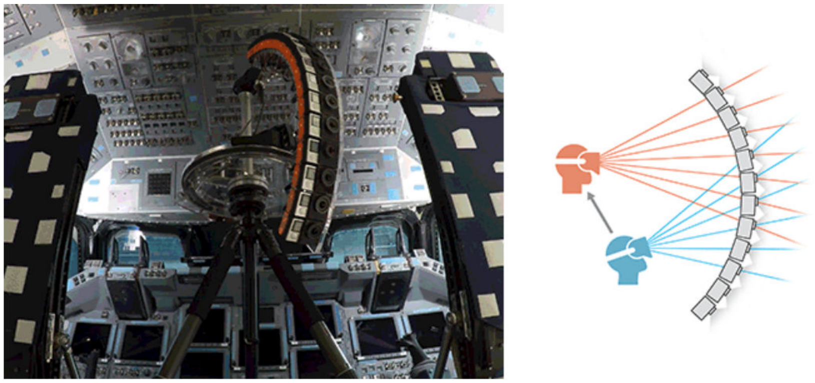


Figure 8: Capture and rendering setup for the space-shuttle [15]

Lightfield rendering and display

To visualize scenes captured as LF-data on a standard display, LF rendering is required to generate new views of the scene. The approaches taken solve the rendering task by finding content in the LF closely matching the angular and spatial properties of each ray collected by a virtual camera. The differences in methodology are thus mostly restricted to the unique properties of the capturing-setup used [25, 33] as well as depth estimation for pixels in the novel view [34, 35].

Displays capable of displaying entire LFs at the same time on the other hand often do not require any rendering-process as the devices themselves restrict the angular-range a pixel is visible from. By construction and design, only scene-content relevant to the viewing-scenario can be observed by a user. The display technology hereby leverages LFs to deliver differing viewing-experiences, which are commonly unable to be emulated by conventional display technology. There is a multitude of novel display-technologies in development incorporating the use of LFs into their design [38, 45, 21].

As Stereo-displays allow for views intended for the right and left eye to be displayed at the same time, LF-displays are able to display an entire LF at the same time. For LF-displays in the form TVs, which views are visible to a user on the screen depend on the viewing angle. This viewing angle differs for the left and right eye and varies for different positions on the screen. Thus, a change in head-position of the user results in a change of viewing-direction on the LF-display. The scene appears to extend in-front and behind the screen and allows for depth perception without glasses for an unlimited amount of simultaneous users [30]. This benefit often comes at a cost, typically display-brightness and the display-resolution visible to the user.

Near-eye LF-displays used in head-mounted devices, such as VR-headsets, allow for the emulation of accommodation and vergence with the use of LFs. This way, objects in the scene can be focused on by the user and accommodation-vergence conflicts common in stereo-vision can be avoided [29, 28]. An example of a near-eye LF-display is shown in figure 9.

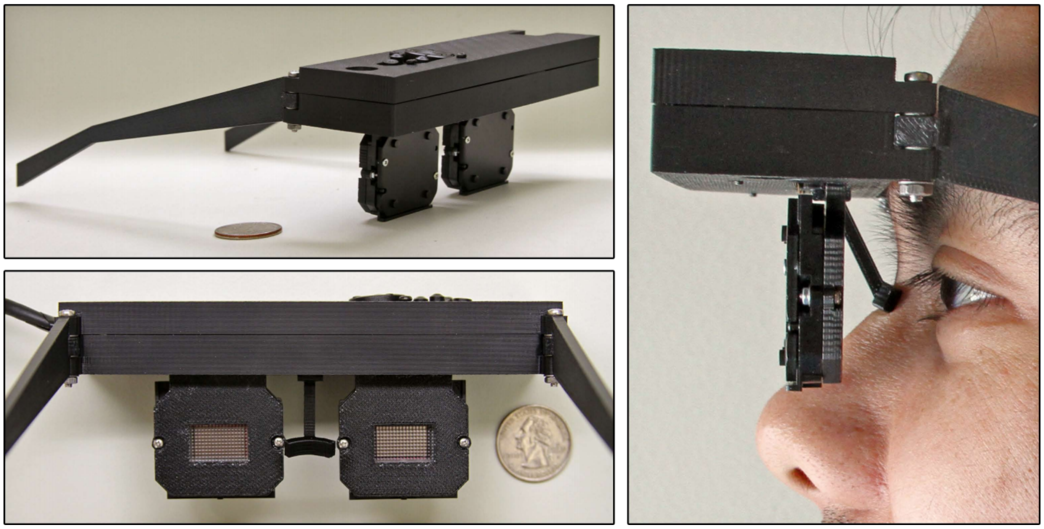


Figure 9: Prototype for a near-eye lightfield-display [29]

Translucent LF-displays capable of 360° of viewing angle allow for spatially-small 3D objects to be observed in a holographic fashion. Figure 10 shows a such a visualization of a tie-fighter model on a 360° LF-display [26], which is reminiscent of scenes from the film-series Star Wars.

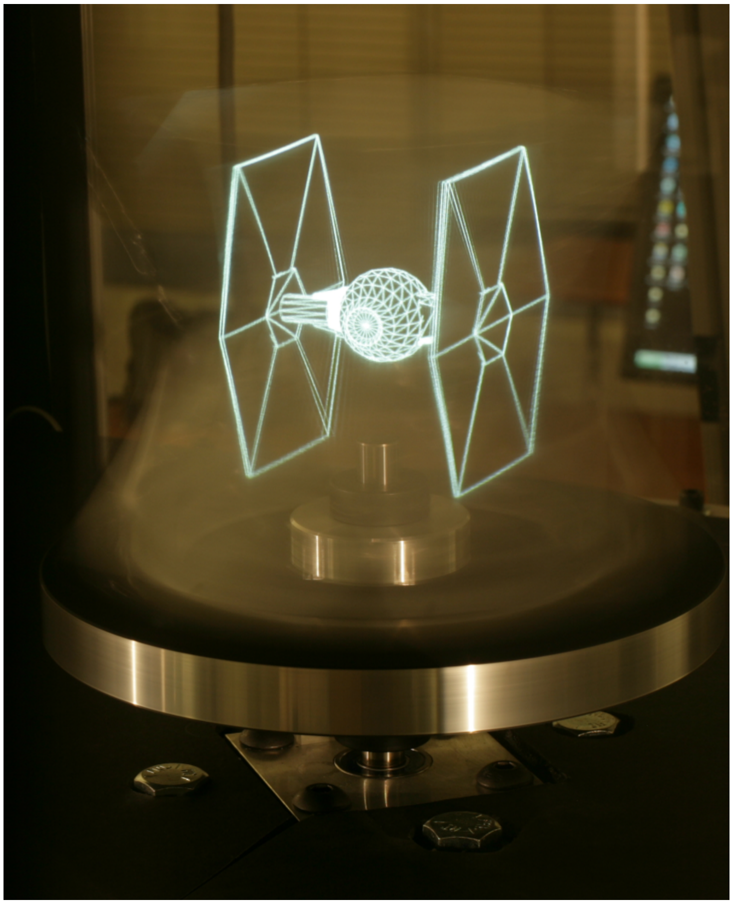


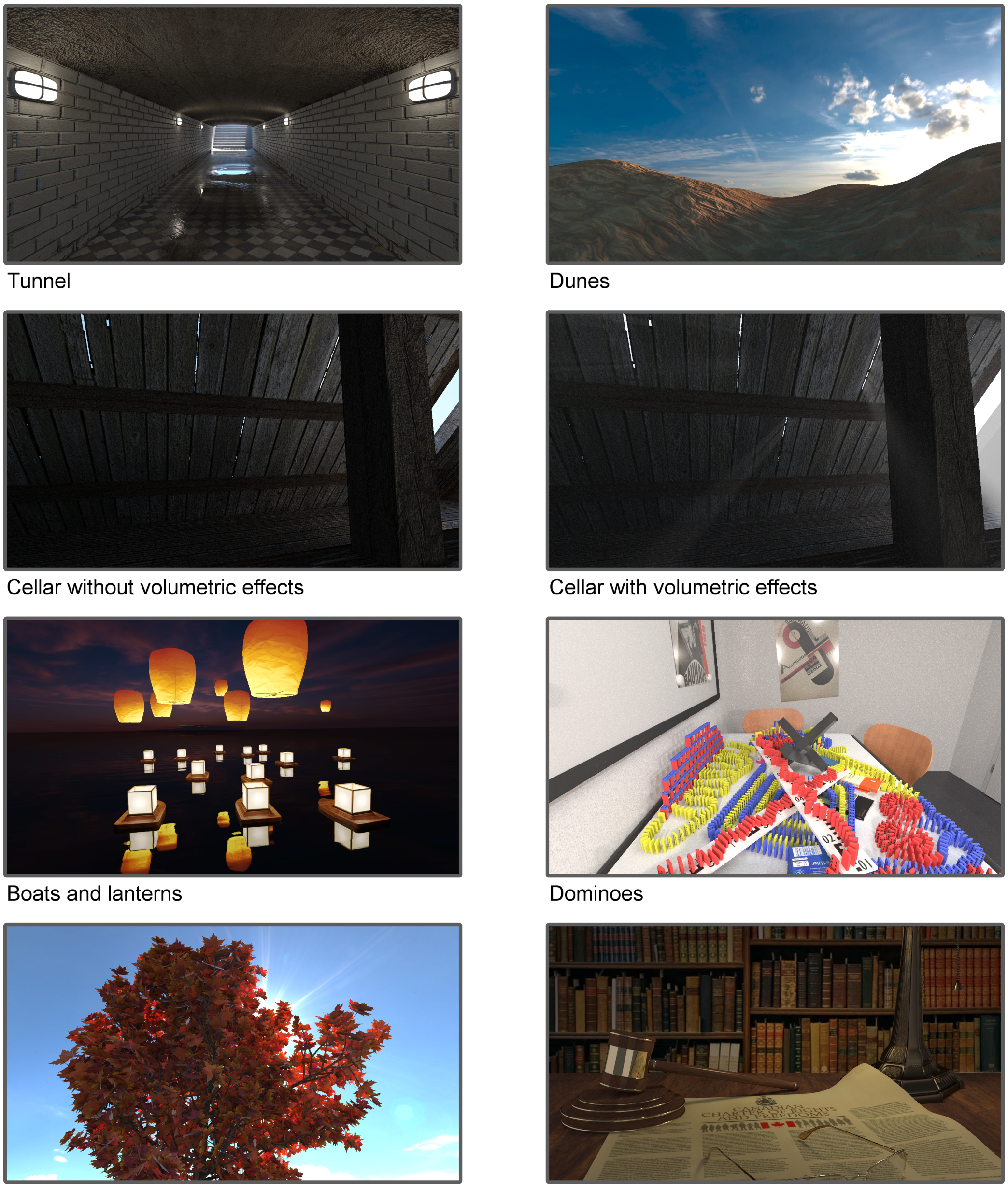
Figure 10: Tie-fighter visualisation on a 360° lightfield display [26]

3 Lightfield rendering system overview

The LF rendering system presented in this thesis has been created to study the application of TMOs to HDR-data in the context of real-time view-synthesis.
The LF-data provided to the LF rendering system includes metadata, either tonemapped or HDR luminance-information as well as disparity-maps for each key-view. From this input-data, a stream of display-ready novel views, captured by a user-controlled virtual camera, is generated.

3.1 Capturing and pre-processing of datasets

The scenes used to generate LF-data have been created in Blender for the purpose of evaluating the LF rendering system. They contain features created specifically to be challenging for both the tonemapping and view-synthesis aspects of the LF rendering system. Four scenes, the Boats and Lanterns scene, the Cellar scene, the Tunnel scene and the Dunes scene were created from scratch for the purpose of this thesis, while the Dominoes scene and the Tree scene were re-purposed from scenes created at the MPI research-group in Saarbrücken. External resources for environment-maps [2] and textures [5] were used for the creation and re-purposing of the scenes. One externally created scene [12], the Desk scene is used for evaluation purposes as well. The Cellar scene comes in two variations, one including volumetric effects and one without volumetric effects so as to be able to study the impact of volumetric effects on tonemapping and view-synthesis. An overview of all scenes is shown in figure 11.



Tunnel

Dunes

Cellar without volumetric effects

Cellar with volumetric effects

Boats and lanterns

Dominoes

Tree

Desk

Figure 11: Scenes available for lightfield rendering

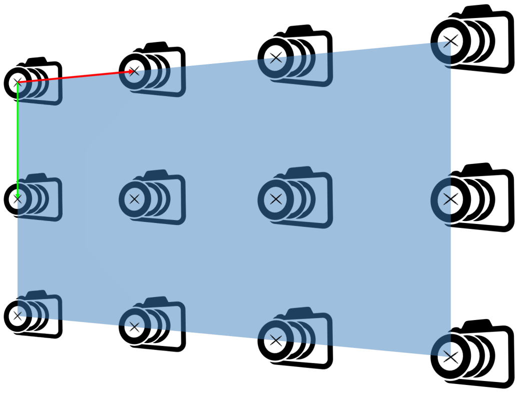


Figure 12: Camera-array with camera-plane (blue) and spanning-vectors (red and green) [1]

The scenes themselves were captured using a 2D-array of displaced pinhole-cameras serving as key-views of the LF. For all scenes, except the Cellar-scene, a grid of 17×9 key-views has been generated. For the Cellar scene, a grid of 17×5 key-views was used instead due to spatial limitations in the scene. The camera-plane is the smallest bounded plane on which the focal points of all pinhole-cameras are located and is perpendicular to the viewing axis of the pinhole-cameras as shown in figure 12. The distance between the focal point of a key-view and its nearest horizontal and vertical neighbours in the camera-plane is fixed. A coordinate system is defined for points on the camera-plane, with the origin corresponding to the focal point of the key-view forming the top-left corner of the camera-plane. The spanning-vectors of the coordinate system are chosen as the vector connecting a key-view with its right neighbour and the vector connecting a key-view with its lower neighbour.
All key-views are recorded with the same spatial resolution and opening angles for both the disparity-map and the HDR-luminance image. The disparity-map provided for each key-view contains the disparity between each pixel and corresponding pixels in adjacent key-views. The disparity-values are provided in the scale of pixel units on the image-plane of the key-view.

The rendering of the key-views itself has been distributed on the MPI-server-cluster due to time constraints. A modification of a python-script, created by the MPI research-group in Saarbrücken, is used to distribute rendering-jobs across available servers from the MPI-server-cluster. In each rendering-job, the camera in the scene is displaced according to the position corresponding to the key-view to be rendered in the camera-plane. A server may perform multiple rendering-jobs corresponding to multiple key-views, which are aggregated in a single job file. The job-file is then executed according to a python-script using the blender python API.

3.1.1 Computation of disparity-maps and handling of depth-edges

When naively rendering a scene in Blender, the luminance-information for pixels on the border between foreground- and background surfaces in the scene, commonly referred to as a z-edge, is collected from both foreground and background. The ratio between foreground- and background-samples hereby depends on the percentage of the background being occluded by the foreground inside the pixel-boundaries. While such a blending between foreground and background is intentional as it provides images without aliasing, in a LF rendering context however, this behaviour is unwanted.
For views synthesized by the LF rendering system, a different part of the background surface is observed bordering the same area of the foreground-surface for different key-views of the LF. Due to this, it is not possible to re-use the interpolated luminance-values along depth-edges found in the key-views. In addition, a single disparity value for pixels with contributions from foreground and background must be assigned in the disparity-map for each key-view. As the luminance-values from foreground and background may differ strongly, even a small contribution of either foreground or background to the pixel value creates a colour visibly different from background and foreground. Assigning the pixel to either foreground or background in the disparity-map will lead to artefacts during view-synthesis.

To resolve these problems present in the input data, both the luminance- and disparity-information has been rendered at a higher resolution than what is later provided to the LF rendering system. The horizontal and vertical resolution has been increased by a factor of four, while the samples per pixel has been reduced by a factor of 16.

For the disparity-maps, multiple samples per pixel are used and the inverse of the z-depth is collected for each sample. The inverse of the z-depth, when re-scaled according to camera-parameters and the distance between neighbouring cameras, creates the disparity-map in pixel-distances. The disparity in real-world units on the screen of a key-view is given as:

$$d = \frac{b \cdot f}{z},$$

where z denotes the z-depth of the pixel, b refers to the baseline-distance, i.e. the distance between two adjacent key-views, and f denotes the focal length of the camera. This equation is later derived in Section 4.1.2. The z-depth of a pixel is the distance between the focal point and the point in the scene visible in the pixel projected orthogonally to the image-plane. To obtain d in pixel units, the amount of pixels per real-world unit on the image-plane of the key-view needs to be determined. Given the camera-parameters, the pixel-density p can be determined from the camera-geometry:

$$p = \frac{w}{f \cdot 2 \cdot tan(fov_h/2)},$$

where w denotes the width of the image in pixels and the horizontal opening-angle is given by fov_h. As w has been set to be four times larger than the image-width provided to the LF rendering system, the original value for w has been used in the equation. d in pixel units can be obtained directly by computing:

$$d = \frac{w \cdot b}{z \cdot 2 \cdot tan(fov_h/2)}$$

The inverse of the z-depth is averaged over multiple samples per pixel during rendering instead of taking the inverse over a z-depth average, because the average of the inverse of the z-depth does not equal the inverse of the average z-depth. This is especially noticeable when dealing with samples of the environment map, for which an infinite distance is assigned. A single sample on the environment-map would lead to infinity being assigned to the pixel as the average z-depth, and thus zero on the disparity-map, even in case samples are mostly collected from the foreground. When averaging over the inverse of z-depth samples, the resulting disparity lies between the disparity of foreground and background, which can be used to detect pixels collecting from both foreground and background.

The inverse of the z-depth for a given sample is collected for the first opaque surface hit by a camera-ray and volumetric effects have been excluded from the scene. This has provided the most consistent results for the test-scenes, but depending on the scene an inclusion of non-opaque surfaces may be beneficial.

The colour- and disparity-values are further processed to obtain luminance- and disparity-values in the resolution of the LF-data. For each pixel in the output image, a block of 4×4 pixels in the input image is processed. In case

the difference between the minimum and maximum disparity values in the block exceeded a user-defined threshold t_1, the block is considered to contain a z-edge. Pixels belonging to foreground, background and pixels containing a contribution of both foreground and background are assumed to be contained in the 4×4 block in this case.
The pixels with the highest and lowest disparity in the block are assumed to belong to the background and foreground exclusively and their disparity-values are stored as foreground and background-disparities. Pixels with a disparity falling within a set range t_2 around the foreground- and background-disparities, are assigned to foreground or background respectively. Pixels outside of this range are assumed to contain samples from both foreground and background and are thus discarded. In case more samples are assigned to the background than foreground in the 4×4 block, the average luminance and disparity-values of the background-pixels are set as the output. Otherwise, the average luminance and disparity-values of the foreground-pixels are set as the output.
In case a pixel is determined to belong to a single surface, the luminance- and disparity-averages of all pixels in the block are set as the output.

This processing results in output images with aliasing along depth-edges in both luminance- and disparity-values. For all scenes, t_1 was chosen as 0.5 pixels and t_2 as 0.05 pixels.

3.2 Lightfield rendering pipeline

The LF rendering system, implemented in c++ 11 using OpenGL 3.3 and QT 5.10, serves as a pipeline generating a stream of display-ready frames from LF-data. LF-data is provided in a LF file, containing key-views and metadata, which is based on previous work from the MPI research-group in Saarbrücken. The LF rendering system itself follows a pipeline architecture consisting of 3 separate processing stages: Pre-processing, view-synthesis and post-processing. The pre-processing stage is executed once upon loading a LF-dataset and both the view-synthesis and post-processing stage are executed once for each novel view generated.

Pre-processing

The first stage of the pipeline, the pre-processing stage, processes the key-views provided in the LF-data and stores generated results to be accessed in later stages of the pipeline. The output of this stage depends on the tonemapping procedure chosen to process the LF.
The colour-data for key-views provided to later stages of the pipeline is either available in a display-ready format or as HDR-luminance-data. In case the tonemapping procedure applies TMOs to key-views in the preprocessing stage display-ready key-views with disparity-maps are the output of this stage. Alternatively, if the tonemapping procedure applies TMOs to synthesized novel views, image-statistics are stored for each of the key-views to aid or enable the application of a TMO in the post-processing stage. The image-statistics chosen are hereby specific to the requirements of the TMO used in the post-processing stage.

View-synthesis

The view-synthesis stage is responsible for synthesizing novel views of the scene captured by a virtual camera from the pre-processed LF-data. The camera-controllers implemented for the LF rendering system are limited to cameras with their focal points located on the camera-plane. To synthesize a view, data from four key-views closest to the focal point on the camera-plane is used. In case image-statistics are provided for the key-views, both the image-statistics for the four views as well as bilinearly-interpolated image-statistics for the novel view are provided to the post-processing stage.

Post-processing

The post-processing stage allows for additional processing of the novel view generated in the view-synthesis stage. For display-ready colour-data, no additional processing is performed, but an implementation of image-enhancements, such as screen-space anti-aliasing is suggested to be implemented in this stage. In case the synthesized view is available as HDR-luminance-data, the tonemapping of the novel view is performed in this stage and the tonemapping can be adjusted to account for past novel views to obtain an adaptation to changes in lighting conditions in accordance with human perception.

4 View synthesis

To visualize sparse LFs as 3D-scenes on a standard display, novel views must be synthesized in real-time. This task is accomplished by aggregating data from the key-views in the LF with the use of disparity-maps.

To synthesize a novel view, two main tasks must be accomplished:

1. For each pixel on the image-plane of the novel view, a ray R is traced into the scene and the first intersection between R and surfaces in the scene at point P must be determined (discussed in section 4.1).

2. The appearance of the surface at P, as observed from the direction $\overrightarrow{d}$ of the ray R, and the appearance of volumes traversed by R must be reconstructed (discussed in section 4.2).

To accomplish these tasks, rays in the LF passing close to P with directions close to $\overrightarrow{d}$ must be found. The ray R is intersected with the camera-plane at a point I. In the camera-plane, we chose up to four neighbouring key-views $V_1..V_4$ with focal points $F_1..F_4$ closest to I. These nearest key-views are used in the view-synthesis process exclusively for two reasons:

1. With the assumption of a Lambertian surface, the appearance of the surface at point P in the viewpoint of novel view can be approximated by the appearance values captured from the viewpoint of nearest key-views. In case P is visible in four nearest key-views, $\overrightarrow{d}$ can be easily interpolated from the viewing-directions $\overrightarrow{F_iP}$. This property enables a close reproduction of surface-appearance at P via bilinear interpolation of pixel values from correspondences A_i in the image-planes of the nearest key-views. Appearance reproduction is discussed in detail throughout section 4.2.

2. For directions $\overrightarrow{d}$, which lie within the range of directions captured by key-view-frustums, the search for A_i in at least one key-view is guaranteed to return an in-frustum result. This property is restricted to LFs, for which the maximum disparity-value from all key-views does not exceed the minimum of horizontal and vertical key-view resolution. This property is further illustrated in section 4.1.3.

4.1 Geometry reproduction

4.1.1 Epipolar geometry

Given the intersection between R and the camera-plane at I, the task of finding A in the key-view V can be transformed into a correspondence-problem for an orthoparallel camera-setup as shown in figure 13. To do so, a copy V' of the frustum of V is translated such that the focal point of V' matches I. The ray R is projected onto the, infinitely extended, image-plane of V' in the point B'. In an orthoparallel camera-setup, location of A is known to be restricted to a position on the epipolar-line e, which is formed by intersecting the epipolar-plane E, on which F,P and I lie, with the image-plane of V [43].

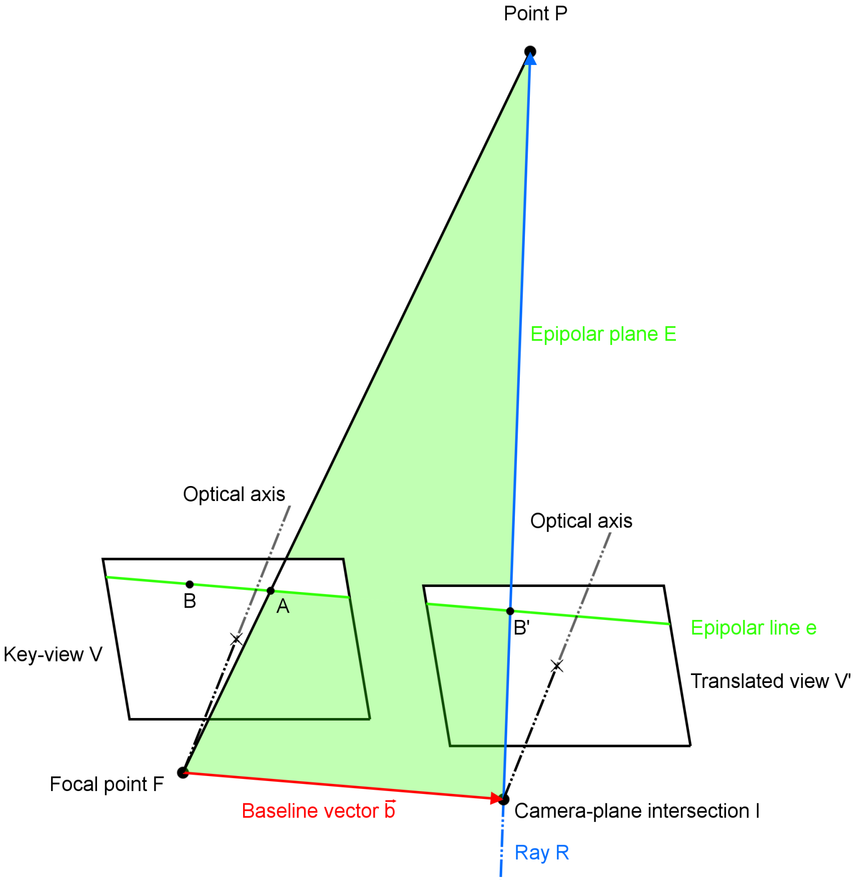


Figure 13: Epipolar geometry for orthoparallel camera-setups

As P is unknown, we span E using two spanning vectors, the direction $\overrightarrow{d} = \overrightarrow{IP}$ of R and the baseline-vector $\overrightarrow{b} = \overrightarrow{FI}$, as well as an origin on the plane, which is chosen as F. If P was an infinite distance away from the camera, then $\overrightarrow{FP}$ would be parallel to $\overrightarrow{d}$ and due to this, the pixel-coordinates of A on the screen of V would match the pixel-coordinates of B' in the point B. This way, the position of A is further restricted to lie in direction $\overrightarrow{d}$ from B on the image-plane of V.

No direct depth-information for B' is available in V' as P is unknown, so A cannot be directly determined and is required to be found via a process called backwards-registration. To determine the position of A via backwards-registration, additional information is required, which is provided in the form of a disparity-map for V.

4.1.2 Correspondence and disparity

For all key-views of the LF, the intersections between camera-rays and the scene are known during rendering in Blender. In the configuration shown in figure 14, the views V and V' baseline $b = |\overrightarrow{b}|$ and focal length f are considered. Points A and B' on the camera plane of V and V' are the projection of their corresponding point P in the scene located at z-depth z. Given that the triangles ΔABF and ΔFIP are similar triangles, the disparity d, which is the distance between two corresponding pixels A and B, can be calculated as:

$$d = \frac{b \cdot f}{z}$$

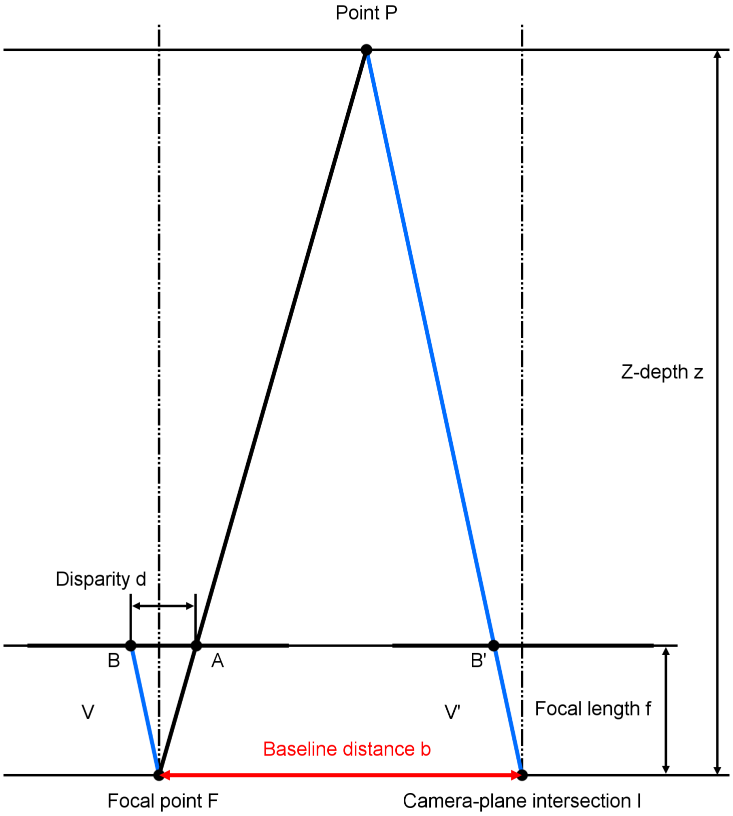


Figure 14: Disparity for orthoparallel camera-setups

During the creation of a disparity-map, both b and B are unknown and for different rays R, these values differ as well. For all rays intersecting P however, A remains the same. Due to this property, a disparity-map can be created for which b is assumed to be 1 and the disparity d is stored in pixel units at A in the disparity-map.

4.1.3 Iterative correspondence search

With the help of the disparity-map of V and the fact that the disparity d is linear in $b = |\overrightarrow{b}|$, A should satisfy the equation

$$disp(A) = disp(B + \overrightarrow{b} \cdot d) = d,$$

where $disp(p)$ denotes the sampling of the disparity-map at point p in pixel-coordinates.

This equation may be fulfilled for multiple different values of d as the ray R may intersect multiple surfaces in the scene. Only the first intersection with the scene is visible to the ray R and as such, the largest disparity value is opted among all possible disparity values fulfilling the equation for A.
To find A, a search in the disparity-map, as shown in figure 15, is performed by considering different candidates for d, starting with d chosen as the maximum disparity d_{max} of all disparity-maps of the key-views. d is iteratively reduced by $1/b$ until the condition $disp(B + \overrightarrow{b} \cdot d) > d$ is fulfilled for the first time. This way it is known that the condition for A must be fulfilled for d between the current and previous candidate for d. A can then be approximated closely by a second pass over this candidate-range with an increased sample-density for d.

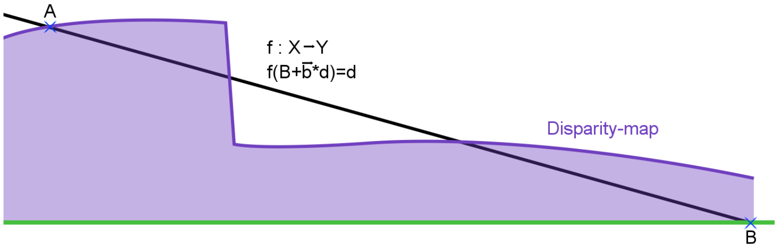


Figure 15: Search for A in the disparity-map

Because the sample-density for d is limited, the search may miss the A belonging to the first intersection between R and surfaces in the scene. In practice this causes a shrinkage of occluders and artefacts at depth-discontinuities in the novel view. A sample-density for d chosen as $1/b$ is used, such that the samples taken in the disparity-map are spaced a distance of one pixel apart from each other. This sample-density for d was found to be one of the smallest sample-densities resulting in an acceptable degree of artefacts visible in the novel view and is used throughout this thesis.

As the search for A is limited by the image boundaries of the image-plane, in case the first sample for d inside image boundaries already fulfils $disp(B + \overrightarrow{b} \cdot d) > d$, A may lie outside of frustum of V and is discarded. For the four nearest key-views, the starting-points of the search for each key-view form a square with a side-length of d_{max} in pixel units in the disparity-map, as the vectors $\overrightarrow{b_i}$ span a unit-square in the coordinate system of the camera-plane. In case d_{max} is smaller than the minimum of horizontal and vertical key-view-resolution, at least one search starts within the image boundaries if B is inside the image boundaries and returns an in-frustum result.

4.1.4 Visibility and occlusions

For the novel views considered in this thesis, the focal point of the virtual camera of a novel view is restricted to lie on the camera-plane. The virtual cameras implemented for this thesis are discussed in section 4.3. Despite restricting the focal point in such a way, an effort has been made to implement a view-synthesis scheme suitable for arbitrary focal points by using backwards-registration. For focal points on the camera-plane, I remains fixed for all rays R and for each of the four nearest key-views a warped image is created based on the correspondences found via backwards-registration is created. These warped images cannot be directly combined to yield the novel view as surfaces occluded in a key-view may be disoccluded in the novel view and vice versa.

In case a disocclusion occurs, correspondences A_i found for points P on the surface disoccluded in the novel view, lie on a disparity-edge between foreground and background in the image-plane of the key-view. This is the case as the foreground fulfils $disp(B + \overrightarrow{b} \cdot d) > d$ during backwards registration, and so A is found between foreground and background as the disparity-map is treated as continuous. A continuous surface between foreground and background is implicitly assumed to exist, for which $disp(A) = disp(B + \overrightarrow{b} \cdot d) = d$ is fulfilled. The difference in the disparity between foreground and background may be large and as such, large areas in warped key-views containing samples on a disparity-edge may exist, forming a pseudo-surface. As the pseudo-surface does not exist in the scene, correspondences on this surface are discarded.

To distinguish between deformations of a single, continuous surface and disocclusions between two different surfaces, the four nearest pixels around each correspondence A in the image-plane of the key-view are examined. In case the difference between the maximum and minimum disparities between the four pixels exceed a fixed threshold, a disocclusion between two different surfaces is assumed and A is discarded. For the results presented throughout this thesis a threshold of 1 was used. Discarding correspondences this way removes occluders, which are less than 2×2 pixels wide in key-views from the scene as all A lie between foreground and background for such an occluder. Fine structures, such as small branches in the Tree scene, are lost.

In case an occlusion of surfaces visible in a key-view occurs in the novel view, correspondences A_i found during backwards-registration may not match the occluder visible in the novel view and are required to be discarded. A correspondence A is found to lie on the background if and only if the occluder is not captured in the key-view. If the disparity of the occluder does not exceed the minimum of horizontal and vertical key-view resolution, the occluder is covered by at least one nearest key-view-frustum and A will be found on the occluder unless the occluder itself is occluded. A is discarded as occluded in case a correspondence A' for the ray R exists in another key-view with $disp(A') > disp(A) + \delta$. A δ has been introduced due to numerics. All remaining correspondences are considered visible in the novel view. Visible correspondences in the warped top-left key-view of the Dominoes scene are shown in figure 16.

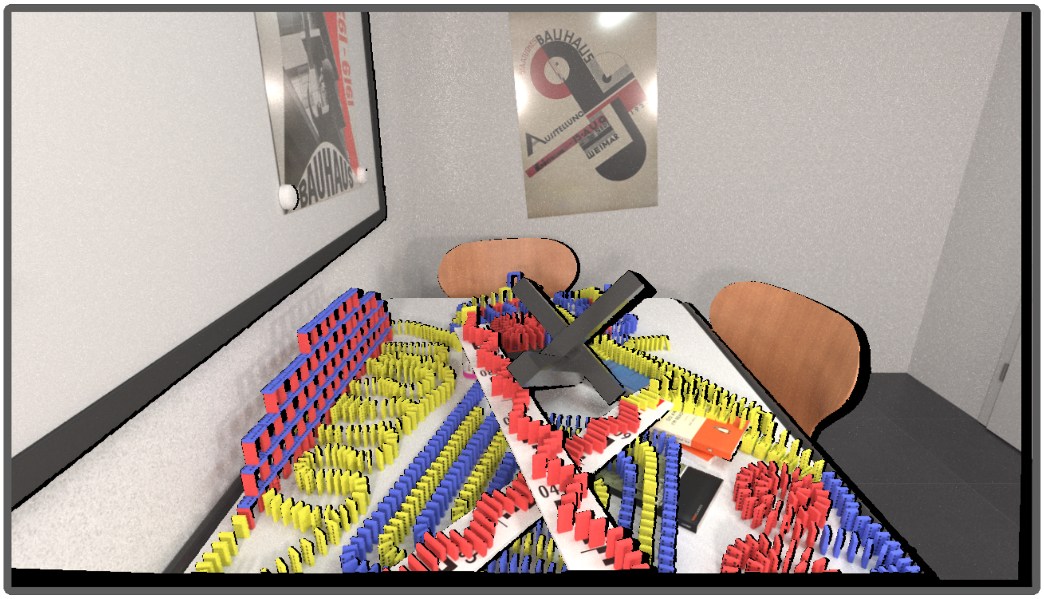


Figure 16: Visible pixels in a warped key-view for the Dominoes scene

4.1.5 Limitations and failure-cases of geometry reproduction

Inherently, any image-based rendering approach is restricted by the capture of the scene. In the case of the view-synthesis scheme used in this thesis, there are parts for the scene, for which no information can be found in any of the for closest key-views. Depending on the viewing scenario, a point P visible to the virtual camera may not be captured in any key-view frustum.

For virtual camera-frustums generating rays R, for which B' lies outside the image-plane of V', it is possible that no correspondences can be established in $V_1..V_4$ via backwards-registration. In this case, no pixel value is assigned to the ray. For all rays, B' can be guaranteed to lie inside the image-plane of V' of the virtual camera-frustum, when translated to the focal point of a key-view-frustum, is covered entirely by the key-view-frustum. The view-synthesis scheme is intended to be used with cameras fulfilling this condition and in the following, B' is assumed to lie inside the image-plane of V' and d_{max} is assumed to not exceed the minimum of vertical and horizontal key-view resolution.

A rendering failure under these assumptions may still occur depending on the topology and amount of occluders in the scene as well as the available number of in-frustum correspondences A_i.
In case a single occluder is present in the scene and four in-frustum correspondences A_i are available, for which a point P in the scene is occluded, the topology of the occluder determines whether or not P may be visible by any ray R with I in-between $V_1...V_4$. Let $P_1...P_4$ be the intersections of rays with the occluder, originating from $F_1...F_4$ in directions $\overrightarrow{F_1P}...\overrightarrow{F_4P}$. $P_1...P_4$ are not visible in the pixels $A_1...A_4$ as no information for P is available in any nearest key-view and as such, $A_1...A_4$ are found between occluder and background instead.

As the line $\overline{IP}$ is covered by the volume spanned between $V_1...V_4$ and P, any R with I in-between $V_1...V_4$ intersects the quad Q spanned by $P_1...P_4$. In case Q is entirely covered by the volume of the occluder, it is known that for any R with I in-between $V_1...V_4$, the occluder must be intersected by R and P is not visible. Q is always covered for occluders with a surface possessing no negative curvature at any point. Thus, a scene containing a single occluder fully visible in all key-views, which possesses no point of negative curvature, is guaranteed to be reproduced without rendering failures occurring. This is shown on the left half of figure 17.

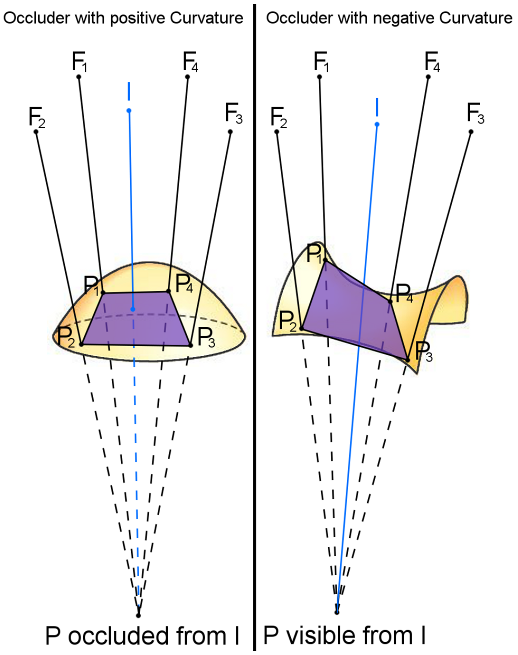


Figure 17: Visibility of P for different occluders [13]

Occluders possessing negative curvature at some points on the surface, such as the Boats and Lanterns in the Boats and Lanterns scene, do not necessarily give rise to rendering failures. Some features of occluders however are especially prone to causing rendering failures. Occluders containing small slits or holes, such as the slits present in the Cellar scene, and occluders possessing silhouettes with a large negative curvature, such as the leavesin the Tree scene are notable examples. An example of a rendering-failure for a surface with negative curvature is shown in the right half of figure 17. For occluders with slits and holes, a different area of the background surface is often visible through the slits or holes for different key-views and the coverage of the background surface is often low. In this case, the parts of the background visible in a novel view may not be visible in any of the four nearest key-views. Occluders possessing silhouettes with negative curvature may mimic the behaviour of slits or holes in this regard, giving rise to an insufficient coverage of the background surface. For occluders with holes or slits, such as the ceiling in the Cellar scene, through which a uniform background is visible, filling slits or holes for the creation of disparity-maps can result in a faithful reproduction of the scene.

In case a scene contains multiple occluders, rendering-failures may occur independently of the topology of objects in the scene. In such scenes, Q many be spanned between two or more occluders while a point P on the background is still visible to R as R passes between occluders in the scene. In the Tree scene, this behaviour can be observed as holes in the foliage can be seen depending on the position of cameras in the scene.

For points P not covered by at least one frustum of a nearest key-view, a rendering failure may occur for a single occluder with arbitrary topology. In case P is not covered by a key-view-frustum, P may be visible from the focal point of the key-view, but the information about P is not available and thus cannot be found during backwards-registration.

In case no information for P is available in any nearest key-view, a recovery of information for P may be possible. Under the conditions that B' lies inside the image-plane of V' and d_{max} is does not exceed the minimum of vertical and horizontal key-view resolution, at least one in-frustum result has been found during backwards registration. Instead of discarding results for which a disocclusion is detected, the pixel, form the set of nearest pixels surrounding A, with the lowest disparity-value is returned as A. If the occluded background is uniform in terms of colour and appearance, the occluded background is restored by interpolating the pixel values of available A_i.

4.2 Interpolation and appearance reproduction

The pixel value of a ray captured by the virtual camera is determined by interpolating the pixel values of available correspondences A_i. The contributions of correspondences from different key-views are taken into account in a multiple weighting approach using a bilinear weight W_{bi}, introduced to reproduce material appearances, and a border-weight W_{bo}, introduced to prevent artefacts at intersections between the surfaces in the scene and the frustums-borders of four nearest key-view-frustums.
The pixel value for an output-pixel is calculated as

$C_{out} = \sum_{i=1}^{4} \frac{W_{vi,i} \cdot W_{bo,i} \cdot W_{bi,i} \cdot C_{A,i}}{\sum_{j=1}^{4} W_{vi,i} \cdot W_{bo,i} \cdot W_{bi,i}}$,
where $C_{A,i}$ denotes the pixel value of A_i obtained via backwards-registration, $W_{vi,i}$ indicates the visibility of A_i and indices i and j are key-views from the set of four nearest key-views. A correspondence A_i has a visibility $W_{vi,i}$ of 1 in case A_i has not been discarded as out-of-frustum or as it is occluded in the novel view. $W_{vi,i}$ is set to zero otherwise. In case disocclusions occur for all in-frustum correspondences and a restoration of the background is attempted, the visibility of background-pixels is set to 1.

4.2.1 Bilinear weighting

The A_i with $W_{vi,i} = 1$ may differ in pixel value despite depicting the same point P in the scene. For luminance-values, these differences usually stem from glossy surfaces, volumetric effects and refractions in the scene as well as noise present in the key-view images. Aside from noise, these differences are not errors and they are required to faithfully reproduce the material appearance of surfaces and volumes in the scene. While differences in luminance-values are necessary for material-reproduction, different sources of luminance-differences may lead to different artefacts in the output image.
In case tonemapped pixel values are provided instead, the tonemapping process may have introduced differences in pixel value not present between the luminance-values. Causes of differences in pixel values introduced by tonemapping and their impact on the visibility of artefacts and the tonemapping-consistency are examined in section 5.

It is important to note that the sources differences in pixel value cannot be distinguished by the view-synthesis and all differences are therefore addressed in the same manner. In the scene, the ray R can be constructed as a bilinear-interpolation of rays R_i originating at F_i and passing through P. The interpolation-weights $W_{bi,i}$ assigned for R are calculated as

$$W_{bi,i} = (1 - |b_{i,x}|) \cdot (1 - |b_{i,y}|),$$

where $b_{i,x}$ and $b_{i,y}$ are the components of the respective baseline-vector $\overrightarrow{b_i}$ in the coordinate system of the camera-plane. These interpolation-weights $W_{bi,i}$ are used to interpolate the pixel values of available correspondences.

For I matching the position of the focal point F of a key-view V in the scene, pixel values are exclusively collected from V for correspondences A associated with a single surface in the scene. This property of bilinear weighting is referred to as key-view interpolation and is desirable as pixel values of V are considered to be ground-truth pixel values matching the material appearance in the scene for rays passing through the focal point of V.
Bilinear weighting provides smooth transitions in output pixel values in case there is a smooth change in the location of I, or the set of nearest key-views changes over time.

Material appearance

For different R_i, the angle between R_i and the surface-normal at P differs. Depending on the viewing angle, the angular distribution of secondary rays used by Monte-Carlo-style ray-tracing is chosen according to the bidirectional reflectance distribution function (BRDF) [31]. The BRDF is part of a representation of the material appearance of the surface at P and a change in angular distribution causes changes in material appearance.

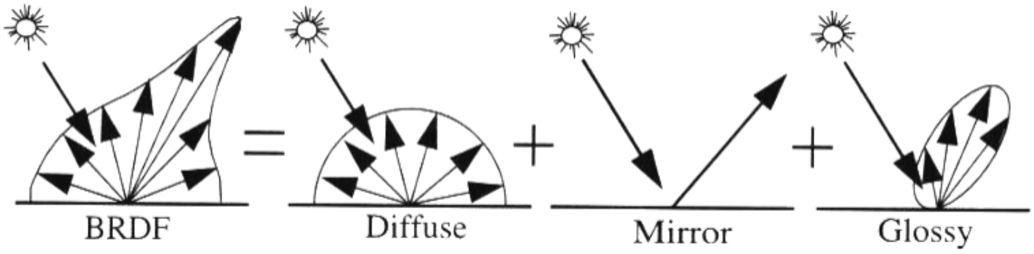


Figure 18: Modelling of a BRDF for use in ray-tracing [31]

As the luminance-information collected from individual secondary rays cannot be recovered from the key-views in the LF, an accurate reconstruction of BRDF results for synthesized views is not possible for a view-synthesis scheme in general. Instead, the view-synthesis scheme presented in this thesis interpolates the BRDF-distributions of the available correspondences by interpolating the pixel values. As the bilinear-weighting mirrors the interpolation of R from R_i, the interpolation of the BRDF results is expected to closely resemble the material appearance.

How closely such an interpolation resembles the material appearance depends on the BRDF used and how strongly the directions d_i of R_i differ. As the R_i used originate from the key-views closest to R, the directions $\overrightarrow{d_i}$ are closest matches surrounding $\overrightarrow{d}$ in the LF. The greater the z-depth of P and the smaller the distance between neighbouring key-views in the LF, the closer $\overrightarrow{d_i}$ match $\overrightarrow{d}$ as can be seen in figure 19.

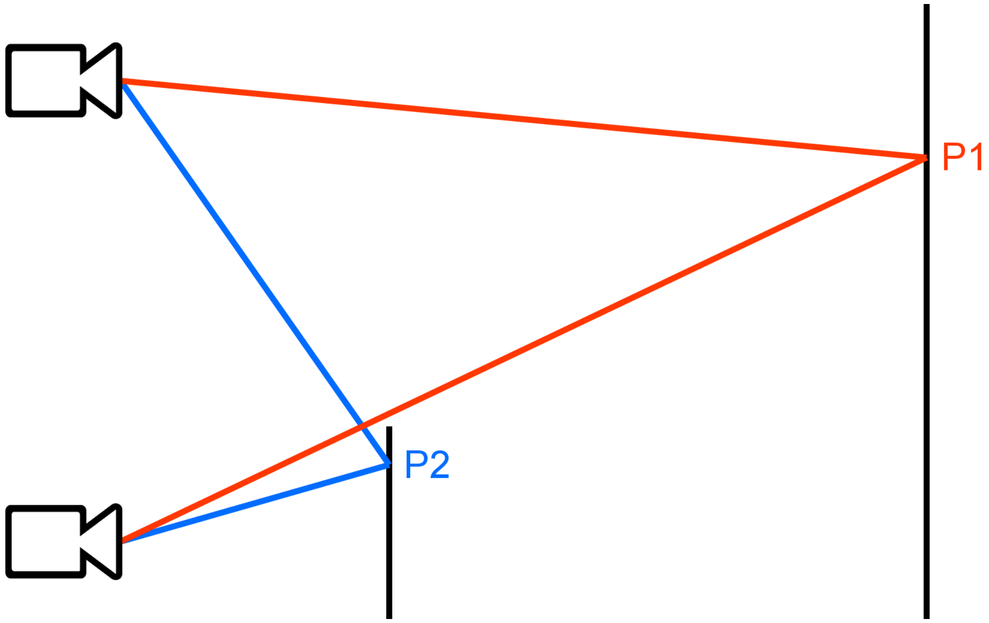


Figure 19: Viewing-scenario for points with different z-depth [6]

In general, highly diffuse surfaces, which posses a mostly angularly uniform BRDF, can be closely approximated by a bilinear interpolation. Highly glossy and mirror-like surfaces or surfaces with mirror-like contributions, such as wet surfaces or car paint, collect from secondary rays near the direction of the reflected camera-ray much more strongly. In such cases, an interpolation of the BRDF-distributions of the key-views generates multiple directions strongly contributing to the colour-result as can be seen in figure 20.

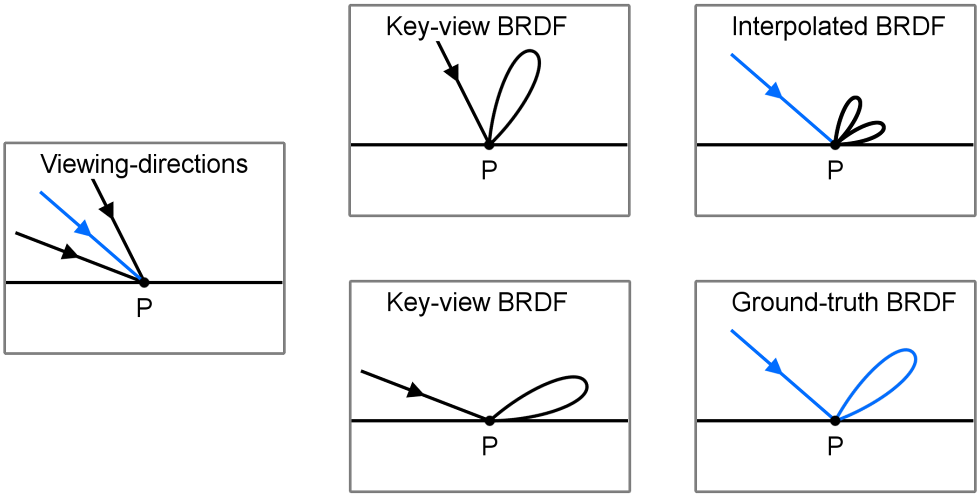


Figure 20: Interpolation of BRDF results for a glossy surface

As a result, multiple copies of reflected scene-content may be visible for surfaces with a low angular sample-density in the output image. The visibility of these copies changes with the camera-motion as the bilinear weights change accordingly. Increasing the spatial density of the LF or avoiding specular surfaces close to the camera-plane can help preventing the introduction of artefacts into novel views.

Volumetric effects

For volumetric effects, no specific disparity-value of the volume can be assigned to a pixel in a key-view and also no straightforward separation between volume and background can be made. The disparity assigned to a ray through a pixel is based on its intersection with the scene-geometry. Thus, in the key-views, the contribution of the volume to a ray R is assigned to the point P in the scene, which is located on a surface. After backwards-registration is performed, volumetric effects no longer align with their causes in the scene and up to four overlapping copies of the same volumetric effect may be visible. Due to bilinear-weighting, the visibility of these copies depends on the position of the virtual camera in the scene and changes upon camera-motion.

Visibility-borders

Differences in pixel values result in visible artefacts in novel views as for different pixel in the novel view, the set of key-views in which the point P is visible in differs. In the novel view, areas of same visibility, in which the associated P is visible by the same set of key-views, are delineated by visibility-borders.

In case a point P on a surface is not visible in at least one of the four nearest key-views, the contributions from key-views with no visibility are missing. As such, the interpolation of the pixel values from correspondences in which P is visible will not reflect the material appearance in the scene. In extreme cases, only one view contains the point on the surface and as such, no interpolation will be possible at all. This creates noticeable artefacts, especially for mirror-like surfaces. On such surfaces, the reflections appear to be painted onto the surface and do not follow the camera-motion.

Even in case such interpolation-deficiencies are not visible within an area of same visibility, they may become apparent along visibility-borders. The visibility of artefacts along visibility-borders depends on the weights and pixel values associated to the key-views, for which the visibility changes across a visibility-border. The larger the differences in pixel value between the set of key-views with and the set of key-views without changes in visibility, the greater the potential visibility of artefacts. How strongly artefacts are visible within a given key-view depends on the contribution assigned to key-views with changes in visibility. As the contributions of key-views change with the bilinear-weights, the visibility of artefacts at visibility-borders changes upon camera-movement.

Border regions

To prevent artefacts at occlusion-borders, a spatial filtering of $W_{vi,i}$ has been considered. Discontinuities in key-view contributions could be lowered by reducing contributions of key-views depending on their distance to nearby visibility-borders. The approach has been discarded it has been expected to require a filter-size prohibitively large for real-time rendering in order to achieve satisfying results.

Changes in visibility in a novel view usually take place either because P is not captured in a key-view frustum or because P is occluded in a key-view. Changes in visibility due to occlusions may be difficult to predict; however, across frustum-boundaries, and as such across image boundaries, the visibility is known to change to zero.

The horizontal and vertical distance between A and the image boundaries can therefore be used to approximate the distance to a visibility-border in the novel view.
To enable a gradual transition in key-view contributions, border regions in key-views of a fixed size around the horizontal and vertical image-borders are defined. For the horizontal and the vertical border region, the weights $W_{bo,h}$ and $W_{bo,v}$ are calculated and the border-weight W_{bo} is set to $W_{bo,h} \cdot W_{bo,v}$. Using linear gradients g_h and g_v from 0 at the frustum-border to 1 at the inner edge of the border region for $W_{bo,h}$ and $W_{bo,v}$, is insufficient as this approach neglects to consider the impact of the bilinear weighting.
Gradual transitions in key-view contributions in border regions are inherently in conflict with the properties of bilinear-weighting, which tend to maintain smoothness in I. When a transition between different sets of nearest key-views occurs, the bilinear weight for the key-views which are being excluded from the set of nearest key-views, will approach zero. To enable gradual transitions in key-view contributions, these key-views are required to have a significant contribution to pixel values in border regions for other key-views. As the set of nearest key-views changes, key-views with a previously significant contribution of pixel values in border regions for other key-views no longer contribute to pixel values.

When using linear gradients, the bilinear weighting dominates key-view contributions when transitioning between different sets of nearest key-views. This results in visible edges in the output image if I is near key-views as no gradual transition in key-view contributions takes place.

One can adjust $W_{bo,h}$ and $W_{bo,v}$ to account for the bilinear weights by using $g_h \cdot (|b_x| + \delta)$ and $g_v \cdot (|b_y| + \delta)$ for $W_{bo,h}$ and $W_{bo,v}$. The δ has been introduced to ensure key-view interpolation, as otherwise $W_{bi} \cdot W_{bo}$ would be zero for an interpolated key-view. When adjusting the border-weight in this fashion, $W_{bo,h}$ and $W_{bo,v}$ are no longer 1 at the inner edge of the border region. This is in conflict with the central region of the image, for which the weight 1 is assigned so as to not affect the reproduction of material appearances. In addition, when transitioning between different sets of nearest key-views, the contributions of key-views changes abruptly.

To allow for a certain amount of smoothness in I, while enabling gradual transitions in key-view contributions in border regions, the linear gradient and the adjusted gradient can be combined via interpolation between both gradients. The interpolation-weights were chosen as g_h^3 and g_v^3 for $W_{bo,h}$ and $W_{bo,v}$ respectively. The interpolation-weights were chosen experimentally so as to achieve the desired result. One obtains $W_{bo,h}$ and $W_{bo,v}$ as

$$W_{bo,h} = g_h^4 + (1 - g_h^3) \cdot g_h \cdot (|b_x| + \delta)$$

and

$$W_{bo,v} = g_v^4 + (1 - g_v^3) \cdot g_v \cdot (|b_y| + \delta).$$

In border regions, as defined by the border-weighting, rapid changes in key-view contributions may occur. However, even at the cost of material-reproduction, a gradual transition of key-view contributions within a novel view is deemed preferable over a noticeable artefact within a novel view. The more noticeable the impact of border-weighting on material-reproduction and smoothness of key-view contributions in I, the more noticeable an artefact at a visibility-border would be without border-weighting.

4.3 Camera-configurations

While in principle a backwards-registration approach to view synthesis allows for arbitrary virtual cameras to render the scene, for the purpose of this thesis, two virtual cameras have been implemented in the LF rendering system. Both of these cameras restrict the focal point to lie on the camera-plane.

Shifted-frustum camera

This virtual camera allows for the camera-frustum to be shifted on the camera-plane and has its viewing axis fixed to be orthogonal to the camera-plane. The camera possesses the same ratio between pixel-size on the virtual screen and focal length as the key-views such that points B' on V' can be directly obtained.
By default, the same resolution, and thus the same opening angles and aspect-ratio as key-views of the LF are chosen. While larger opening-angles are possible, the use of larger opening-angles is not intended as in this case some B' will lie outside of the image-plane of V'.

Fixed-screen camera

This virtual camera has been created to simulate the viewing-scenario of a person sitting in front of a monitor while moving their head-position parallel to the screen. The intent is to give the user the experience of the scene extending in-front of and behind the screen during head-motion. In the implementation done in this thesis, the head-motion is simulated by mouse-movement, but a use of head-tracking is intended for real-world applications. Given the real-world horizontal and vertical distance between key-views as well as their camera-opening angles and resolutions in the form of metadata, distances and directions in the scene can be related to real-world distances and positions in the key-view images. The simulated monitor-screen is fixed in the scene while the focal point is moved on the camera-plane.

implementation

For both cameras, a scene containing a screen-filling quad, representing the image-plane of the virtual camera, is rendered. The vertices of the quad have texture-coordinates assigned to them, such that texture-coordinates on the rendered quad directly map to the positions B' on the image-plane of V'.
To obtain the texture-coordinates, the vectors spanning the frustum of the virtual camera are projected onto the, infinitely extended, image-plane of V'. The pixel-coordinates obtained by the projections are mapped to texture-coordinates.
In addition to the texture-coordinates, key-views with disparity-maps, the baseline-vectors $\overrightarrow{b_i}$ and the maximum-disparity in the LF are supplied as uniform inputs to a fragment-shader implementing the view-synthesis as described in this section.

5 Tonemapping of lightfields

This section examines in which stage of the LF rendering pipeline and in what manner the TMOs introduced in section 2.1.2 can be applied and how this choice impacts the tonemapping results. In section 5.1 an application of Reinhard 2005, bilateral filtering and histogram adjustment to key-views is discussed. The tonemapping results for the novel view are interpolated from tonemapped key-views during view-synthesis as discussed in section 4.2. Improvements to Reinhard 2005 and bilateral filtering are considered to remove artefacts in novel views. Section 5.2 considers an application of Reinhard 2005 to synthesized views to allow for a rapid adaptation to changes in lighting conditions to take place without generation of artefacts. A processing method to adjust key-views, tonemapped by arbitrary TMOs, to be more suitable for LF rendering is presented and discussed in section 5.3.

The goals of tonemapping for LFs can be summarized as follows:

Tonemapping consistency within a novel view

In the series of novel views generated by the LF rendering pipeline, no visible artefacts should be introduced to individual novel views due to an interplay between the TMO and the view-synthesis. As discussed in section 4.2, the view-synthesis is prone to introducing artefacts into novel views in case differences in pixel value exist between correspondences in the four nearest key-views. Visibility-borders in novel views are especially prone to the creation of artefacts as the set of key-views contributing to the pixel value changes across these borders.

In case luminance-differences do exist in the HDR key-views, the differences are assumed to be intentional so as to emulate the properties of volumes and materials in the scene. As such, artefacts visible in areas with strong luminance-differences are permissible as their source is assumed to be primarily related to the view-synthesis process and general limitations of LF rendering.

Temporal tonemapping-consistency

The novel views produced by the LF rendering pipeline can be visualized in the form of a video. As such, surfaces in the scene ought to keep an appearance consistent with their material properties upon camera-movement. However, TMOs, in particular TMOs adaptive to changes in the lighting conditions of the scene, can result in unexpected changes of surface-appearance in the video. These adaptations are commonly observed as variations in the global brightness level but may be local in nature for local TMOs. The change in the appearance of a surface in the scene over time is considered permissible and intentional as long as the change is consistent with both, the material properties of the surface and the changes in lighting conditions.

A change in appearance due to material properties is dependent on the change in viewing angle of the surface over time. The change in viewing angle itself depends on the camera-movement as well as the distance between the focal point of the virtual camera and the point on the surface. For example in case of mostly diffuse surfaces, only small changes in appearance, due to the material properties, are observed, while for specular surfaces, close to the virtual camera, the changes in appearance are large.

For TMOs adaptive to changes in lighting conditions, a change in tonemapping behaviour over time should be proportional to a change in lighting conditions. Especially (dis-)occlusions of strong light sources, such as the sun, may trigger an intentional rapid adaptation of the tonemapping behaviour. In case no change in lighting conditions has taken place, no change in tonemapping behaviour should be visible in the output video.

5.1 Key-view tonemapping

The most basic application of TMOs examined in this thesis is a direct application of the TMOs on each of the key-views of the LF with the same tonemapping parameters. The application of TMOs in this is case performed in the pre-processing stage of the LF rendering pipeline and no post-processing is applied to novel views.

5.1.1 Tonemapping-consistency for key-view tonemapping

This style of application has implications on which properties a TMO is required to satisfy so as to produce results in line with the consistency-goals for tonemapping.

Tonemapping-consistency within a novel view

The key-views tonemapped during the pre-processing stage are assumed to be free of visible artefacts. Tonemapping-inconsistencies in novel views stem from differences in pixel values introduced by tonemapping the key-views and the changing contributions of key-views within a novel view.

Due to a bilinear-weighting of key-view-contributions to the pixel values, within surface-areas visible from the same subset of the four nearest key-views, the tonemapping is consistent in the novel view. While border-weighting adjusts the key-view-contributions in the novel view, it does so in a locally smooth manner, leading to locally smooth changes in tonemapping.

Abrupt changes in key-view contributions, such as across visibility-borders, may lead to visible artefacts in the novel view. Especially for TMOs adaptive to changes in lighting conditions or TMOs using image-statistics with no or low robustness towards outliers as discussed in section 2.1.2, large changes in tonemapping between adjacent key-views can be expected. The greater the differences in tonemapping between adjacent key-views and the greater the contributions of key-views for which the visibility changes across visibility-borders, the greater the visibility of artefacts. A TMO suitable for key-view tonemapping ought to produce key-views with low or limited changes in tonemapping between neighbouring key-views.

Temporal tonemapping-consistency

As the key-views themselves are tonemapped, the temporal tonemapping-consistency is dependent on the view-synthesis process as well as differences in tonemapping between key-views. Due to the smoothly varying bilinear-weighting for smoothly varying camera-positions on the camera-plane, the key-view contributions vary smoothly over time. However, the greater the change in camera-position over successive frames, the greater the change in key-view contributions due to bilinear-weighting. In case the differences in pixel value between correspondences in key-views are sufficiently large and stem from the tonemapping process, noticeable inconsistencies can be observed upon rapid camera-movement.

The visibility of artefacts within a novel view varies over time as the key-view contributions change. The higher the contributions of key-views, for which the visibility of the surface changes across visibility-borders, the more noticeable the inconsistencies in tonemapping.

A large change in key-view contributions may also occur over small changes in camera-position due to the conflicting goals of border-weighting and bilinear-weighting. When closely approaching transitions between different sets of nearest key-views, the bilinear-weighting dominates over the border-weighting such that the contributions of key-views, which are not part of the new set of closest key-views, approach zero. However, even at slightly larger distances to the closest transition between sets of nearest key-views, the impact of border-weighting supersedes the impact of bilinear-weighting to enable gradual transitions in key-view contributions in border regions. For sufficiently strong differences in pixel value as a result of tonemapping, a rapid change in tonemapping behaviour can be observed.

As the temporal tonemapping-consistency is dependant on the change in key-view contributions, a change in tonemapping behaviour may not be consistent with its cause in the scene. A rapid adaptation to a (dis-)occlusion of a light source, for instance, is not handled and the adaptation towards such a change in lighting conditions gradually takes place between key-views. As such, a brightening or darkening of novel views may be observed before a change in lighting conditions takes place in the output video.

5.1.2 Naive bilateral tonemapping operator

The bilateral TMO, as introduced in 2.1.2, separates the input luminance into two layers of information, the detail-layer and the base-layer. This separation can be seen as an approximation of a separation of luminance into texture, contained in the detail-layer and lighting, contained in the based layer. While this approximation is not perfect, for an application to single images this does not result in visible artefacts. For LF-rendering, artefacts may be introduced to novel views due to the behaviour of the bilateral filter across depth edges, which commonly correspond to luminance-edges, in key-views.

An application of the bilateral TMO [24] to each key-view of the LF, produces Cornsweet-profiles in the detail-layer at transitions between two areas with differing average luminance in case the difference in average luminance is sufficiently small. The creation of Cornsweet-profiles is in this case intentional so as to preserve low-contrast edges. For these low-contrast edges, the Cornsweet-profiles are of low contrast as well and do not introduce halo artefacts. The bilateral filtering used for this TMO prevents the creation of Cornsweet-profiles across high-contrast edges as these Cornsweet-profiles would otherwise introduce halo artefacts.

The bilateral filter [42], which is applied to luminance-values in the logarithmic domain to obtain the base-layer, computes a weighted average for each pixel p over a local neighbourhood S. The weight assigned to a neighbourhood-pixel q is a product of a luminance-weight W_l and a distance-weight W_s. The luminance-weight has been introduced to reduce contributions depending on the contrast between p and q so as to prevent the creation of unwanted Cornsweet-profiles. Both weights are computed by application of a Gaussian distribution with a mean of zero to the respective differences in luminance and position between the central pixel and a neighbourhood pixel:

$$W_s = G_{\sigma_s}(||p - q||)$$

$$W_l = G_{\sigma_l}(log(L_p) - log(L_q)),$$

where G_{σ_s} and G_{σ_l} are gaussian-distributions with variance σ_s^2 and σ_l^2 respectively and L_p, L_q are the luminance-values at p and q respectively. The result of the bilateral filter $BF(p)$ yields the base-layer in the logarithmic domain and is given by

$$BF(p) = \frac{1}{W_p} \cdot \sum_{q \in S} W_s \cdot W_l \cdot log(L_q),$$

where

$$W_p = \sum_{q \in S} W_s \cdot W_l.$$

The visibility of Cornsweet-profiles in the key-views depends on the parameters chosen for the Gaussian distributions used in the bilateral filter as well as the compression applied to the base layer. The Cornsweet-profiles themselves are contained in the detail-layer and are not compressed. Poor parameter-choices may still give rise to halo-artefacts in the key-views. The general processing scheme applied to a low-contrast edge resulting in the generation of Cornsweet-profiles is illustrated in figure 21 and results for a key-view of the cellar scene are shown in figure 22.

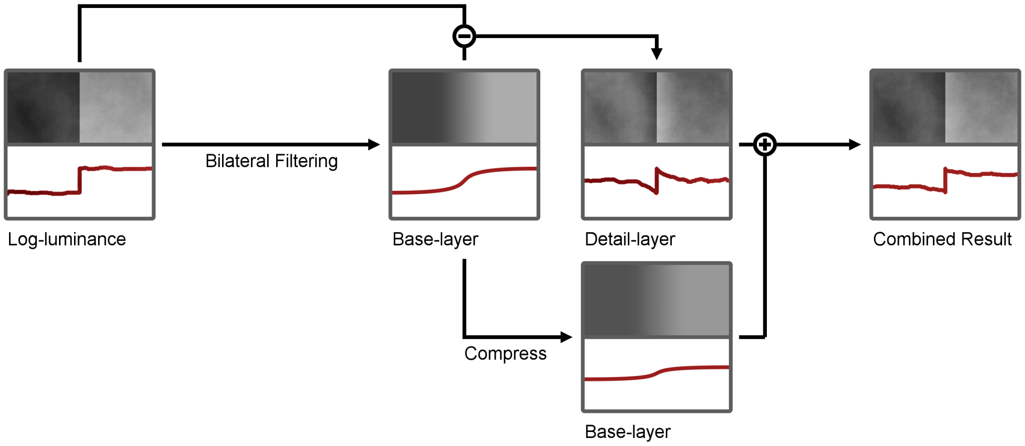
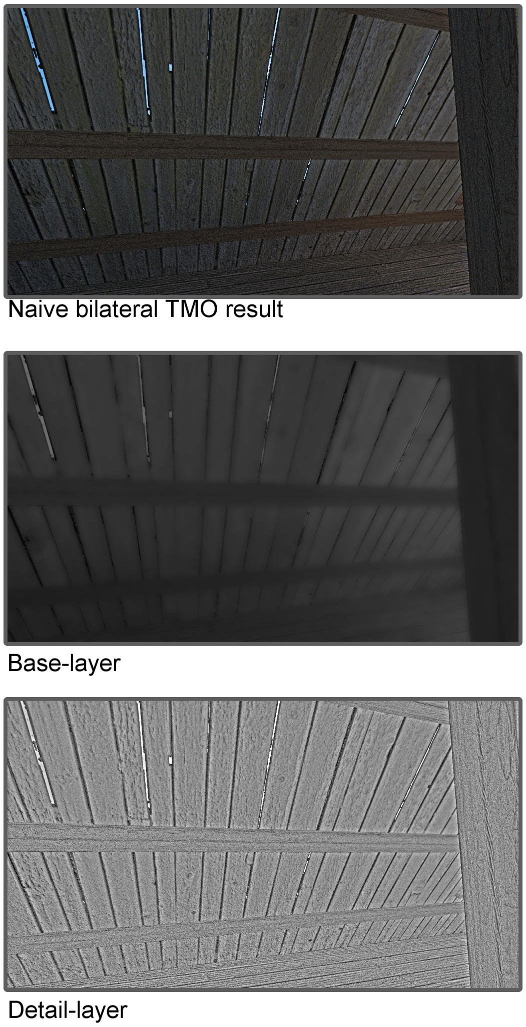


Figure 21: Processing-scheme for the naive bilateral tonemapping operator for a low-contrast edge (contrasts are exaggerated to improve visibility)

While the creation of Cornsweet-profiles is both intentional and necessary to achieve the goal of preserving local contrasts, a problem arises as these profiles are generated on depth-edges in key-views. The position of these profiles on the foreground tends to remain consistent between adjacent key-views of the LF, however the positions of the profiles on the background lie near different visibility-borders in novel views. The differences in pixel values across visibility-borders are often large enough to introduce visible artefacts into novel views despite the visibility of halo-artefacts being negligible in key-views.



Naive bilateral TMO result

Base-layer

Detail-layer

Figure 22: Application of the naive bilateral TMO to a key-view

For areas sufficiently distanced from visibility-borders, the tonemapping consistency in a novel view is maintained, as the local neighbourhoods used during bilateral-filtering correspond to similar surface-areas in the scene.

Besides artefacts near visibility-borders, which fade in and out depending on key-view contributions, the output-frames are consistent in the temporal domain.

5.1.3 Enhanced bilateral tonemapping operator

In the LF rendering system introduced in this thesis, disparity-maps are given as inputs, enabling the creation of an enhanced version of bilateral tonemapping, which prevents the occurrence of unwanted Cornsweet-profiles at visibility-borders for low-contrast edges. To account for disparity-differences between the central-pixel p and a neighbourhood pixel q, a disparity-weight W_d is introduced in the TMO. The disparity weight is generated by application of a Gaussian distribution to the disparity-difference between the pixels as

$$W_d = G_{\sigma_d}(disp(p) - disp(q)).$$

For the enhanced bilateral filter EBF used to obtain the enhanced base-layer, the contribution of a neighbourhood-pixel is modified such that

$$EBF(p) = \frac{1}{W_p} \cdot \sum_{q \in S} W_s \cdot W_l \cdot W_d \cdot log(L_q),$$

where

$$W_p = \sum_{q \in S} W_s \cdot W_l \cdot W_d.$$

This way, contributions across disparity-edges in the key-views are heavily reduced and no Cornsweet-profiles across visibility-borders are generated.

This modification of the bilateral TMO generates hard edges in the modified base-layer at disparity-edges in the key-views for low-contrast edges if parameters are chosen to fully remove Cornsweet-profiles across visibility-borders. For LF rendering, this removes the cause of pixel-differences across visibility-borders on the background, however the contrast between foreground and background is compressed as it is now contained in the modified base-layer as illustrated in figure 23. In extreme cases, this can cause the contrast to fall below a threshold discernible by the human eye and reduce depth-perception in the scene as the necessary pictorial cues are missing [17].

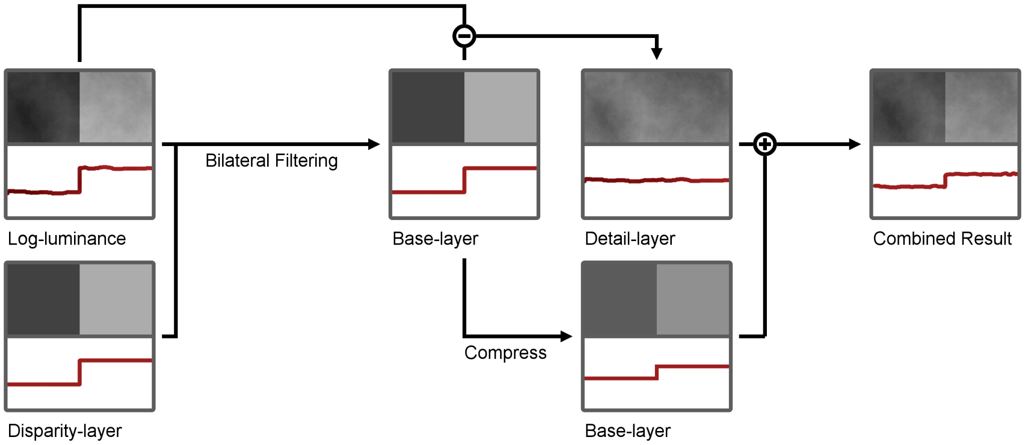


Figure 23: Compressed contrast between foreground and background for base-layer modification applied to a low-contrast edge (contrasts are exaggerated to improve visibility)

To preserve contrast across disparity-edges in novel views, the background cannot be used in Cornsweet-profiles to prevent artefacts in novel views. However, neighbouring key-views typically do agree on the position of Cornsweet-profiles for the foreground. This way, contrast can be preserved to a certain extent by using half of the Cornsweet-profile on the foreground while removing the other half of the profile on the background in each key-view.

To achieve this goal, three layers of luminance-information are created. Apart from the base- and detail-layer, a foreground-layer is introduced. The foreground-layer contains the Cornsweet-profiles on the foreground, which were originally part of the detail-layer in the naive bilateral TMO. To create the foreground-layer, a temporary layer is computed using bilateral filtering, for which contributions of the background to pixels on the foreground do not take the disparity-weight into account as it was the case in the naive bilateral TMO. For the contributions of the foreground to pixels on the background however, the contributions are penalized according to the disparity-weight as in the creation of the modified base-layer. This behaviour has been implemented by setting the disparity-difference between the central pixel and a neighbourhood-pixel to zero in case the disparity-difference is negative for the creation of the temporary layer.

The disparity-weight is modified, such that

$$W'_d = G_{\sigma_d}(maxdisp(p) - disp(q), 0).$$

The bilateral filter FBF used to obtain the temporary-layer is calculated as

$$FBF(p) = \frac{1}{W_p} \cdot \sum_{q \in S} W_s \cdot W_l \cdot W'_d \cdot log(L_q),$$

where

$$W_p = \sum_{q \in S} W_s \cdot W_l \cdot W'_d.$$

In the temporary layer, the foreground closely matches the base layer in the naive bilateral TMO and the background closely matches the modified base layer. This way, subtracting the temporary layer from the modified base-layer yields the foreground-layer containing half-Cornsweet-profiles on the foreground. The foreground-layer is scaled independently of the modified base-layer by multiplication with a scale factor in the logarithmic domain to adjust the visibility of Cornsweet-profiles and low-contrast edges. The sum of these three layers yields the desired luminance in the logarithmic domain. The scale factor for the foreground-layer and standard deviation for the Gaussian distribution used for the disparity-weight are given as additional parameters of the enhanced bilateral TMO. An overview for the processing performed in the enhanced bilateral TMO is shown in figure 24 and an application to a key-view is shown in figure 25.

The modifications introduced in the enhanced bilateral tonemapping help to achieve the desired spatial and temporal tonemapping-consistency. A lack of adaptability towards the lighting conditions of the scene when using this TMO can be seen as a major downside and makes this TMO less suitable for LFs with changes in lighting conditions.

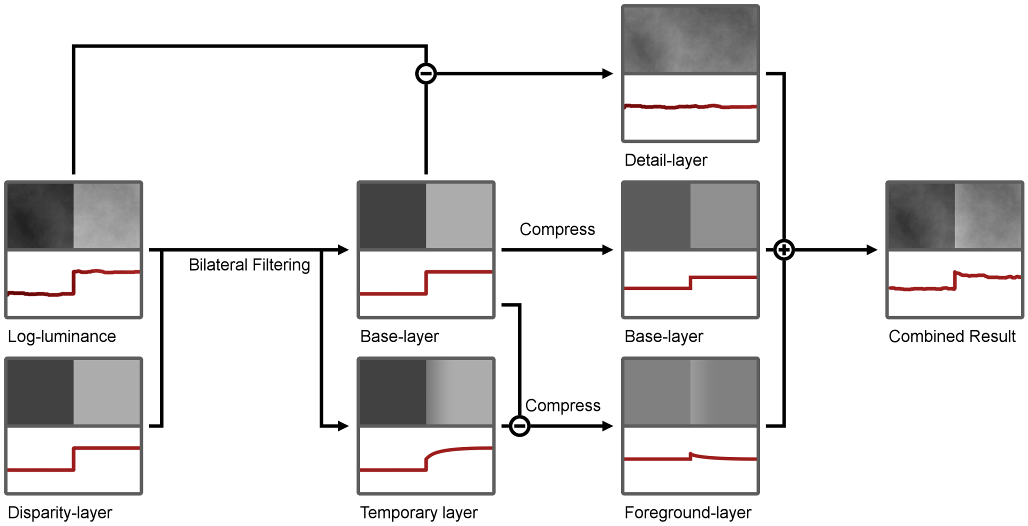
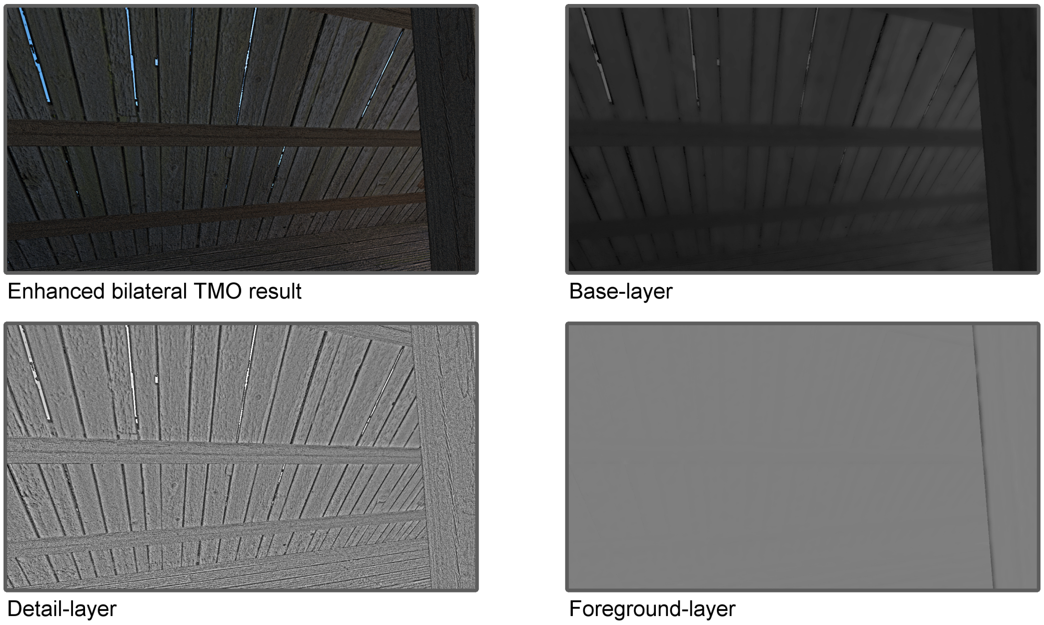


Figure 24: Processing-scheme for the enhanced bilateral tonemapping operator for a low-contrast edge (contrasts are exaggerated to improve visibility)



Enhanced bilateral TMO result

Base-layer

Detail-layer

Foreground-layer

Figure 25: Application of the enhanced bilateral TMO to a key-view

5.1.4 Histogram adjustment

Histogram adjustment [41] allows for a tonemapping process, which is both considered robust as well as adaptive to the lighting-condition in the key-views. In the implementation for this thesis, the logarithmic luminance spectrum has been subdivided into 4096 bins of equal size to generate an image-histogram for each key-view. Given the minimum luminance L_{min} and maximum luminance in the image L_{max}, a pixel with luminance L is sorted into the bin b according to:

$$b = \frac{log(L) - log(L_{min})}{log(L_{max}) - log(L_{min})} \cdot 4096$$

Bins in the histogram are repeatedly trimmed until less than 2.5% of the number of pixels in the image were trimmed during the last iteration. The number of pixels to which bins in the histogram are trimmed depends on the capabilities of the display, in particular the static contrast the display can produce. For current display-technologies contrast ratios c of around 1000:1 are common, which is also used in this work. For a number of pixels in the histogram T before the next trimming is performed, all bins b with occurrence-frequencies $f(b)$ exceeding the trimming limit t are set to contain t amounts of pixels, calculated as:

$$t = \frac{log(c) \cdot T}{(log(L_{max}) - log(L_{min}) \cdot 4096)}$$

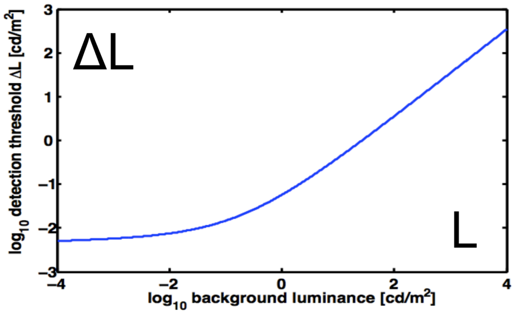


Figure 26: Just noticeable differences function for human perception [36]

Such a trimming with a fixed t for all bins is appropriate if the response of the HVS to stimuli is roughly logarithmic in the sense that multiplicative changes in luminance are perceived as approximately fixed increments in luma. For bright scenes, the human perception of luminance follows a logarithmic response, however for dark scenes this not the case, as can be seen in figure 26, and a trimming using a fixed ceiling is not appropriate. To limit the perceived contrasts on the screen to be at most as large as the perceived contrasts when viewing the real scene, human perception in terms of just noticeable differences $\Delta L_t(L)$ is taken into account for luminance-values of both, display and scene. For the peak-brightness L_{dmax} of the screen, 300 cd/m^2 are used in this thesis as this is a typical value for current displays and L_{dmin} is set to 0.3 cd/m^2 according to the contrast ratio. The world-luminance L_w of pixels in bin b is known and the luminance on the display L_d assigned to bin b can be calculated for the histogram of the previous iteration using the cumulative distribution function

$$p(b) = \frac{\sum\limits_{b_i<b} f(b_i)}{T}$$

as $log(L_d) = log(L_{dmin}) + [L_{dmax} - L_{dmin}] \cdot p(b)$.
The trimming limit, when accounting for human perception of luminance, is calculated as:

$$t = \frac{\Delta L_t(L_d) \cdot L_w}{L_d \cdot \Delta L_t(L_w)} \cdot \frac{log(c) \cdot T}{(log(L_{max}) - log(L_{min}) \cdot 4096)}$$

To apply the calculated display-luminance for each bin to the key-views, the cumulative distribution function $p(b)$ of the repeatedly trimmed histogram is stored as texture such that texture-coordinates between zero and one refer to input luminance values in the logarithmic domain between $log(L_{max})$ and $log(L_{min})$. The display-luminance L_d of a pixel in the image of luminance L can be obtained by using the texture-coordinate

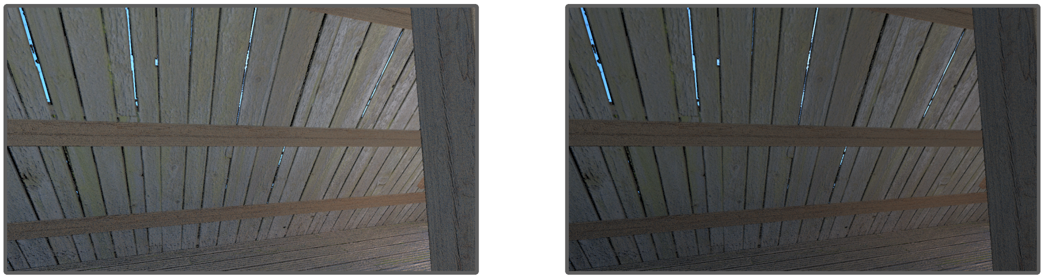
$$\frac{log(L) - log(L_{min})}{log(L_{max}) - log(L_{min})}$$

to obtain $p(b)$ for the current pixel, which in turn is used to calculate L_d. The luma-value corresponding to L_d is determined using the dynamic range of the display as well as gamma-correction.

As the images produced by multiplication of the colour-values with the target-luma tend to produce over-saturated images, a colour-correction has been applied to the relative colour-values of a pixel, which are obtained by division of channel-values by L. The relative colour-values are raised to the power of α, which is usually chosen from values ranging from 0.5 to 0.9, reducing the saturation in the output.

Major parts of the scene-content captured in a key-view are expected to also be captured by adjacent key-views. A histogram over the shared scene-content is expected to show similar luminance-distributions between adjacent key-views. This makes histogram-adjustment especially suitable for denser LFs as adjacent key-views contain more shared scene-content when compared to more sparse LFs.

The consistency of the histograms directly translates into a consistent tonemapping behaviour between key-views for bright scenes, with only small differences in pixel value for corresponding pixels in the key-views. Due to this, no visible artefacts are introduced into novel views due to differences in tonemapping and temporal consistency is achieved. For dark scenes, such as the Cellar scene, bins in the histogram are strongly trimmed during each iteration when accounting for human perception. This way, the trimming of the bins ends after a few iterations and differences in tonemapping between iterations are noticeable as seen in figure 27. Setting the number of iterations performed during trimming to a fixed number restores tonemapping-consistency within novel views and over time. While the enhanced bilateral filter posses similar qualities in terms of tonemapping-consistency, a histogram adjustment has the additional benefit to allow for an adaptation towards the lighting conditions in the key-views. This adaptation is aimed at reproducing visibility of contrasts in all parts of the scene and not aimed at simulating adaptation-behaviour in the HVS. As such, the visibility of light sources occupying only a small part of a key-view, such as the sun, only has little impact on the tonemapping result despite triggering a strong adaptation for the HVS.



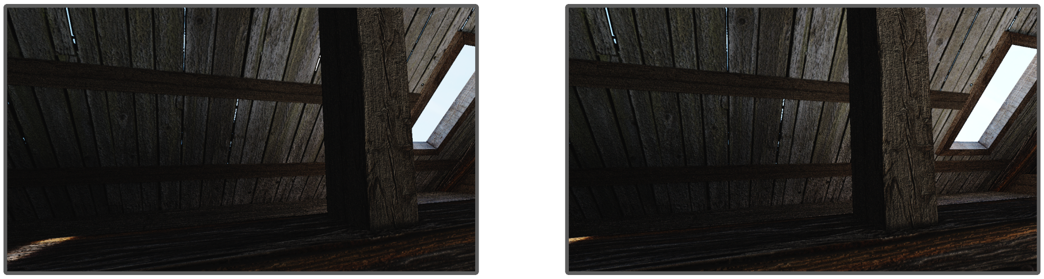
Histogram adjustment (4 iterations)

Histogram adjustment (5 iterations)

Figure 27: Application of histogram adjustment to a key-view for different amounts of trimmings

5.1.5 Naive Reinhard 2005

The Reinhard 2005 TMO [16] uses non-robust image-statistics, as introduced in section 2.1.1, in the form of peak- and minimum-luminance to calculate a contrast-metric and applies a non-robust luma-normalization to the output image. Due to this, the tonemapping may vary strongly between key-views. This is especially noticeable for (dis-)occlusions of a light source, such as the sun, between adjacent key-views. For example, a direct view of the sun is likely to result in a high peak-luma and a high minimum-luma in the output, which leads to a large reduction of overall brightness after luma-normalization. In case bright light sources are occluded, a luma-normalization typically increases the overall brightness. Even in case no change in lighting-condition occurs between adjacent key-views, outliers may still lead to noticeable differences in tonemapping as can be seen in figure 28.



Naive Reinhard 2005 (left key-view)

Naive Reinhard 2005 (right key-view)

Figure 28: Naive application of Reinhard 2005 to neighbouring key-views (brightness adjusted for print)

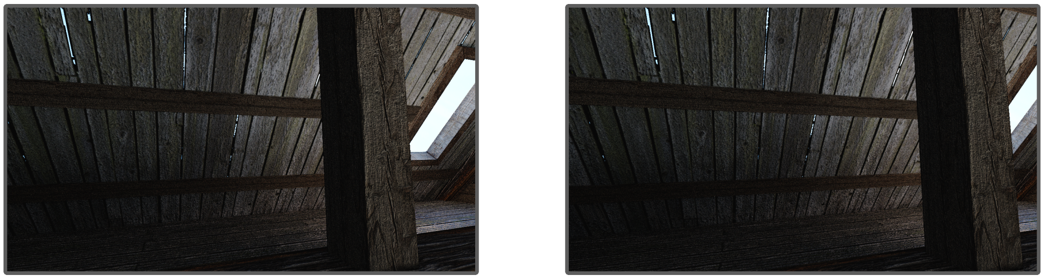
In the context of tonemapping-consistency within a single frame, such behaviour potentially results in artefacts observable as noticeable changes in tonemapping across visibility-borders. Unlike bilateral filtering, the inconsistencies are not limited to regions near the visibility-borders and surface-areas with the same visibility from the four nearest key-views posses visibly different tonemapping.

Due to potentially large differences in tonemapping between adjacent key-views, the temporal consistency of the TMO is low. Even in surface-areas, which are visible in all four nearest key-views, changing contributions of the key-views results in noticeable changes in tonemapping upon camera-movement. For comparatively dense LFs, changes in tonemapping due to outliers are especially noticeable as the real world distance between key-views shrinks and thus the same camera-movements have a larger impact on the change in bilinear-weights.
Changes in tonemapping are even more pronounced near key-views as the contributions of key-views are prone to rapid changes in border regions. These rapid changes stem from the conflicting goals of key-view interpolation and border-weighting.
Tonemapping behaviour between surface-areas of same visibility in the nearest four key-views changes differently as for different areas the surface is occluded in different key-views with different associated bilinear-weights. This leads to changes in tonemapping behaviour at visibility-borders to fade out in case the contributions of key-views, with changes in visibility across the visibility-border, are low.
Overall, the Reinhard 2005 TMO is ill-suited for a naive application to key-views of the LF in many scenarios.

5.1.6 Joint Reinhard 2005

The Reinhard 2005 TMO can be adapted to be more suitable for per-key-view application by considering the entirety of the key-views of the LF as parts of a single image. This way, the same image-statistics, aggregated over all key-views, are utilized to obtain the same tonemapping behaviour for all key-views. Due to this, the adaptivity to the lighting conditions in each individual key-view, which is considered an intentional feature of the Reinhard 2005 TMO, is lost entirely.

In case of the Reinhard 2005 TMO, the initial result of the operator is further processed as a luma-normalization is performed. This processing, when performed for each individual key-view, introduces new tonemapping-inconsistencies between key-views as the maximum- and minimum-luma used for the normalization differ between key-views. Maximum- and minimum-luma are not robust image-statistics and as such prone to noise and outliers. The result after a per-key-view luma-normalization is often not better than performing a naive application of the Reinhard 2005 TMO.
A possible solution appears to be using the maximum- and minimum-luma, found via a search in all key-views. In many cases, values for maxima and minima are found to be far from the maxima and minima present in the individual key-views. This way, the resulting images lack in luma-contrast. In case the minima and maxima are very close to the display-limitations, a luma-normalization has very little effect. For good results, minimum- and maximum-luma values, to which the key-frames are normalized, can be set manually. A result for two neighbouring key-views can be seen in figure 29



Joint Reinhard 2005 (left key-view)

Joint Reinhard 2005 (right key-view)

Figure 29: Joint application of Reinhard 2005 to neighbouring key-views (brightness adjusted for print)

5.2 Tonemapping of synthesized views

For TMOs based on robust image-statistics, introduced in section 2.1.1, a per-key-view application can be expected to deliver good results. This is the case as a majority of the scene-content captured by a key-view can be expected to be captured in adjacent key-views. In this case, shared scene-content is a main contributor towards robust statistics and ensures these statistics vary smoothly over the key-views in the LF, resulting in a smooth tonemapping behaviour.

For TMOs using non-robust image-statistics, such as the Reinhard 2005 TMO, a tonemapping of key-views is prone to introducing artefacts to novel views, especially at visibility-borders. To prevent such artefacts within novel views, while simultaneously maintaining the ability of TMOs to adapt towards the lighting conditions in the scene, the tonemapping can be applied to the synthesized views instead.
To interpolate pixel values in a perceptually-linear colour-space during view-synthesis, a gamma-correction has been applied to the HDR-key-views. After view-synthesis, the gamma-correction is undone. All colour-values remain continuous floating-point values throughout this process.

As the same tonemapping is applied to all pixels in an output-frame, the tonemapping in the output-frame is consistent, independent of the TMO used. Achieving temporal tonemapping-consistency in the generated output video is the main challenge for tonemapping of synthesized views.

5.2.1 Tonemapping-interpolating Reinhard 2005

To obtain a tonemapping behaviour, which is smoothly varying with the movement of the camera-position on the camera-plane, a bilinear-weighting approach can be utilized. The synthesized-view is tonemapped four times, once according to the image-statistics of each of the four nearest key-views. The tonemapping of the synthesized view is obtained by combining the four results, using the bilinear-weights determined during view-synthesis. Luma-normalization is applied afterwards to obtain the final tonemapping result.

Due to the luma-normalization being applied to the output image, additional adaptivity towards the scene-content can be achieved. Events, such

as (dis-)occlusions of a light source in the output video result in an immediate dimming or brightening. While luma-normalization does allow for rapid changes in tonemapping behaviour, in case these changes are directly linked to events, these changes are perceived as intentional adaptation-behaviour by a user. This is not the case for a naive application of Reinhard 2005 to the key views, as luma-normalization has been applied to the key views. Thus, changes in tonemapping behaviour are dependent on the changing contributions of key-views during view-synthesis.
Still, the use of non-robust image-statistics allows for outliers to impact the tonemapping behaviour, leading to rapid changes in tonemapping, which are not tied to changes in lighting conditions and are noticeable as artefacts in the video. Additionally, the interpolation of tonemapping results used to obtain the intermediate tonemapping of the synthesized view is related to the spatial density of the LF. For rapid camera-movements, this results in rapidly changing contributions and thus, visible inconsistencies of tonemapping in the output video.

5.2.2 Statistic-interpolating Reinhard 2005

To improve the tonemapping behaviour with regards to rapid camera-movement as well as sensitivity to outliers, the intermediate tonemapping result can be generated using interpolated image-statistics instead of interpolating tonemapping results. For the novel view, the image-statistics used in the tonemapping process are generated by interpolation between the image-statistics used in the previous output-frame, as well as a set of image-statistics interpolated between the four nearest key-views. The interpolation-weight for interpolation between metrics of the previous and current frame was chosen as a time-dependent exponential. For a computation time t between successive frames, the weight w, assigned to the previous image-statistics, is determined as $w = e^{-t/t_0}$ to mimic adaptation in the HVS [39]. t_0 was chosen to be 150 milliseconds, which corresponds to the adaptation-behaviour of rods. This was done as Reinhard 2005 simulates the behaviour of rods and not the behaviour of cones to obtain tonemapping results and as such, using the behaviour of rods is more in line with the tonemapping intent of HVS simulation.

While such a time-adaptation is possible using an interpolation of tonemapping results, a history of past key-view image-statistics and weights is required. This is the case as, unlike with an interpolation of image-statistics, the tonemapping applied to the previous output frame is not directly available. To obtain the tonemapping used in the previous frame, the tonemappings used in all preceding frames are required in a recursive manner as the tonemapping in each frame considers the tonemapping of its predecessor. Due to computation-limitations, only limited history can be considered for this, which is why an interpolation of image-statistics is used instead. An interpolation of image-statistics does not necessarily result in an interpolation of tonemapping results, however in case of Reinhard 2005, the perceived difference is small enough to allow a use image-statistics instead of tonemapping results.

Despite the improvements in temporal consistency of the tonemapping behaviour, the luma-normalisation of output-frames is still prone to outliers. To adjust for this, the minimum and maximum luma-values for normalization are interpolated between the current frame and the last frame using w. As a result, a temporally consistent tonemapping behaviour is obtained while allowing for a high degree of adaptivity towards the lighting conditions in the scene.

5.3 Key-view adjustment

The insight gained into the interplay between TMOs and view-synthesis through the examination of previously proposed approaches has inspired the creation of a novel processing method for LFs. This method has been dubbed "Key-view adjustment" and processes the LF in the epipolar domain by making use of the geometrical relationship between key-views. It adapts key-views, tonemapped by arbitrary TMOs, to be more suitable for LF rendering. The aim is to reduce pixel-differences for correspondences in neighbouring key-views caused by differences in tonemapping to improve tonemapping consistency within novel views and over time for LF rendering. Material-appearances and adaptation to the lighting conditions should be preserved during large camera-movements. The computation time of this processing method is significant, restricting it to offline-application speeds.

The processing goals are achieved by considering the correspondences q of a pixel p in other key-views, which can be obtained directly as the disparity-value of p is known. Given the positions of key-views on the camera-plane F_p and F_q, q can be calculated as

$$q = p + (F_p - F_q) \cdot disp(p).$$

A correspondence-map CM containing correspondences of p in all other key-views can be formed using forward image-warping. In this correspondence-map, not every pixel may be available as not all correspondences lie within the image boundaries. Additionally, an occluder to the point in the scene may be found instead. The differences in pixel values between the pixels corresponding to the same point in the scene may be a result of tonemapping adaptive to the lighting conditions in the key-views or a result of changes in material appearance and volumetric effects. In order to differentiate, the HDR key-views are provided as an input as they only contain differences due to material appearance, volumetric effects and noise.

The processing-goal is to reduce the luma-contrast between the tonemapped pixel value of p and the pixels in the correspondence-map, after a processing of all pixels in all key-views has been performed. With considerations made for occlusions and material appearance, this reduction of luma-contrast represents a reduction in pixel-differences for correspondences in adjacent key-views.

Inspired by the generation of the base-layer in the improved bilateral TMO, a bilateral filter is applied to p in the correspondence map. The contribution assigned to a pixel in the texture is the product of partial weights for distance W_s, disparity W_d and luminance W_l.
A weighting for occlusions through W_d lowers contributions from pixels q, for which the disparity indicates that an occluder to the scene-point was found in the key-view. By choosing σ_d appropriately, the contribution from occluders is reduced to the point that only pixels corresponding to the same point in the scene as p contribute in a meaningful way. W_d is calculates as

$$W_d = G_{\sigma_d}(disp(p) - disp(q)).$$

To preserve the material appearance of the surface, a luminance-weight W_l has been introduced. In case significant disagreements between the HDR-luminance of p and a neighbourhood-pixel q exist, the contribution of q is required to be reduced so as to preserve the material appearance at p. W_l is calculated as

$$W_l = G_{\sigma_l}(log(L_p) - log(L_q)),$$

where L_p and L_q are the HDR-luminance-values for pixels p and q.
A spatial weight W_s is introduced to reduce local differences in luma-values in the correspondence-map while preserving luma-differences over large scales in the correspondence-map. Local differences in luma-values are to be reduced, as differences in luma-values between adjacent key-views are prone to introducing artefacts into novel views. W_s is calculated as

$$W_s = G_{\sigma_s}(||p - q||).$$

As a result, the differences in luma-values between adjacent key-views are reduced, while differences in luma values over large distances are mostly preserved. While this results in a greater tonemapping-consistency within novel views and over time, an adaptation of the tonemapping behaviour towards changes in lighting conditions, occurring between adjacent key-views, is reduced. For scenes in which a rapid adaptation-behaviour is desired, a use of synthesized-view tonemapping is recommended instead.

The weights assigned to the pixels are used to accumulate a weighted sum of luma-values from the tonemapped key-views over the correspondence-map. The result of the bilateral filter BF_1 yields a target-luma value and is given as

$$BF_1(p) = \frac{1}{W_p} \cdot \sum_{q \in CM} W_s \cdot W_l \cdot W_d \cdot C(q),$$

where $C(q)$ refers to the tonemapped pixel value at q and where

$$W_p = \sum_{q \in CM} W_s \cdot W_l \cdot W_d.$$

Dividing the target luma-value for p by the current luma-value of p yields a luma-adjustment factor.

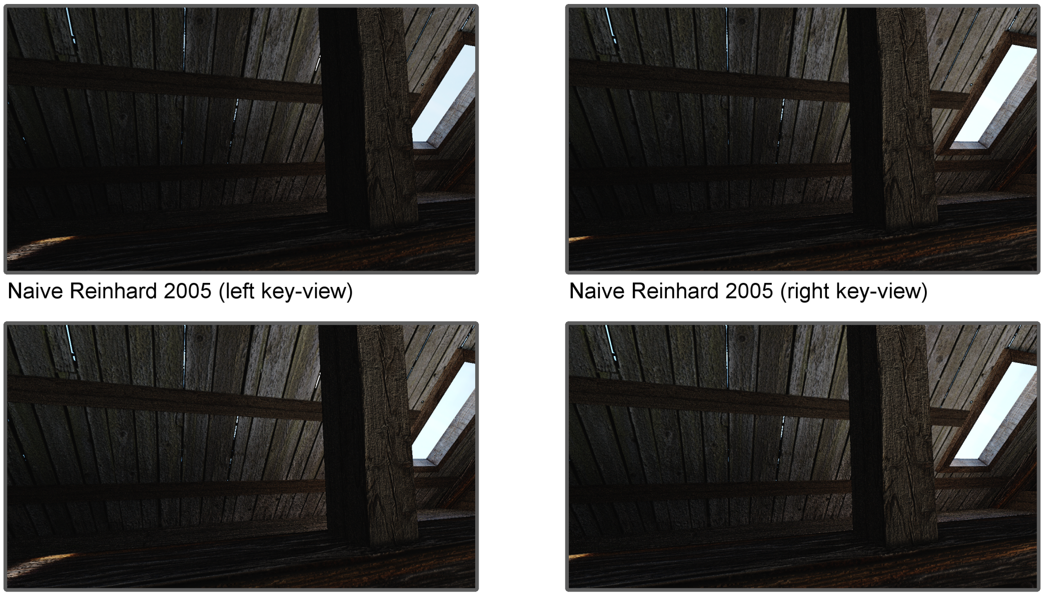
If one was to directly apply the luma-adjustment factor L'_a to all pixels of a key-view, visibility-borders would be created in the key-view. This is the case as the occlusion-weights of correspondences differ for different pixels p in a key-view in accordance with the visibility of the scene-point corresponding to p in the key-views. To avoid the introduction of artefacts, images containing the luma-adjustment factors are processed by a second bilateral filter BF_2 in the neighbourhood S around p, using weights for distance W'_s as well as for disparity W'_d to blur the luma-adjustment factors within same surfaces only. This way, visibility-borders present on surfaces are turned into gradual transitions, which are no longer noticeable artefacts in the key-views. The result of BF_2 yields the processed luma-adjustment factors as

$$BF_2(p) = \frac{1}{W_p} \cdot \sum_{q \in S} W'_s \cdot W_l \cdot W'_d \cdot L'_a,$$

where

$$W_p = \sum_{q \in S} W'_s \cdot W_l \cdot W'_d.$$

Once the luma-values in all key-views are multiplied with the corresponding L'_a, the resulting key-views posses significantly lowered pixel-differences between correspondences in adjacent key-views, while remaining free of visible artefacts. Material-appearances and adaptation to the lighting conditions over large distances on the camera-plane are preserved to a large extent. In figure 30 large differences in tonemapping between two adjacent key-views are present for the naive application of Reinhard 2005. After performing a key-view adjustment, the differences in tonemapping are greatly reduced without introducing visible artefacts into the key-views. Key-view adjustment has been implemented using Matlab [3].



Naive Reinhard 2005 (left key-view)

Naive Reinhard 2005 (right key-view)

Key-view adjustment (left key-view)

Key-view adjustment (right key-view)

Figure 30: Reduction of pixel-differences for adjacent key-views via key-view adjustment (brightness and contrast adjusted for print)

A major benefit of this processing method is the fact that it is applicable to all TMOs without requiring a modification of TMOs to be viable for LF-rendering. The method can also be used for other HDR-LF related tasks, such as preparation of an HDR-LF for use on a specialized LF-display.

6 Results and discussion

In order to asses and compare the qualities of different tonemapping approaches as well as strike a comparison between ground-truth views and views generated during view-synthesis, synthetic LFs were captured from 3D-scenes. The scenes used in this section are already introduced in section 3.1.

As different scenes are challenging for different aspects related to view-synthesis and tonemapping, the individual aspects are discussed and matching examples from the different scenes are provided. For all scenes views were generated using a shifted-frustum camera with the same resolution as the key-views.

The view-synthesis is evaluated separately from tonemapping-related aspects in section 6.1. Geometry reproduction and appearance reproduction are discussed in subsections 6.1.1 and 6.1.2 respectively. The tonemapping of LFs is discussed throughout section 6.2. An evaluation of key-view tonemapping in regards to the visibility of artefacts within novel views is performed in subsection 6.2.1. Temporal aspects of all TMOs adaptive to changes in lighting conditions are analysed in subsection 6.2.2. Lastly, the suitability of TMOs for different scenes outside of a LF rendering context is discussed in section 6.3.

EPI analysis

To better analyse the temporal behaviour of view-synthesis and tonemapping, an epipolar-imaging (EPI) analysis has been performed. First, for a fixed horizontal camera-motion, a set of successive frames is obtained. When stacking successive frames on-top of each other in 3D-space, a 3D-representation of the output video, with one axis representing time and two axis representing space, can be created. The epipolar-images analysed in this thesis are generated by taking horizontal slices through the frames in the 3D-representation of the output video. Each slice details the movement of positions in image-space, shown as lines in the slice, over time. The slope of a line is tied to its disparity-value in the scene and remains constant throughout the EPI in the ground-truth data. A change in pixel value over time along the lines in ground-truth data signifies a change in surface-appearance, which ought to be preserved and emulated in synthesized data. A change in pixel value for tonemapped EPI-slices, for which no corresponding change in the ground-truth data can be observed, indicates a change in tonemapping behaviour over time.

In the following sections, green lines in the EPI indicate which position(s) in the EPI-slice view(s) from the image sequence correspond to. Conversely, green lines in view(s) indicate the cut through the image used to generate the EPI. Red squares indicate which area of the view or EPI has been enlarged.

6.1 View-synthesis

To asses the quality achieved by the view-synthesis employed in the LF rendering system, comparisons are made between ground-truth views rendered in Blender and synthesized views. A gamma-correction has been applied to the views to allow a comparison independent of tonemapping influences. The analysis of the view-synthesis is split into two parts, corresponding to the view-synthesis tasks of geometry reproduction and appearance reproduction as presented in 4. In particular, failure cases and limitation in regards to the visibility of surfaces and reproduction of appearance for volumes and surfaces, further detailed in sections 4.1.4 and 4.2, are examined. To analyse artefacts in novel views, for which the visibility changes upon camera-movement, an EPI is included. In case an artefact-free reconstruction of the scene is achieved or the visibility of artefacts does not change upon camera-movement, no EPI is included.

6.1.1 Geometry reproduction

To asses the reproduction of geometry in a scene, a rendering of the visibility of surfaces in novel views from nearest key-views is included. In the generated false-colour image, green areas indicate a visibility of the surface in all four nearest key-views, while for red areas the background surface is not visible in any nearest key-view. Colours in-between green and red indicate that the surface is visible only in some key-views, with the colour indicating the number of key-views in which the surface is visible. Luma-values used in the image stem from adjusted luma-values of novel views to establish a correspondence between the visibility in the scene and geometry in the scene.

Visibility of surfaces in the scene

A total lack of visibility in the four nearest key-views for surfaces visible in a novel view, may cause rendering-failures, resulting in the introduction of artefacts into novel views.

For the Cellar scene, such a rendering failure can be observed for the vertical timber-beam in figure 31. For some sets of nearest key-views, the timber-beam is not covered by some of the nearest key-view frustums, while other nearest key-view frustums cover parts of the timber-beam. For some points P in the scene, in each key-view P is either occluded by the timber-beam or not covered by the key-view frustum. However, P might be visible from the perspective of some novel views not covering the timber-beam, despite the timber-beam not possessing surface-areas of negative curvature.

In order to restore the background, simply interpolating pixel values of background-samples as described in section 4.1.5 introduces artefacts in the novel view as the background is not uniform.

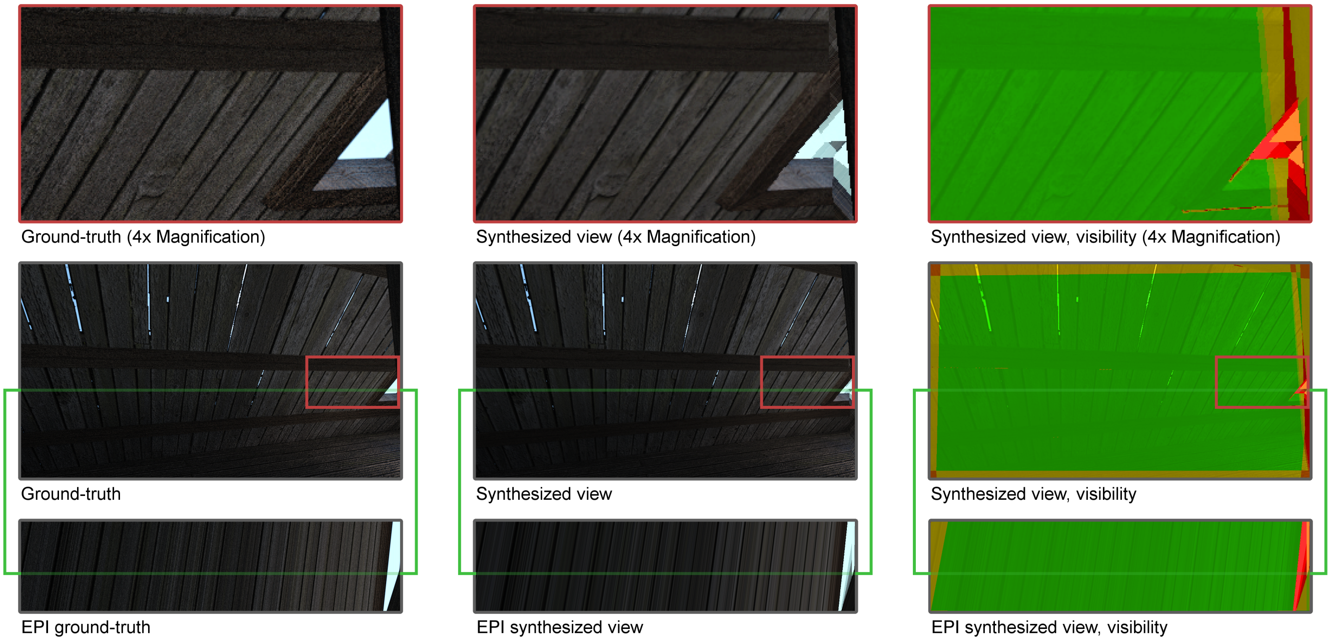
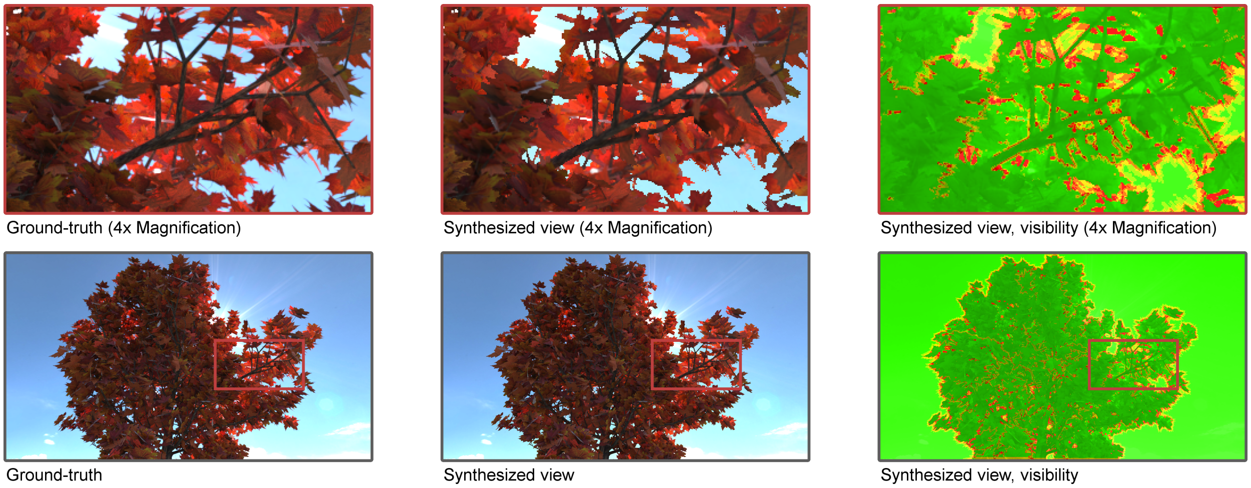

Ground-truth (4x Magnification)
Synthesized view (4x Magnification)
Synthesized view, visibility (4x Magnification)
Ground-truth
Synthesized view
Synthesized view, visibility
EPI ground-truth
EPI synthesized view
EPI synthesized view, visibility

Figure 31: Rendering failure due to missing information for background surfaces (brightness and contrast adjusted for print)

A recovery of the background is possible in the Tree scene, as the occluded parts of sky are mostly uniform as shown in figure 32. An interpolation of background-samples yields a reconstruction of the background which appears to be artefact-free. The source of occlusions, in this case, is the formation of holes in the foliage through multiple leaves acting as occluders as well as single leaves possessing silhouettes with negative curvature.


Ground-truth (4x Magnification)

Synthesized view (4x Magnification)

Synthesized view, visibility (4x Magnification)

Ground-truth

Synthesized view

Synthesized view, visibility

Figure 32: Rendering success despite missing information for background surfaces

Loss of fine structures

In the Tree scene, thin branches of the Tree and fine structures of leaves are lost during view-synthesis as shown in figure 33. For surfaces smaller than 2×2 pixels, correspondences in all nearest key-views are always found to lie between foreground and background. In most cases, the background is chosen as the correspondence, which results in removing the branch from the synthesized view.

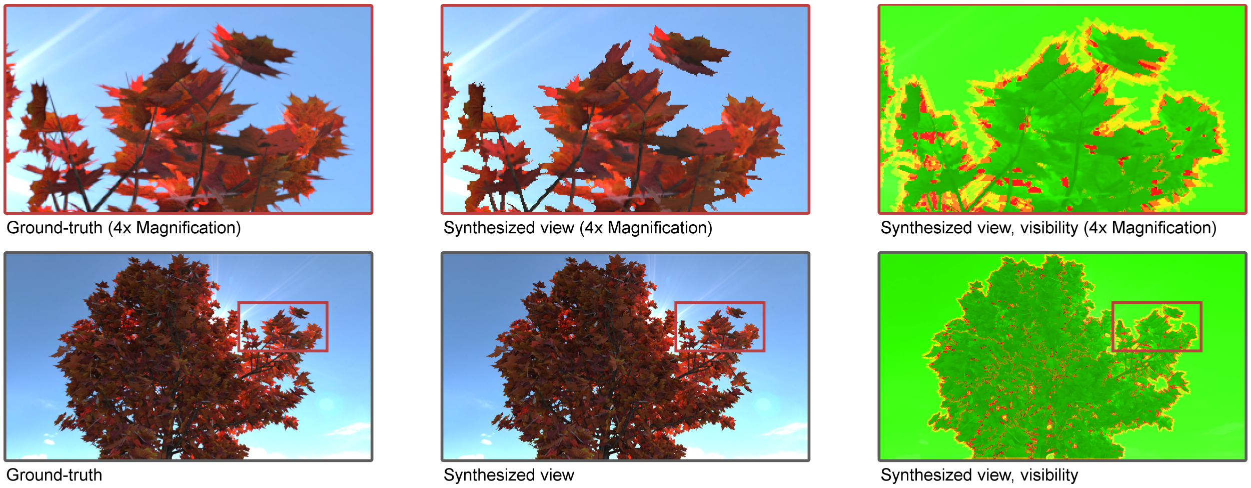

Ground-truth (4x Magnification)
Synthesized view (4x Magnification)
Synthesized view, visibility (4x Magnification)
Ground-truth
Synthesized view
Synthesized view, visibility

Figure 33: Rendering failure for fine structures

6.1.2 Appearance reproduction

Volumetric effects

For volumetric effects, the contribution of the medium to pixels in the key-views is attributed to the background surface through which the medium is observed. For different key-views however, the position on the background surface, to which the contribution of the medium is attributed to, differs. As the contributions of the medium are treated as if they were part of the background surface, the bilinear-interpolation performed during view-synthesis causes copies of the volumetric effect to fade in and out over time during camera-movement. In the Cellar scene with volumetric effects this behaviour can be observed, as shown in figure 34. The fading of copies of volumetric effects can be seen in the EPI.

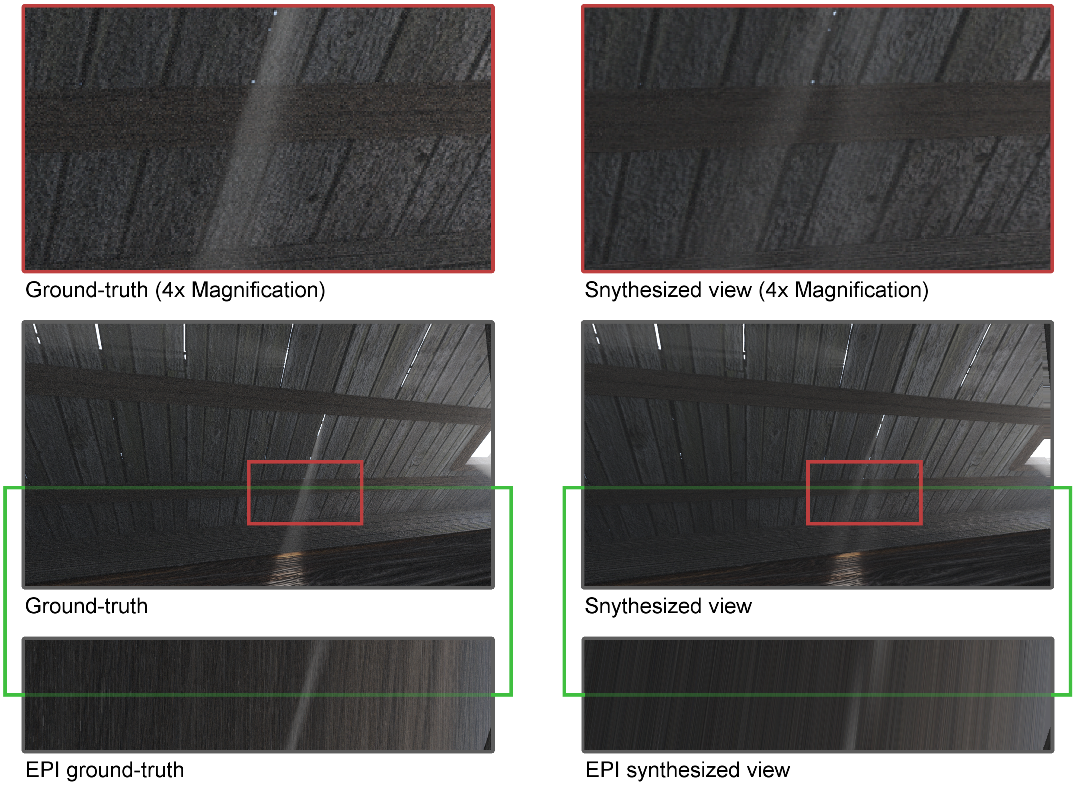
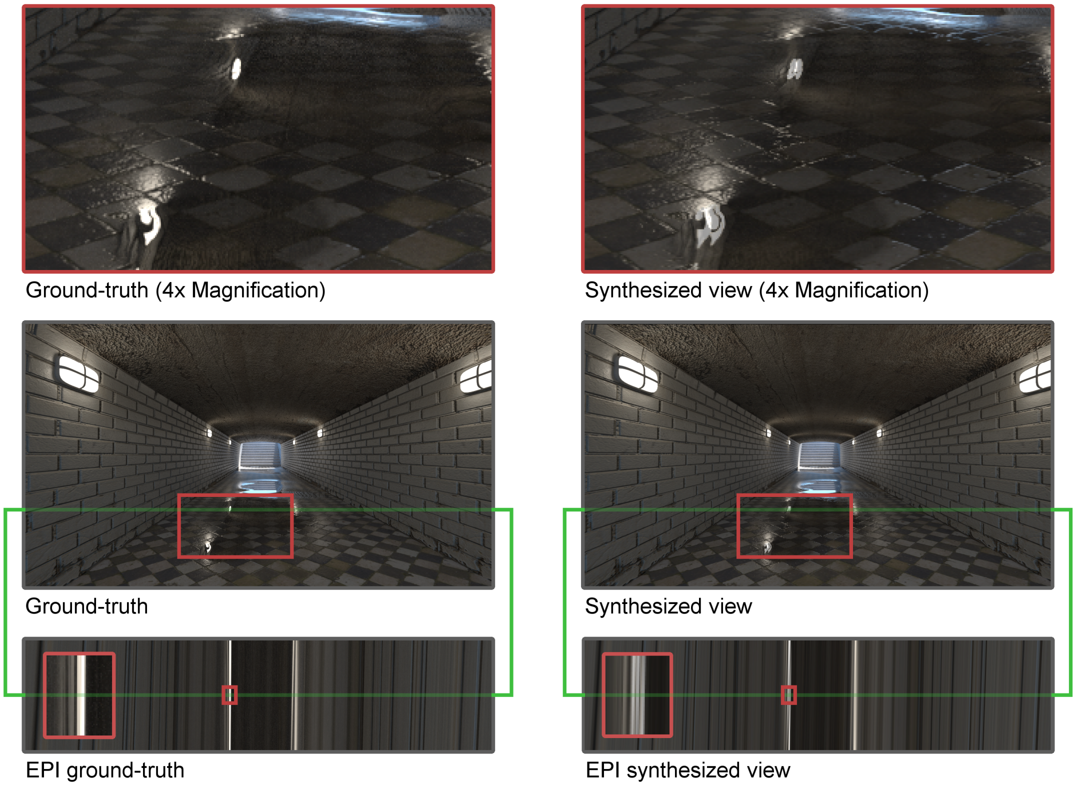

Ground-truth (4x Magnification)
Synthesized view (4x Magnification)
Ground-truth
Synthesized view
EPI ground-truth
EPI synthesized view

Figure 34: Rendering failure for volumetric effects (brightness and contrast adjusted for print)

Glossy and mirror-like BRDF

For surfaces with a glossy and mirror-like BRDF, a reproduction of material appearance may fail as the bilinear interpolation of pixel values from the nearest four key-views fails to reproduce the distribution of secondary rays of the BRDF. Similar to what can be observed for volumetric effects, copies of reflected scene-content fade in and out on during camera-movement. This is especially visible in the Tunnel scene, as shown in figure 35, for correspondences strongly differing in pixel value, such as for highlights created by reflections of light sources or for differing occlusions of the sky. For the Tunnel scene, material-reproduction is not satisfactory in such areas of novel views.



Ground-truth (4x Magnification)

Synthesized view (4x Magnification)

Ground-truth

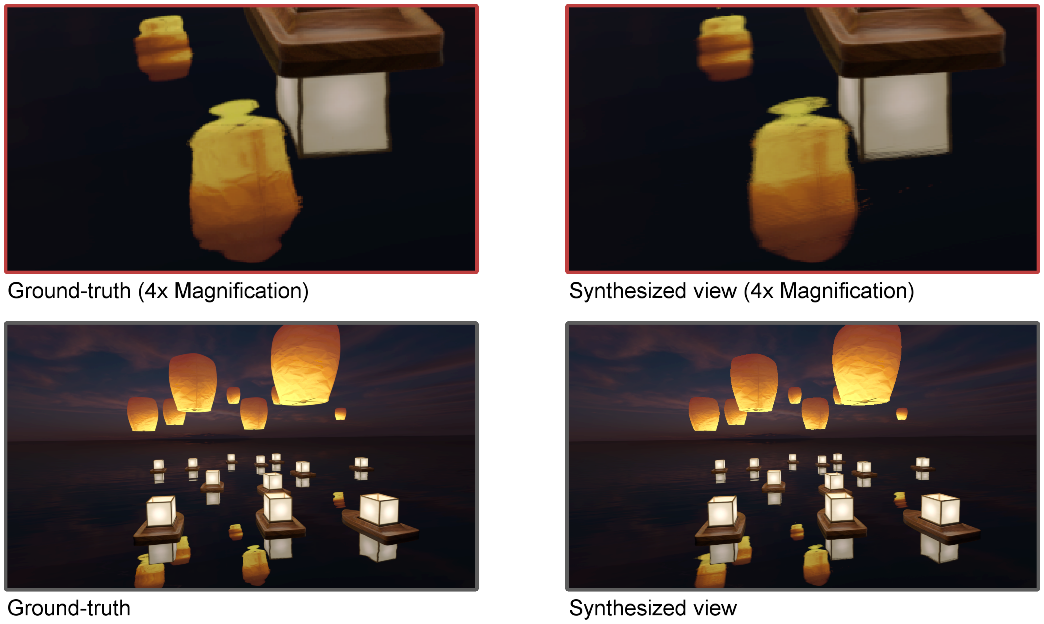
Synthesized view

EPI ground-truth

EPI synthesized view

Figure 35: Rendering failure for a surface with mirror-like contributions

In the Boats and Lanterns scene, artefacts on the water are less visible as shown in figure 36. Due to small distortions on the water surface and a higher spatial density of key-views, areas in which the pixel values change drastically across neighbouring key-views are small. Artefacts are visible as slightly blurry outlines of reflected scene-content instead of overlapping copies of the same reflection and are not noticeable without a ground-truth comparison.



Ground-truth (4x Magnification)

Synthesized view (4x Magnification)

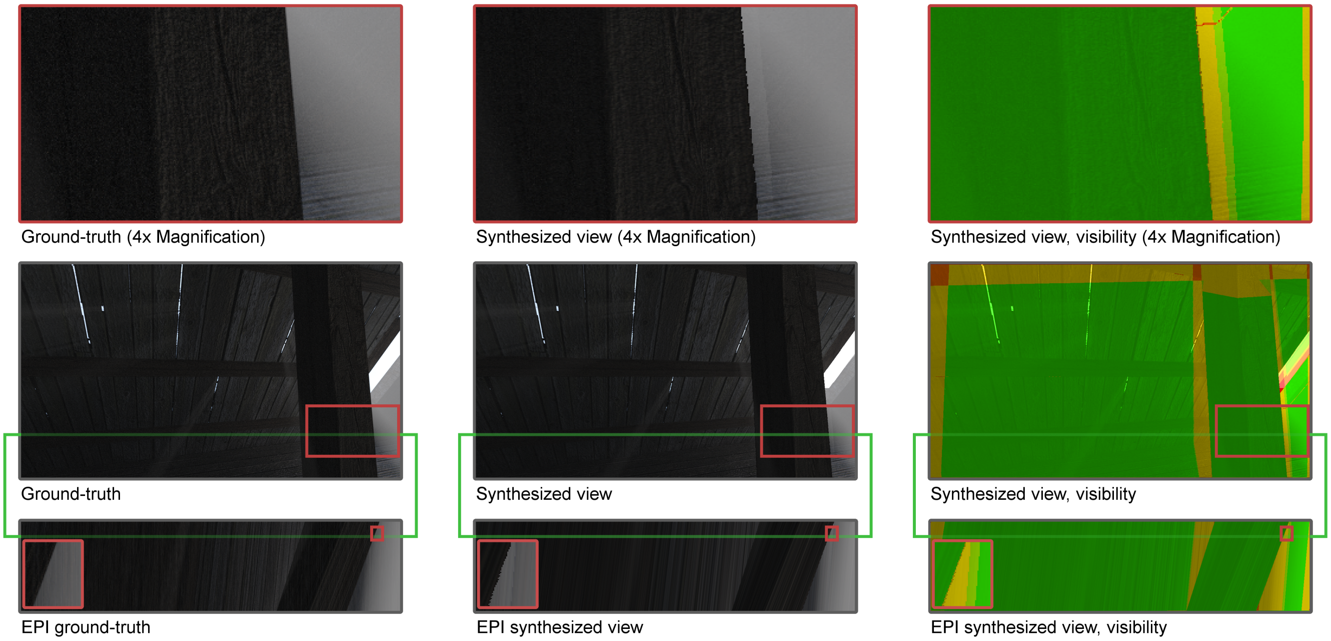
Ground-truth

Synthesized view

Figure 36: Rendering success for a water-surface

Dis-occlusions

While artefacts near visibility-borders caused by changes in key-view contributions are not noticeable for changes in surface appearance in the test-scenes, disocclusions give rise to artefacts in the Cellar scene with volumetric effects as shown in figure 37. As pseudo-surfaces created by disocclusions are removed during the view-synthesis process, the contribution of god-rays is missing for those areas of the novel view. As a result, a sharp edge cutting off a god-ray at a visibility-border can be observed. The edge can be observed to fade over time due to changes in key-view contributions in the EPI. A rendering of surface-visibility is shown to highlight visibility-borders in the novel view.



Ground-truth (4x Magnification)

Synthesized view (4x Magnification)

Synthesized view, visibility (4x Magnification)

Ground-truth

Synthesized view

Synthesized view, visibility

EPI ground-truth

EPI synthesized view

EPI synthesized view, visibility

Figure 37: Rendering failure for disocclusions

6.2 Tonemapping

6.2.1 Tonemapping consistency within novel views

Tonemapping, when applied to key-views in the LF, instead of an application to synthesized views, may introduce tonemapping-inconsistencies into novel views. For TMOs used in this thesis, inconsistencies were visible for the naive applications of Reinhard 2005 and the bilateral TMO. The enhanced bilateral TMO is free of noticeable tonemapping-inconsistencies and for Reinhard 2005 a use of a joint application of Reinhard 2005 is free of tonemapping-inconsistencies while losing adaptivity towards changes in lighting condition. A key-view adjustment for Reinhard 2005 achieves artefact-free novel views at the cost of introducing globally varying tonemapping-inconsistencies, which are not perceptible as artefacts. Adaptation towards changes in lighting conditions taking place over large distances in the LF is preserved by a key-view adjustment.

Comparison between naive bilateral tonemapping and enhanced bilateral tonemapping

Artefacts at visibility-borders are particularly pronounced for the Cellar scene, as the vertical wooden beam posses low enough differences in luminance compared to the ceiling, that Cornsweet-profiles are generated for the naive bilateral TMO. For figure 38 settings for the bilateral filter were chosen to exaggerate the effect for print. As the position of profiles on the ceiling differs between key-views, the half of the Cornsweet-profiles on the ceiling is no longer connected with the other half of the Cornsweet-profiles on the beam in synthesized views. When accounting for disparity during bilateral-filtering in the enhanced bilateral TMO, Cornsweet-profiles on the ceiling have been reduced to the point where no artefacts are visible.

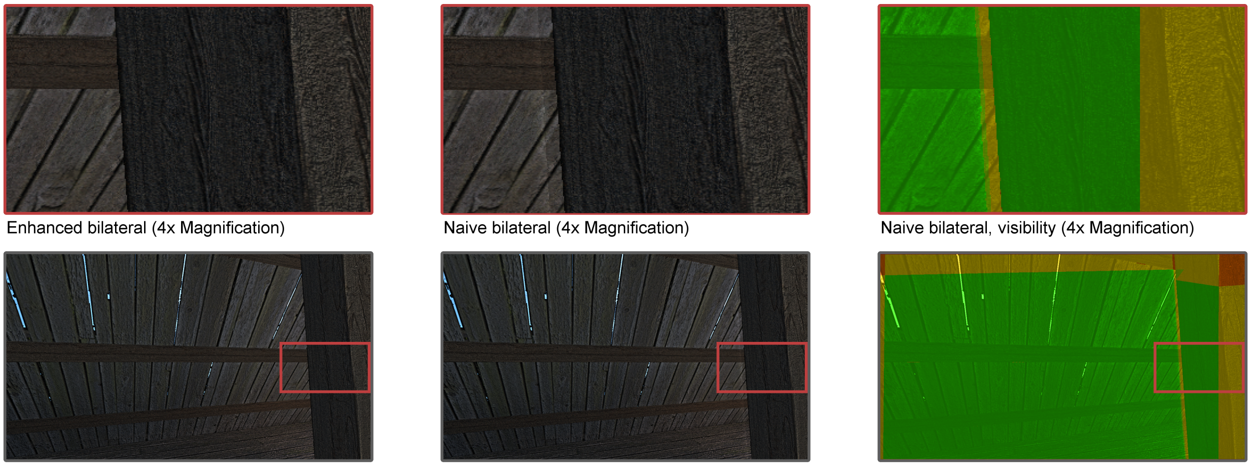

Enhanced bilateral (4x Magnification)
Naive bilateral (4x Magnification)
Naive bilateral, visibility (4x Magnification)
Enhanced bilateral
Naive bilateral
Naive bilateral, visibility

Figure 38: Tonemapping failure recovered in enhanced bilateral-filtering

Scenes with no or minor inconsistencies for naive Reinhard 2005

For scenes without change in lighting conditions and for which the peak luminance in the scene is not affected by noise or outliers, the naive application of Reinhard 2005 provides satisfying results without inconsistencies visible within novel views. Even in scenes heavily affected by noise, such as the Dominoes scene, shown in figure 39, the relatively noise-free highlight on a poster provides a consistent source of peak luminance.

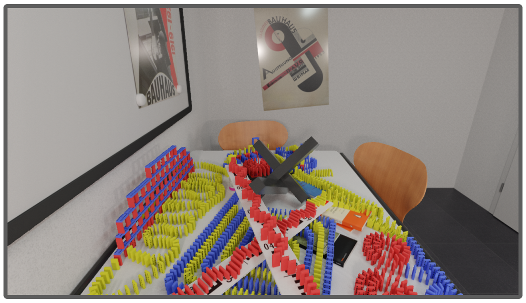
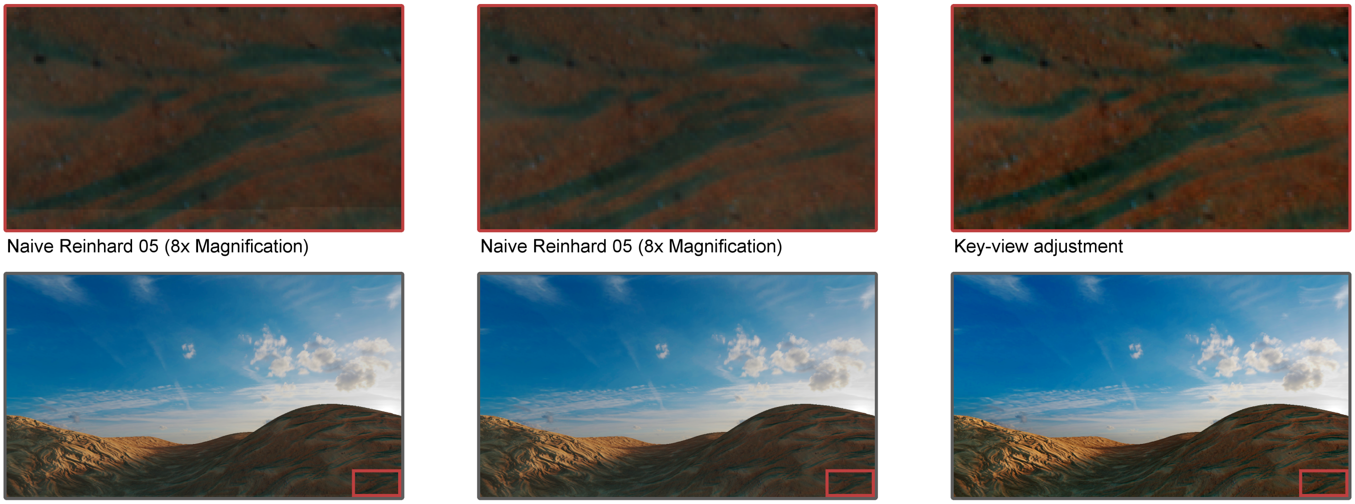


Figure 39: Tonemapping success for naive Reinhard 2005

In scenes with gradually changing lighting conditions, such as the Dunes scene shown in figure 40, the adaptation towards the lighting conditions in the scene is similarly gradual. In this case, a use of naive Reinhard 2005 can be considered preferable to the joint Reinhard 2005 as an adaptation towards changes in lighting conditions is desirable. Differences in pixel values of correspondences resulting from this adaptation are noticeable as artefacts in border regions of key-views for the naive Reinhard 2005 as the key-view contributions vary in the novel view. An application of key-view adjustment eliminates artefacts caused by border-weighting successfully. When disabling border-weighting, a tonemapping-inconsistency in the form of a sharp border is visible in novel views for a naive application of Reinhard 2005. In this particular capture of the Dunes scene, border-weighting causes a brightening of the image in the border region as the cut-off key-view is noticeably darker than other key-views contributing to pixel values.


Naive Reinhard 05 (8x Magnification)

Naive Reinhard 05 (8x Magnification)

Key-view adjustment

Naive Reinhard 05 without boarder-weighting

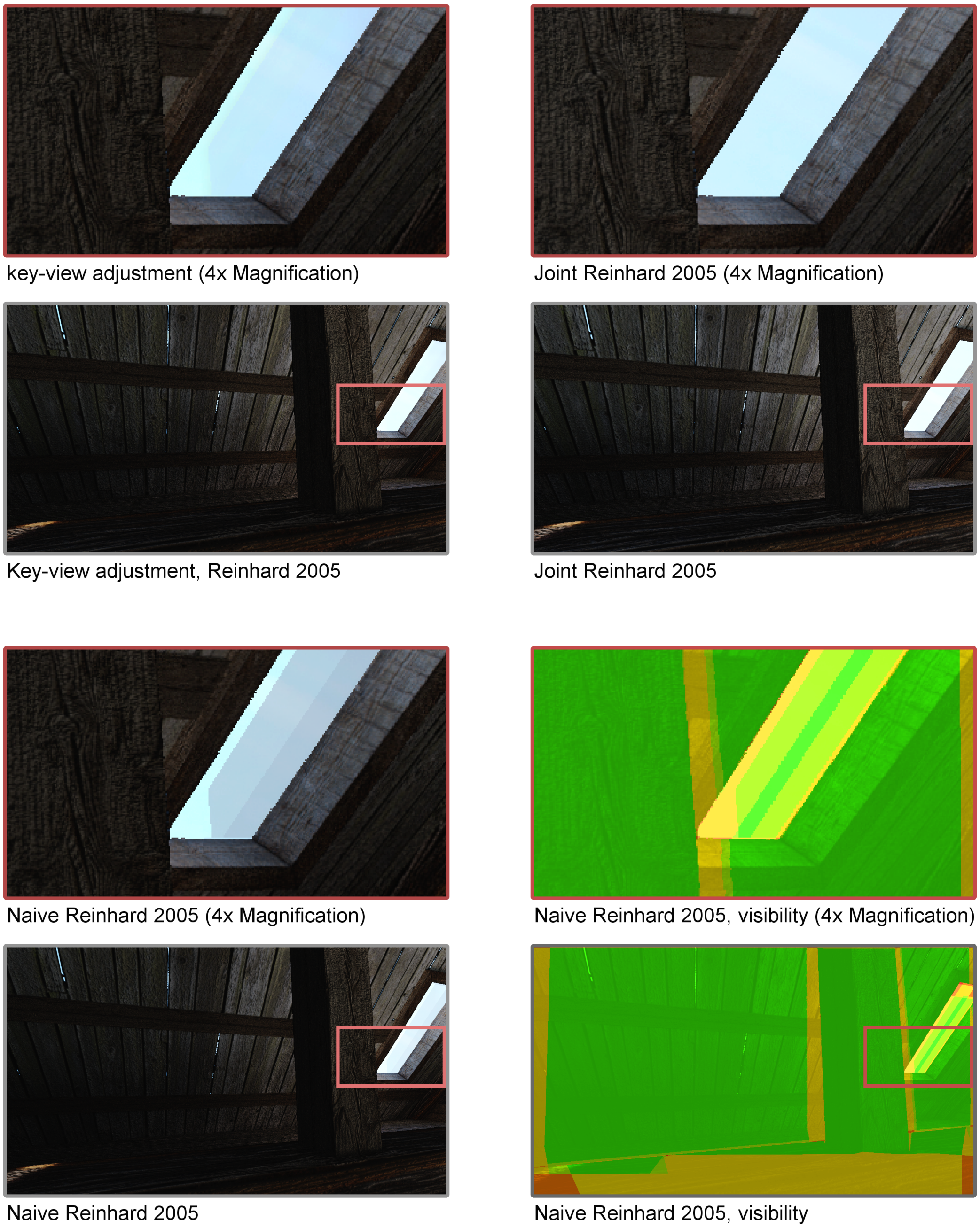
Naive Reinhard 05 with boarder-weighting

Key-view adjustment without border-weighting

Figure 40: Comparison of artefacts in border regions

Scenes with inconsistencies for naive Reinhard 2005

When either rapidly changing lighting conditions are present in the scene or the peak luminance is heavily impacted by noise or outliers, a use of naive Reinhard 2005 introduces artefacts to novel views. In the Cellar scene shown in figure 41, artefacts are particularly noticeable as direct sunlight is visible through slits in the ceiling in some key-views, while being occluded in other key-views. For key-views with a differing visibility of direct sunlight, the tonemapping for a naive application Reinhard 2005 differs noticeably. An application of key-view adjustment generates key-views with barely perceptible changes in tonemapping within key-views, while removing artefacts at visibility-borders from novel views. An application of joint Reinhard 2005 has been provided as an additional comparison as it is free from any inconsistencies in tonemapping. Parameters for luma-normalization were chosen manually for the joint application of Reinhard 2005. A rendering of the visibility of surfaces from the nearest four key-views is included to highlight artefacts at visibility-borders. In figure 41 adjustments for brightness and contrast were made for print. For the novel views, the brightness was increased, while in the magnifications the brightness was increased to allow for a good visibility of artefacts for print.


key-view adjustment (4x Magnification)

Joint Reinhard 2005 (4x Magnification)

Key-view adjustment, Reinhard 2005

Joint Reinhard 2005

Naive Reinhard 2005 (4x Magnification)

Naive Reinhard 2005, visibility (4x Magnification)

Naive Reinhard 2005

Naive Reinhard 2005, visibility

Figure 41: Comparison of artefacts at visibility-borders (brightness and contrast adjusted for print)

6.2.2 Temporal tonemapping-consistency

In terms of temporal consistency, both the variants of Reinhard 2005 and histogram adjustment are of interest as they posses an adaptivity towards changes in lighting conditions. The Reinhard 2005 variants allows for a more rapid adjustment to changing lighting conditions compared to histogram adjustment via the use peak luminance and luma-normalization. For scenes in which the peak luminance is heavily impacted by noise, Reinhard 2005-variants may deliver adaptation-behaviour inconsistent with the lighting conditions. For such scenes, a high spatial density of the LF is especially detrimental as image-statistics are generated on a per-key-view basis. As both the tonemapping- and image-statistic-interpolating Reinhard 2005 variants behave similarly, the statistic-interpolating variant has been chosen for comparison due to its superior temporal consistency for fast camera-movements and improved robustness during luma-normalization.

Unintended adaptation-behaviour for scenes with no change in lighting conditions

In the Desk scene in particular, a change in tonemapping behaviour due to outliers can be observed for Reinhard 2005-variants, while histogram adjustment delivers temporarily consistent results as shown in figure 42. A use of key-view adjustment for Reinhard 2005 delivers consistent results as changes in tonemapping behaviour are preserved on large scales only. For the statistic-interpolating Reinhard 2005 TMO, the change in tonemapping behaviour is less gradual compared to the naive Reinhard 2005. This is the case, since as soon as the peak-luma of the output changes due to one outlier becoming occluded, the tonemapping-behavior rapidly adapts due to luma-normalization. For the naive Reinhard 2005, the adaptation-behaviour is tied to the change in key-view contributions over time and thus more gradual. In the EPI, this change can be seen as an overall change in brightness between the selected key-frames of the image sequence.

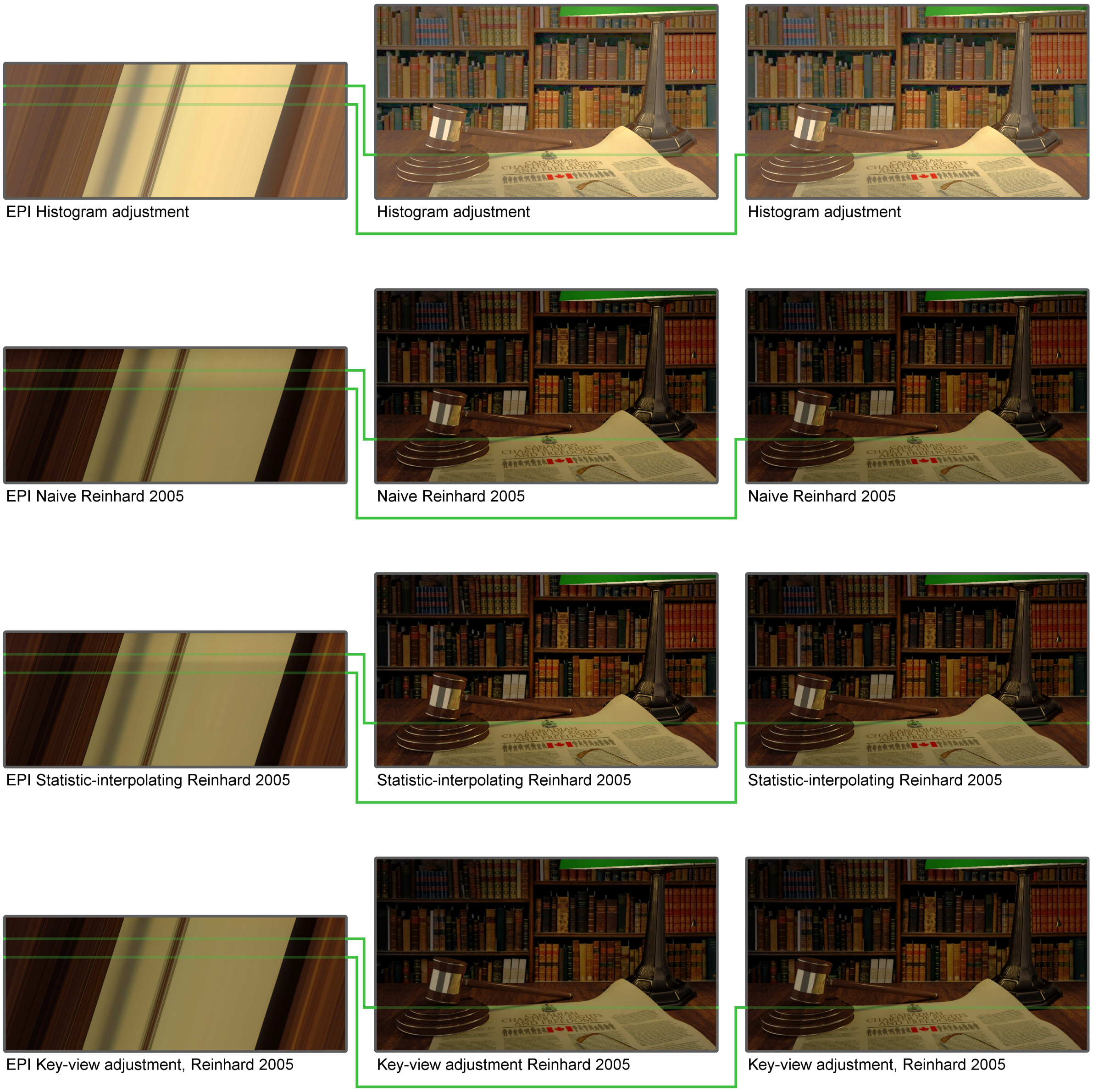

EPI Histogram adjustment
Histogram adjustment
Histogram adjustment
EPI Naive Reinhard 2005
Naive Reinhard 2005
Naive Reinhard 2005
EPI Statistic-interpolating Reinhard 2005
Statistic-interpolating Reinhard 2005
Statistic-interpolating Reinhard 2005
EPI Key-view adjustment, Reinhard 2005
Key-view adjustment Reinhard 2005
Key-view adjustment, Reinhard 2005

Figure 42: Comparison of adaptation-behaviour over time for the Desk scene

Rapid changes in lighting conditions

For the Cellar scene, shown in figure 43, peak luminance values change depending on the visibility of the sun through the slits in the ceiling. The direct sunlight is not subject to noise and as such, changes in tonemapping behaviour due to outliers are less noticeable for statistic-interpolating Reinhard 2005 TMO when compared to the Desk scene. A particularly large change in lighting conditions takes place when a slit providing direct sunlight becomes occluded by the vertical timber-beam. For the statistic-interpolating Reinhard 2005, this causes an intentional rapid adaptation to take place. For naive Reinhard 2005 this change is gradual once more and starts before a change in lighting conditions takes place. As for different key-views the visibility of the slit differs, transitions in tonemapping behaviour are visible as artefacts at visibility-borders in novel views for the naive application of Reinhard 2005. Due to the changing key-view contributions over time, the visibility of artefacts varies over time. This behaviour is highlighted in the enlarged section of the EPI, where visibility-borders are visible in the sky as well as in the first key-view selected from the image sequence. A key-view adjustment yields results without visible artefacts at visibility-borders and provides a slow adaptation to the lighting conditions as the camera is moved towards or away from the window. A rapid adaptation to the occlusion of direct sunlight through the beam is largely lost.

As the bright slits in the ceiling contribute little to the histogram of key-views, a histogram adjustment results in a very slight, slow adaptation to the change in lighting conditions near the window. The number of trimmings performed during histogram adjustment is set to 4 for the Cellar scene as in dark scenes bins in the histogram are strongly trimmed during each trimming. A differing number of trimmings performed between neighbouring key-views introduces visible artefacts to novel views.

A rendering of the visibility of surfaces from the nearest four key-views is included in figure 43 to highlight artefacts at visibility-borders as well as changes in visibility over time in the corresponding EPI. In figure 43 adjustments to brightness and contrast for print were made to the Reinhard 05 variants. In the novel views the brightness was increased and in the magnifications brightness and contrast were adjusted to allow for a good visibility of artefacts for print.

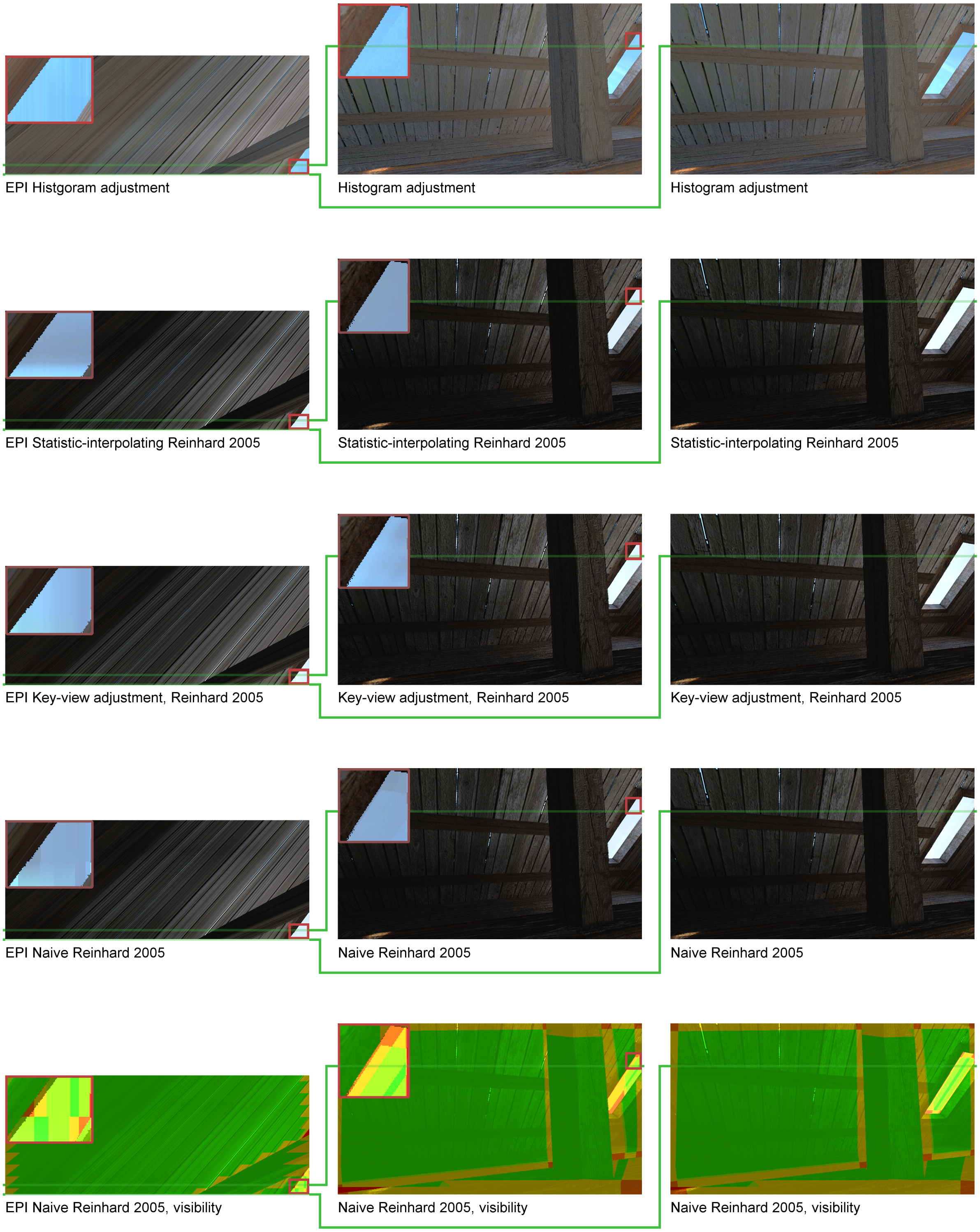

EPI Histgoram adjustment
Histogram adjustment
Histogram adjustment
EPI Statistic-interpolating Reinhard 2005
Statistic-interpolating Reinhard 2005
Statistic-interpolating Reinhard 2005
EPI Key-view adjustment, Reinhard 2005
Key-view adjustment, Reinhard 2005
Key-view adjustment, Reinhard 2005
EPI Naive Reinhard 2005
Naive Reinhard 2005
Naive Reinhard 2005
EPI Naive Reinhard 2005, visibility
Naive Reinhard 2005, visibility
Naive Reinhard 2005, visibility

Figure 43: Comparison of adaptation-behaviour over time for the Cellar scene (brightness and contrast adjusted for print)

Gradual changes in lighting conditions

In the Dunes scene, shown in figure 44, peak luminance values smoothly vary as more or less of the sun is visible behind the Dunes. Due to this, both the naive Reinhard 2005 TMO as well as the histogram adjustment TMO provide consistent results with a slow adaptation over time, which is less pronounced for histogram adjustment. The statistic-interpolating Reinhard 2005 TMO behaves similarly to the naive Reinhard 2005. For the Reinhard 2005 TMO, artefacts introduced by border-weighting are visible when transitioning between different sets of nearest key-views. This can be observed in the enlarged section at the lower border of the EPI in figure 44 (brightness and contrast adjusted for print), which exhibits a different tonemapping behaviour over time when compared to the rest of the EPI. A key-view adjustment is able to remove those inconsistencies, however the adaptation towards the lighting conditions is reduced as well due to the limited extent of the camera-plane. The adaptation-behaviour for a key-view adjustment of Reinhard 2005 is overall more consistent over time when compared to the other Reinhard 2005 variants, which show a slightly irregular adaptation-behaviour. Particularly the statistic-interpolating Reinhard 2005 exhibits a for this scene undesirable rapid adaptation. In practice, inconsistencies in adaptation-behaviour are not noticeable as artefacts and are only apparent in the EPI. The EPI recorded for in figure 44 is created along a vertical cut through the 3D-representation of the image sequence, combined with a vertical camera-movement. This is done as the adaptation-behaviour is most strongly observed for vertical camera-movements in the scene. In figure 44 the magnifications in the EPI were adjusted for print via an increase of brightness and contrast. In the EPI for histogram adjustment, brightness and contrast were increased for print as well.

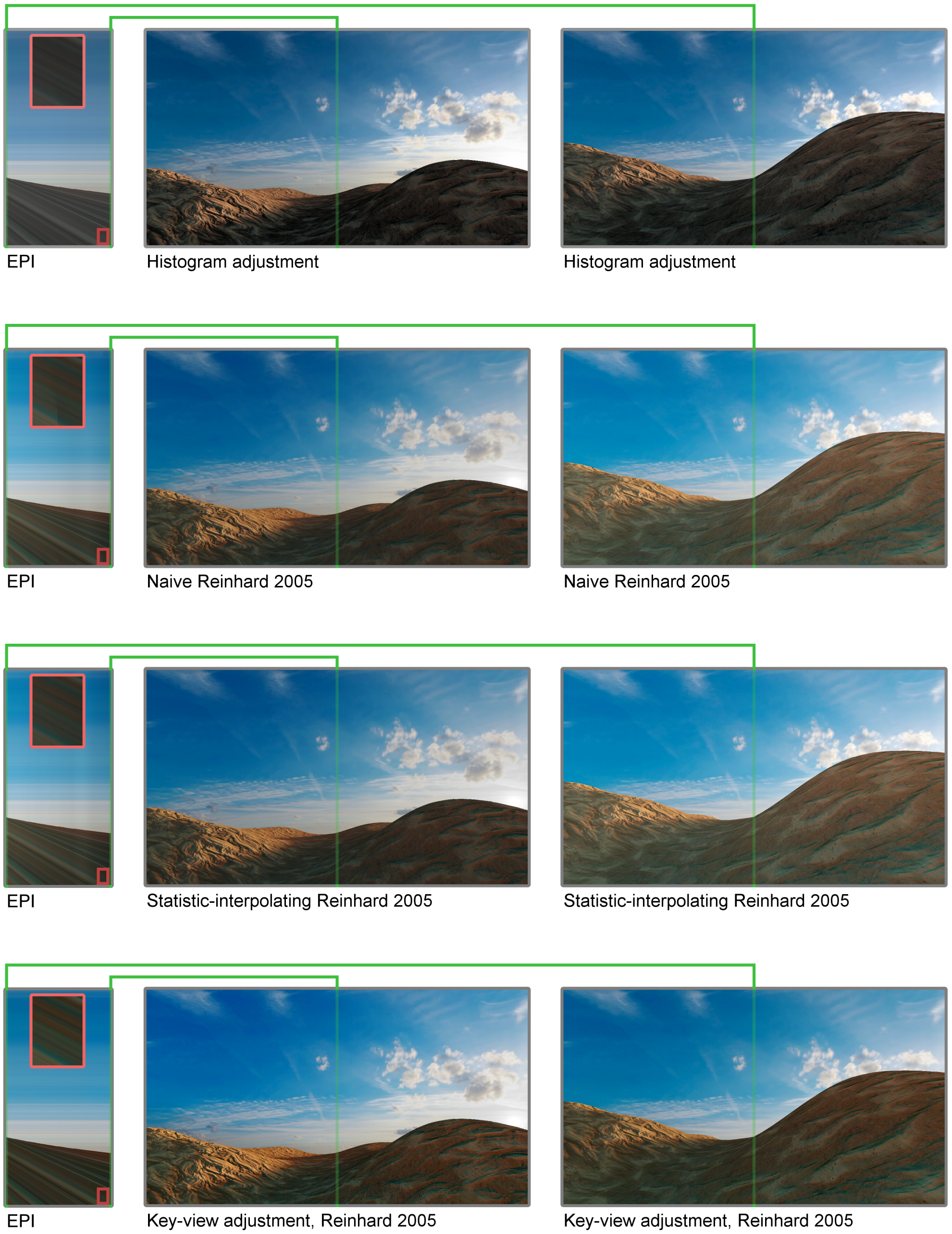

EPI
Histogram adjustment
Histogram adjustment
EPI
Naive Reinhard 2005
Naive Reinhard 2005
EPI
Statistic-interpolating Reinhard 2005
Statistic-interpolating Reinhard 2005
EPI
Key-view adjustment, Reinhard 2005
Key-view adjustment, Reinhard 2005

Figure 44: Comparison of adaptation-behaviour over time for the Dunes scene (adjusted brightness and contrast for print)

6.3 Suitability of tonemapping operators for individual scenes

This section evaluates the tonemapping behaviour of operators based on how suitable the TMOs are in regards to the requirements of the different scenes. This evaluation is detached from the view-synthesis aspect of LF rendering and is performed on the results delivered for selected key-views in the scene. As such, naive Reinhard 2005, enhanced bilateral filtering and histogram adjustment are used in this evaluation. As most TMOs deliver appropriate results for most scenes, this section will focus on failure-cases instead.

Histogram adjustment

Histogram adjustment is focused on preserving the visibility of scene-content. For scenes containing indoor and outdoor content, visibility is preserved within areas containing indoor and outdoor at the cost of reducing the contrast between indoor and outdoor content. Histogram adjustment does not possess any parameters to control or enforce a minimum preservation of contrast between indoor and outdoor content and as such, these contrasts may be strongly reduced. This reduction of contrast may be noticeable as an artefact generated by the TMO and has been especially noticeable for the Cellar scene without volumetric effects as shown in figure 45. In this scene, the contrast between the comparatively bright sky visible through the window and the comparatively dark interior of the Cellar is very low. The dark interior appears much too bright for this scene and does not reflect the lighting conditions.

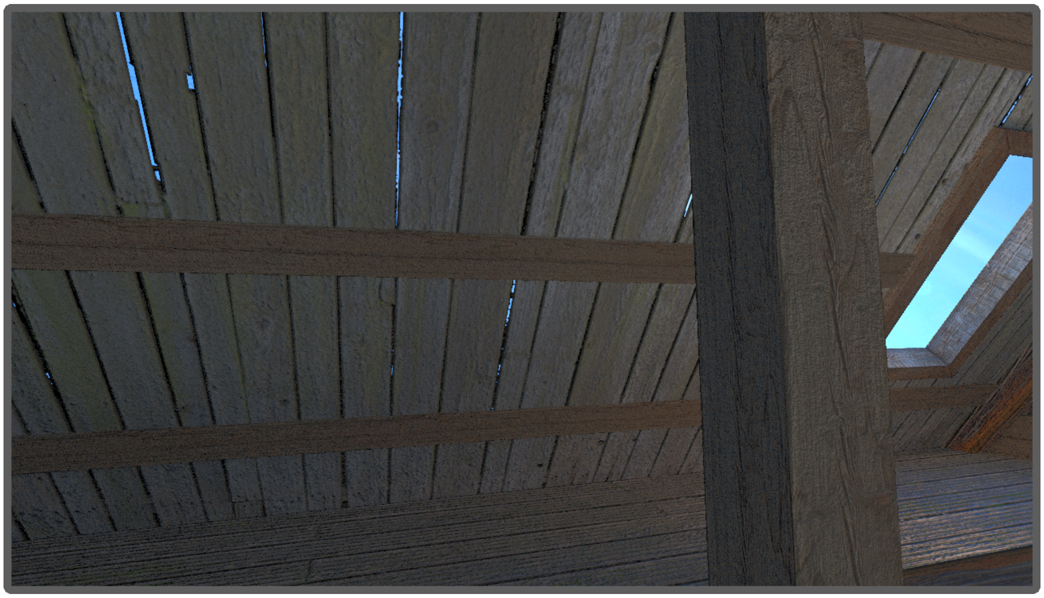


Figure 45: Tonemapping-failure for histogram adjustment

Enhanced bilateral-filtering

As enhanced bilateral-filtering is not adaptive towards the lighting conditions in the scene and makes use of a linear-scaling of the base-layer in the logarithmic domain, it is poorly suited for scenes containing areas with strongly differing luminance-values. In the Boats and Lanterns scene, shown in figure 46, a compression of the base-layer is required to assign appropriate luma-ranges to both the bright Boats and Lanterns as well as the dark water and sky. Large parts of the scene, such as the water, contain only little high-frequency luminance content and as such, a preservation of local contrasts has little contribution to the appearance of those areas. Due to the compression of the base layer, the output lacks global contrasts within the dark and bright parts of the scene, especially the water, and a large part of the dynamic range of the display remains unused.

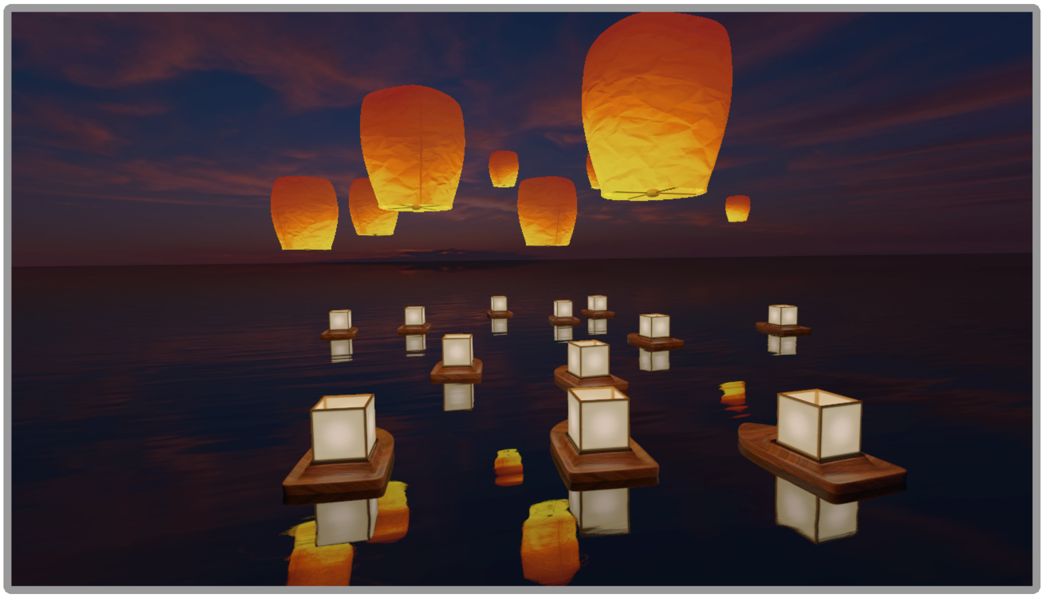


Figure 46: Tonemapping-failure for bilateral filtering

Naive Reinhard 2005

Due to the inherent s-shape of the sigmoid-curve used to compress the dynamic range, bright and dark regions of a scene have their luminance-range compressed into a disproportionally small luma-range. In the Cellar scene with volumetric effects in particular, shown in figure 47, the high intensity of direct sunlight causes the floor and ceiling to lack brightness and contrasts even when choosing large values for the brightness parameter. A choice of a large brightness-parameter leads to an increased compression of the tonemapping-curve for high luminance-values and the visibility of contrasts in bright regions of the image is poor. For the Cellar-scene with volumetric effects, this causes the god-rays to strongly mask textures and edges, making it difficult to distinguish surfaces and structures behind god-rays. Especially near the window in figure 47 a strong masking of textures and edges by the god-ray can be observed.

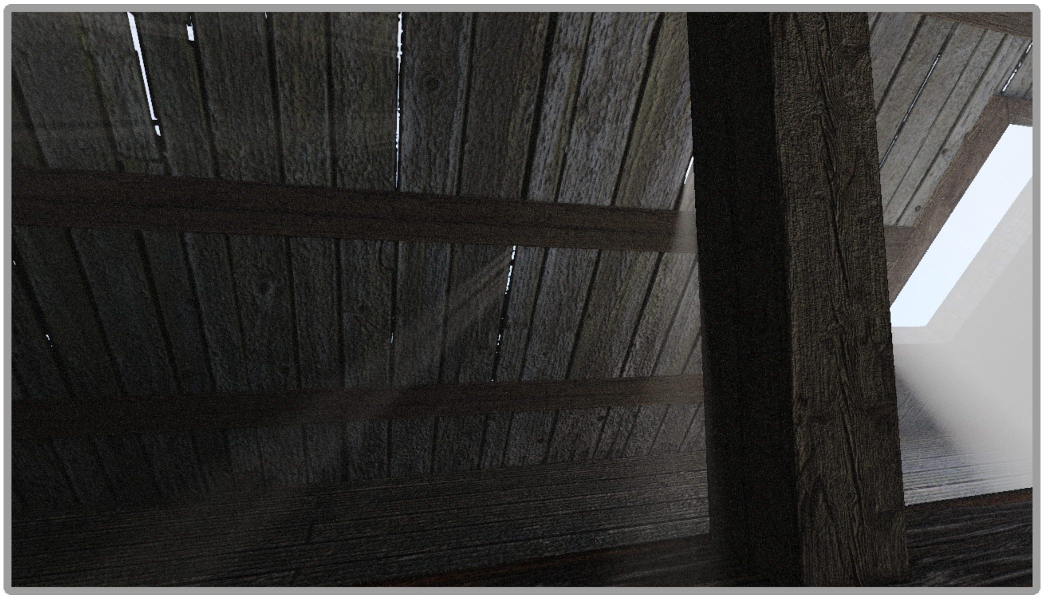


Figure 47: Tonemapping-failure for Reinhard 2005

7 Conclusion

A real-time LF rendering system suitable for sparse HDR-LFs has been developed over the course of this thesis. An analysis of three common TMOs and their use and application in a LF rendering system has been performed. During this analysis, the different techniques employed, aside from histogram adjustment, were shown to require modifications to be suitably applied to HDR-LF data or novel views. A processing method, applicable to various TMOs, has been developed in the form of key-view adjustment and allows for artefact-free LF rendering. To achieve this result, differences in tonemapping behaviour between neighbouring key-views are reduced and as such, a rapid adaptation towards changes in lighting conditions is lost.
This restriction of adaptation-behaviour is ultimately rooted in the use of per-key-view tonemapping, which inherently introduced artefacts to novel views in case the tonemapping behaviour of neighbouring key-views differs noticeably. While a synthesized-view tonemapping has been shown to be capable of a rapid adaptation towards changes in lighting conditions, this rapid adaptation has been made possible by the use of non-robust image-statistics. In scenes in which these image-statistics strongly differ between key-views due to noise or outliers, the lack of robustness results in a tonemapping behaviour inconsistent with the lighting conditions in the scene.

To further improve on tonemapping for HDR-LF rendering, the development of a problem-aware synthesized-view tonemapping-scheme, taking advantage of the possibles for pre-processing the LF-data, holds promise. The use of image-statistics, which allow for a strong adaptivity towards the lighting conditions in the scene, while simultaneously possessing a sufficient robustness towards outliers, are likely key to creating such a tonemapping-scheme.
While for LF rendering further improvements are foreseeable, in case display-ready key-views are required as an input for novel LF-displays or other applications, the key-view adjustment processing method presented in this thesis already delivers promising results. Improvements aimed at preserving a greater degree of adaptivity towards the lighting conditions, while simultaneously enforcing limits to differences in pixel value for correspondences, may further improve upon the key-view adjustment processing scheme.

A use of LF-rendering for VR applications and stereo vision has not been examined in this thesis and the creation of TMOs suitable for stereo-vision is a field in which future work is likely to take place. This is especially the case as an application of LFs is inherently suitable for stereo and VR applications and novel LF-displays include stereo as key-feature of the display. An incorporation of tonemapping-goals related to preserving visual cues for stereo-vision, such as preserving contrasts across depth-edges, as well as ensuring a consistent tonemapping behaviour for both eyes, while allowing for a rapid adaptation to changes in lighting conditions are likely fields of study.

In conclusion, while a development of general tonemapping-solutions for HDR-LF can be expected to take place in the future, superior results are likely to be achieved when tailoring the tonemapping-solution towards the specific application and intended use of the HDR-LF data. Especially for novel LF-displays, new solutions are both required and likely to be created in a display-specific fashion.

List of Figures

Bibliography

[1] Camera-icon made by freepik. https://www.flaticon.com/free-icon/photo-camera-outline-in-perspective-variant_39112.

[2] Environement-maps used in scenes (demo version). https://blendermarket.com/products/pro-lighting-skies.

[3] matlab.

[4] Rgbe format (used in lf file). https://github.com/banterle/HDR_Toolbox.

[5] Textures used in scenes. cgtextures.com.

[6] acdesign55. Camera icon.

[7] Adelson and W. et al. Light field inside a camera lenslet-based light field camera. https://www.slideshare.net/cameraculture/coded-photography-ramesh-raskar.

[8] V. K. Adhikarla, M. Vinkler, D. Sumin, R. Mantiuk, K. Myszkowski, H.-P. Seidel, and P. Didyk. Towards a quality metric for dense light fields. In *Proceedings of the IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2017.

[9] S. H. Ahn. Histogram, 2016. http://www.songho.ca/dsp/histogram/histogram.html.

[10] K. Akeley. Light-field imaging approaches commercial viability, 2015. http://informationdisplay.org/IDArchive/2015/NovemberDecember/FrontlineTechnologyLightFieldImaging.aspx.

[11] S. Anstis, I. Howard, and B. Rogers. A craik-o'brien-cornsweet illusion for visual depth. *Vision research*, 18(2):213217, 1978.

[12] blenderjunky. Legal desktop (desk scene).

[13] E. Britannica. Curvature.

[14] R. Butler. Lytro light field camera. https://www.dpreview.com/reviews/lytro.

[15] P. Debevec. Experimenting with light fields, 2018. https://www.blog.google/products/google-ar-vr/experimenting-light-fields/.

[16] K. Devlin and E. Reinhard. Dynamic range reduction inspired by photoreceptor physiology. *IEEE Transactions on Visualization and Computer Graphics*, 11:13–24, 01 2005.

[17] P. Didyk, T. Ritschel, E. Eisemann, K. Myszkowski, H.-P. Seidel, and W. Matusik. A luminance-contrast-aware disparity model and applications. *ACM Trans. Graph.*, 31(6):184:1–184:10, Nov. 2012.

[18] F. Durand. Applications siggraph 2017, 2007. http://people.csail.mit.edu/sparis/bf_course/slides/04_applications_simple_bf.pdf.

[19] G. Eilertsen, R. K. Mantiuk, and J. Unger. A comparative review of tone-mapping algorithms for high dynamic range video. *Computer Graphics Forum*, 36(2):565–592.

[20] A. K. et al. Multiview imaging and 3dtv. 10, November 2007.

[21] B. M. et al. A survey on computational displays: Pushing the boundaries of optics, computation, and perception.

[22] M. L. et. al. The stanford multi-camera array. http://graphics.stanford.edu/projects/array/.

[23] N. B. et al. Light-field imaging and display systems, 2016. http://informationdisplay.org/IDArchive/2016/JulyAugust/FrontlineTechnologyLightFieldImaging.aspx#R6.

[24] D. F. and D. J. Fast bilateral filtering for the display of high-dynamic-range images. *ACM Transactions on Graphics*, (3):257–266, 12 2002.

[25] S. B. K. Harry Shum. Review of image-based rendering techniques, 2000.

[26] A. Jones, I. McDowall, H. Yamada, M. Bolas, and P. Debevec. Rendering for an interactive 360&deg; light field display. *ACM Trans. Graph.*, 26(3), July 2007.

[27] S. B. Kang, J. A. Webb, C. L. Zitnick, and T. Kanade. A multibaseline stereo system with active illumination and real-time image acquisition. In *Proceedings of IEEE International Conference on Computer Vision*, pages 88–93, June 1995.

[28] G. Kramida. Resolving the vergence-accommodation conflict in head-mounted displays. 22:1912 – 1931.

[29] D. Lanman and D. Luebke. Near-eye light field displays. *ACM Trans. Graph.*, 32(6):220:1–220:10, Nov. 2013.

[30] D. Lanman, G. Wetzstein, M. Hirsch, W. Heidrich, and R. Raskar. Polarization fields: Dynamic light field display using multi-layer lcds. *ACM Trans. Graph.*, 30(6):186:1–186:10, Dec. 2011.

[31] H. Lensch. Brdfs and texturing.

[32] M. Levoy. Light fields and computational imaging. *Computer*, 39(8):46–55, Aug 2006.

[33] M. Levoy and P. Hanrahan. Light field rendering. In *SIGGRAPH '96 Proceedings of the 23rd annual conference on Computer graphics and interactive techniques*, pages 31–42, 1996.

[34] I.-C. Lin. Image-based modeling and rendering.

[35] Z. Lin and H.-Y. Shum. A geometric analysis of light field rendering. *International Journal of Computer Vision*, 58(2):121–138, Jul 2004.

[36] R. Mantiuk. Tone mapping. `https://www.cl.cam.ac.uk/~rkm38/pdfs/tone_mapping.pdf`.

[37] u. s. MIP research group. Original .lf data-format.

[38] B. Neal. Survey of alternative displays, 2016.

[39] S. N. Pattanaik, J. Tumblin, H. Yee, and D. P. Greenberg. Time-dependent visual adaptation for fast realistic image display. In *Proceedings of the 27th Annual Conference on Computer Graphics and Interactive Techniques*, SIGGRAPH '00, pages 47–54, New York, NY, USA, 2000. ACM Press/Addison-Wesley Publishing Co.

[40] K. M. Rafa K. Mantiuk and H.-P. Seidel. High dynamic range imaging. In M. K. Publishers, editor, *Wiley Encyclopedia of Electrical and Electronics Engineering*. 2010.

[41] H. Rushmeier, C. Piatko, and G. W. Larson. A visibility matching tone reproduction operator for high dynamic range scenes. *IEEE Transactions on Visualization and Computer Graphics*, 3:291–306, 10 1997.

[42] C. Tomasi and R. Manduchi. Bilateral filtering for gray and color images. In *Sixth International Conference on Computer Vision (IEEE Cat. No.98CH36271)*, pages 839–846, Jan 1998.

[43] J. Weickert. 3-d reconstruction ii: Stereo. password and username: ipcv16.

[44] J. Wen, X. Yan, X. Jiang, Z. Yan, Y. Wang, and J. Wang. Nonlinear mapping method for the generation of an elemental image array in a photorealistic pseudoscopic free 3d display. *Appl. Opt.*, 57(22):6375–6382, Aug 2018.

[45] F. Yaraş, H. Kang, and L. Onural. State of the art in holographic displays: A survey. *J. Display Technol.*, 6(10):443–454, Oct 2010.