Computer Graphics

HDR Imaging

Philipp Slusallek

Overview

- **HDR Acquisition**
- **Tone-Mapping**

High Dynamic Range Imaging

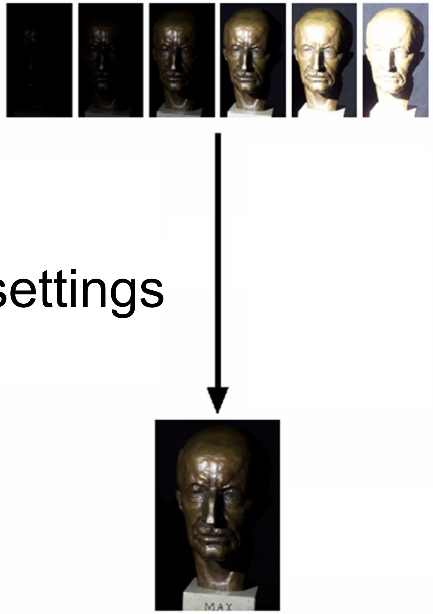
- **Contrast Handling**
  - Input: HDR intensities in real-world scenes (e.g., from rendering)
  - Output: Typically, LDR devices
- **Acquisition of HDR input**
  - HDR cameras
    - Still rather exotic
  - LDR cameras
    - Requires multiple exposures to fully cover the high dynamic range
    - Now often integrated (e.g., into smartphones)
- **Display**
  - HDR displays
    - Modern displays are now getting more and more HDR capable
    - Sometime >1000 individual LED in backlight or micro-LED display
  - Display on LDR monitors
    - *Tone mapping* to perceptively compress HDR to LDR

Part I

HDR Acquisition

Acquisition of HDR from LDR

- **Limited dynamic range of cameras is a problem**
  - Shadows are underexposed
  - Bright areas are overexposed
  - Sensor's temporal sampling density is not sufficient → saturation
- **Good sign**
  - Some modern CMOS imagers have a higher (and often sufficient) dynamic range than most traditional CCD sensors



- **Basic idea of multi-exposure techniques**
  - Combine multiple images with different exposure settings
  - Makes use of available sequential dynamic range
- **Other techniques available**
  - E.g., HDR video

Exposure Bracketing

- **Acquiring HDR from LDR input devices**
  – Take multiple photographs with different times of exposure

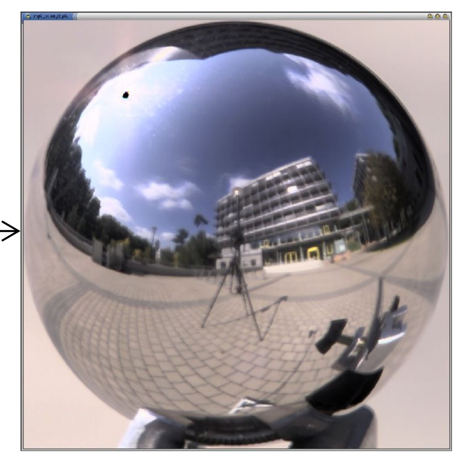
- **Issues**
  – How many exposure levels?
  – How much difference between exposures?
  – How to combine them?

Application

- **Capture HDR env. maps from series of input images**

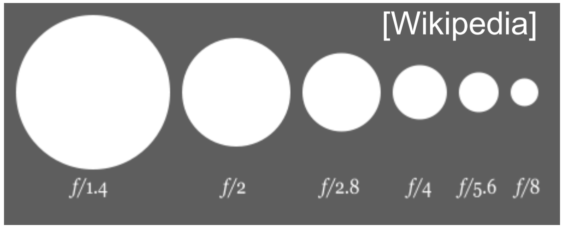
1/2,000s 1/500s 1/125s 1/30s 1/8s

- **Used to illuminate virtual scenes with real-world environment**

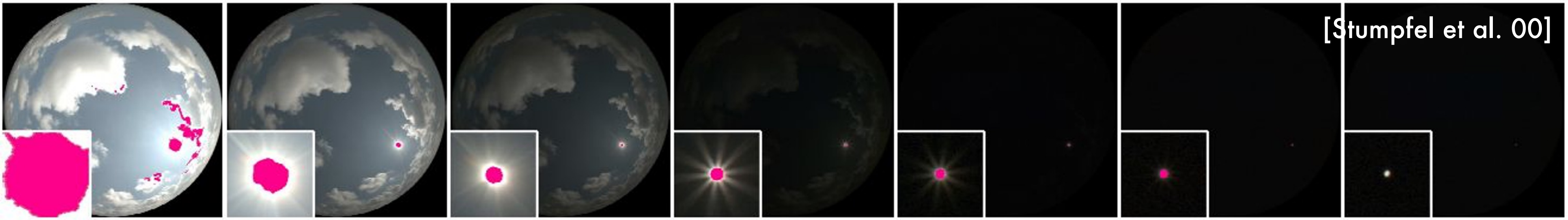


HDR in Real World Images

- **In photography**
  - F-number = focal length / aperture diameter
  - 1 f-stop incr.: f-# * √2 → aperture area / 2
- **Natural scenes**
  - 37 stops (~10 orders of magnitude)
  - 18 stops (2^{18} = ~262 000) at given time of day
- **Humans**
  - After adaptation: 30 stops (~9 orders of magnitude)
  - Simultaneously: 17 stops (~4 orders of magnitude)
- **Analog cameras**
  - 10-16 stops (~3 orders of magnitude)
  - Fish-eye pix of sky with diff. exposures show saturation (e.g., sun)



Doubling the f-number decreases the aperture area by a factor of four (i.e., need to quadruple exposure time to preserve same brightness)



f/4, 1s | f/4, $\frac{1}{4}$s | f/4, $\frac{1}{32}$s | f/16, $\frac{1}{16}$s | f/16, $\frac{1}{125}$s | f/16, $\frac{1}{1000}$s | f/16, $\frac{1}{8000}$s

Dynamic Range of Cameras

- **E.g. photographic camera with standard CCD sensor**

– Dynamic range of sensor	1:1,000
– Exposure time (handheld cam.): 1/60s – 1/6,000s	1:100
– Varying aperture: f/2.0 – f/22.0	1:100 (appro.)
– Electronic: exposure bias / varying “sensitivity”	1:10
– Total (sequential) dynamic range	1:100,000,000

- **But simultaneous dynamic range still only 1:1,000**
  - Aperture: varying depth of field
  - Exposure time: only works for static scenes

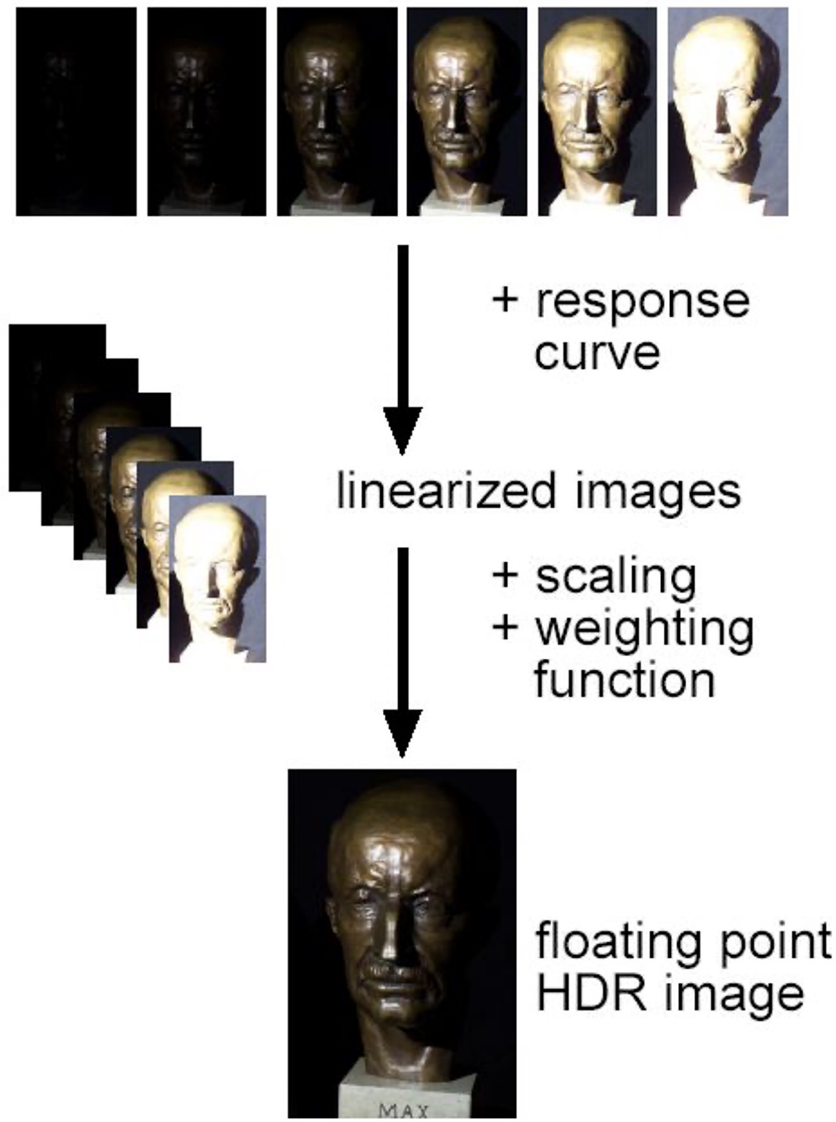
- **Similar situation for analog cameras**
  - Chemical development of film instead of electronic processing

  $\rightarrow$ Allows for varying sensitivity

Multi-Exposure Techniques

- **Analog film**
  - Several emulsions of different sensitivity levels [Wyckoff 1960s]
    - Dynamic range of about 10^8
- **Digital domain**
  - Similar approaches for digital photography
  - Commonly used method [Debevec et al. 97]
    - Select a small number of pixels from all images
    - Perform optimization of response curve with smoothness constraint
  - Newer method by [Robertson et al. 99]
    - Optimization over all pixels in all images
- **General idea of HDR imaging**
  - Combine multiple images with different exposure times
    - Pick for each pixel a well-exposed image
    - Response curve needs to be known to calibrate values betw. images
    - Change only exposure time, not aperture due to diff. depth-of-field !!

Multi-Exposure Techniques



HDR Imaging [Robertson et al. 99]

- **Principle of the approach**
  - Calculate an HDR image using the given response curve
  - Optimize response curve to better match resulting HDR image
  - Iterate till convergence: approx non-linear process w/ linear steps
- **Input**
  - Series of images *i* with exposure times t_i and pixels *j*
  - Response curve *f* applied to incident energy yields pixel values y_{ij}

$$y_{ij} = f\left(I_{y_{ij}}\right) = f\left(t_i x_j\right)$$

- **Task**
  - Recover response curve: $f^{-1}\left(y_{ij}\right) = I_{y_{ij}}$
  - Determine irradiance x_j at pixel *j* from energies $I_{y_{ij}}$:

$$x_j = I_{y_{ij}} / t_i$$

HDR Imaging [Robertson et al. 99]

- **Calculate estimates of HDR input values x_j from images via maximum-likelihood approach**

$$x_j = \frac{\sum_i w_{ij} t_i^2 x_{ij}}{\sum_i w_{ij} t_i^2} = \frac{\sum_i w_{ij} t_i I_{y_{ij}}}{\sum_i w_{ij} t_i^2}$$

- **Use a bell-shaped weighting function $w_{ij} = w(y_{ij})$**
  - Do not trust as much pixel values at extremes
    - Under-exposed: high relative error prone to noise
    - Over-exposed: saturated value
- **Use an initial camera response curve**
  - Simple assumption: linear response

HDR Imaging [Robertson et al. 99]

- **Optimizing the response curve $I(y_{ij})$**
  - Minimization of objective function O (sum of weighted errors)

$$O = \sum_{i,j} w_{ij} \left(I_{y_{ij}} - t_i x_j \right)^2$$

  - Using standard Gauss-Seidel relaxation yields

$$I_m = \frac{1}{Card(E_m)} \sum_{i,j \in E_m} t_i x_j$$

$$E_m = \{(i,j): y_{ij} = m\}$$

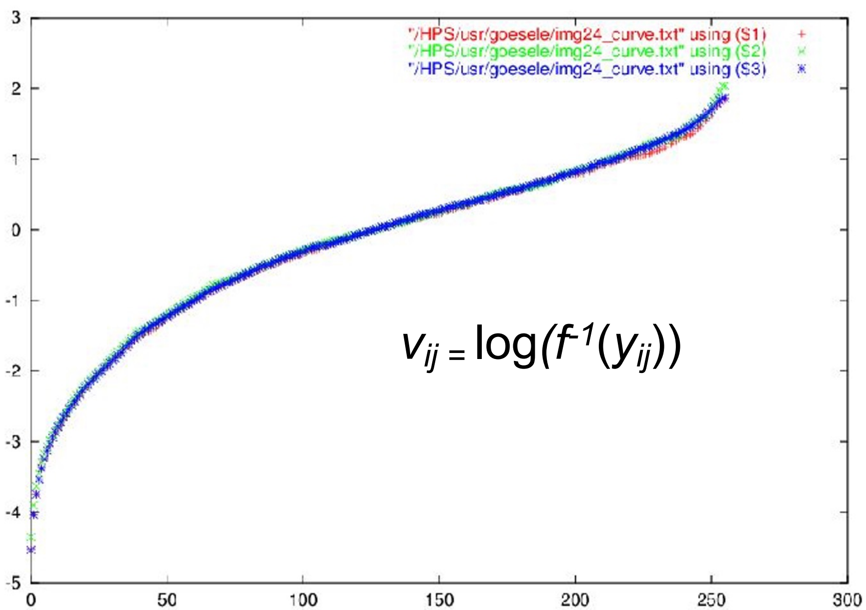
  - Normalization of I so that $I_{128} = 1$

HDR Imaging [Robertson et al. 99]

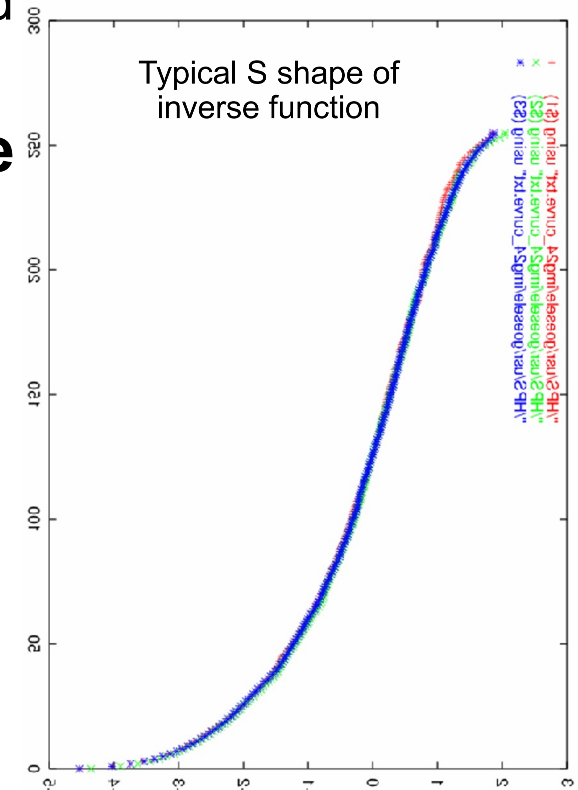
- **Both steps ...**
  - Calculation of an HDR image using *I*
  - Optimization of *I* using the HDR image

  … are now iterated until convergence
  - Criterion: decrease of *O* below some threshold
    - Usually about 5 iterations are enough
- **Logarithmic plot of the response curve**

$$y_{ij} = f(\exp(v_{ij}))$$



$$I_{ij} = \exp(v_{ij})$$
$$v_{ij} = \log(I_{ij})$$



Choice of Weighting Function

- ***w*(*y*$_{ij}$) for response [Robertson et al. 99]**

$$\mathrm{w}_{ij} = \exp\left(-4\frac{\left(y_{ij} - 127.5\right)^2}{127.5^2}\right)$$

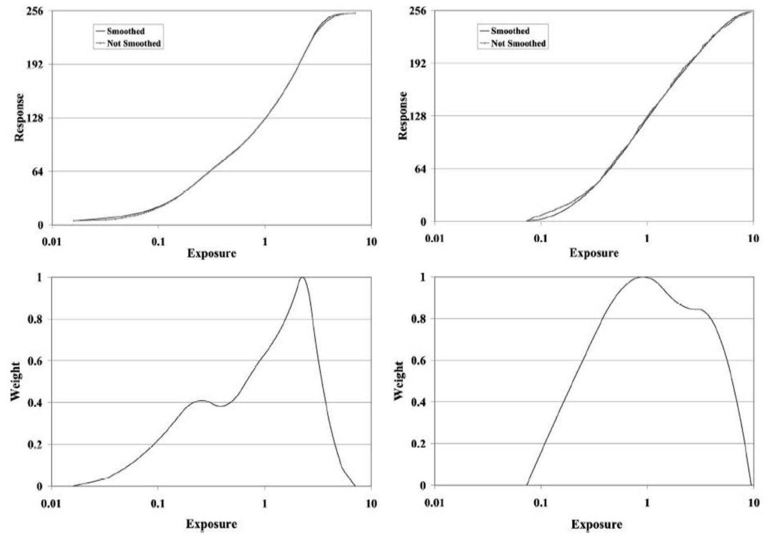
  – Gaussian-like bell-shaped function
  – For 8-bit images, centered around $(2^8 - 1) / 2 = 127.5$
  – Possible width correction at both ends: over/under-exposure
  – Motivated by general noise model: downweight high relative error

- ***w*(*y*$_{ij}$) for HDR reconstruction [Robertson et al. 03]**
  – Introduce certainty function *c* as derivative of response curve with logarithmic exposure axis: S-shape response→bell-shaped curve
  – Approxim. response curve with cubic spline to compute derivative

$$w_{ij} = w(y_{ij}) = c\left(I_{y_{ij}}\right)$$

Weighting Function

- **Consider response curve gradient**
  - Higher weight where response curve maps to large extent



[Robertson et al. 2003]

- **Difference between exposures levels**
  - Ideally such that respective trusted regions (central part of weighting function) are roughly adjacent

HDR Generation

- **What difference to pick between exposures levels?**
  - Most often a difference of 2 stops (factor of 4) between exposures is sufficient
  - See [Grossberg & Nayar 2003] for more details
- **How many input images are necessary to get good results?**
  - Depends on dynamic range of scene illumination and on quality requirements
  - Often 3 images are fine (normal + lower & higher exposure)

Algorithm of Robertson et al.

- **Discussion**
  - Method is very easy
  - Doesn’t make assumptions about response curve shape
  - Converges quickly
  - Takes all available input data into account
    - As opposed to [Debevec et al. 97]
  - Can be extended to > 8-bit color depth
    - 16 bits should be followed by smoothing
    - Quantization to 8 bits eliminates large amount of noise
    - Higher precision with 16 bits more likely to still contain notable noise

Part II

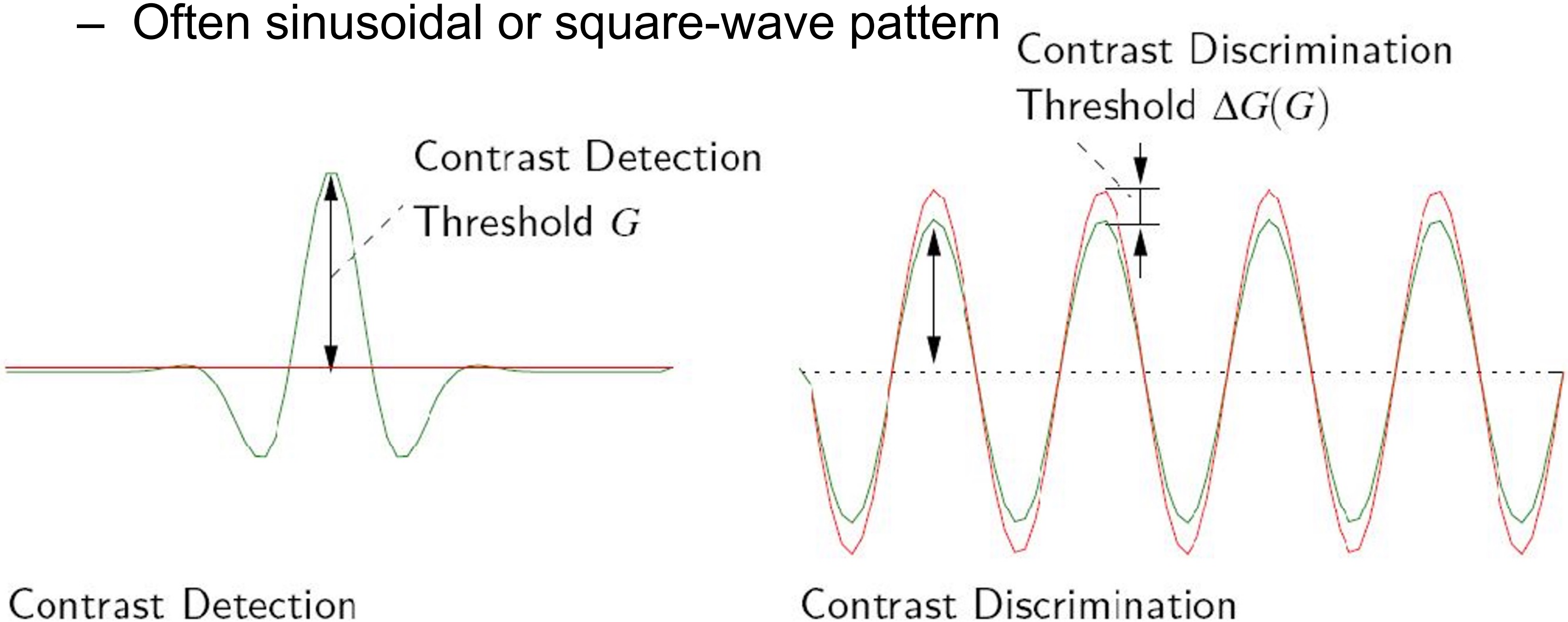
Tone Mapping

Terms and Definitions

- **Dynamic range**
  - Factor between the highest and the smallest representable value
  - 2 strategies to increase dynamic range:
    - Make white brighter, or make black darker (more practical)
    - Reason for trend towards reflective rather than diffuse displays
- **Contrast**
  - Simple contrast: $\mathrm{C_S} = \frac{L_{max}}{L_{min}}$
  - Weber fraction: $\mathrm{C_W} = \frac{\Delta L}{L_{min}}$ with $\Delta L = L_{max} - L_{min}$
  - Michelson contrast: $\mathrm{C_M} = \frac{|\, L_{max} - L_{min}\,|}{L_{max} + L_{min}}$
  - Logarithmic ratio: $\mathrm{C_L} = \log_{10}\left(\frac{L_{max}}{L_{min}}\right)$
  - Signal to noise ratio (SNR): $\mathrm{C}_{SNR} = 20 \cdot \log_{10}\left(\frac{L_{max}}{L_{min}}\right)$

Contrast Measurement

- **Contrast detection threshold**
  - Smallest detectable intensity difference in a uniform field of view
  - E.g., Weber-Fechner perceptual experiments
- **Contrast discrimination threshold**
  - Smallest visible difference between two similar signals
  - Works in supra-detection-threshold domain (i.e., signals above it)
  - Often sinusoidal or square-wave pattern



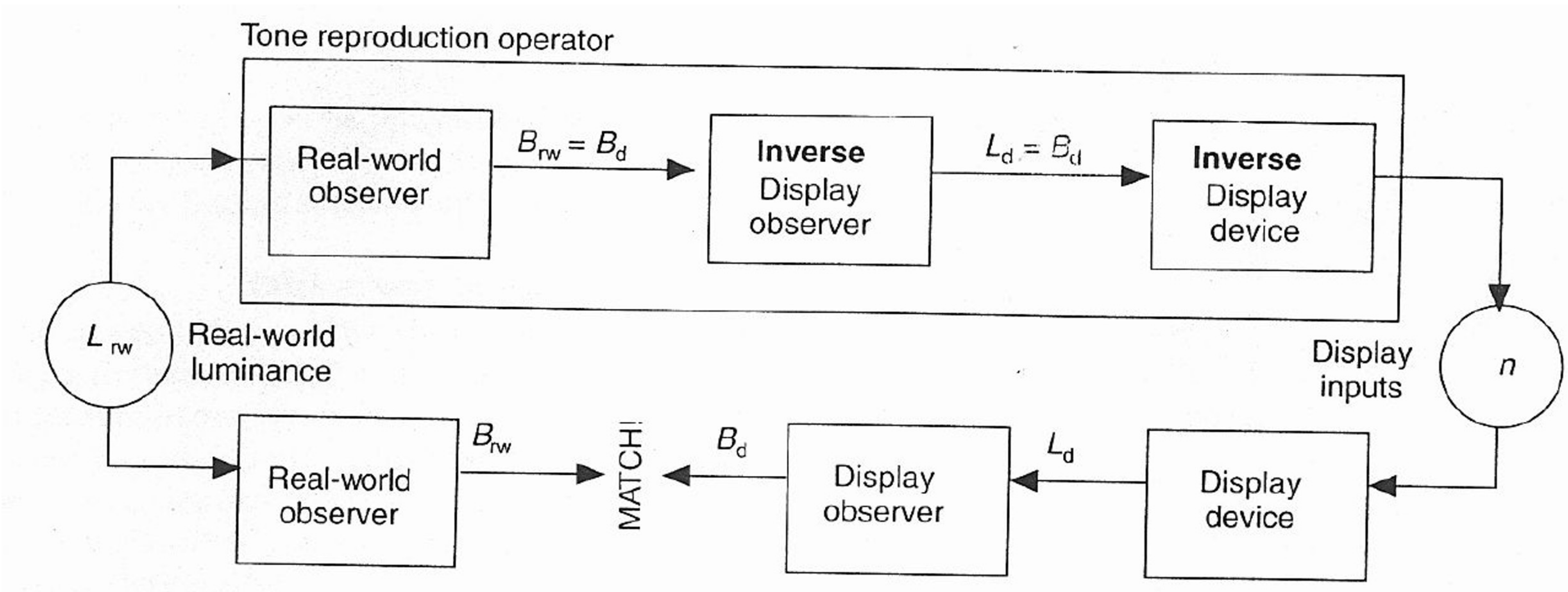
Why Tone Mapping?

- **Mapping HDR radiance values to LDR pixel values?**
  - Luminance range for human visual perception
    - Min 10^{-5} cd/m^2: shadows under starlight
    - Max 10^{5} cd/m^2 : snow in direct sunlight
  - Luminance of typical desktop displays
    - Typically, up to ~500 cd/m^2 : about 2-3 orders of magnitude

- **Goal**
  - Compress the dynamic range of an input image to fit output range
  - Reproduce HVS to closely match perception of the real scene
    - Brightness and contrast
    - Adaptation of the eye to environment
    - Bright/dark input: glare, color perception, loss of visual acuity, …

General Principle

- **Original approach [Tumblin/Rushmeier]**
  - Create model of the observer
  - Requirement: observer looking at displayed virtual image should perceive the same brightness as when staring at the real scene
  - Compute tone-mapping as concatenation/inversion of operators
  - Model usually operates only on luminance (not on color)



- **Other models aim for visually pleasing images**

Heuristic Approaches

- **Linearly scale brightest value to 1 (in gray value)**
  - Problem: light sources are often several orders of magnitude brighter than the rest → the rest will be black
- **Linearly scale brightest non-light-source value**
  - Capping light source values to 1
  - Scale the rest to a value slightly below 1
  - Problem: bright reflections of light sources

- **General problem of simple linear scaling**
  - Absolute brightness gets lost
  - Scaling of light source intensity gets factored out → has no effect
- **Much better: linear scaling in the logarithmic domain**
  - Linear scaling of perceived brightness instead of input luminance
  - Much closer to human perception
  - Typically using $\log_{10}$

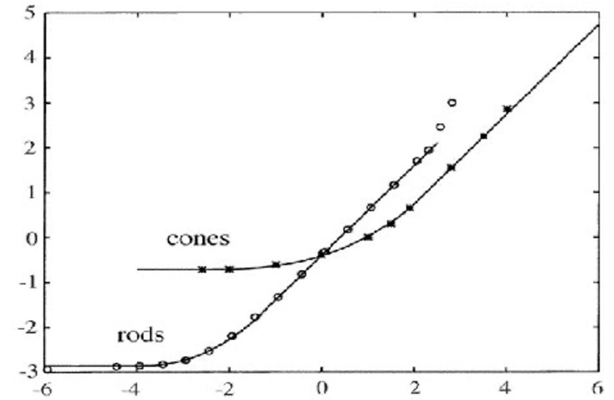
Maintaining Contrast

- **Contrast-based linear scaling factor [Ward 94]**
  - Make just visible differences in real world just visible on display
    - Preserve the visibility in the scene based on Weber's contrast
  - Just noticeable contrast differences according to Blackwell [CIE 81] (subjective measurements)

$$\Delta L(L_a) = 0.0594(1.219 + L_a^{0.4})^{2.5}$$

    - Minimum discernible difference in luminance for given visual adaptation level L_a

Threshold ΔL [log cd/m²]



Adaptation luminance L_a [log cd/m²]

  - Goal: proportionality constant *m*
    - Relates world luminance values L_w to display luminance values L_d
    - $L_d = m\ L_w$

Maintaining Contrast

- **Approach using "just noticeable difference" (JND)**
  - Find m such that JND $\Delta L(L_{wa})$ at world adaptation luminance L_{wa} and JND $\Delta L(L_{da})$ at display adaptation luminance L_{da} verify

$$\Delta L(L_{da}) = m(L_{wa})\Delta L(L_{wa})$$

  - Substitution results in

$$m(L_{wa}) = \left[\frac{1.219 + L_{da}^{0.4}}{1.219 + L_{wa}^{0.4}}\right]^{2.5}$$

  - Compute L_{da} from maximum display luminance: $L_{da} = L_{dmax} / 2$
  - Normalize scaling factor *sf* in [0, 1]

$$sf = \frac{1}{L_{dmax}}\left[\frac{1.219 + (L_{dmax}/2)^{0.4}}{1.219 + L_{wa}^{0.4}}\right]^{2.5}$$

Maintaining Contrast

- **Deriving the real-world adaptation L_{wa}**
  - Depends on light distribution in field of view of observer
  - Simple approximation using a single value
    - Eyes try to adjust to average incoming brightness
    - Brightness *B* based on input luminances:
      - $B = k\, L_{in}^{a}$: Power-law [Stevens 61]
    - Comfortable brightness based on average of input luminances:
      - $\log_{10}(L_{wa}) = E\{\log_{10}(L_{in})\} + 0.84 \Rightarrow L_{wa} = 10\string^(\sum_n \log_{10}(L_{in}) / n)$
- **Problems of this approach**
  - Single factor for entire image
    - Does not handle different adaptation for different locations in image
    - We do not perceive absolute differences in luminance: neighborhood
  - Brightness adaptation mainly acts on 1° field of view of fovea rather than periphery $\rightarrow$ would require eye tracking
  - Adaptation to average results in clamping for too bright regions

Histogram Adjustment

- **Optimal mapping of the dynamic range [Ward 97]**
  - Compute an adjustment image
    - Assume known viewpoint with respect to the scene
    - Blur input image with distance-dependent kernel
      - Filter (average) non-overlapping regions covering 1° field of view, i.e., foveal solid angle of adaptation
      - Reference uses simple box filter
    - Reduce resolution
  - Compute the histogram of the image
    - Bin the luminance values
  - Adjust the histogram based on restrictions of HVS
    - Limit contrast enhancement

⇒ Distributes contrast in the image in a visually meaningful way, but does not try to model human vision per se as outlined by [Tumblin/Rushmeier]

Histogram Adjustment

- **Definitions**
  - $B_w = \log(L_w)$: compute world brightness from world luminance
  - b_i : create N bins i corresponding to ranges of B_w
  - $f(b_i)$: number of B_w samples in bin b_i: $\propto$ PDF
  - $P(b)=\Sigma f(b_i)/T$: normalized sum of $f(b_i)$ for $b_i < b$: CDF ($\int$ of PDF)
  - T : sum over all $f(b_i)$, i.e., total number of samples

$$T = \sum f(b_i)$$

$$\Delta b = \frac{log(L_{w}max) - log(L_{w}min)}{N}$$

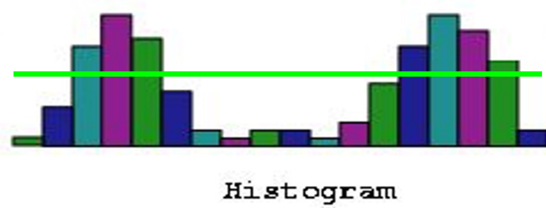
  - Bin step size Δb (in $\log(cd/m^2)$) defined by min/max world luminance for the scene and number of histogram bins N
  - Therefore, the PDF is

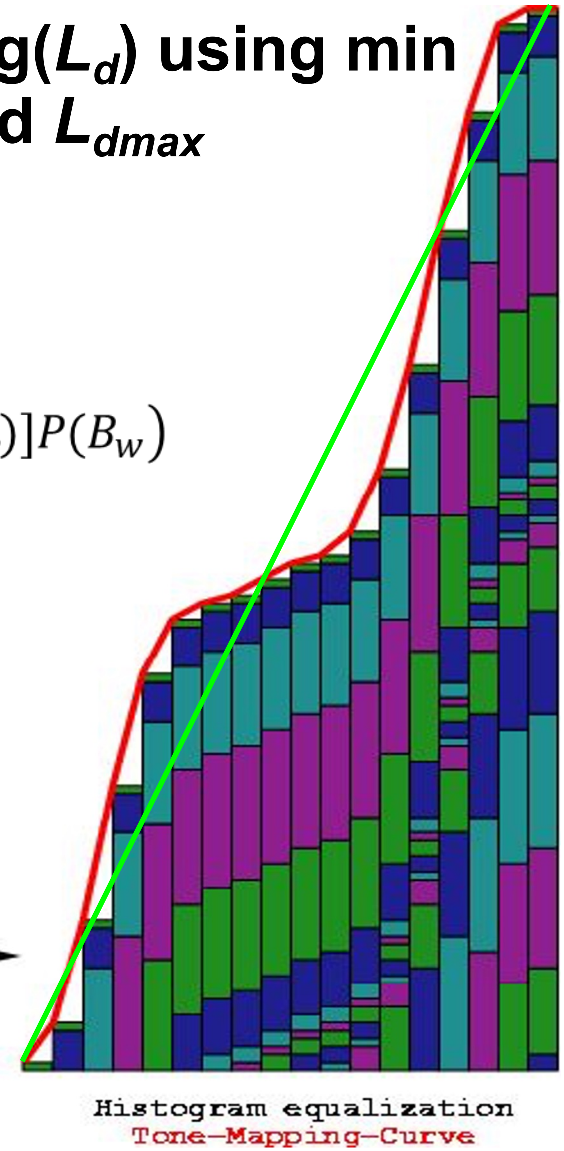
$$dP(b) / db = f(b_i) / (T \Delta b)$$

Naïve Histogram Equalization

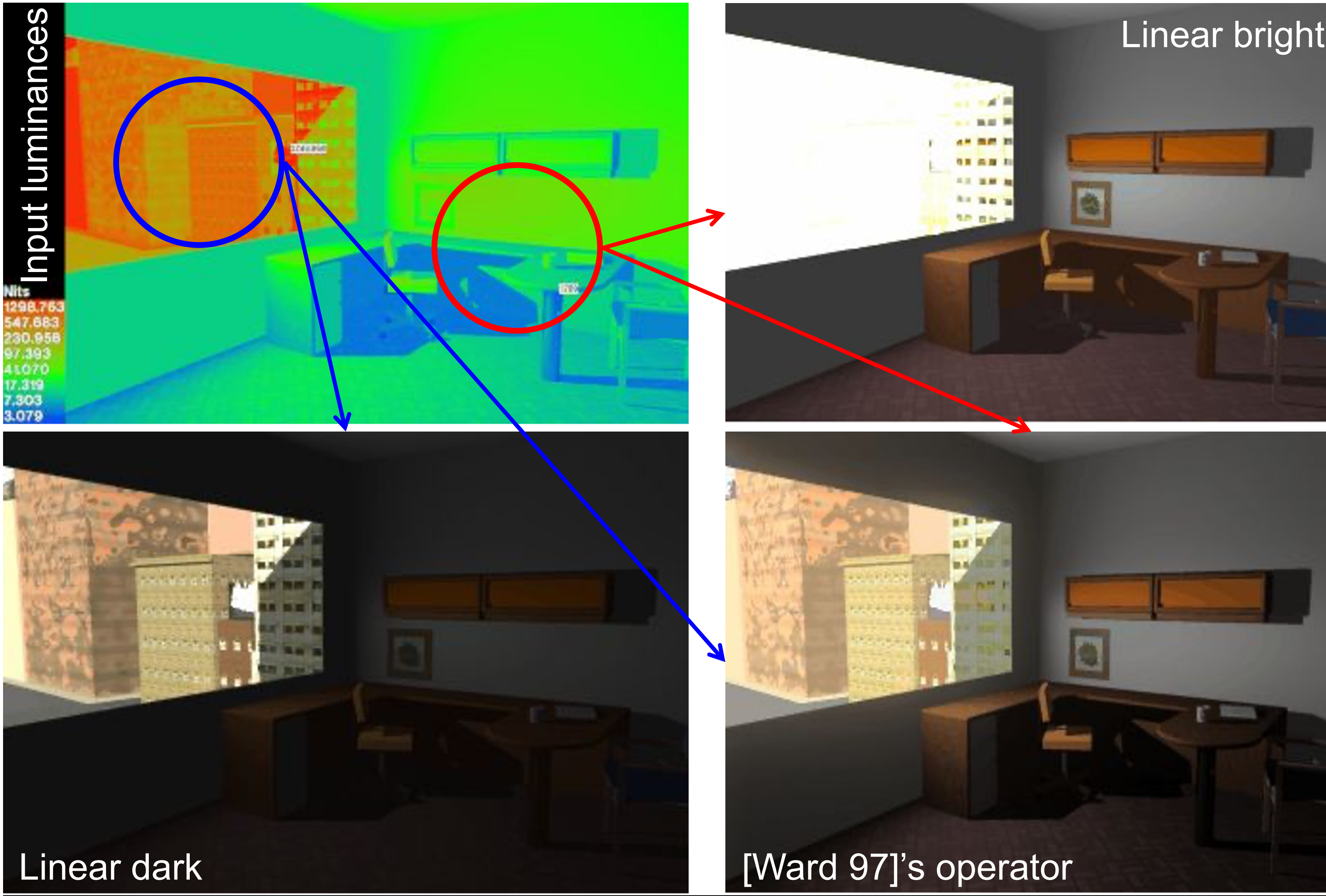
- **Compute display brightness $B_d = \log(L_d)$ using min and max display luminance L_{dmin} and L_{dmax}**

$$B_d = \log(L_{dmin}) + [\log(L_{dmax}) - \log(L_{dmin})]P(B_w)$$



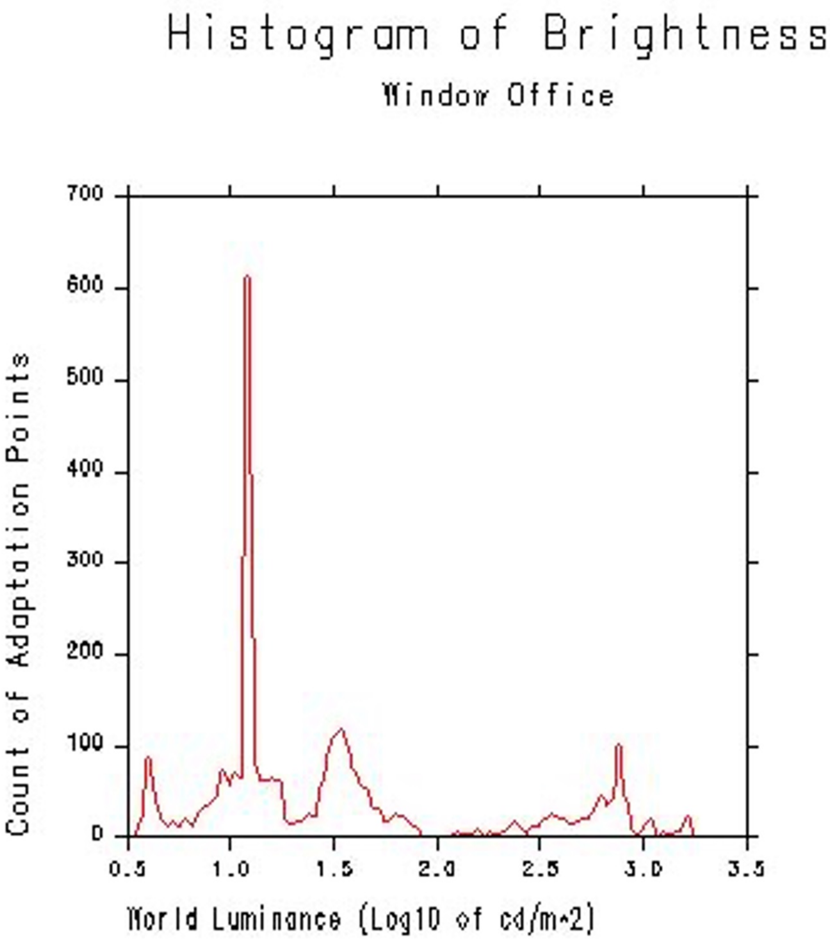


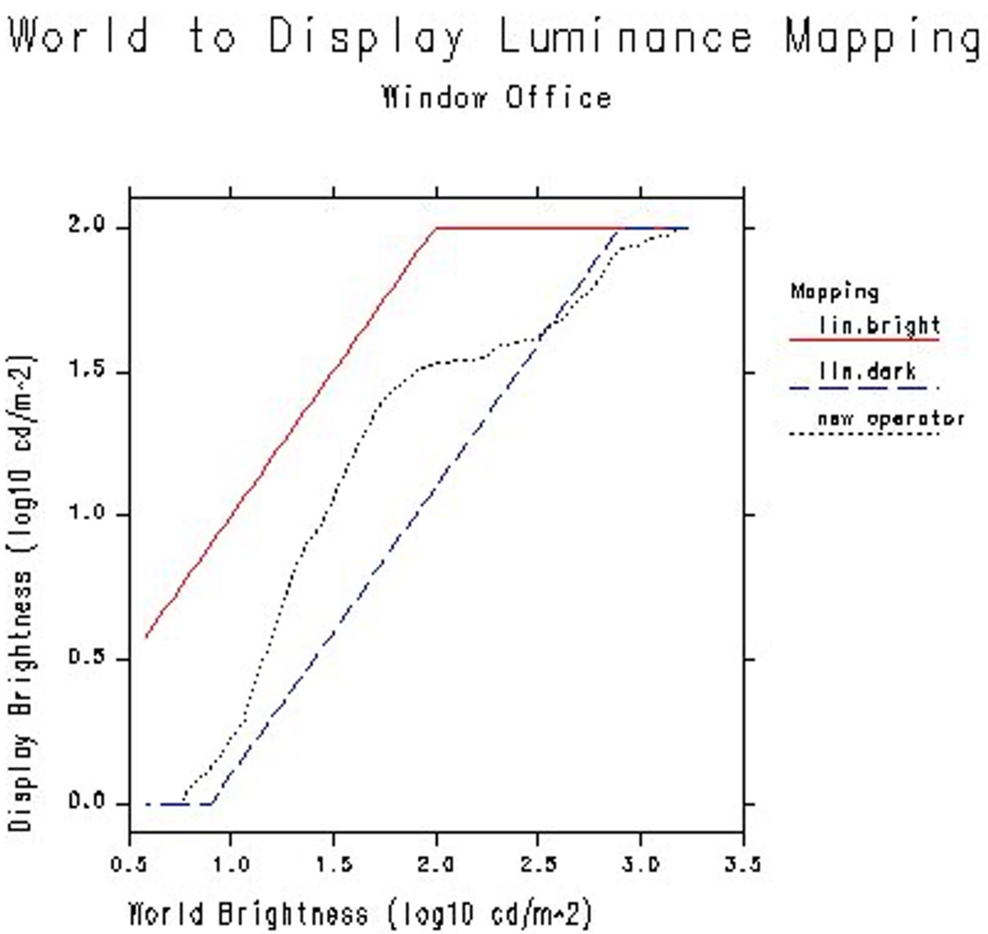
Histogram Adjustment



Histogram Adjustment

- **Linear mapping (scaling) vs. histogram adjustment**





Histogram Adj. w/ Linear Ceiling

- **Problem**
  - Too exaggerated contrast in large highly-populated regions of the dynamic range: enhances features more than the HVS would
- **Idea**
  - Contrast-limited histogram equalization using a linear ceiling (linear scaling works well for low contrast images)

$$\frac{dL_d}{L_d} \leq \frac{dL_w}{L_w} \Rightarrow \frac{dL_d}{dL_w} \leq \frac{L_d}{L_w}$$

  - Differentiate $L_d = \exp(B_d)$ with respect to L_w using the chain rule

$$\frac{dL_d}{dL_w} = \exp(B_d)\frac{f(B_w)}{T\Delta b}\frac{\log(L_{dmax})-\log(L_{dmin})}{L_w} \leq \frac{L_d}{L_w}$$

- **Result**
  - Limiting the sample count per bin in the histogram

$\Leftrightarrow$ limit the magnitude of the PDF, i.e., the slope of the CDF

$$f(B_w) \leq \frac{T\Delta b}{\log(L_{dmax})-\log(L_{dmin})}$$

Histogram Adj. w/ Linear Ceiling

- **Implementing the contrast limitation**
  - Truncate too large bins w/ redistribution to neighbors (repeatedly)
  - Ditto without redistribution (gives better results)
  - Use modified $f(B_w)$ in histogram equalization vs. naïve approach

Histogram Adj. w/ Linear Ceiling

Linear mapping (simple scaling)

Naïve histogram equalization

Histogram adjustment with linear ceiling on contrast

HA based on Hum. Contr. Sensi.

- **Adjustment for JND**
  - Limiting the contrast to the ratio of JNDs (global scale factor)

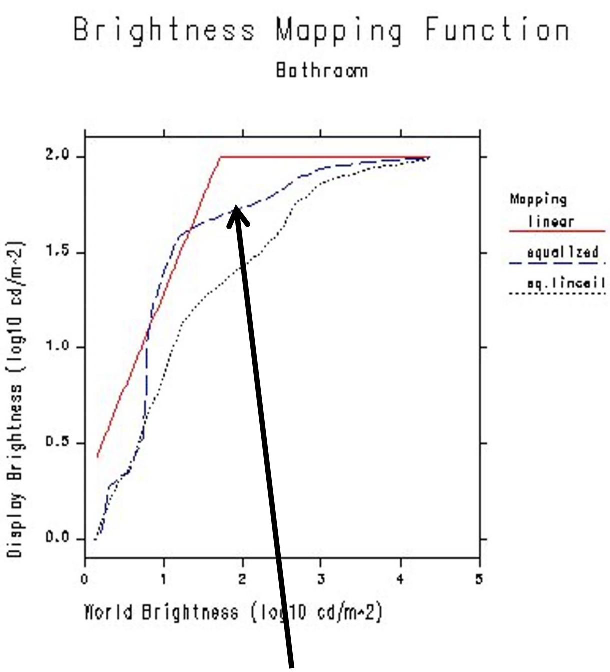
$$\frac{dL_d}{dL_w} \leq \frac{\Delta L_t(L_d)}{\Delta L_t(L_w)}$$

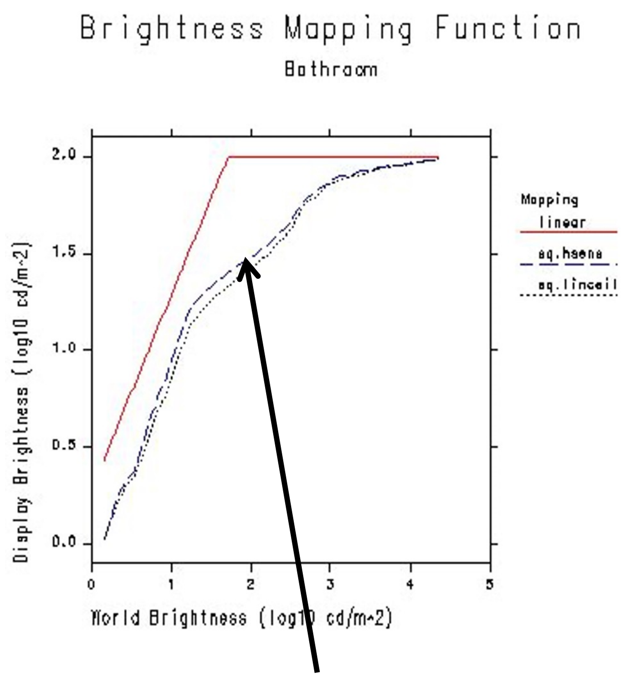
  - That results in

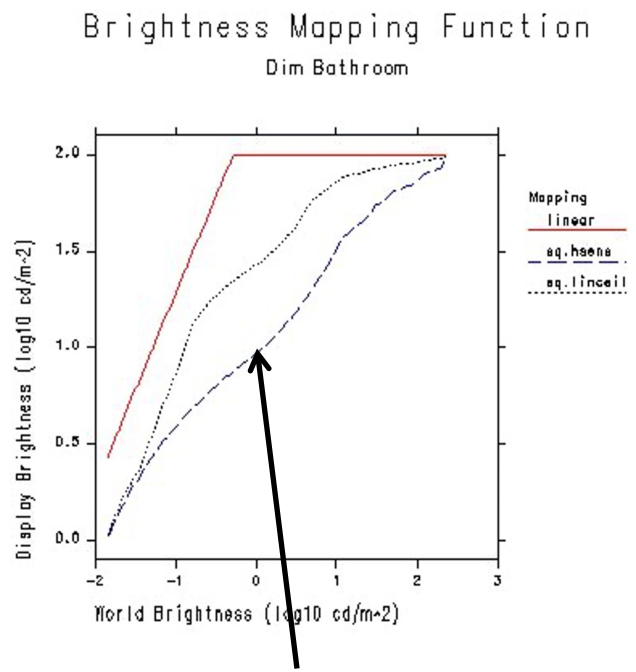
$$f(B_w) \leq \frac{\Delta L_t(L_d)}{\Delta L_t(L_w)} \frac{L_w}{L_d} \frac{T \Delta b}{[\log(L_{d\,max}) - \log(L_{d\,min})]}$$

  - Implementation is similar as for previous histogram equalization

HA based on Hum. Contr. Sensi.







Naïve histogram equalization

HA with human sensitivity in bright bathroom

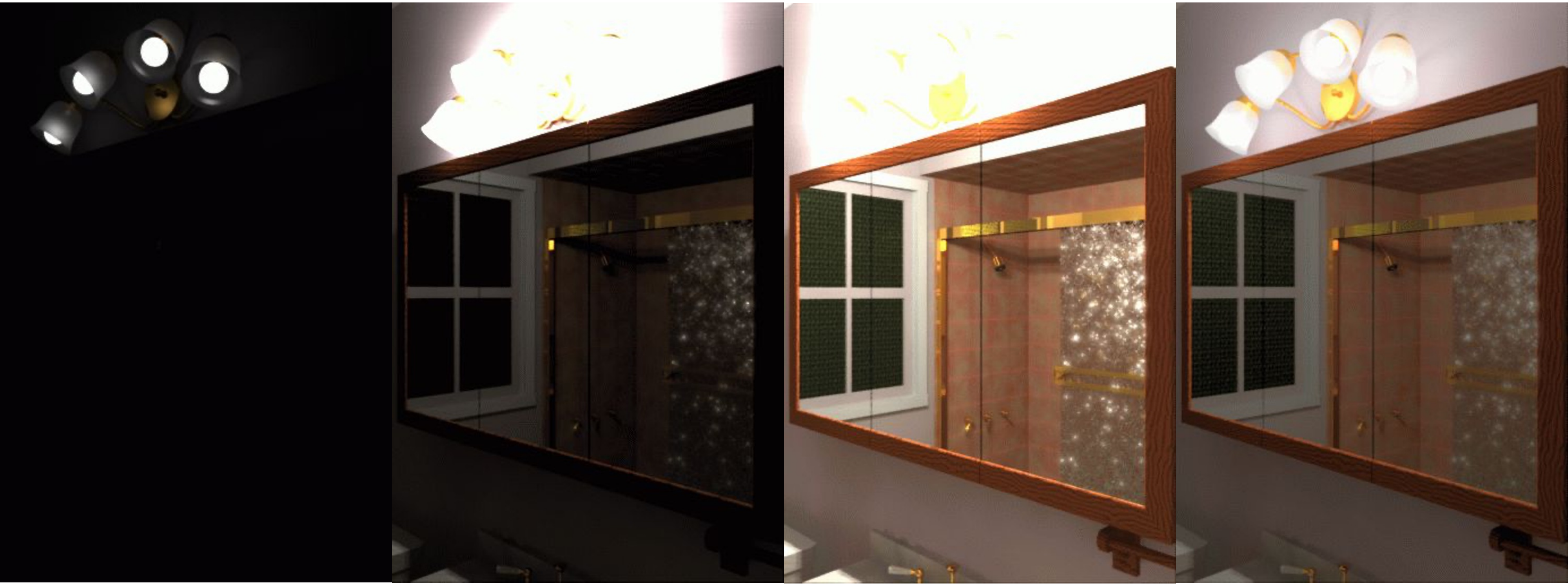
HA with human sensitivity in dim bathroom

HA based on Hum. Contr. Sensi.

- **Reduction of contrast sensitivity in dark scenes**

Comparison

- **[Tumblin/Rushmeier]**
  - Sound methodology from a theoretical standpoint
  - Maybe not optimal models of HVS used in practical experiments

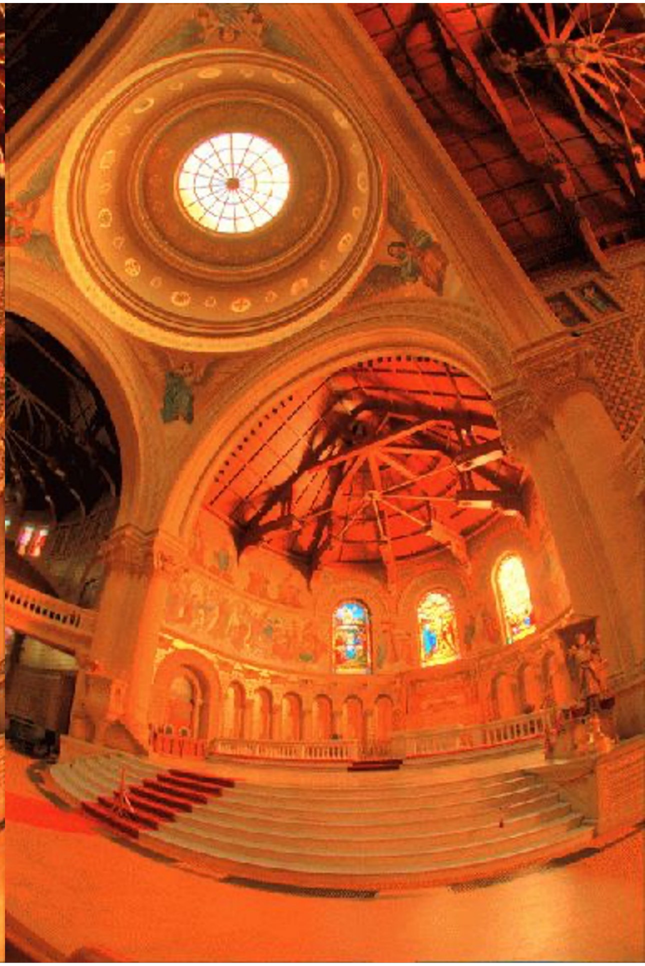


Maximum linear scaling tone mapping | **[Tumblin/Rushmeier] tone mapping** | **Contrast-based lin. scal. [Ward 94] tone mapping** | **Histogram adjustment [Ward 97] tone mapping**

Comparison

[Tumblin/Rushmeier] tone mapping

Contrast-based linear scaling [Ward 94] tone mapping



Histogram adjustment [Ward 97] tone mapping

Local Tone Mapping

- **Usual contrast enhancement techniques**
  - Global tone-map. operator: apply same operation on entire image
  - Either enhance everything or require manual intervention
  - Change image appearance
- **Tone map. often gives numerically optimal solution**
  - No dynamic range left for enhancement
- **Local operators**
  - HVS adapts locally $\Rightarrow$ apply $\neq$ tone-mapping operators in $\neq$ areas

HDR image (reference)

Restore missing contrast

by doing local processing

Tone-mapping result

[Krawczyk 06]

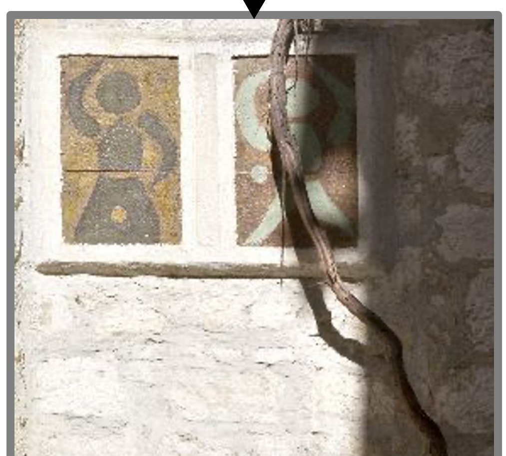
Idea: Enhance Local Contrast

Reference HDR image

Measure lost contrast at several feature scales (preserve small-scale details but adjust overall large-scale contrast)

Tone-mapped image

Enhance lost small-scale contrast in tone-mapped image (best allocation of LDR contrast rather than simulate HVS)



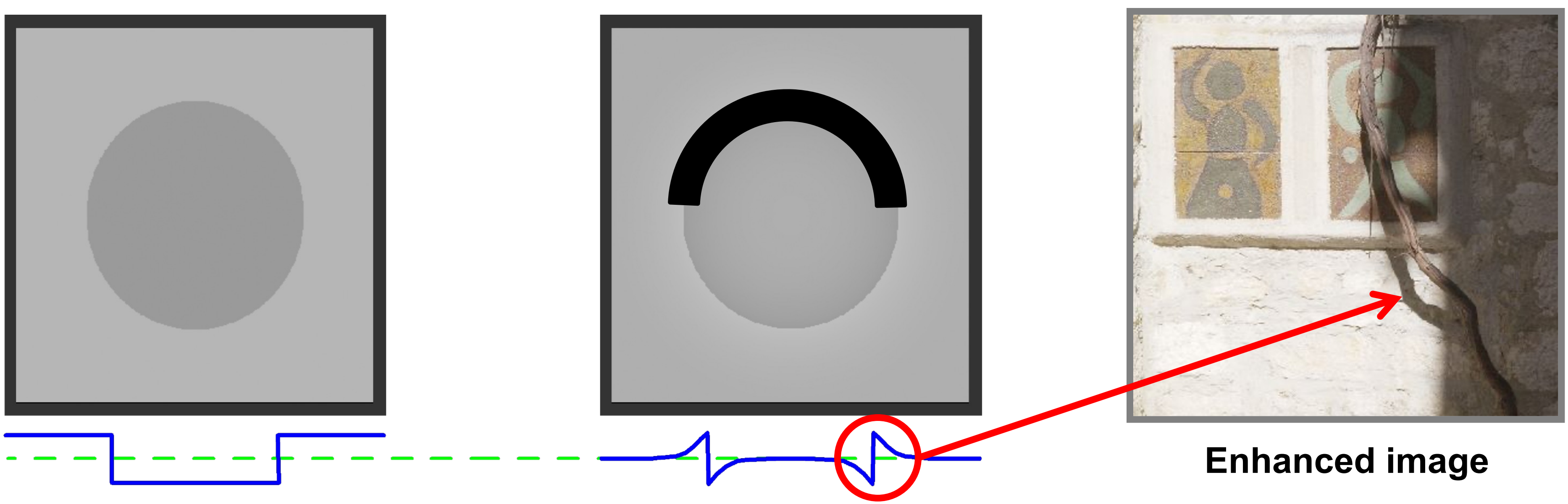
Enhanced tone-mapped image

Communicate lost image contents

Maintain image appearance

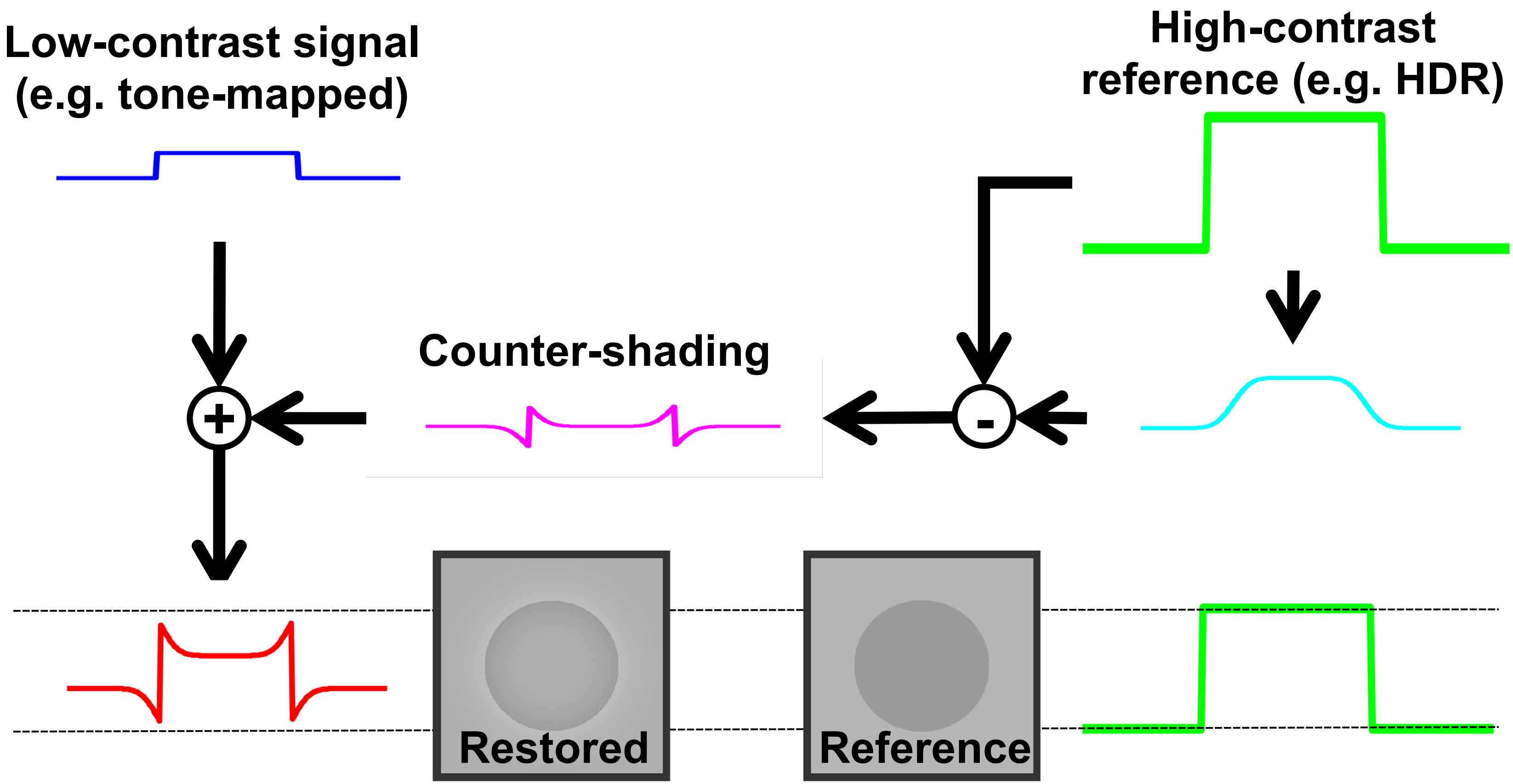
Adaptive Counter-Shading

- **Create apparent contrast based on Cornsweet illusion**
  - Introduce sharp visible edges between similar-brightness regions
- **Countershading**
  - Gradual darkening / brightening towards a contrasting edge
  - Restore contrast of small features with economic use of dyn. range



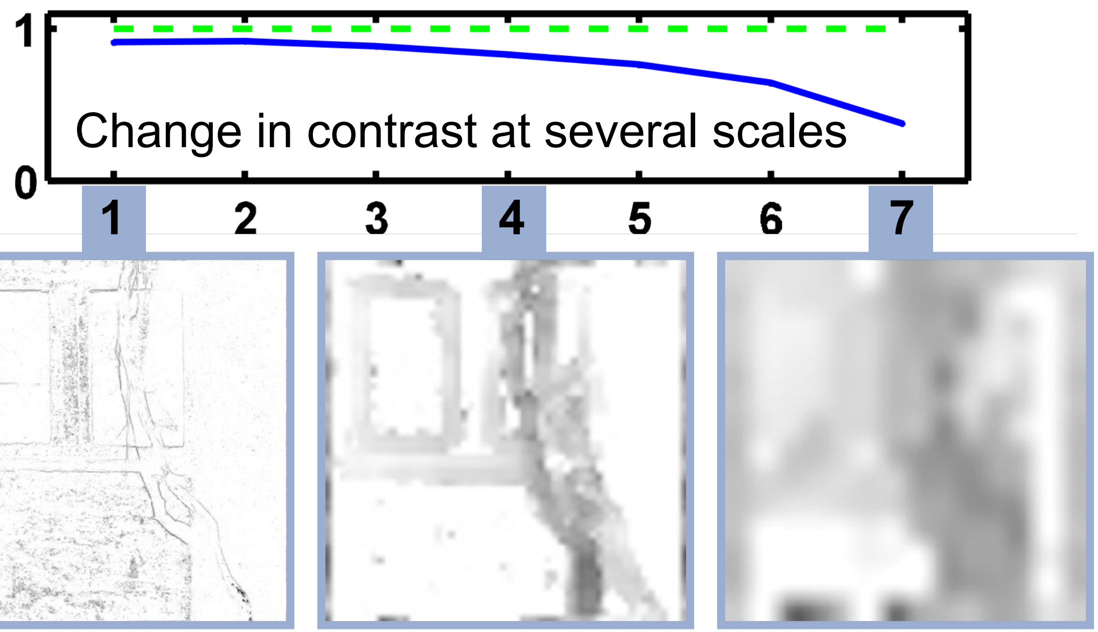
Construction of Simple Profile

- Profile from low-pass filtered reference
- Size and amplitude adjusted manually
- This is unsharp masking



Where to Insert the Profiles?

- **Measure lost contrast at several feature scales**



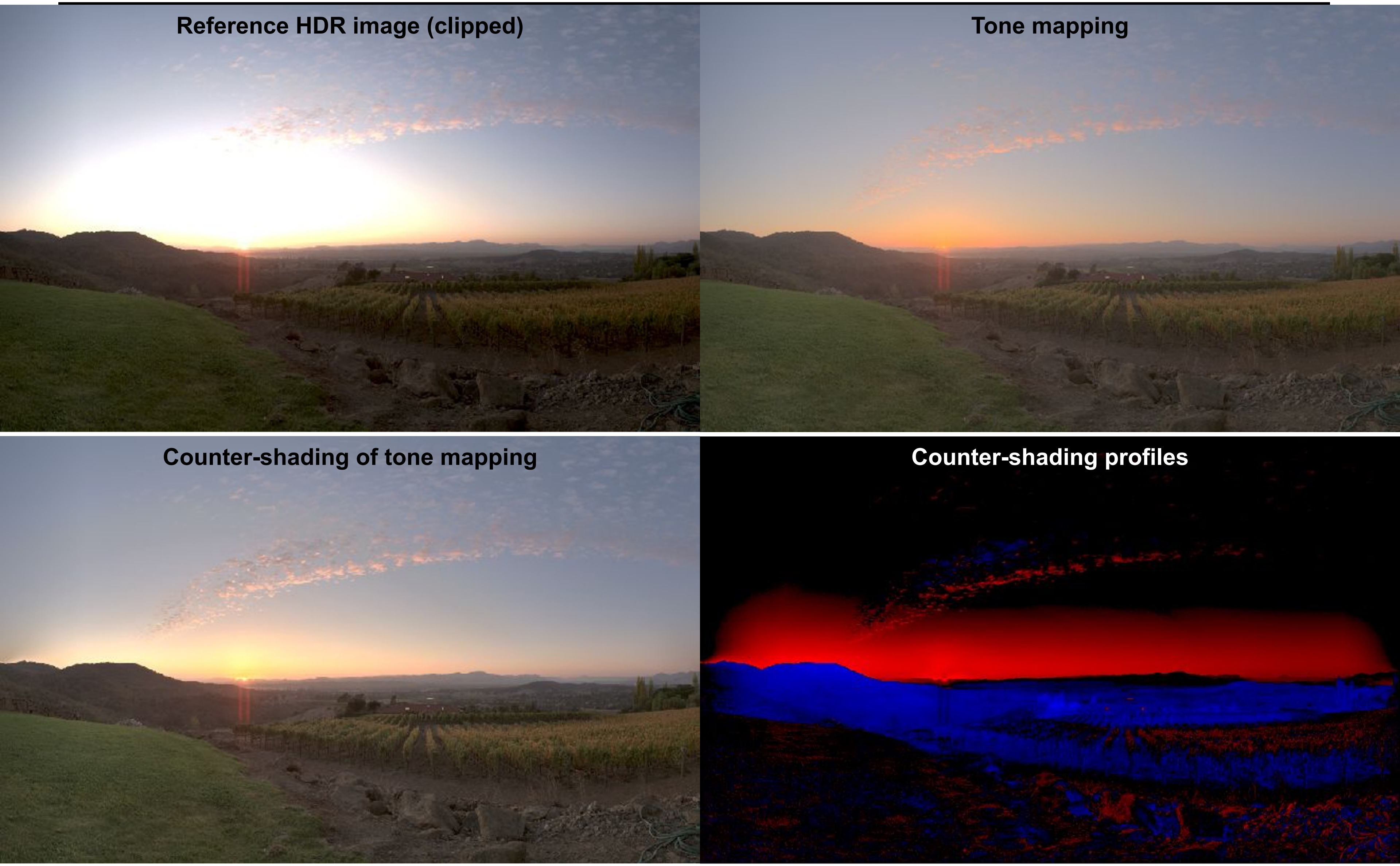
Adaptive Counter-Shading

- **Objectionable visibility of counter-shading profiles**

Progress of restoration

Final contrast restoration

Subtle Correction of Details



Improved Separation