



# **Two-scale Tone Management for Photographic Look**

Soonmin Bae, Sylvain Paris, and Frédo Durand

MIT CSAIL

SIGGRAPH2006

# Ansel Adams

---



Ansel Adams, *Clearing Winter Storm*



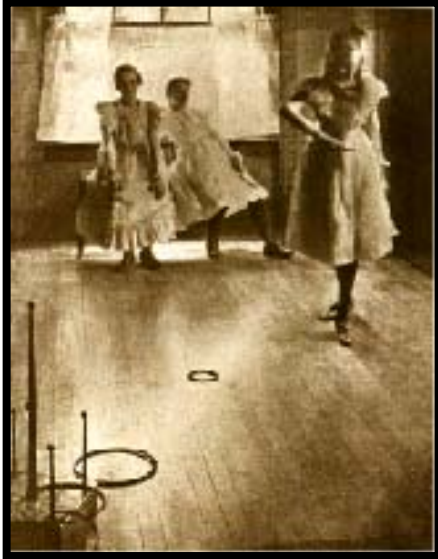
# An Amateur Photographer

---



# A Variety of Looks

---



# Goals

---

- Control over photographic look
- Transfer “look” from a model photo

For example,

we want



with the look of



# Aspects of Photographic Look

---

- Subject choice
  - Framing and composition
  - Specified by input photos
- 
- Tone distribution and contrast
  - Modified based on model photos



Input



Model

# Tonal Aspects of Look

---



Ansel Adams

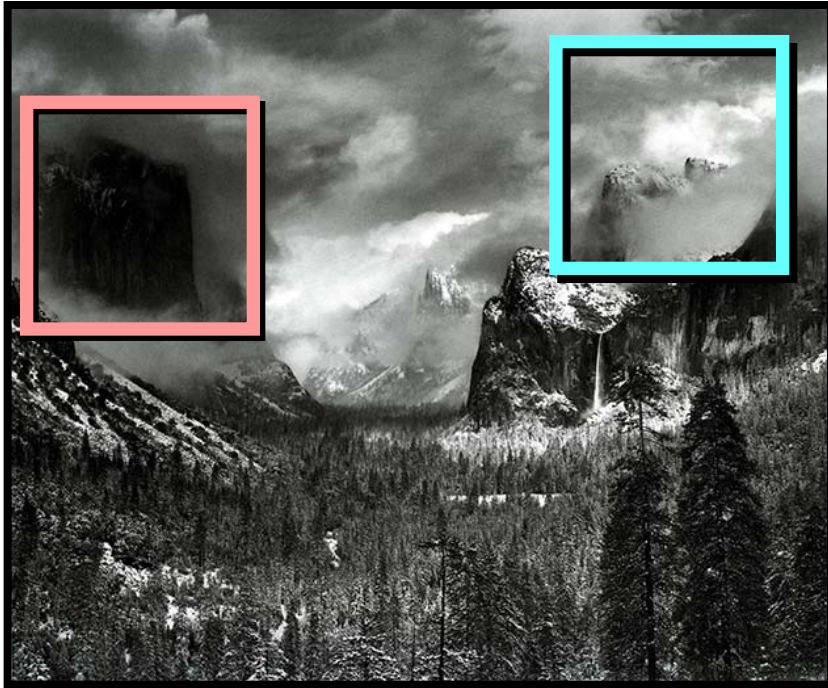


Kenro Izu



# Tonal aspects of Look - Global Contrast

---



Ansel Adams



Kenro Izu

**High Global Contrast**

**Low Global Contrast**

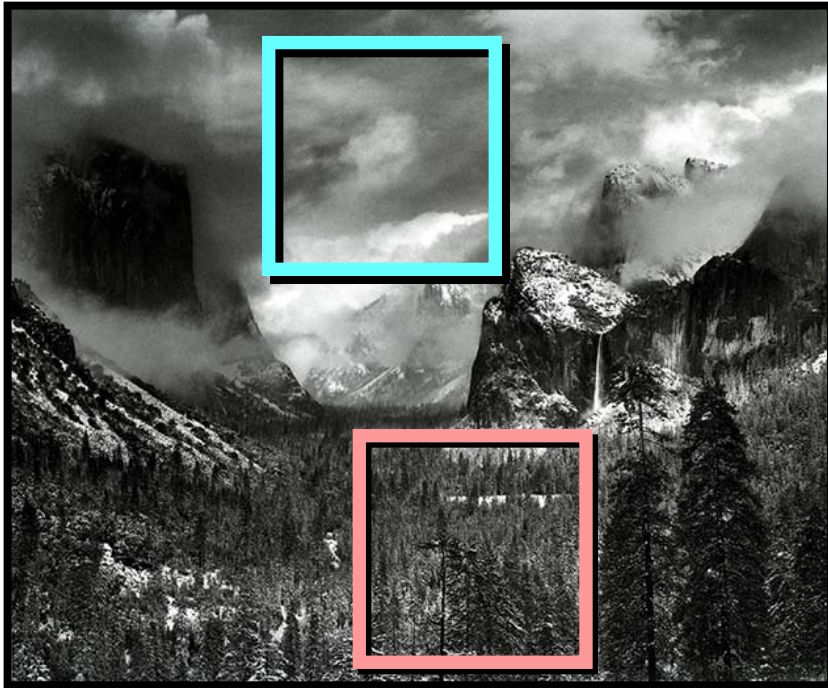




# Tonal aspects of Look

## - Local Contrast

---



Ansel Adams



Kenro Izu

**Variable amount of texture**

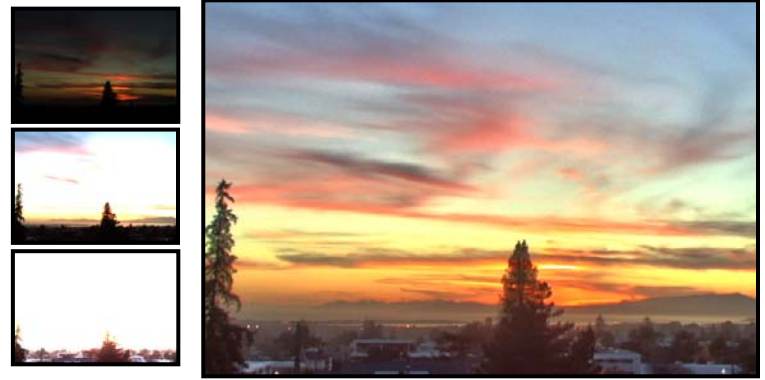
**Texture everywhere**



# Related Work - Tone Mapping

---

- Reduce global contrast  
[Pattanaik 98; Tumblin 99; Ashikhmin 02;  
Durand 02; Fattal 02; Reinhard 02; Li 05]
- Seeks neutral reproduction
  - ✗ Little control over look



[Durand 02]

In contrast,  
we want to achieve particular looks

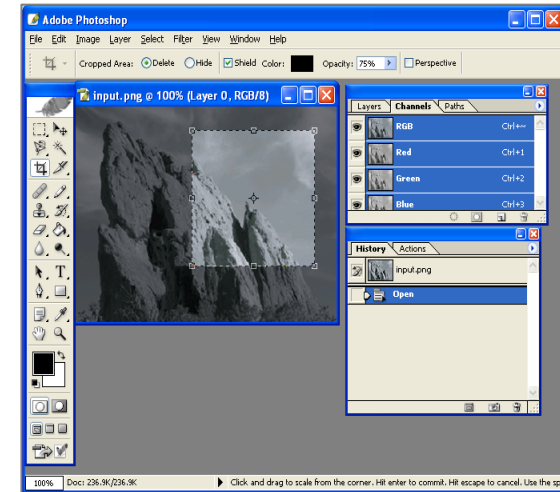


# Related Work – Professional tools

- Image editing software

e.g. Adobe Photoshop

- need skills
- tedious

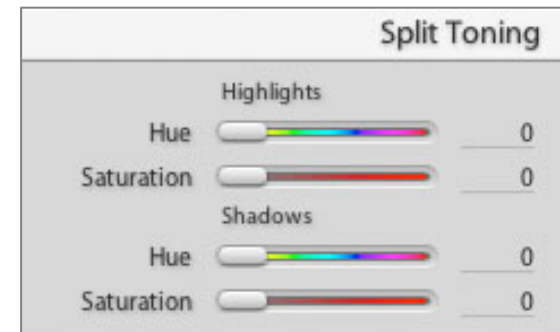


Adobe Photoshop

- Photo management tools

e.g. Adobe Lightroom, Apple Aperture

- optimizes user efficiency (workflow)
- but has limited control



Adobe Lightroom



# Our work



*Input Image*



*Model*



*Result*

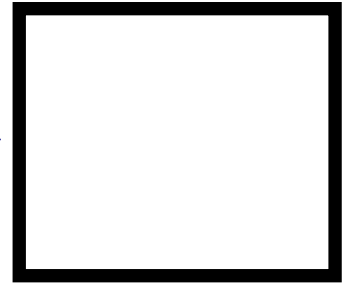
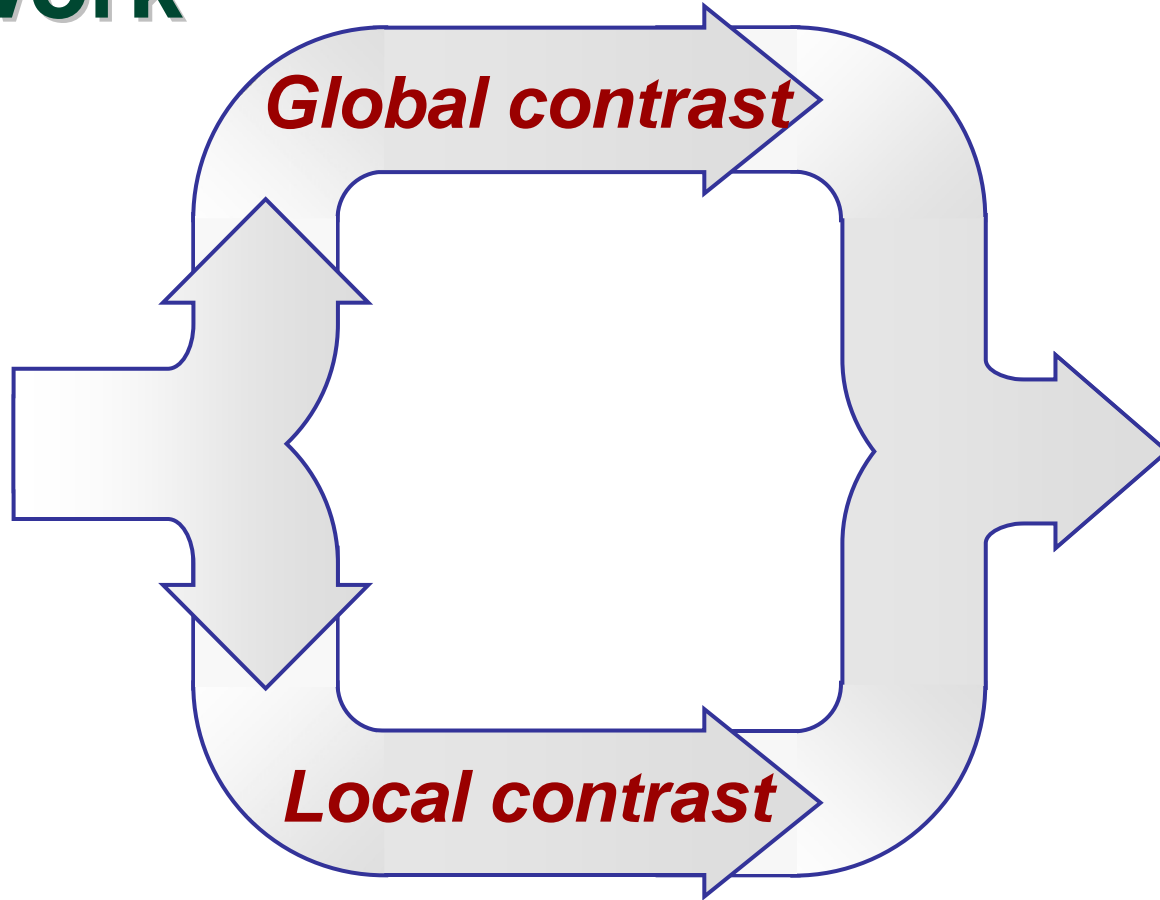
- Transfer look between photographs
  - Tonal aspects



# Our work



*Input  
Image*



*Result*

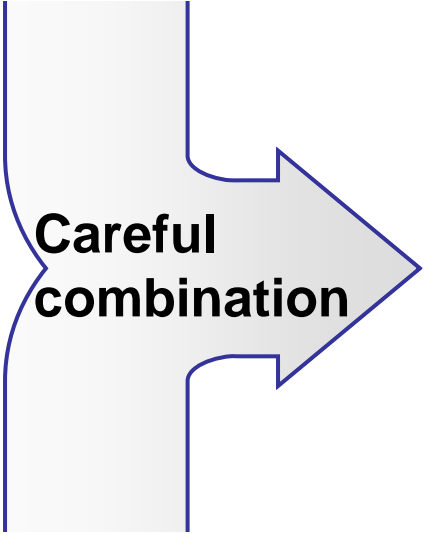
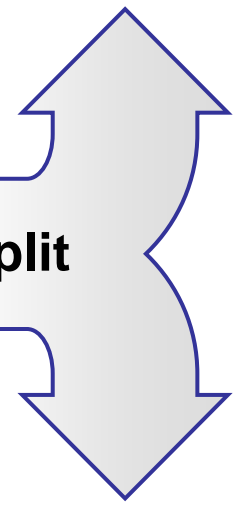
- Separate global and local contrast



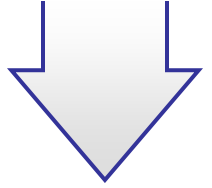
# Overview



*Input Image*



**Post-process**

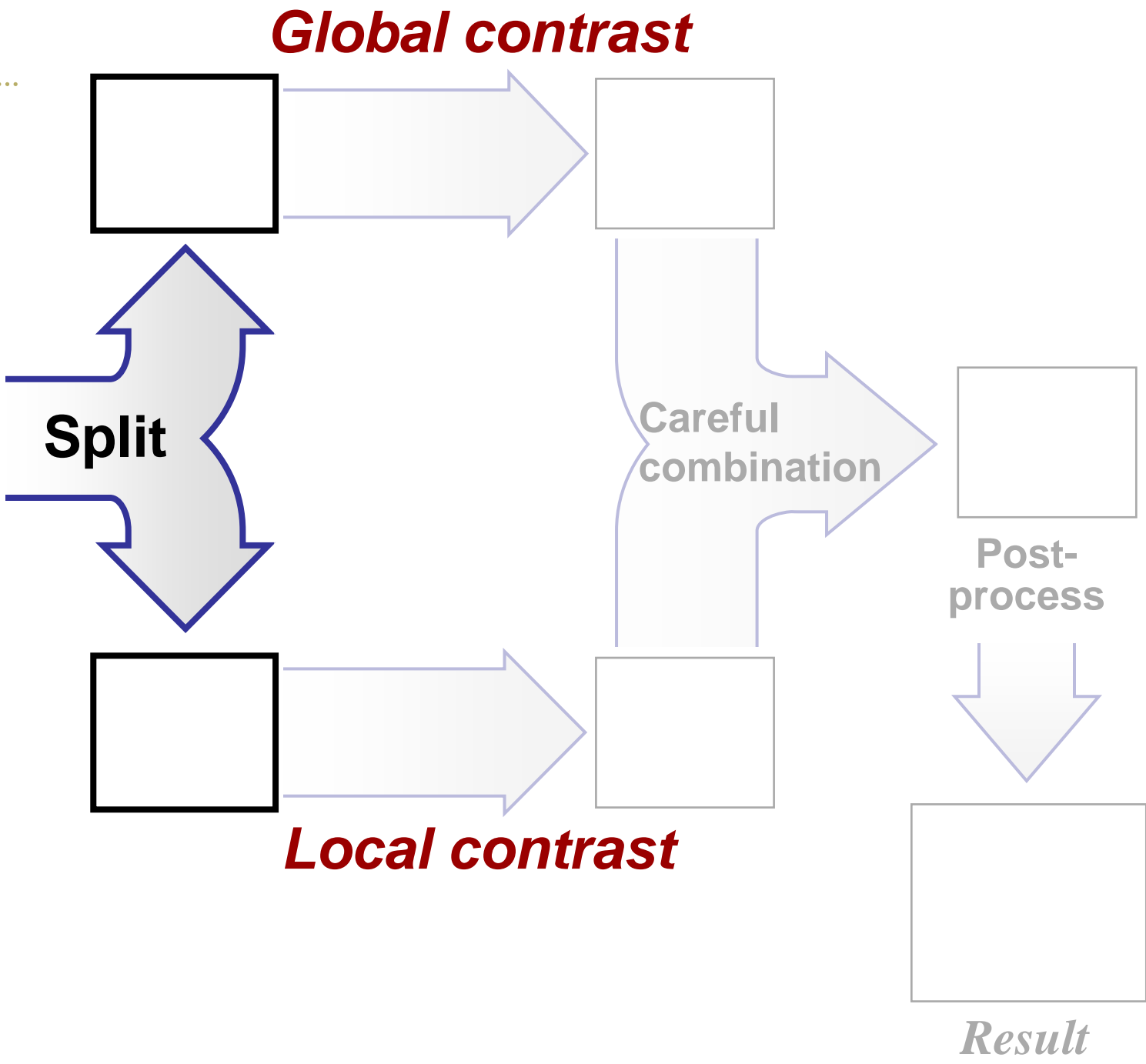


*Result*

# Overview



*Input Image*



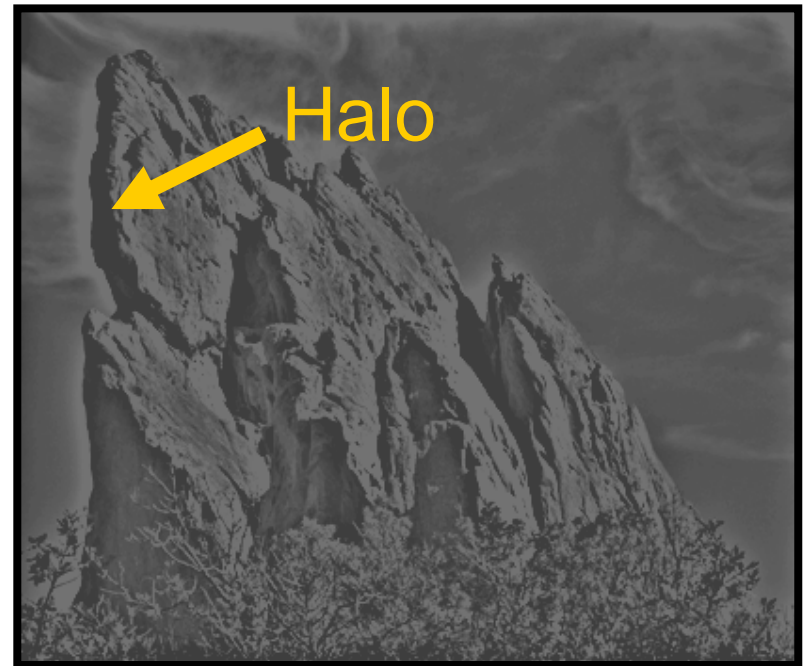
# Split Global vs. Local Contrast

---

- Naïve decomposition: low vs. high frequency
  - Problem: introduce blur & halos



Low frequency  
***Global contrast***



High frequency  
***Local contrast***



# Bilateral Filter

---

- Edge-preserving smoothing [Tomasi 98]
- We build upon tone mapping [Durand 02]



After bilateral filtering  
***Global contrast***



Residual after filtering  
***Local contrast***

# Bilateral Filter

---

- Edge-preserving smoothing [Tomasi 98]
- We build upon tone mapping [Durand 02]

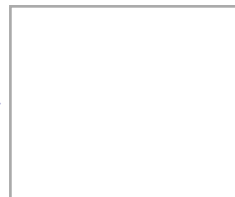


After bilateral filtering  
***Global contrast***

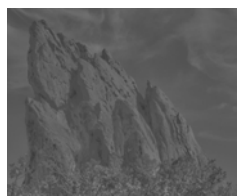


Residual after filtering  
***Local contrast***

***Global contrast***

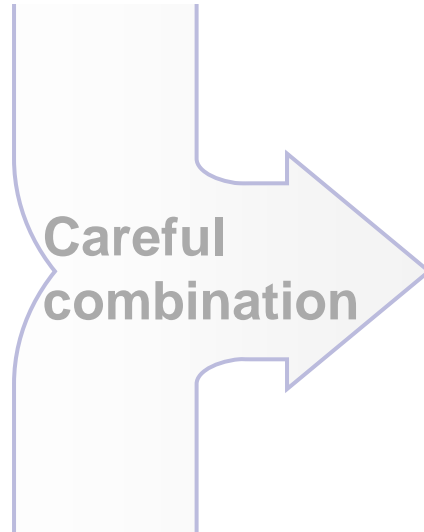


**Bilateral  
Filter**

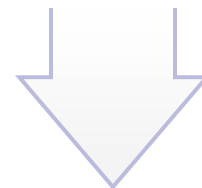


***Local contrast***

**Careful  
combination**

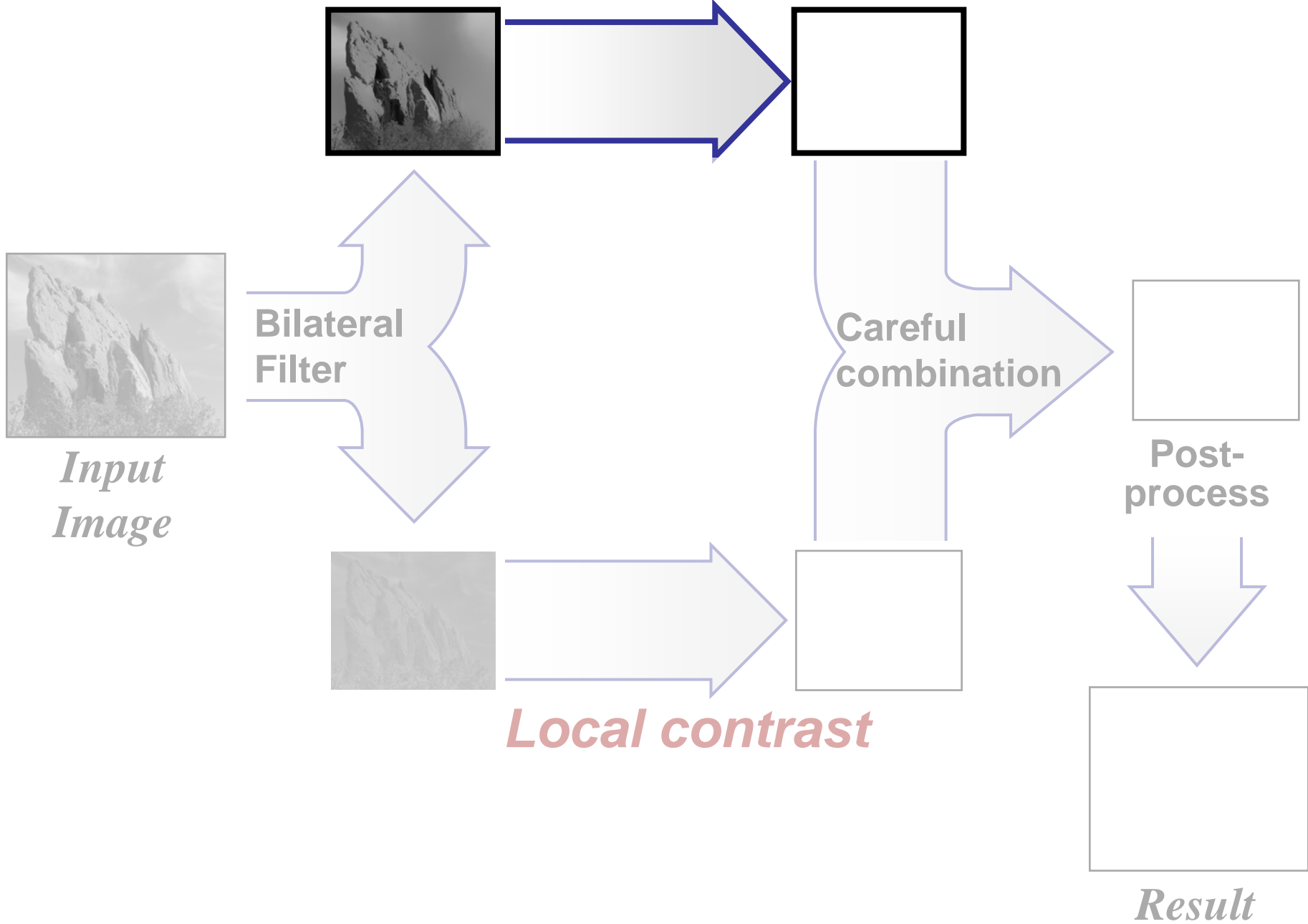


**Post-  
process**



***Result***

# ***Global contrast***



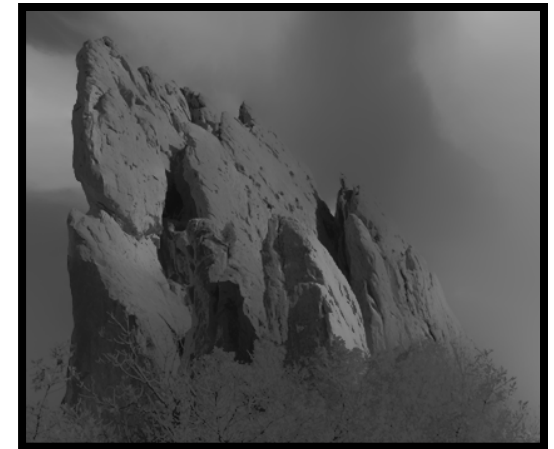
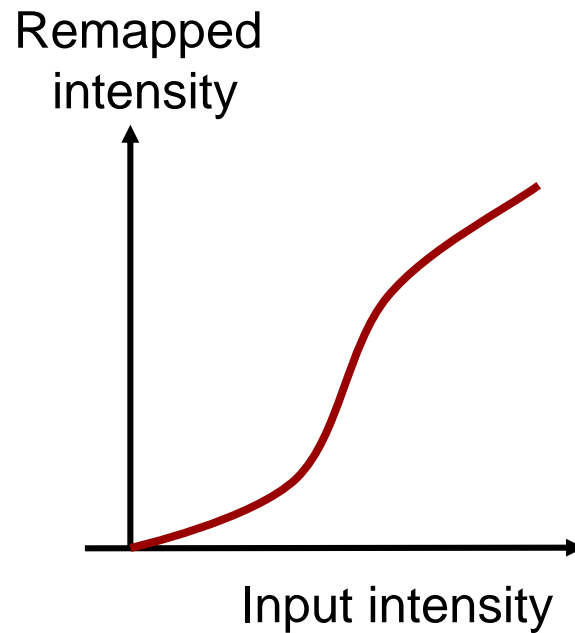
# Global Contrast

---

- Intensity remapping of base layer



Input base



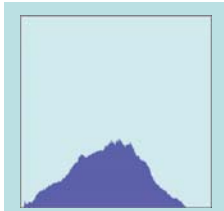
After remapping



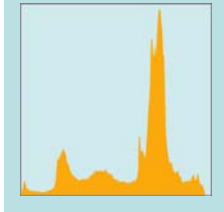
# Global Contrast (Model Transfer)



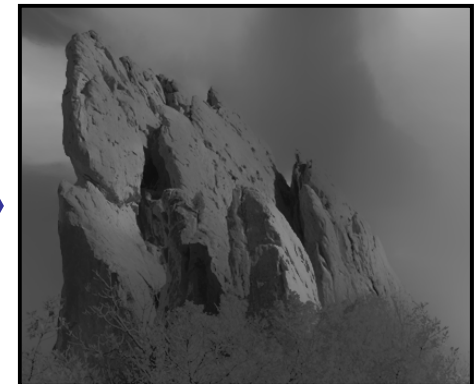
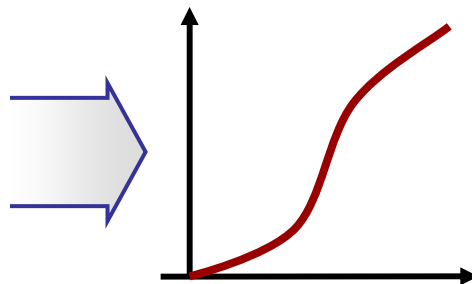
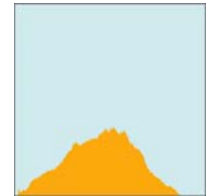
Model  
base



Input  
base



Output  
base



- Histogram matching
  - Remapping function given input and model histogram

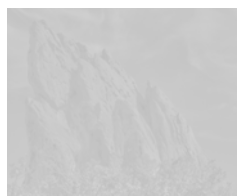
# ***Global contrast***



**Intensity  
matching**



**Bilateral  
Filter**

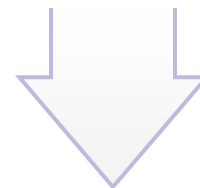


# ***Local contrast***

**Careful  
combination**



**Post-  
process**

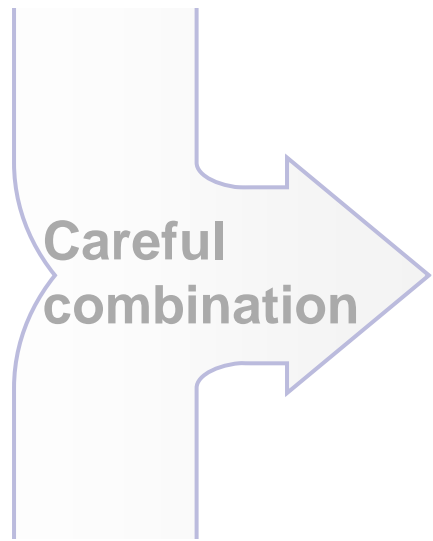
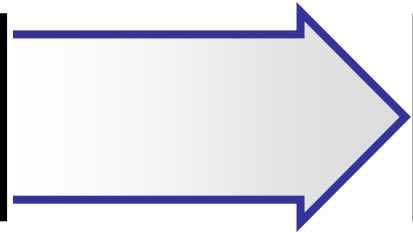
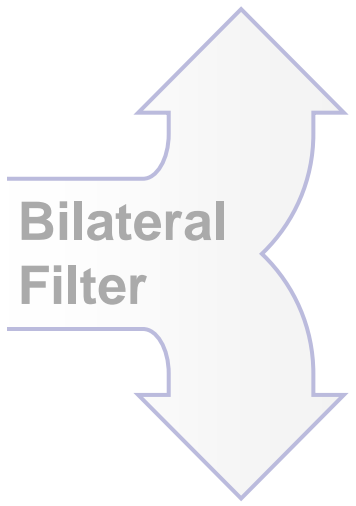


***Result***

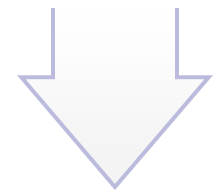
***Input  
Image***

***Result***

# *Global contrast*



Post-process



Result

# *Local contrast*



# Local Contrast: Detail Layer

---

- Uniform control:
  - Multiply all values in the detail layer



Input

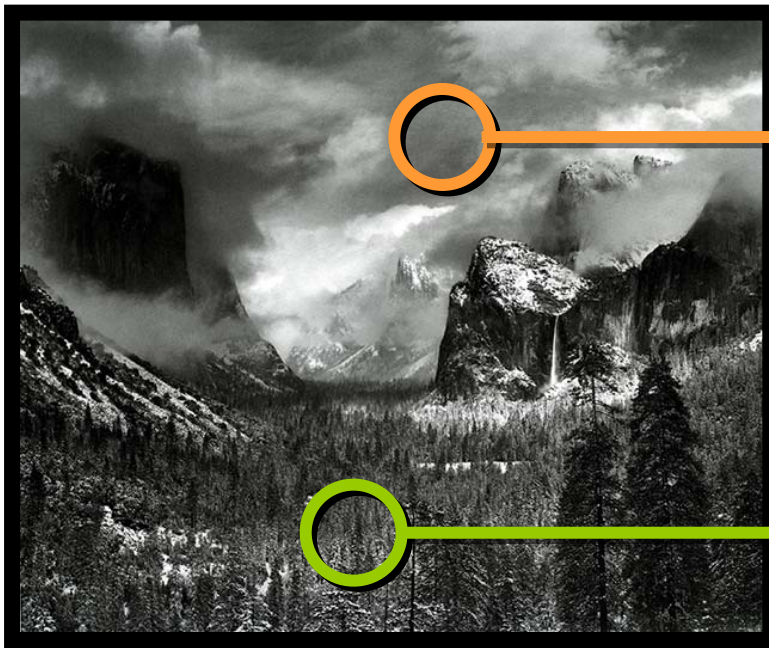


Base + 3 × Detail



# The amount of local contrast is not uniform

---



Smooth region

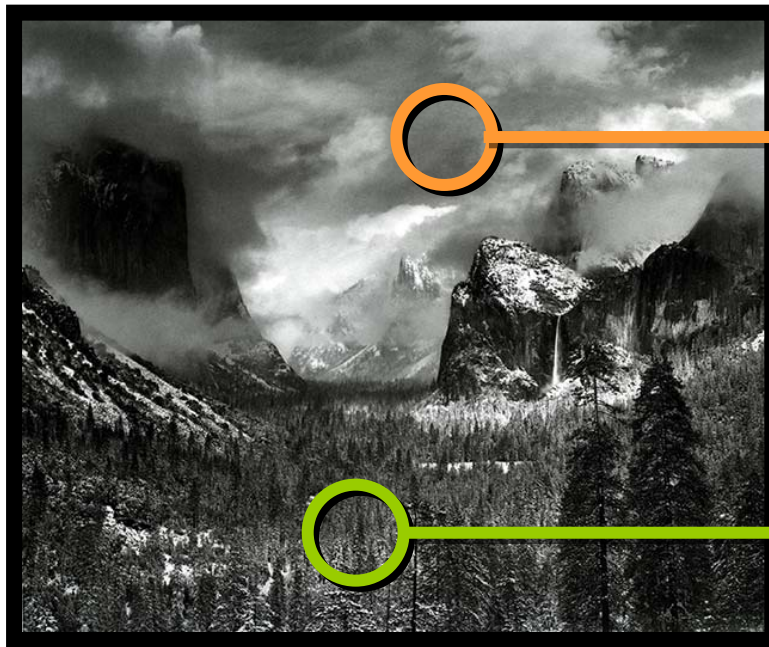
Textured region



# Local Contrast Variation

---

- We define “textureness”: amount of local contrast
  - at each pixel based on surrounding region



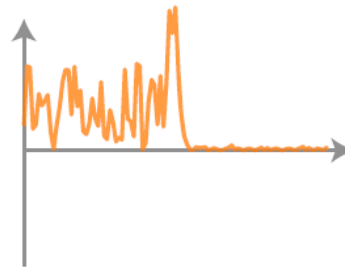
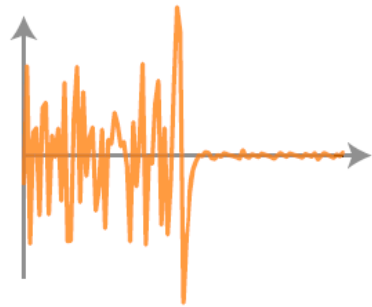
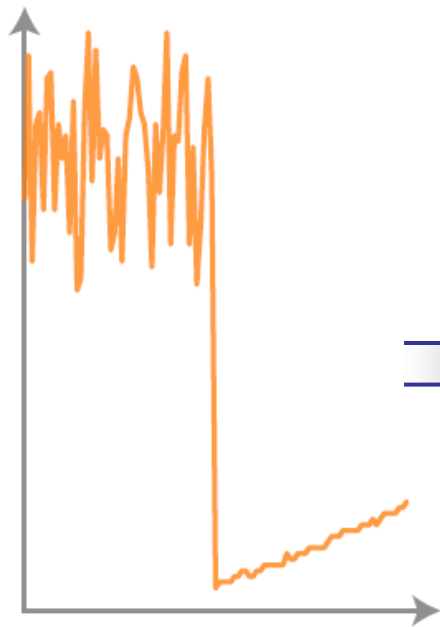
Smooth region  
⇒ Low textureness

Textured region  
⇒ High textureness

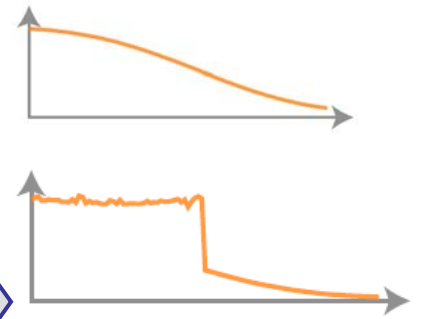


# "Textureiness": 1D Example

---



Previous work:  
Low pass of  $|H|$   
[Li 05, Su 05]



Input signal

High frequency  $H$

Amplitude  $|H|$

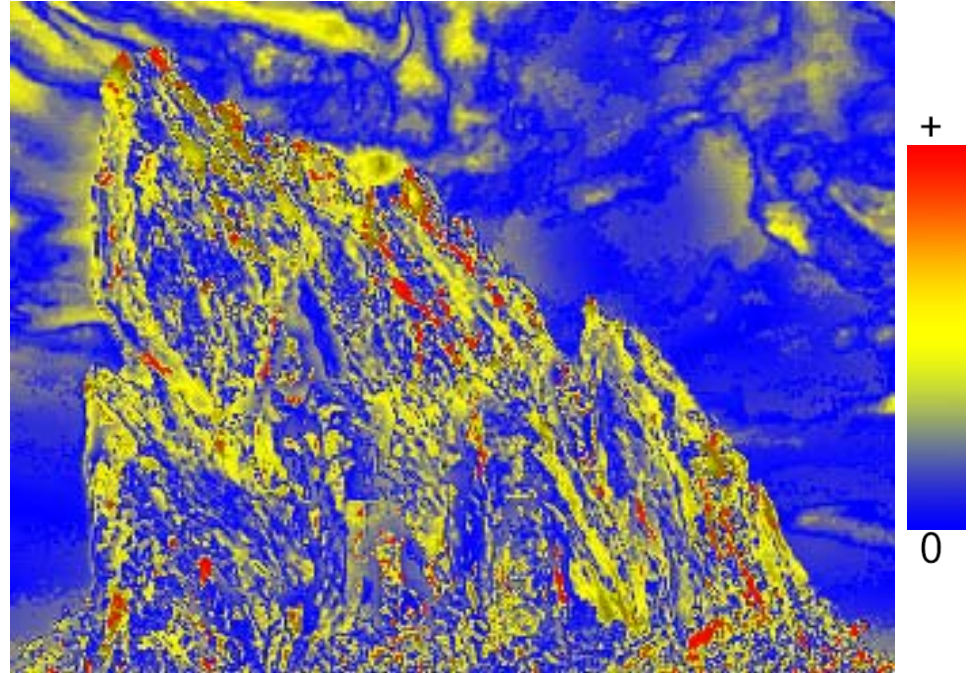
Edge-preserving  
filter

# Textureness

---



Input



Textureness

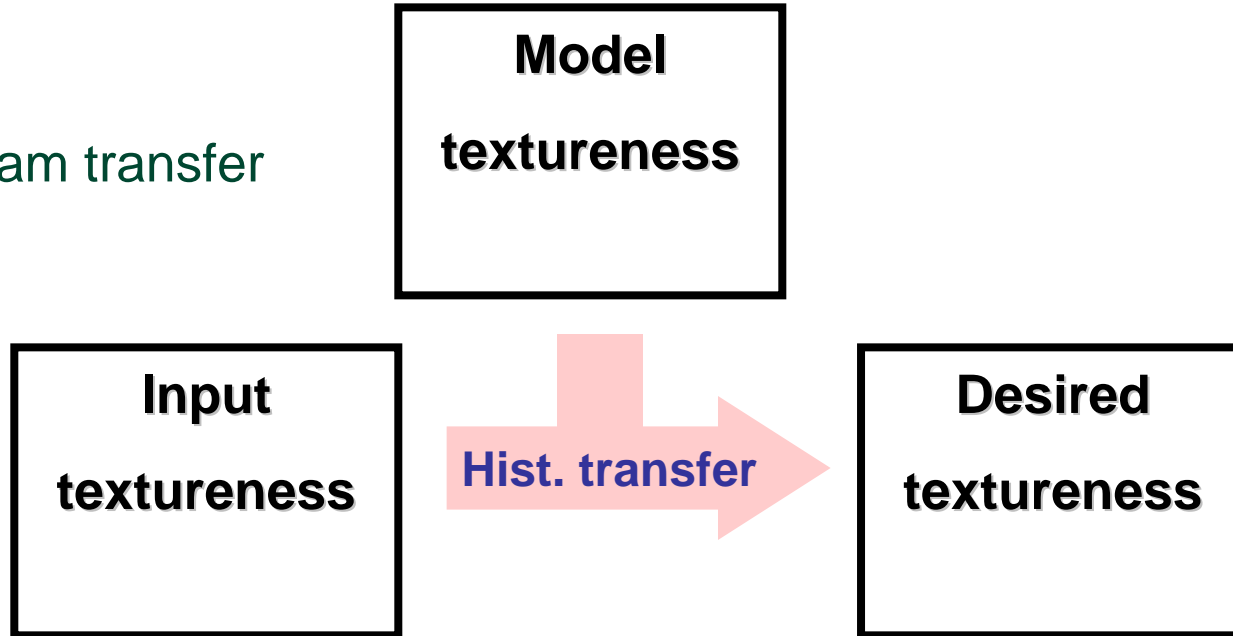


# Textureness Transfer

---

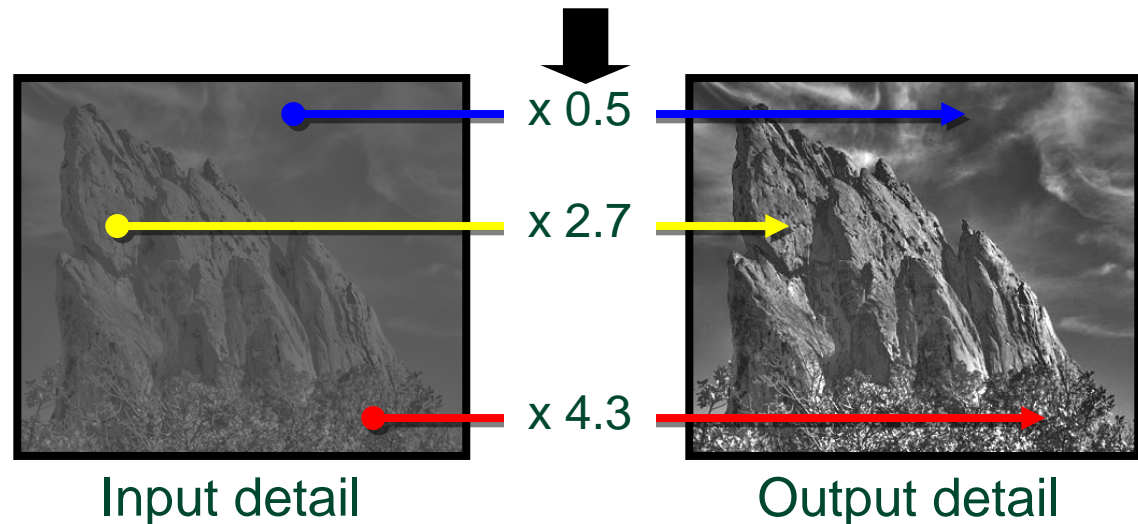
Step 1:

Histogram transfer



Step 2:

Scaling detail layer  
(per pixel) to match  
desired textureness



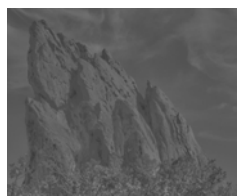
# *Global contrast*



Intensity  
matching



Bilateral  
Filter



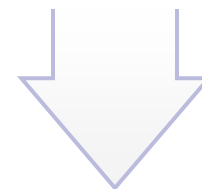
Texture  
matching



Careful  
combination



Post-  
process



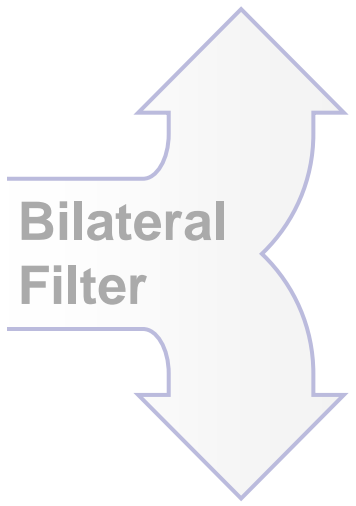
# *Local contrast*

*Input  
Image*

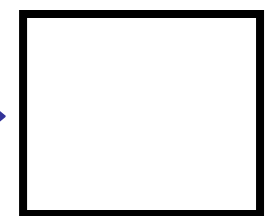
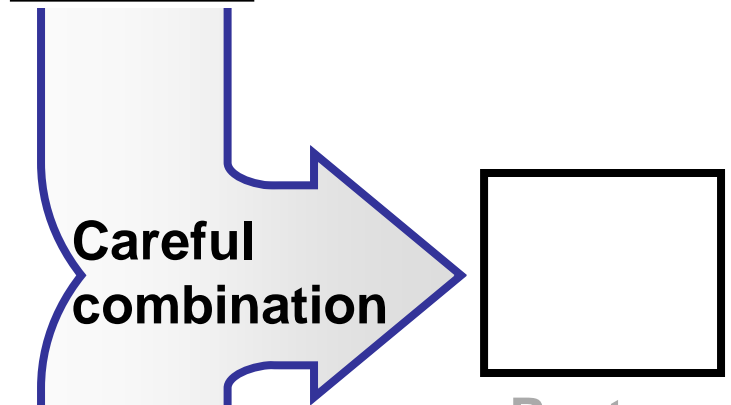
*Result*



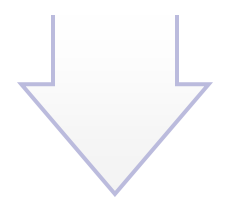
# ***Global contrast***



# ***Local contrast***



Post-process



*Result*

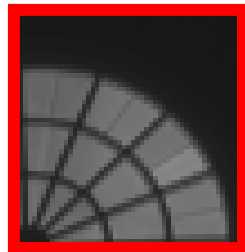


# A Non Perfect Result

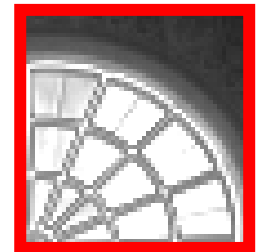
---

- Decoupled and large modifications (up to 6x)  
→ Limited defects may appear

input (HDR)



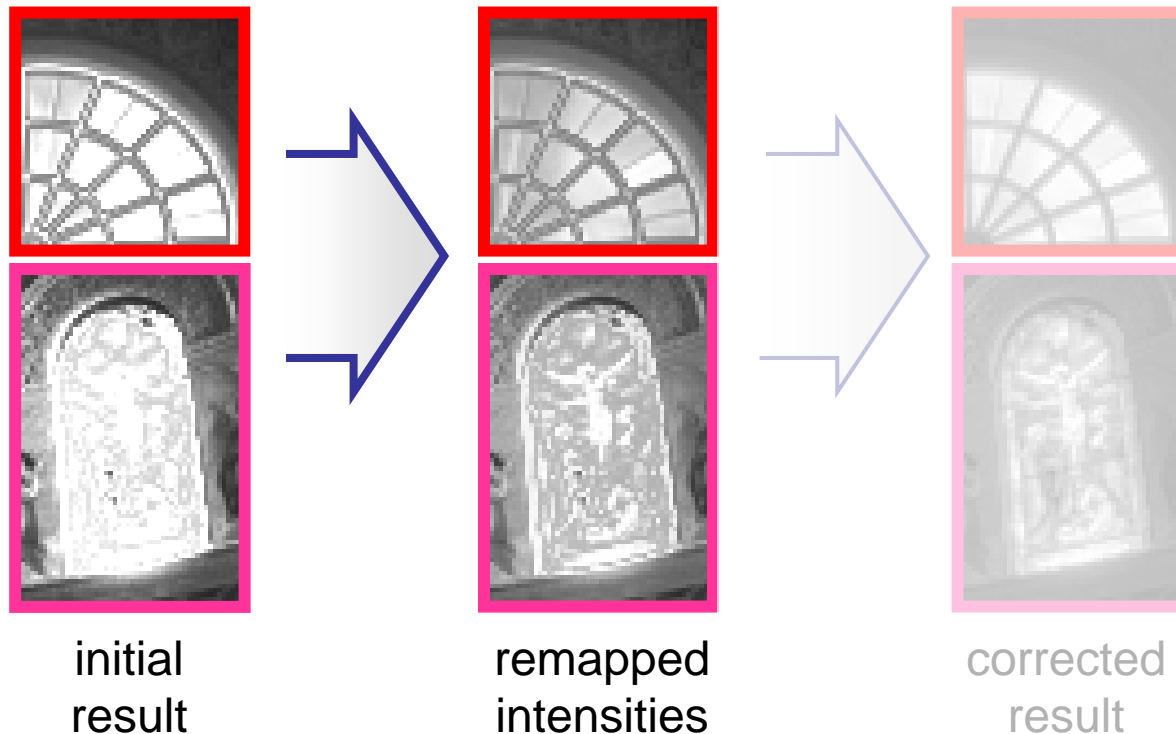
result after  
global and local adjustments



# Intensity Remapping

---

- Some intensities may be outside displayable range.  
→ Compress histogram to fit visible range.

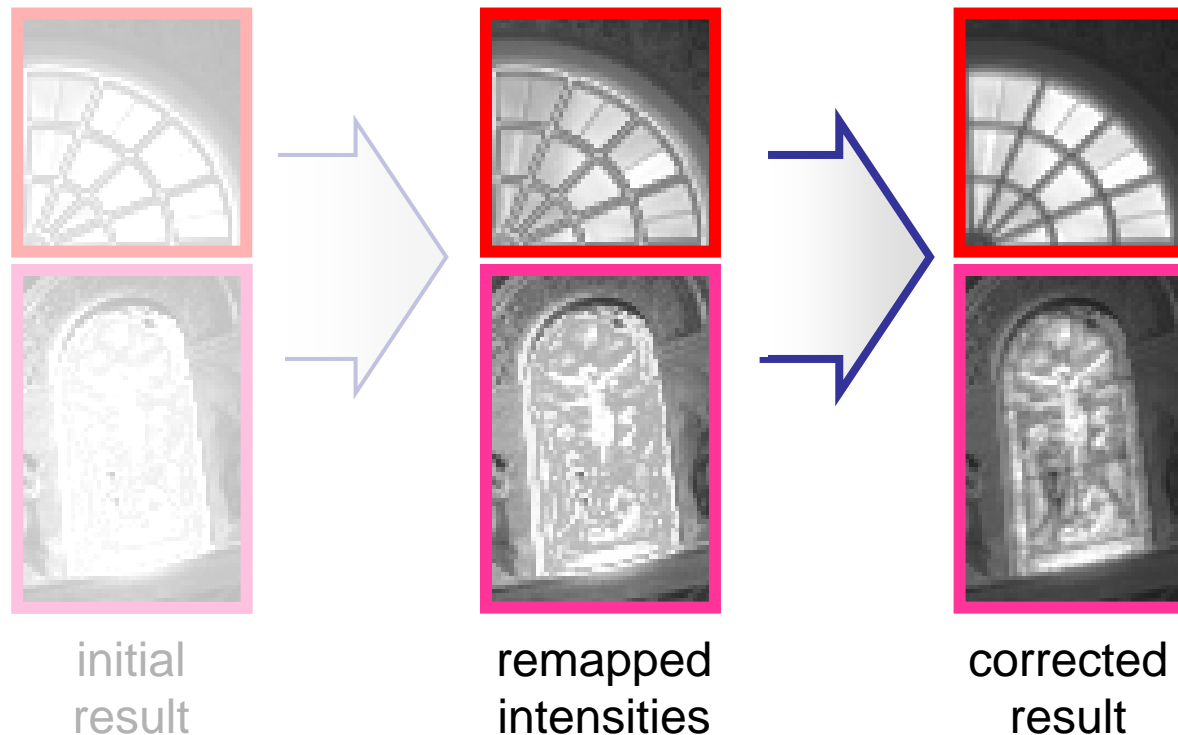


# Preserving Details

## 1. In the **gradient domain**:

- Compare gradient amplitudes of input and current
- Prevent extreme reduction & extreme increase

## 2. Solve the **Poisson equation**.



# Effect of Detail Preservation

---

uncorrected result



corrected result



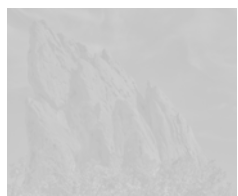
# ***Global contrast***



Intensity  
matching



Bilateral  
Filter



Textureness  
matching

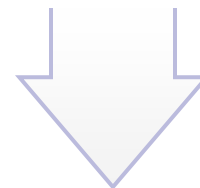


# ***Local contrast***

Constrained  
Poisson



Post-  
process



*Result*



*Input  
Image*

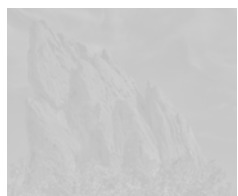
# *Global contrast*



Intensity  
matching



Bilateral  
Filter



Textureness  
matching

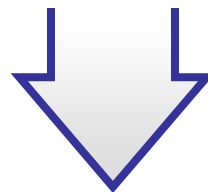


# *Local contrast*

Constrained  
Poisson



Post-  
process



*Result*



*Input  
Image*

# Additional Effects

model

- **Soft focus** (high frequency manipulation)
- **Film grain** (texture synthesis [Heeger 95])
- **Color toning** (chrominance =  $f$ (luminance))



before  
effects



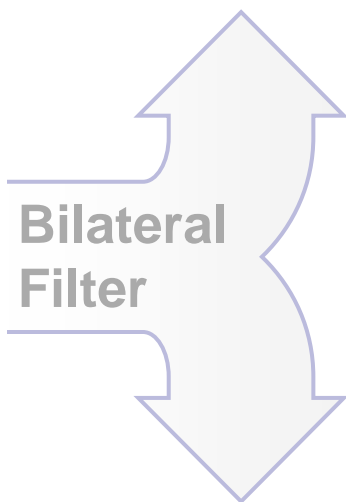
after  
effects



# *Global contrast*



*Input Image*



# *Local contrast*

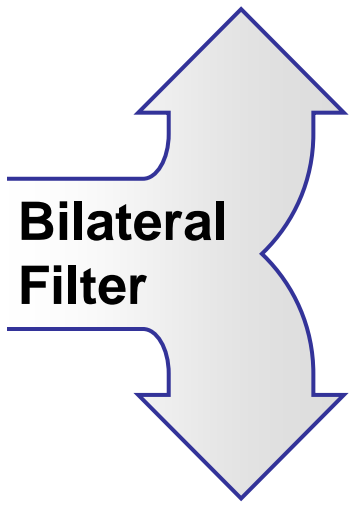


*Result*



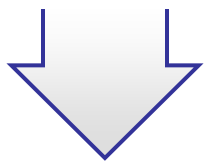
# Recap

## Global contrast



## Local contrast

Soft focus Toning Grain



# Results

---

User provides input and model photographs.

→ Our system **automatically** produces the result.

Running times:

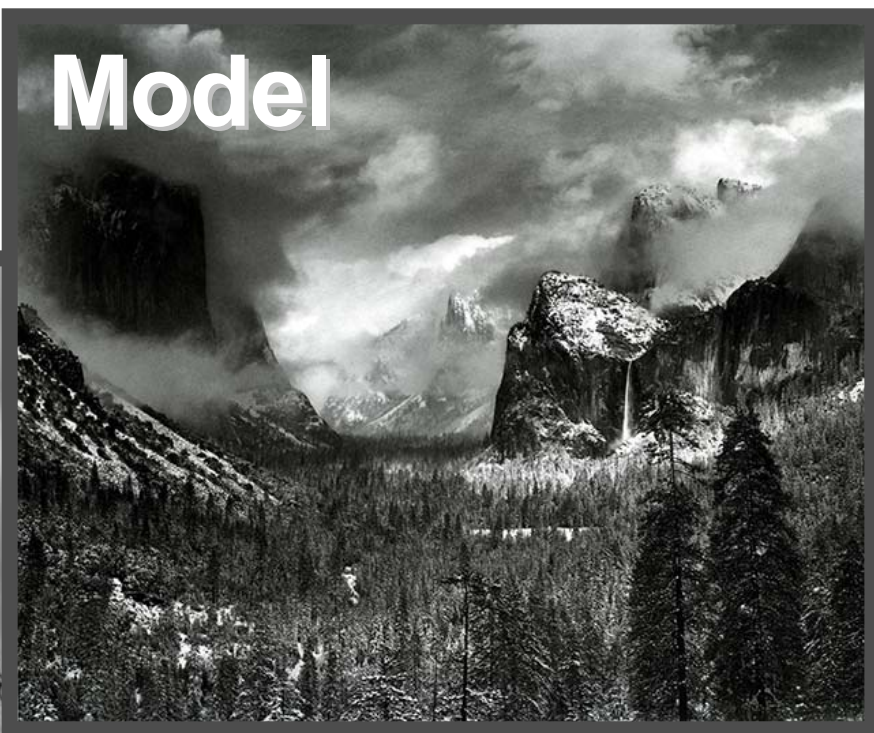
- 6 seconds for 1 MPixel or less
- 23 seconds for 4 MPixels
- multi-grid Poisson solver and fast bilateral filter [Paris 06]



# Input



# Model



# Result

# Model



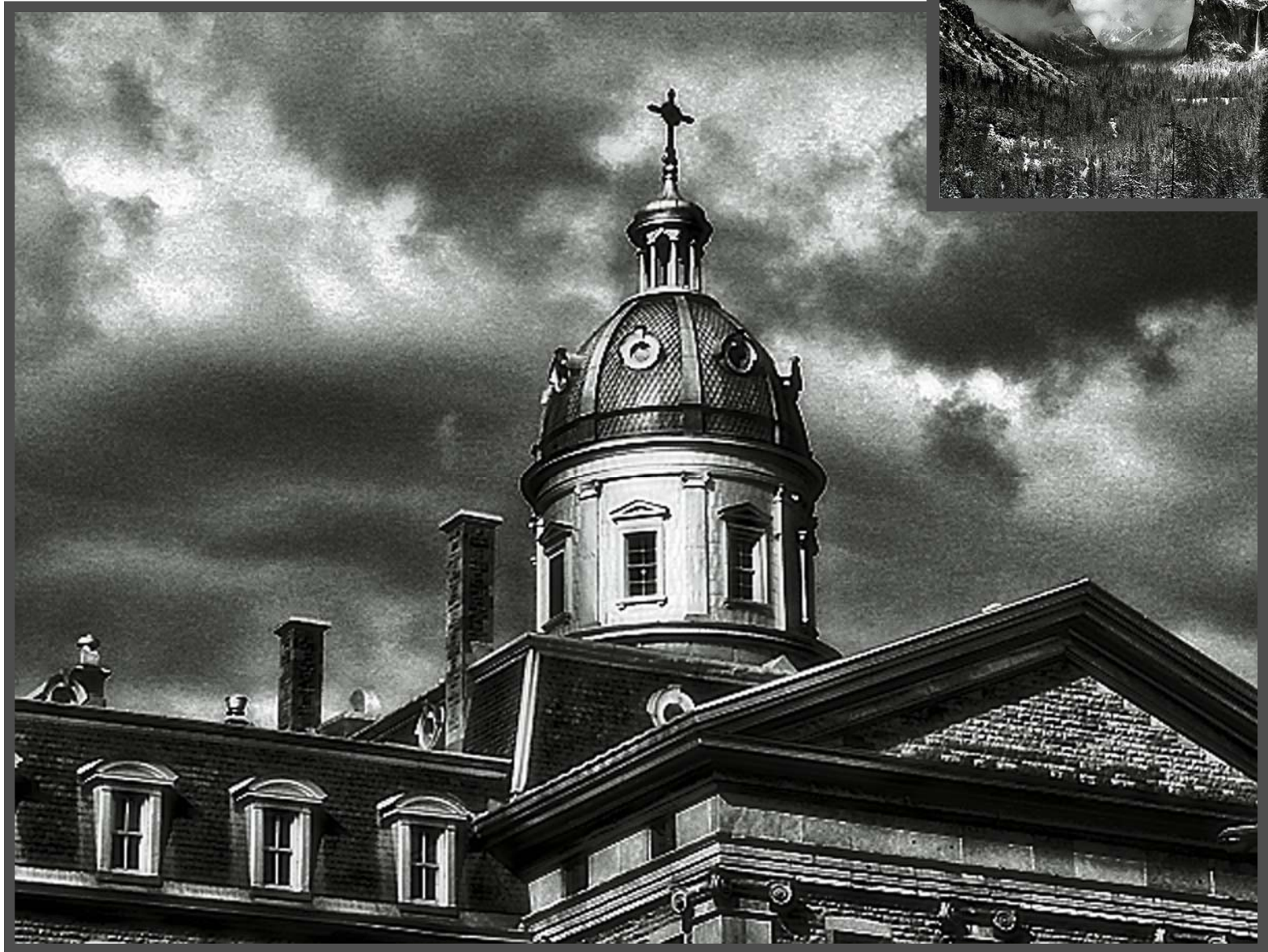
# Input

---

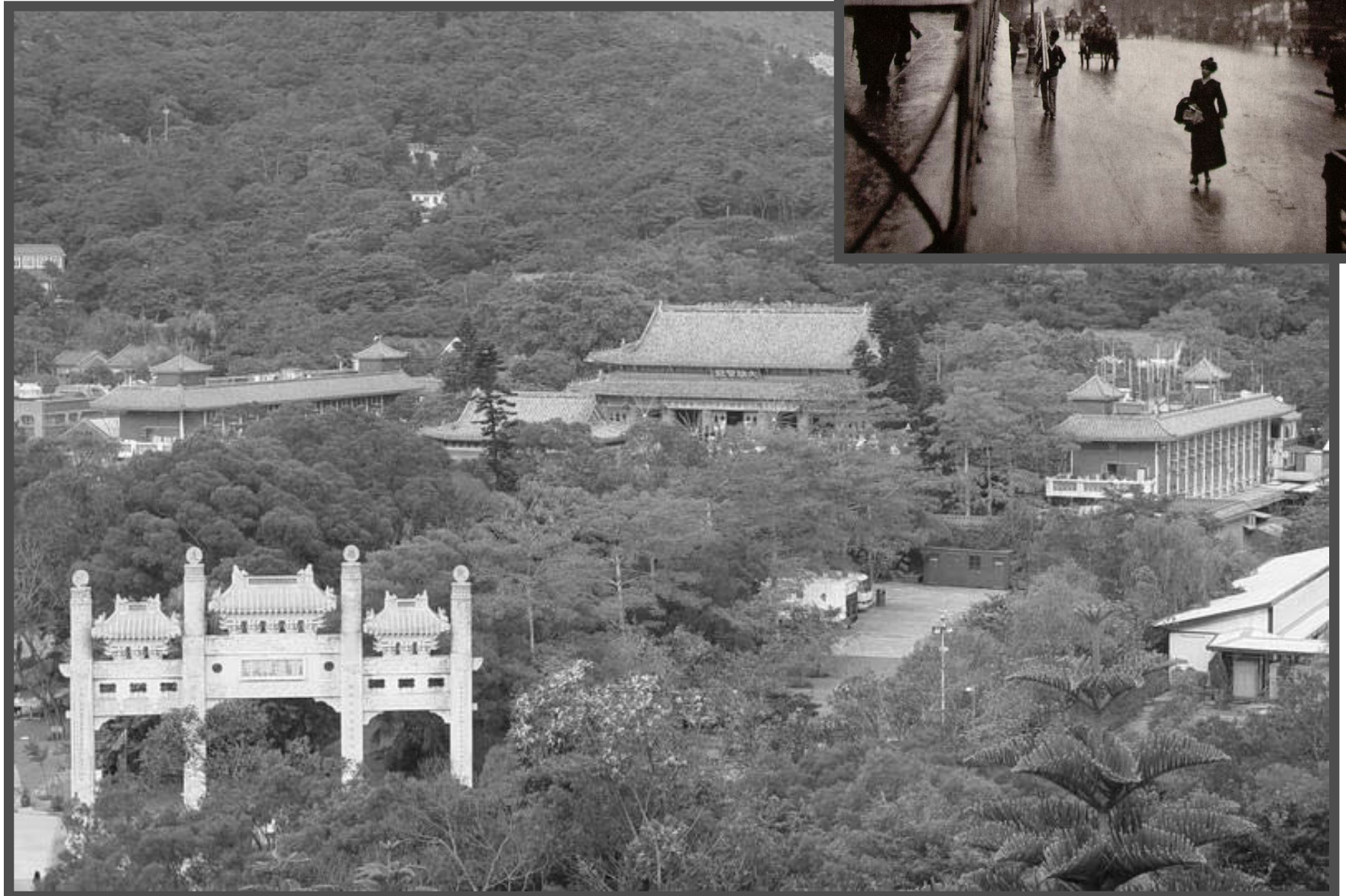


# Result

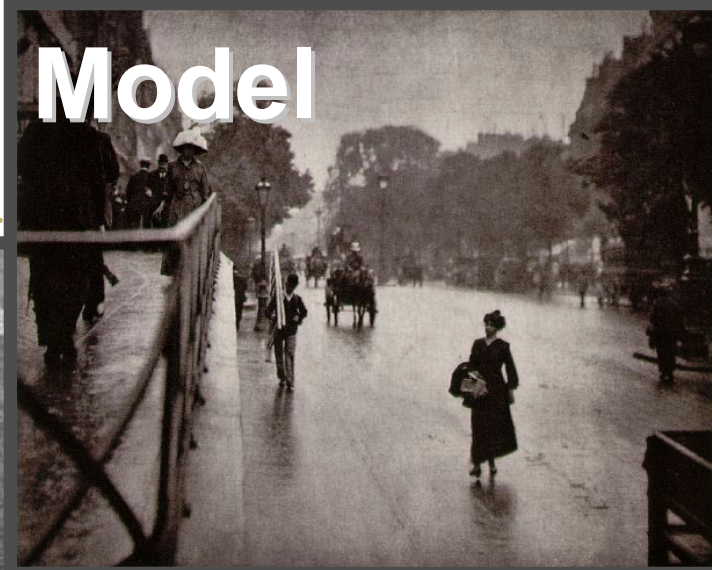
---



# Input

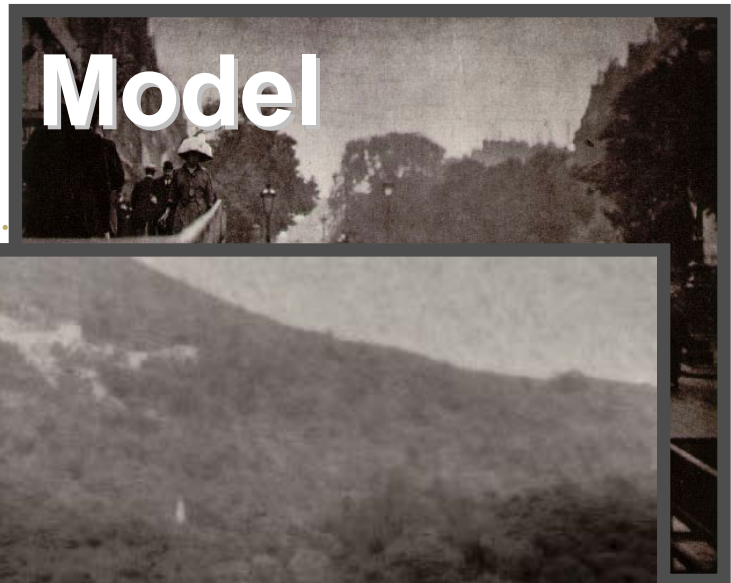
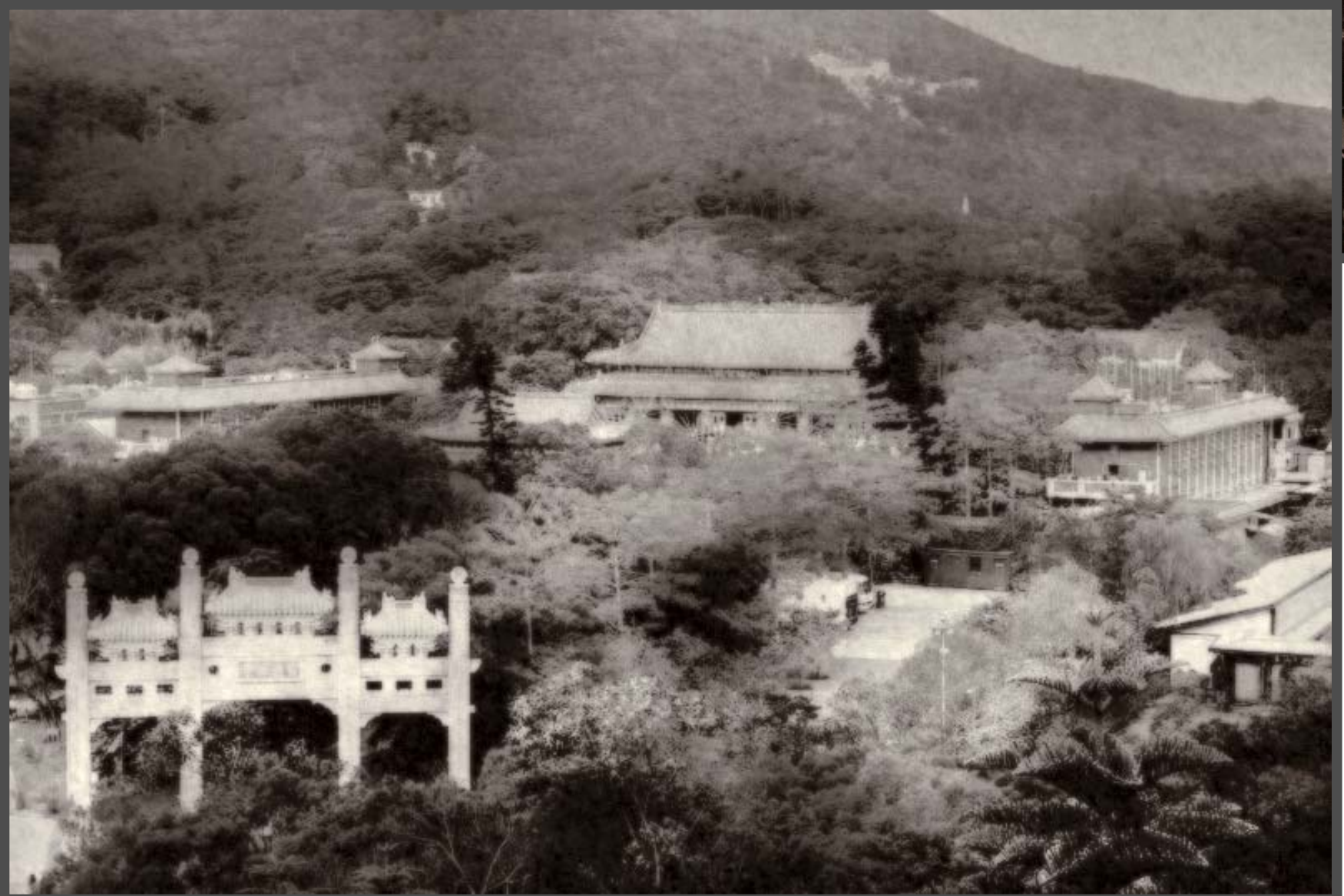


# Model



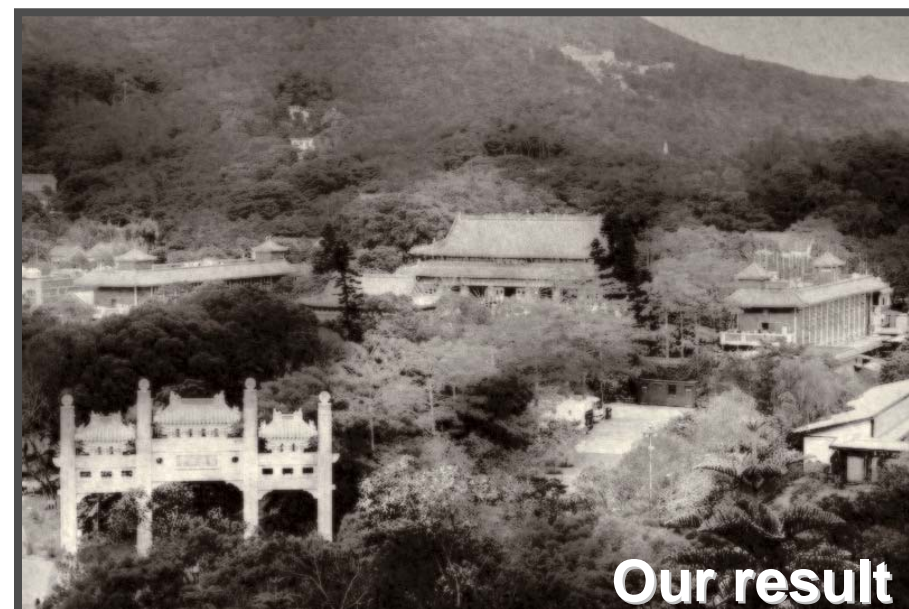
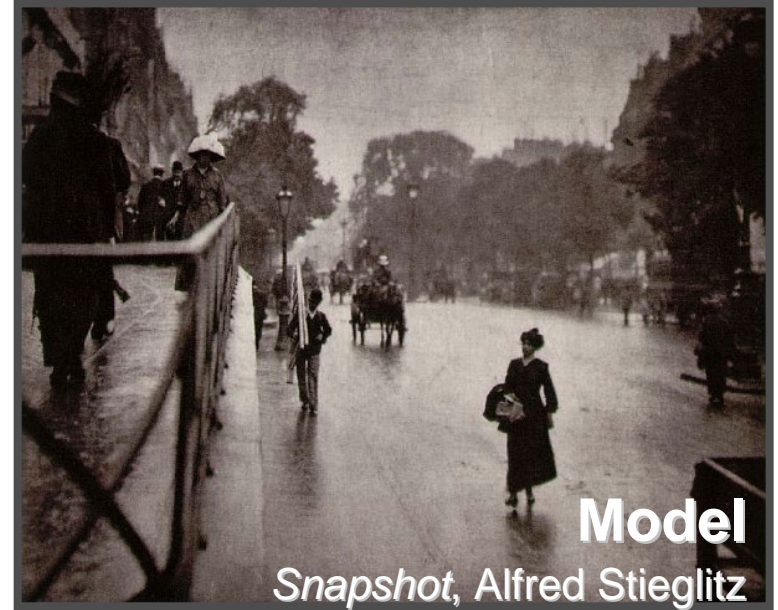
# Result

# Model



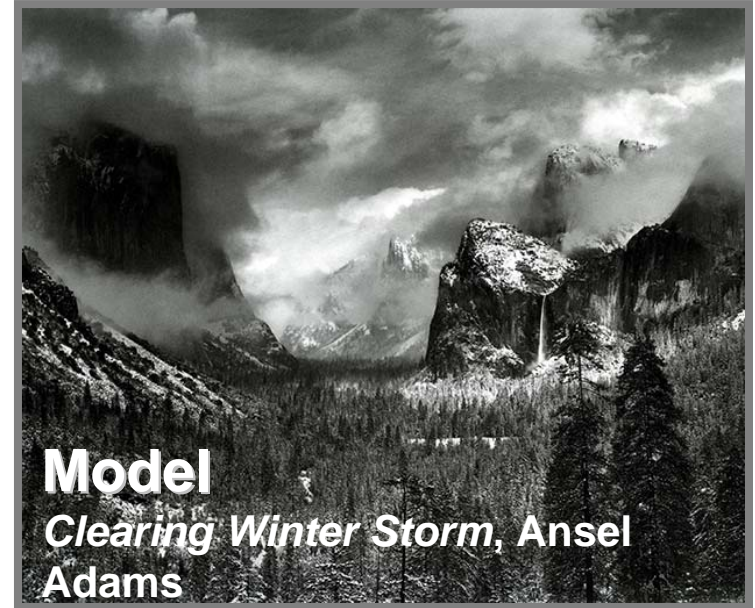


# Comparison with Naïve Histogram Matching



Local contrast, sharpness unfaithful

# Comparison with Naïve Histogram Matching



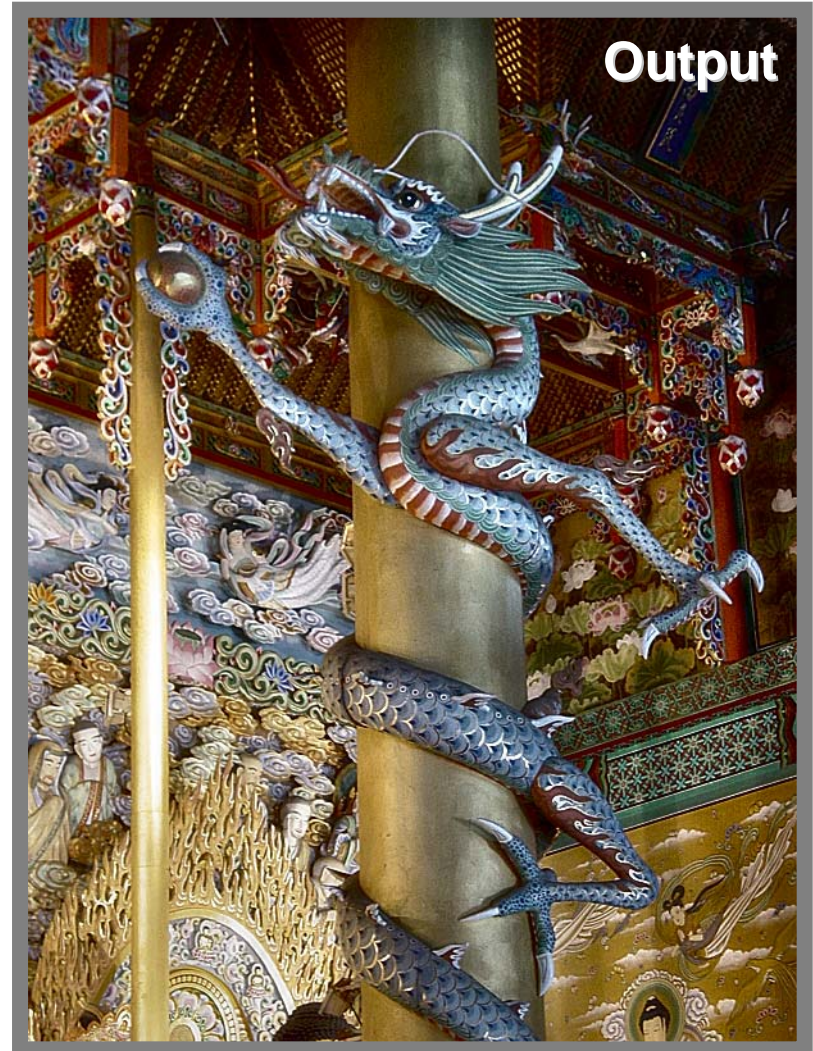
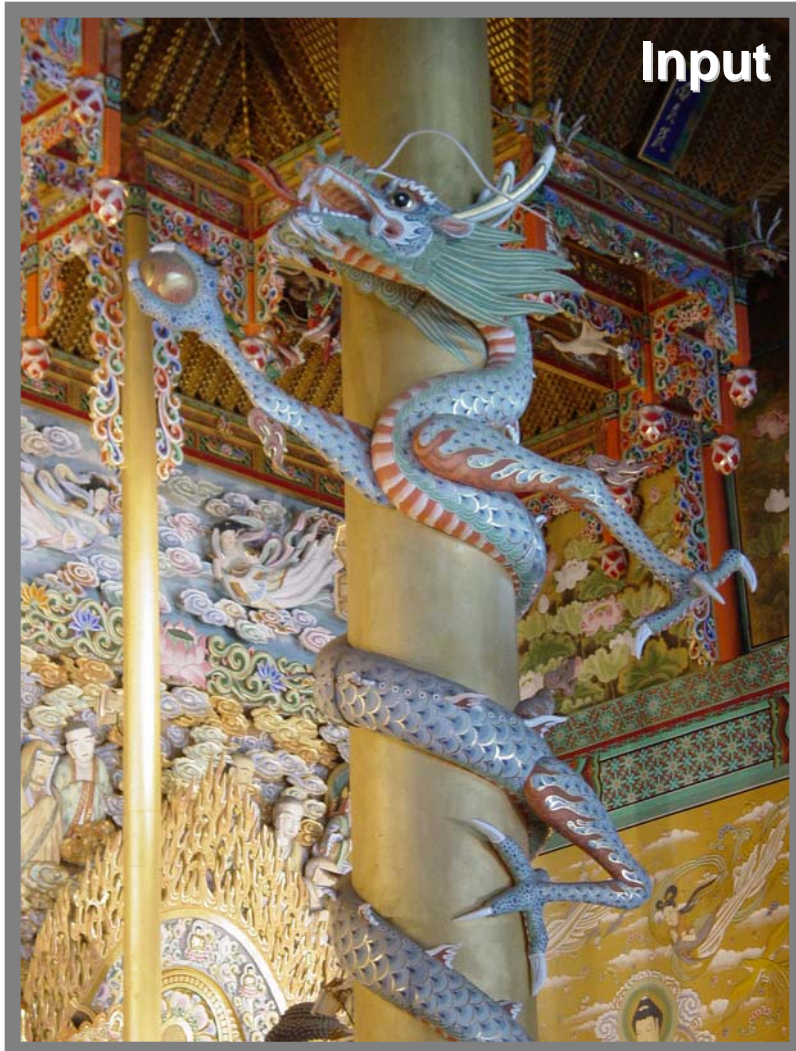
Local contrast too low



# Color Images

---

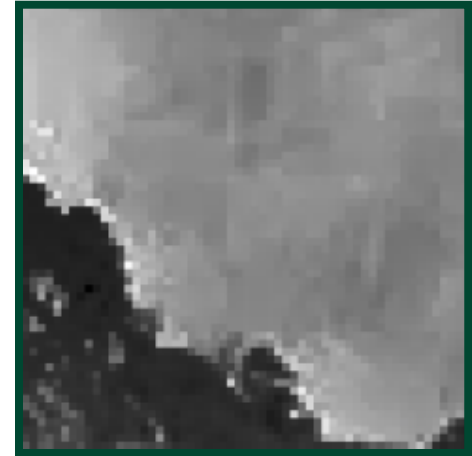
- Lab color space: modify only luminance



# Limitations

---

- Noise and JPEG artifacts
  - amplified defects
  
- Can lead to unexpected results if the image content is too different from the model
  - Portraits, in particular, can suffer



# Conclusions

---

- Transfer “look” from a model photo
- Two-scale tone management
  - Global and local contrast
  - New edge-preserving textureness
  - Constrained Poisson reconstruction
  - Additional effects

