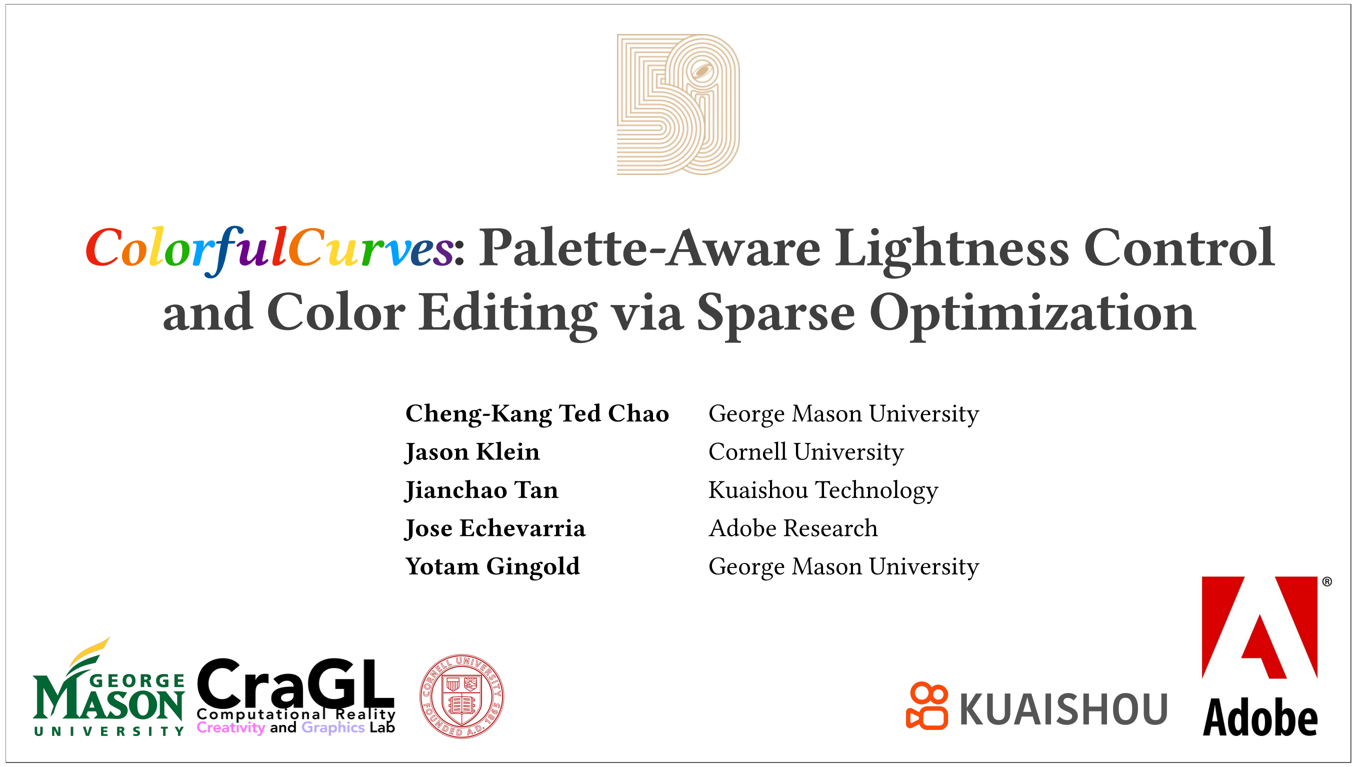


ColorfulCurves: Palette-Aware Lightness Control and Color Editing via Sparse Optimization

Cheng-Kang Ted Chao George Mason University
Jason Klein Cornell University
Jianchao Tan Kuaishou Technology
Jose Echevarria Adobe Research
Yotam Gingold George Mason University

(15s)

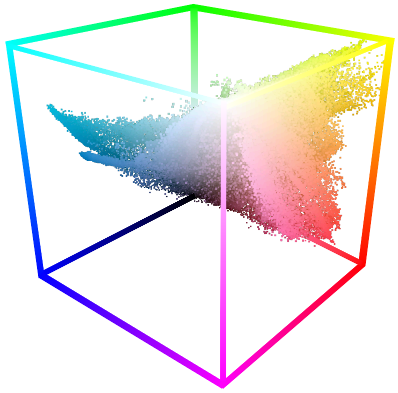
Hi, I’m Ted from George Mason. This is our project ColorfulCurves .....
This is a joint work with Jason from Cornell, Jianchao from Kuaishou Tech, Jose from Adobe Research, and Yotam from George Mason.

Palette-based editing

(10s)

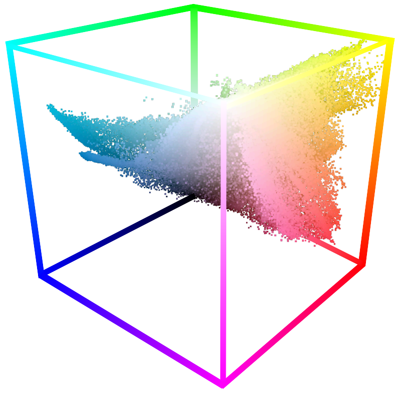
So let’s have a quick recap on palette-based editing.
There are many existing works on finding representative colors from an image such as clustering-based, or looking at color histogram or data-driven approach.

Palette-based editing



Palette-based editing

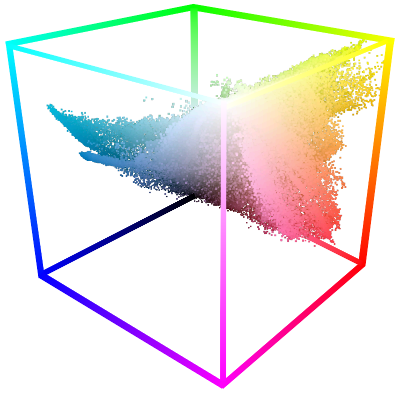
Clustering-based: [Chang et al. 2015; Nguyen et al. 2017; Zhang et al. 2017]



Palette-based editing

Clustering-based: [Chang et al. 2015; Nguyen et al. 2017; Zhang et al. 2017]

Histogram: [Morse et al. 2007]

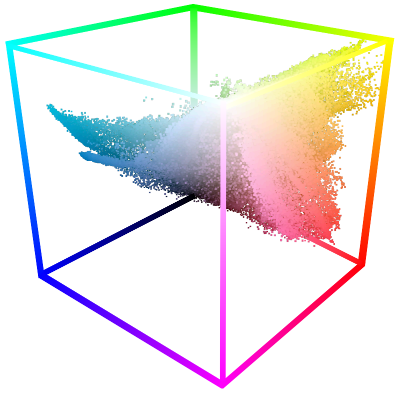


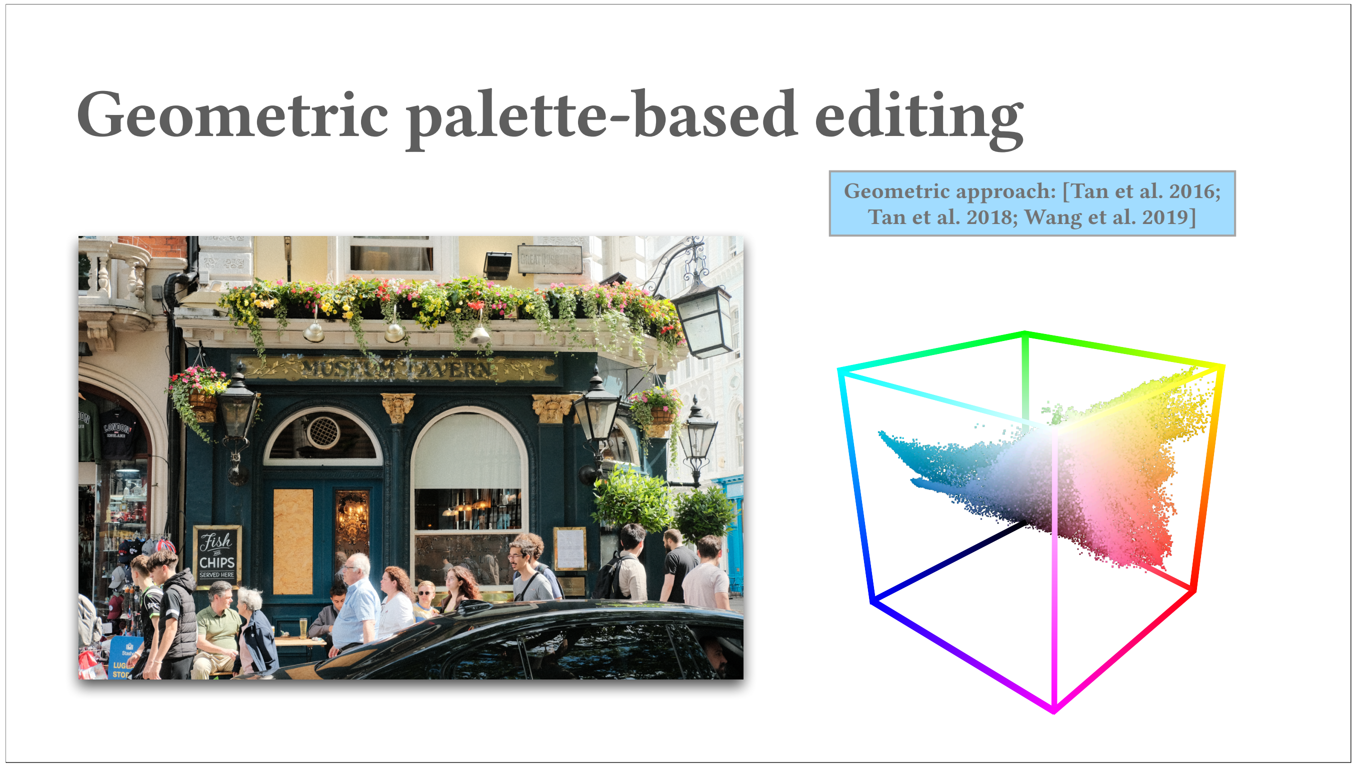
Palette-based editing

Clustering-based: [Chang et al. 2015; Nguyen et al. 2017; Zhang et al. 2017]

Histogram: [Morse et al. 2007]

Data-driven: [Lin & Hanrahan 2013; O'Donovan et al. 2011]





Geometric palette-based editing

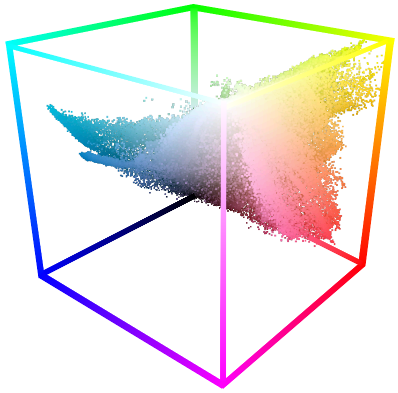
Geometric approach: [Tan et al. 2016; Tan et al. 2018; Wang et al. 2019]

(40s)

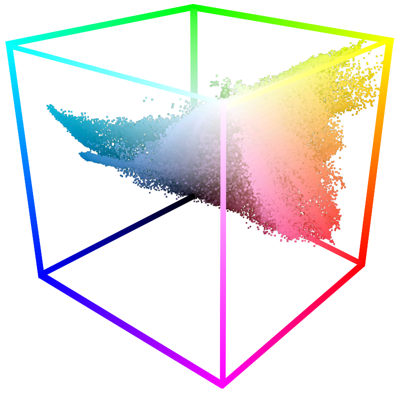
But in this work, we focus on geometric approach to extract palette.
Specifically, we build upon Tan et al. 2018 due to its efficient formulation for getting fast optimization in our pipeline and also because the approach provides spatially coherent editing.
They compute the palette as a simplified convex hull and they compute weights through GBC with respect to the palette. The spatial coherence comes from the tessellation in RGBXY space.
Therefore, after the computation, we can write the decomposition as I=WP. By fixing the weights W, users can change the palette P to perform global color edits.

Geometric palette-based editing

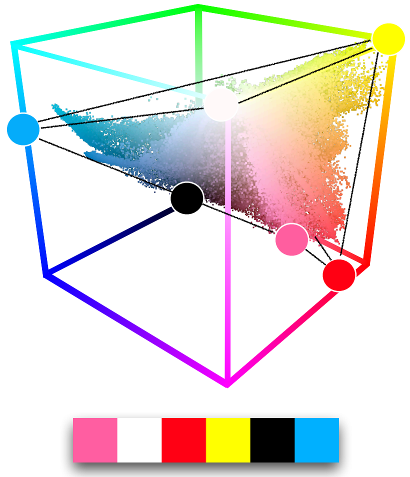
Geometric approach: [Tan et al. 2016; Tan et al. 2018; Wang et al. 2019]



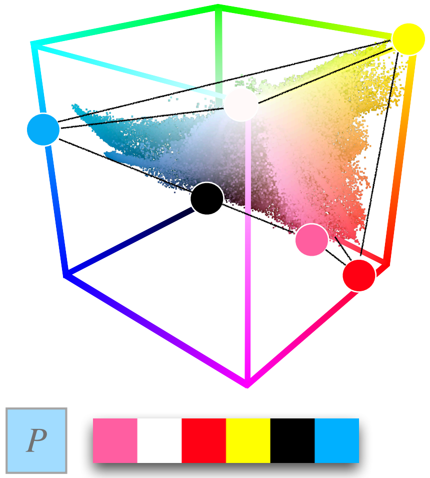
Geometric palette-based editing



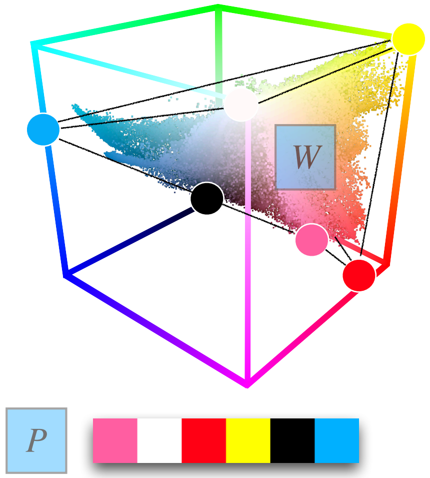
Geometric palette-based editing



Geometric palette-based editing

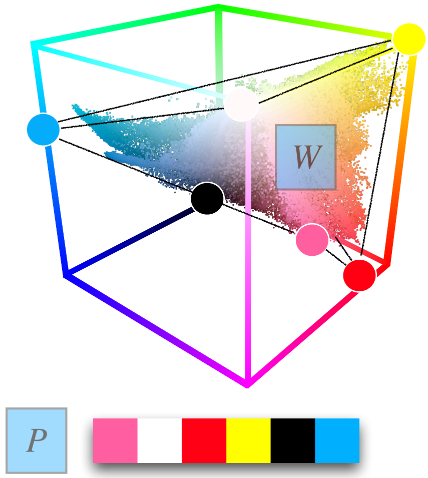


Geometric palette-based editing



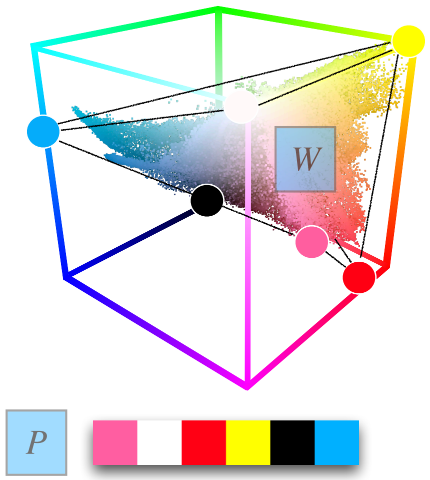
Geometric palette-based editing

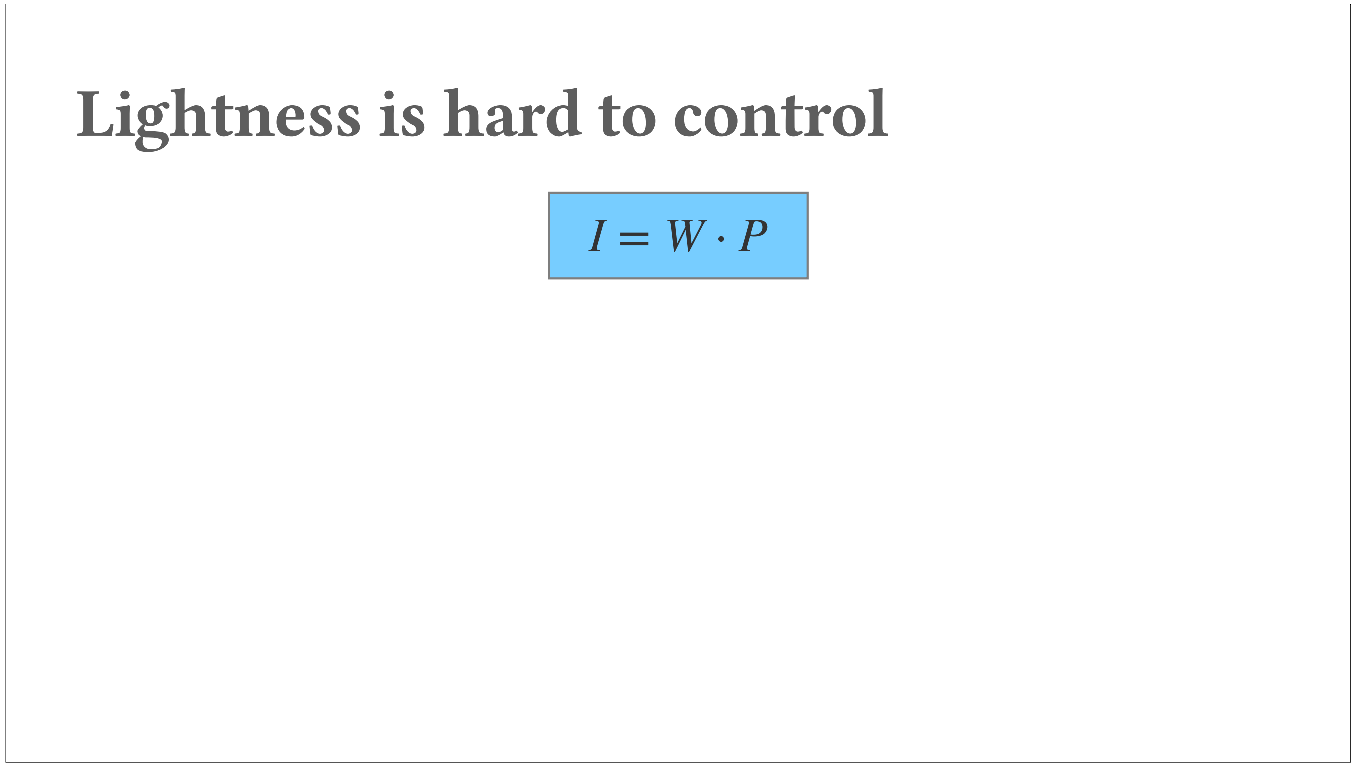
$I = W \cdot P$



Geometric palette-based editing

$I = W \cdot P$





Lightness is hard to control

$$\boxed{I = W \cdot P}$$

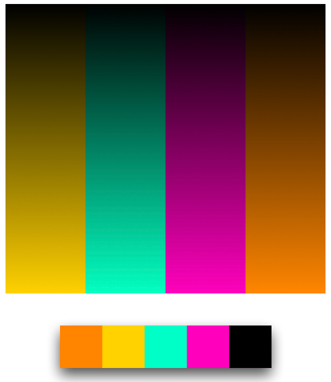
(20s)

While this nice color mixing formulation provides spatially coherent weights, the flexibility of its control on lightness is insufficient because of its linearity.
Our Colofulcurves can perform flexible multi-tone lightness editing. And I will show how in the later slides.

(We might also be able to consider nonlinear scheme, but it might result in a more complicated color mixing model for controlling lightness.)

Lightness is hard to control

$$I = W \cdot P$$

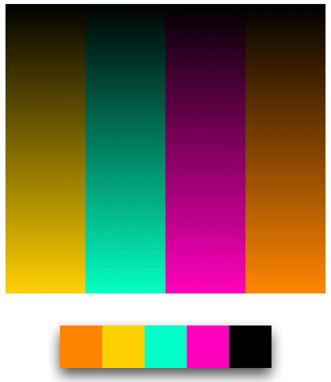


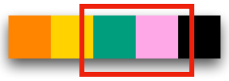
Input

Lightness is hard to control

$$I = W \cdot P$$

Input

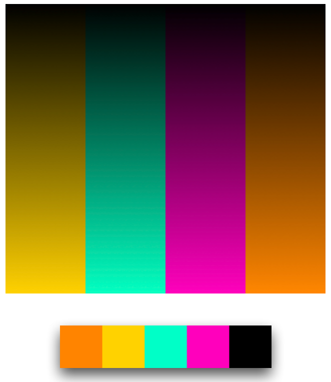




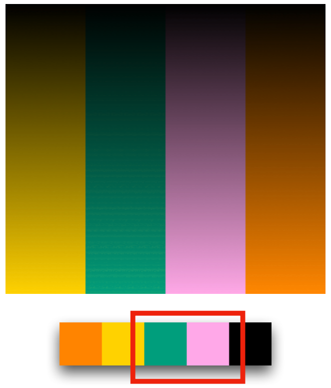
Lightness is hard to control

$$I = W \cdot P$$

Input



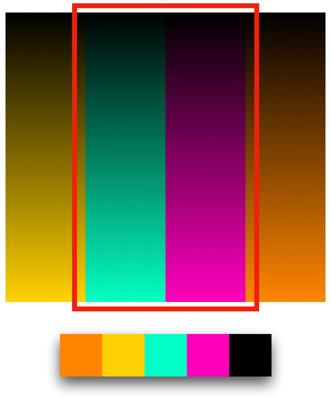
[Tan et al. 2018]



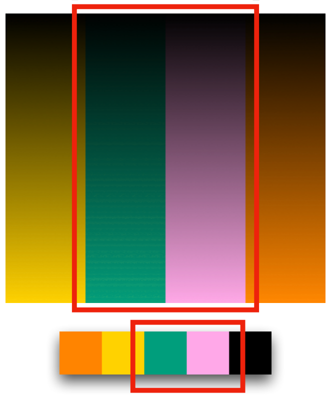
Lightness is hard to control

$$I = W \cdot P$$

Input

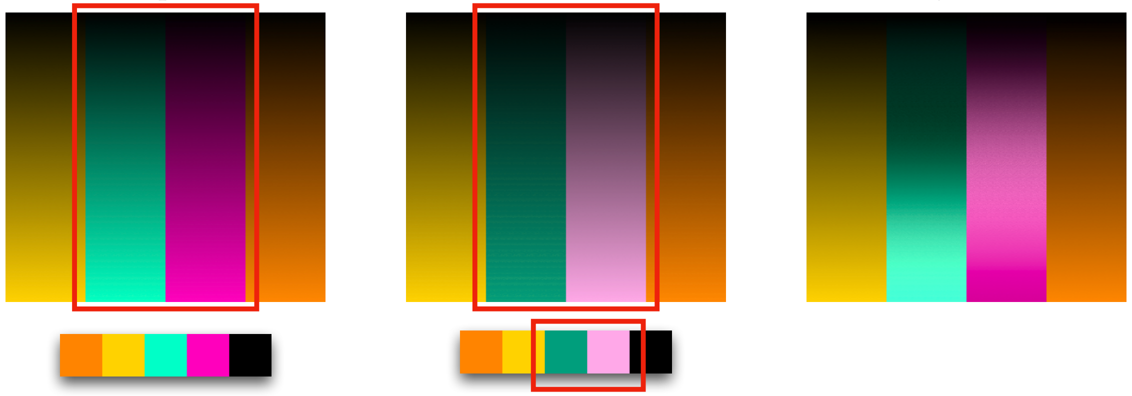


[Tan et al. 2018]



Lightness is hard to control

$$I = W \cdot P$$



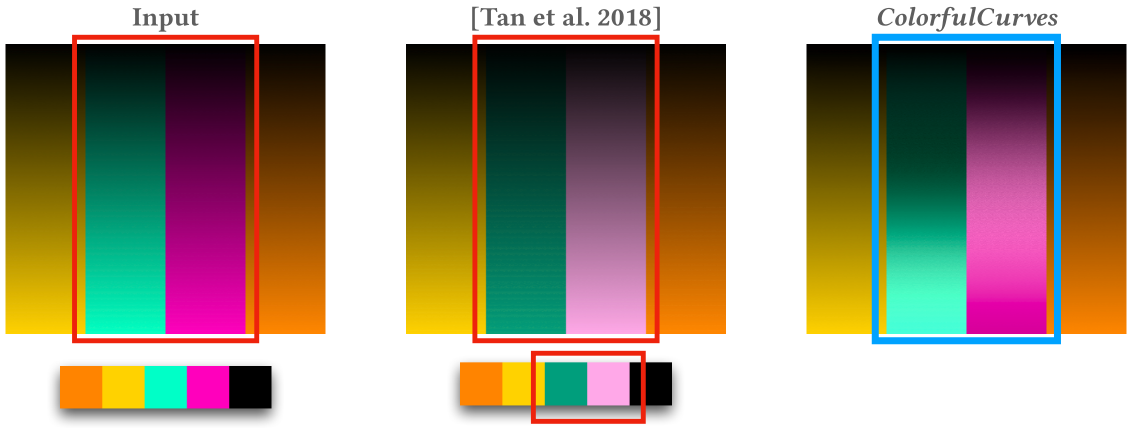
Input

[Tan et al. 2018]

ColorfulCurves

Lightness is hard to control

$$I = W \cdot P$$

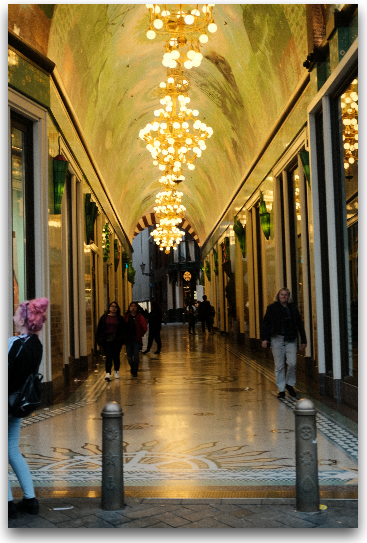


Input

[Tan et al. 2018]

ColorfulCurves

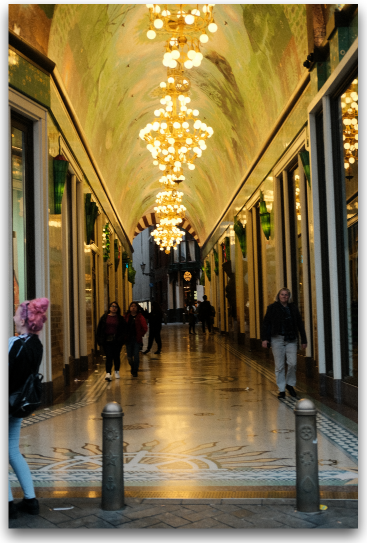
Tone curves aren’t local or color-aware

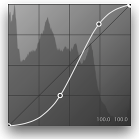


(35s)

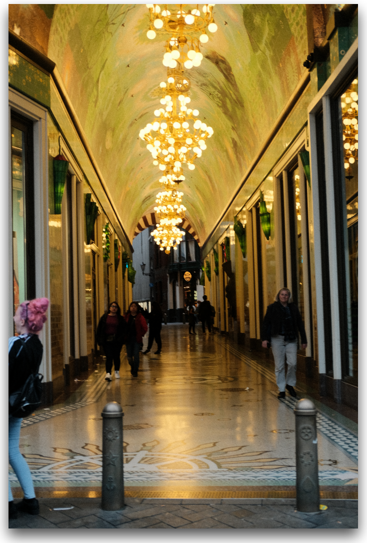
Ok, but on the other hand, if we don’t use palette-based editing framework, and just open up lightroom to edit lightness through tone curves.
For this example, if you wish to make some contrast over the light in this town hall, the edits would affect all over the places like the woman’s hair.
So you might consider masking things up, but if your masks are not properly applied, there might be artifacts like the incoherence between the unmasked and the masked region. Especially like this example, the light is scattered all around the town hall, so it’s very tedious to make masks over these.

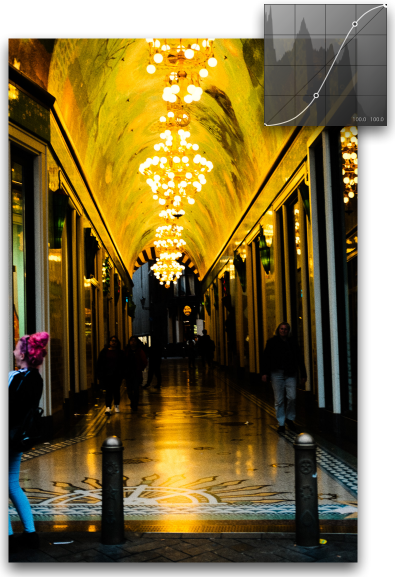
Tone curves aren’t local or color-aware



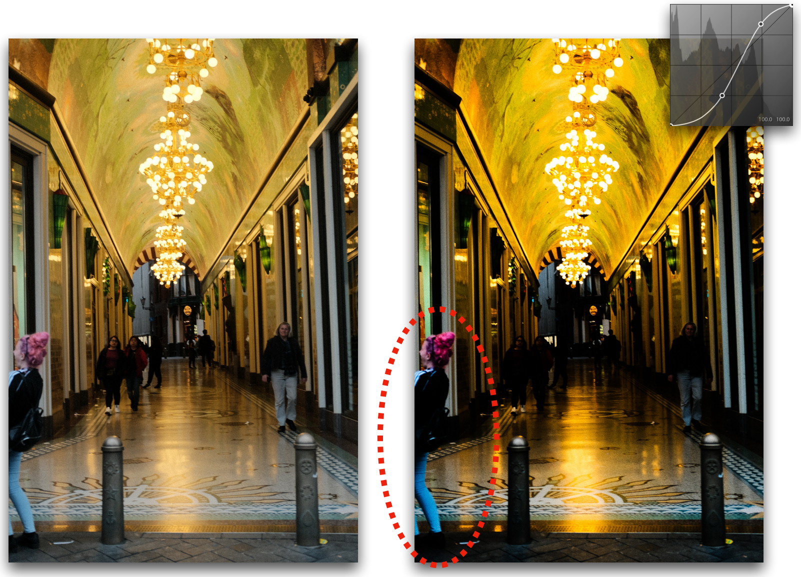


Tone curves aren’t local or color-aware

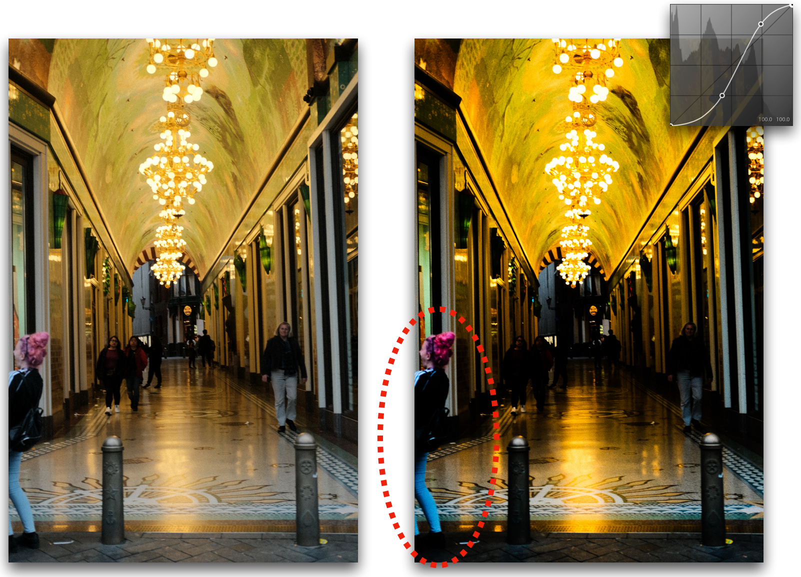




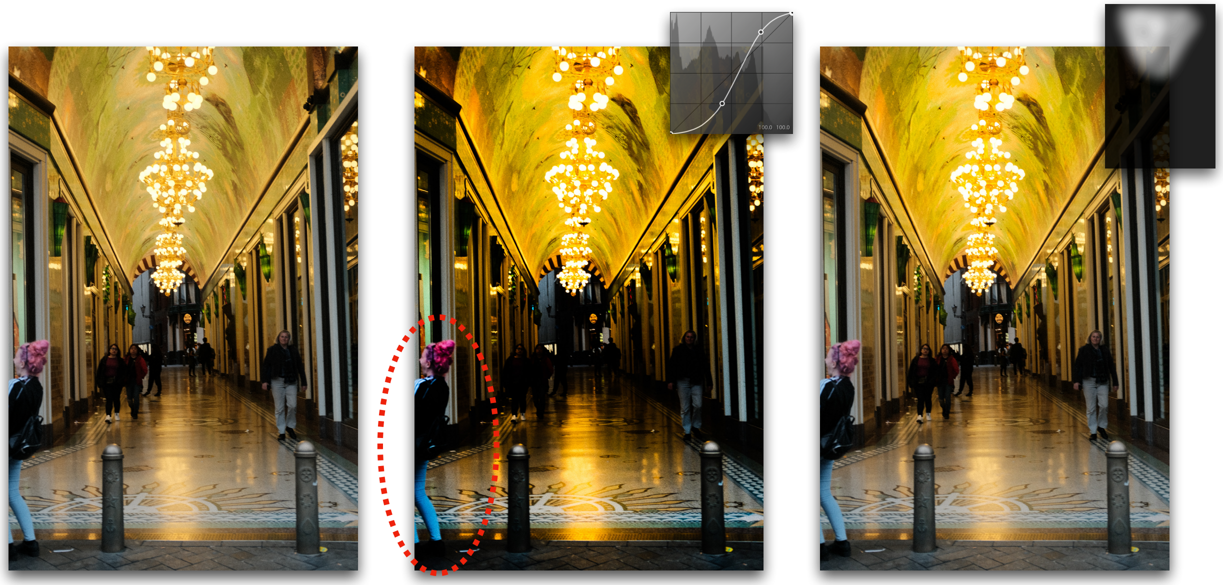
Tone curves aren’t local or color-aware



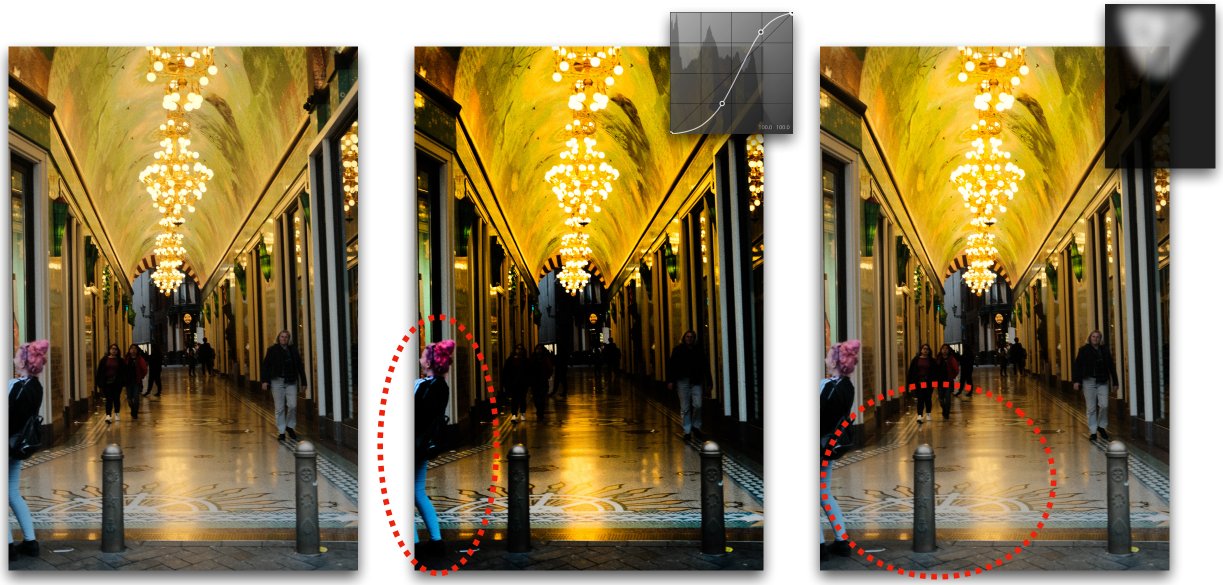
Tone curves aren’t local or color-aware



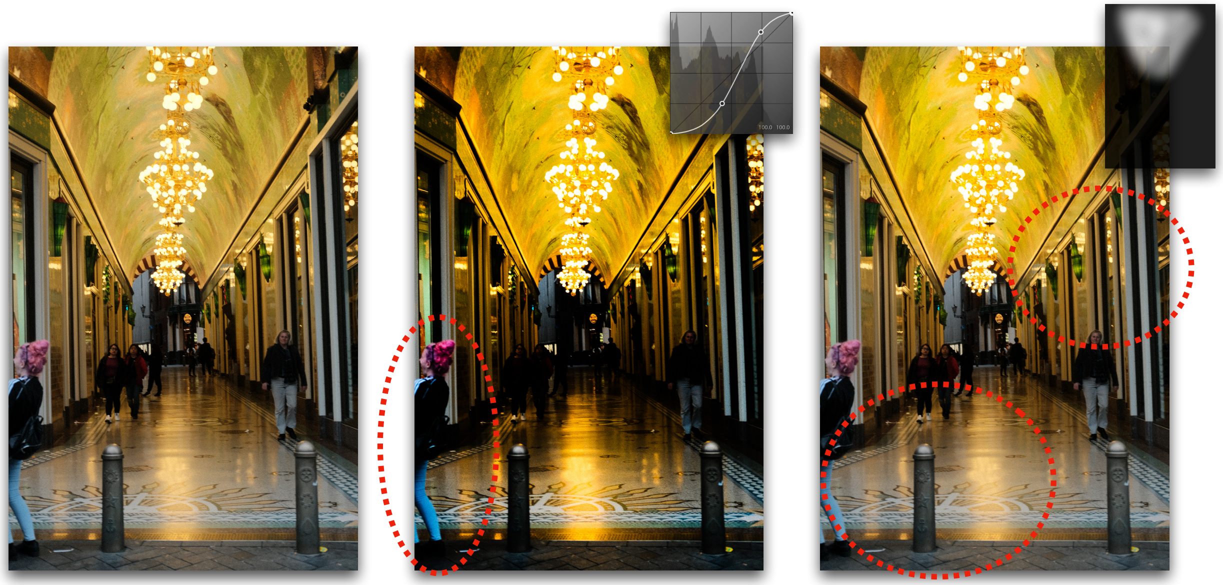
Tone curves aren’t local or color-aware

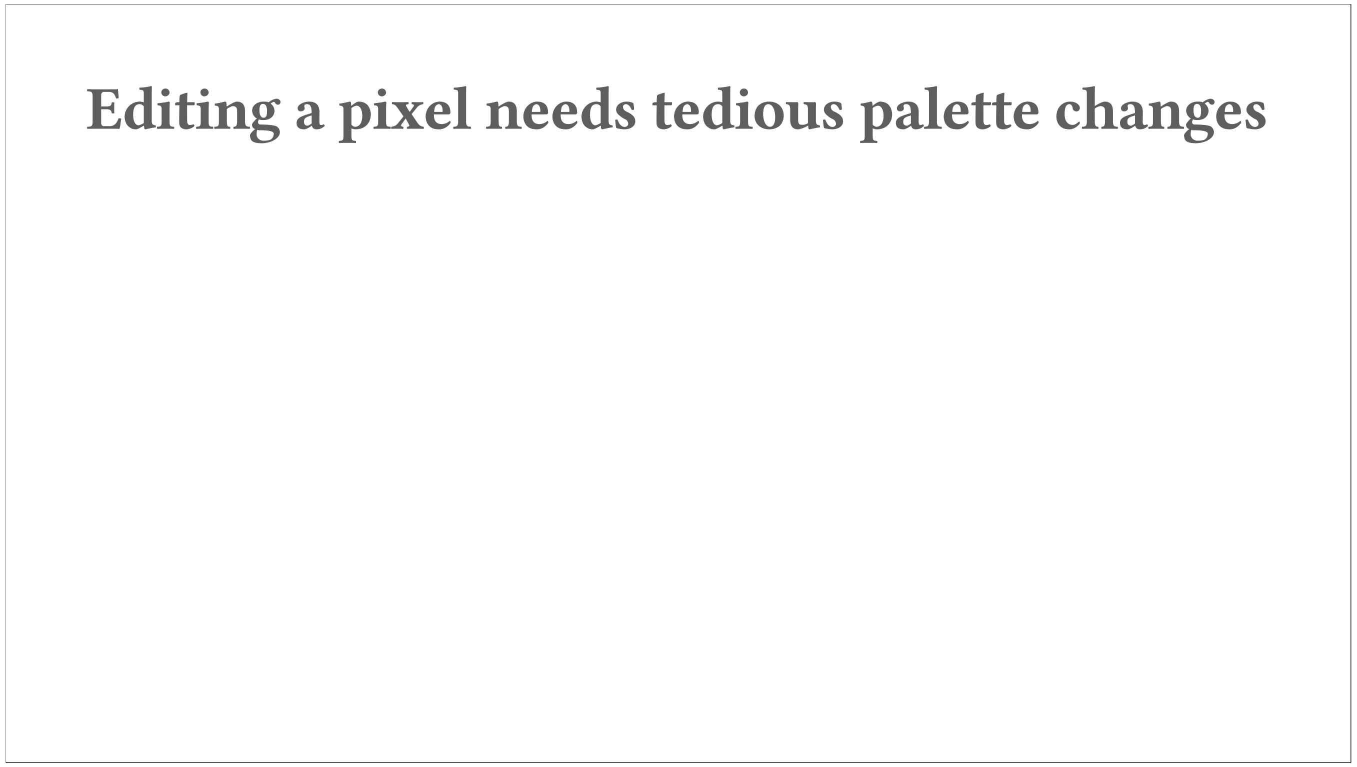


Tone curves aren’t local or color-aware



Tone curves aren’t local or color-aware





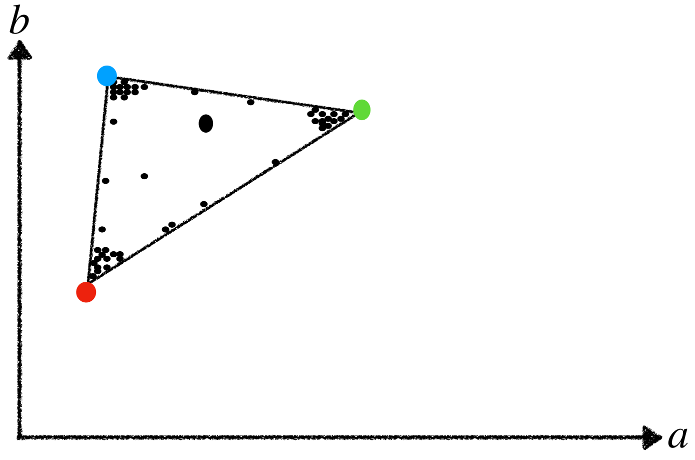
Editing a pixel needs tedious palette changes

(40s)

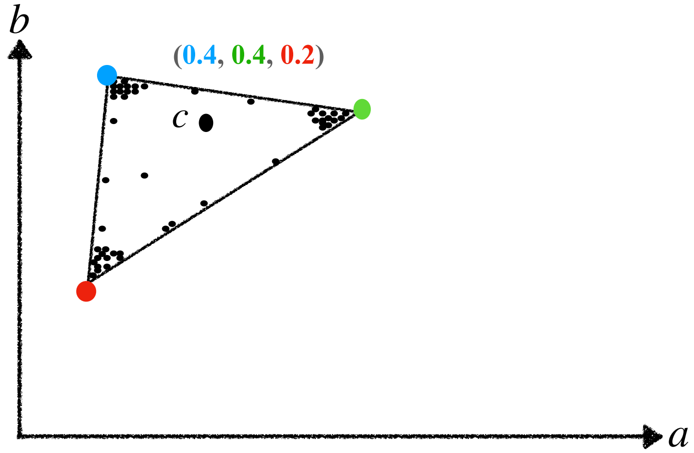
But, one common drawback of palette-based editing is that it’s incompatible with directly changing pixel colors.
For example, even in 2D, if you want to change this specific pixel color c with certain weights into another color c’, there are infinitely many solutions.
To be more specific, in practical editing on images, directly attempting to change this pixel color into another color would need to iterate palette many times since the color mixture of this pixel is non-obvious among the extracted palette.

Editing a pixel needs tedious palette changes

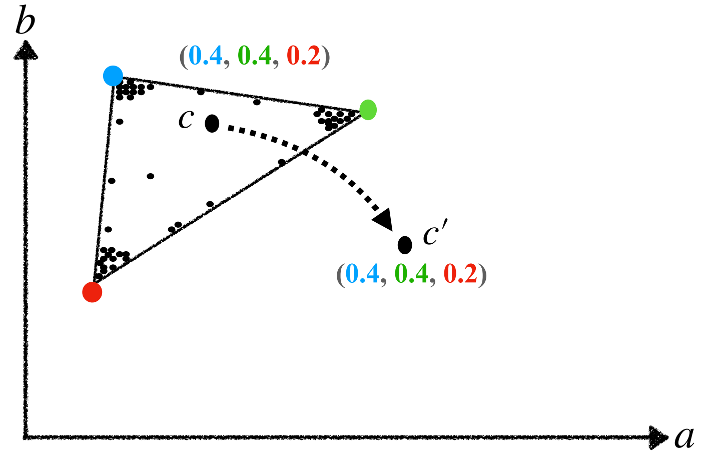
Editing a pixel needs tedious palette changes



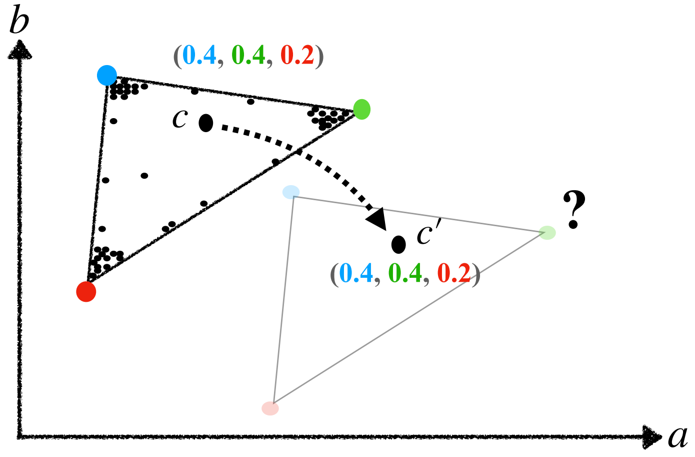
Editing a pixel needs tedious palette changes



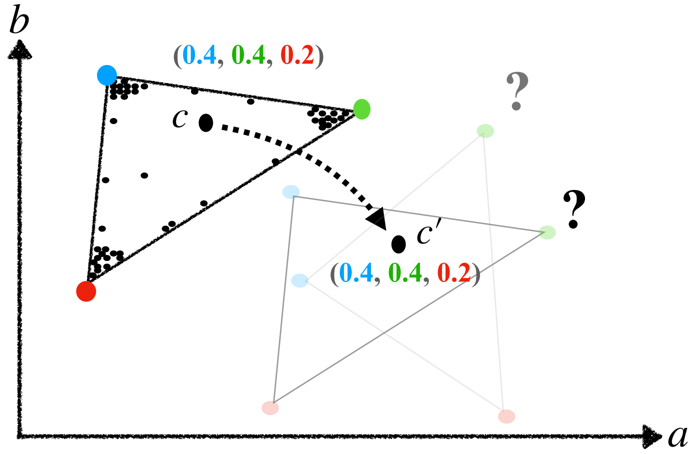
Editing a pixel needs tedious palette changes



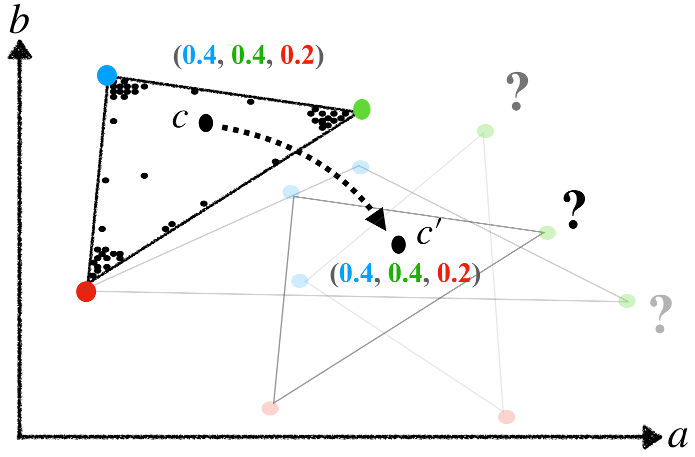
Editing a pixel needs tedious palette changes



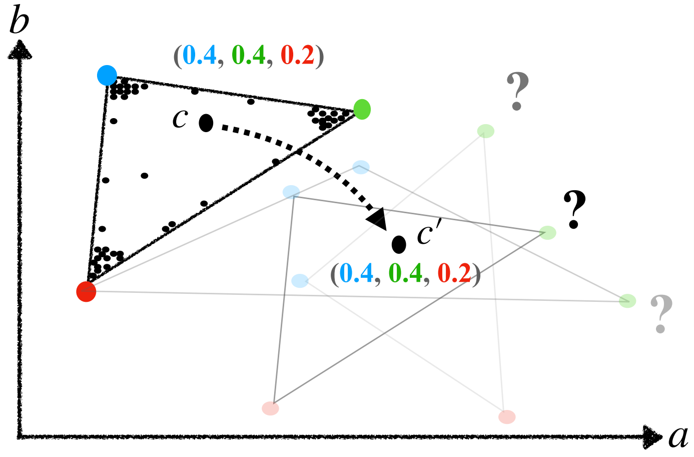
Editing a pixel needs tedious palette changes



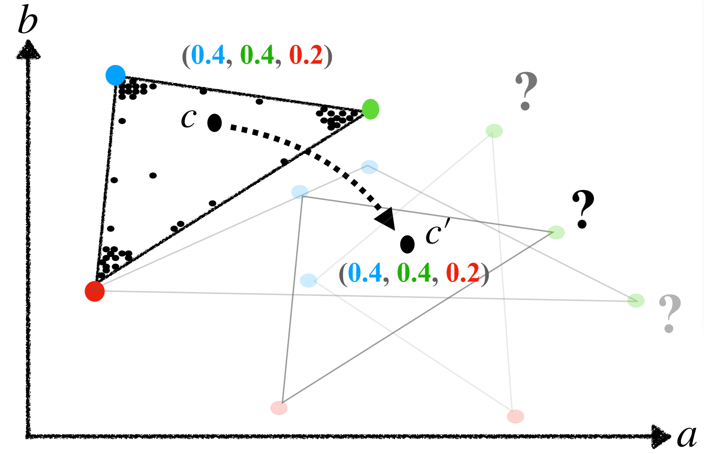
Editing a pixel needs tedious palette changes



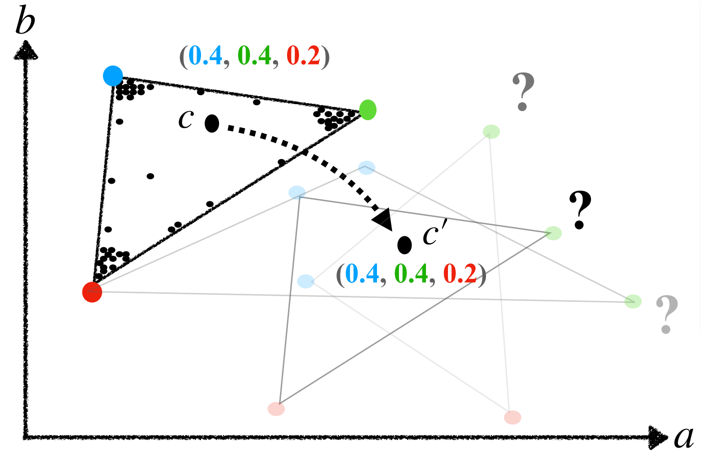
Editing a pixel needs tedious palette changes

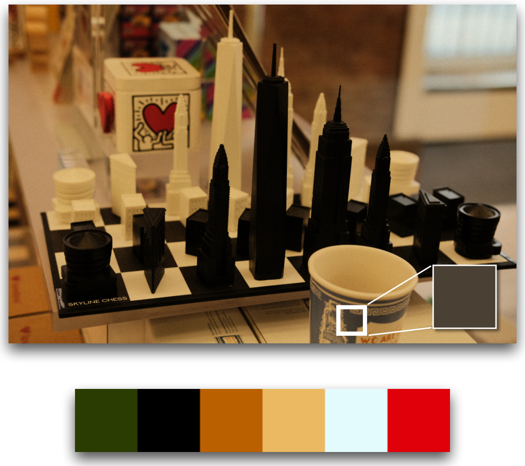


Editing a pixel needs tedious palette changes

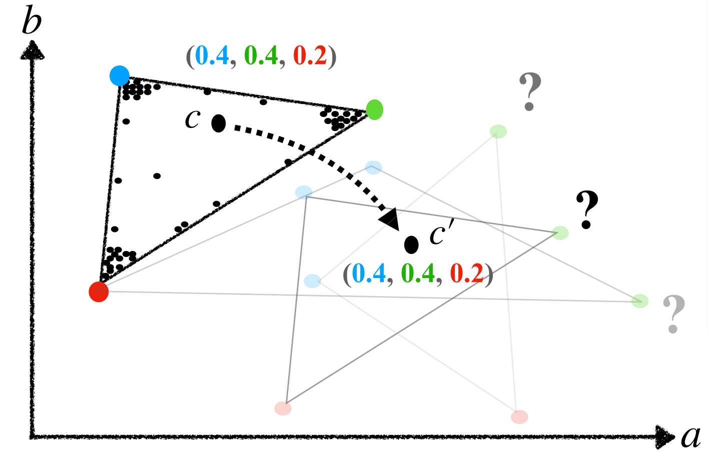


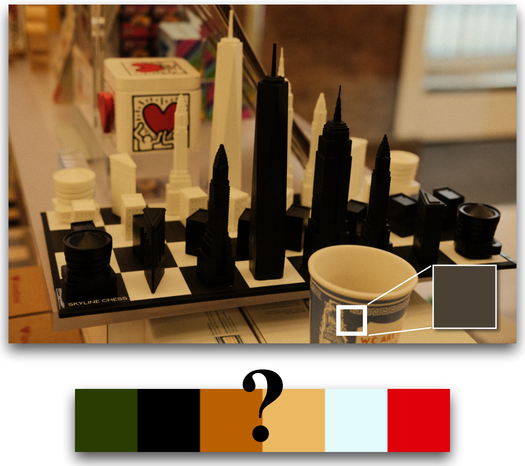
Editing a pixel needs tedious palette changes





Editing a pixel needs tedious palette changes





High-level summary

(35s)

So, in short, we know that palette-based approach provides spatially coherent color edits without manually masking, but the lightness is hard to control and it’s incompatible with directly changing pixel colors. Tone curves edit provides good control on the lightness, but it needs manually masking.
Our approach unites palette-based with tone curves editing to provide flexible control on both color and lightness, and we do this by…

High-level summary

Palette-based editing

✓ **Spatially coherent color edits**

High-level summary

Palette-based editing

✓ Spatially coherent color edits

✗ Hard to control lightness

High-level summary

Palette-based editing

- ✓ Spatially coherent color edits
- ✗ Hard to control lightness
- ✗ Incompatible with directly changing pixel colors

High-level summary

Palette-based editing

- ✓ Spatially coherent color edits
- ✗ Hard to control lightness
- ✗ Incompatible with directly changing pixel colors

Tone curve editing

- ✓ Easy to control lightness

High-level summary

Palette-based editing

- ✓ Spatially coherent color edits
- ✗ Hard to control lightness
- ✗ Incompatible with directly changing pixel colors

Tone curve editing

- ✓ Easy to control lightness
- ✗ Need manual masking

High-level summary

Palette-based editing	Tone curve editing
✓ Spatially coherent color edits	✓ Easy to control lightness

High-level summary

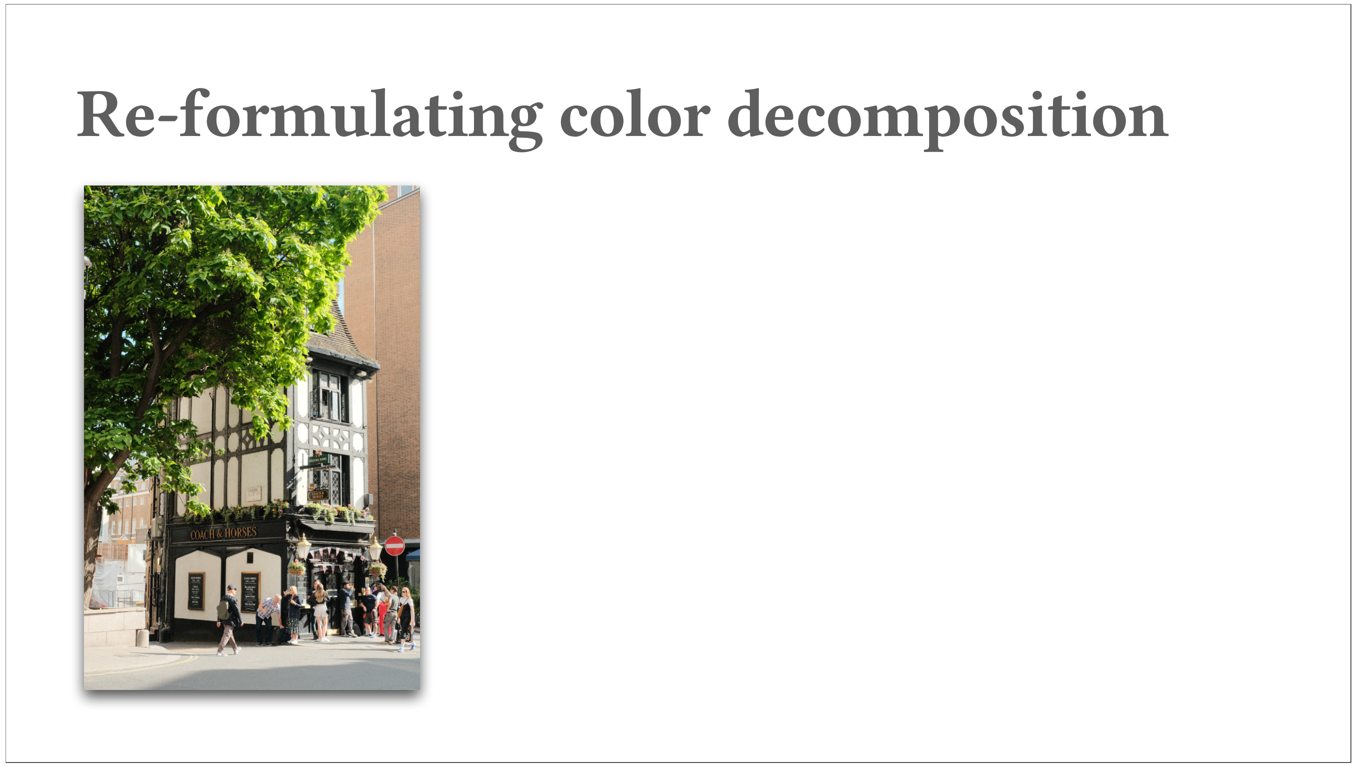
ColorfulCurves

Palette-based editing

+

Tone curve editing

- ✓ Easy to control lightness
- ✓ Spatially coherent color edits
- ✓ Compatible with directly changing pixel colors



Re-formulating color decomposition

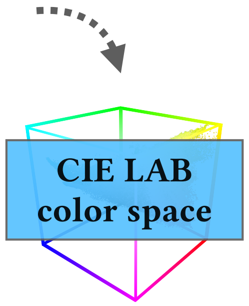
(80s)

Re-formulating the color decomposition.

Our work is in Lab space due to its natural disentanglement between “hue and chroma” and lightness.
We extract a 2D hue-chroma palette by computing simplified convex hull over ab-portion of the image. Here, we also concatenate a grey palette color which is (0,0) in ab-space to provide sparser color control. Then, we compute weights with respect to hue-chroma palette.
We then assign tone curve, which initially is a simple real-valued identity mapping, to each palette color, and I denote them as box symbols here but they are just identity functions mapping original lightness to themselves. In practice, we use biharmonic functions with symmetric boundary conditions on 0 and 1. Please see our paper for more detailed reasoning.

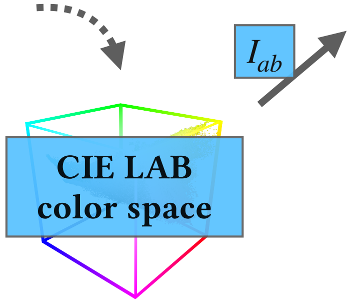
For the lightness portion, we down-scale the contribution from grey palette color in the hue-chroma weights and renormalize it. The renormalized weights then be used to blend the per-palette tone curves. Notice that this formulation perfectly reconstruct the image lightness since each pixel has a weight sum to 1 and per-palette tone curve initially is a identity function. So, given original image lightness as input, it just maps to the original lightness.

Re-formulating color decomposition



CIE LAB
color space

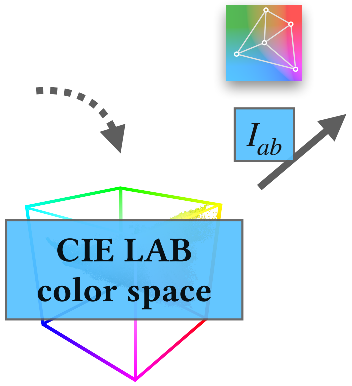
Re-formulating color decomposition



CIE LAB color space

I_{ab}

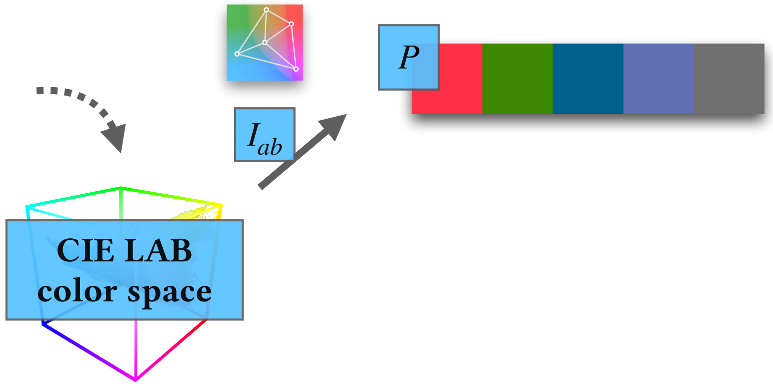
Re-formulating color decomposition



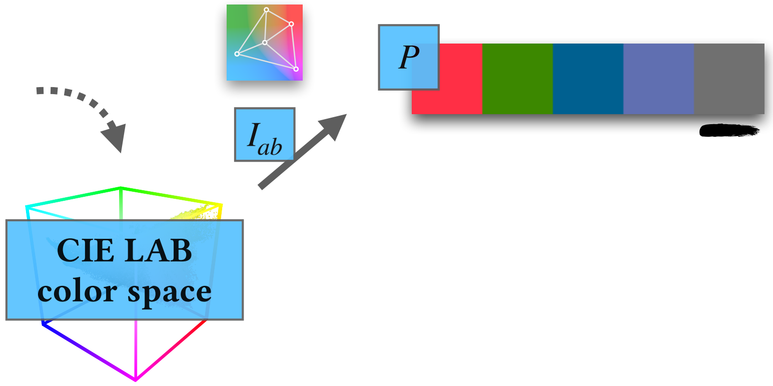
I_{ab}

CIE LAB color space

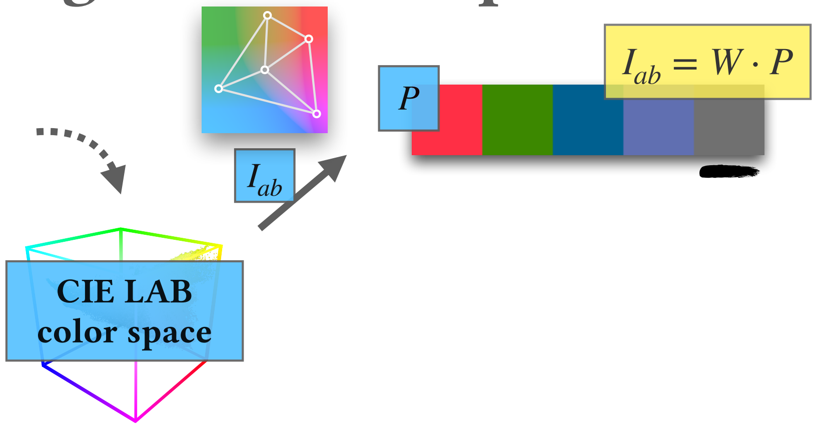
Re-formulating color decomposition



Re-formulating color decomposition



Re-formulating color decomposition



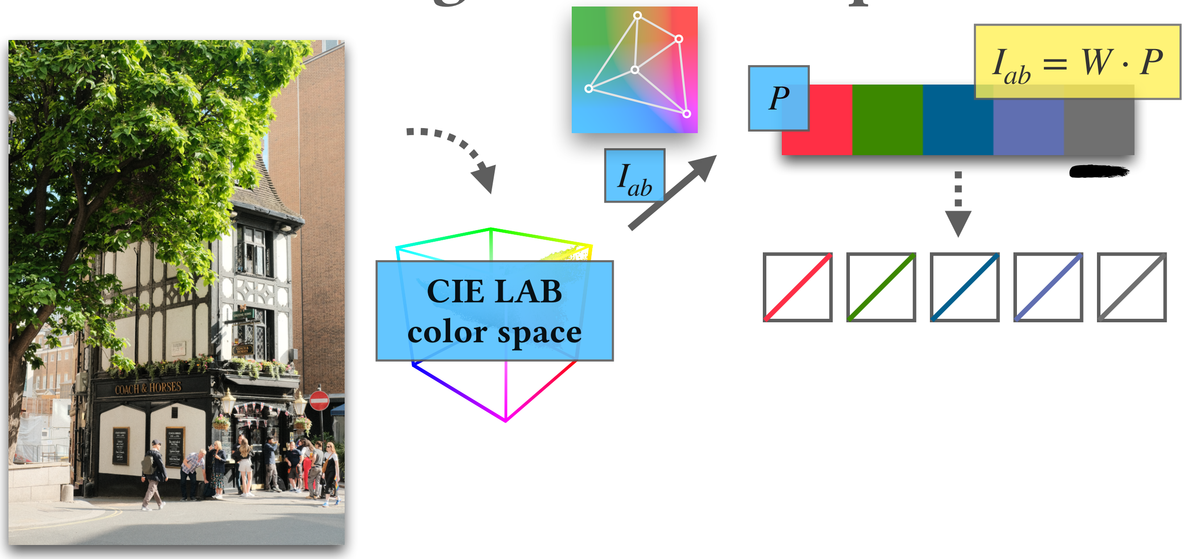
CIE LAB color space

I_{ab}

P

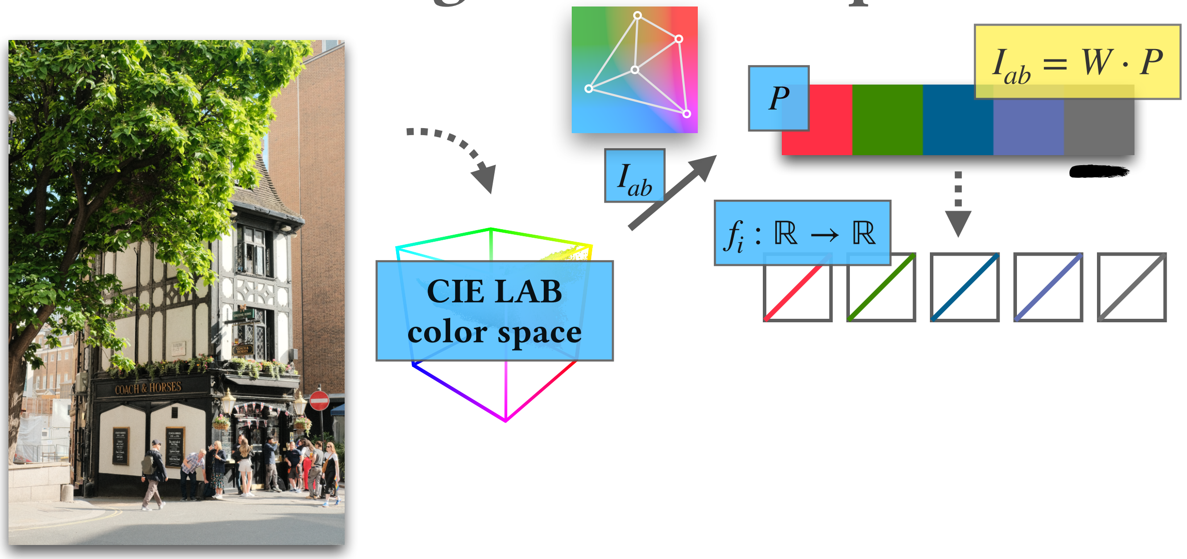
$I_{ab} = W \cdot P$

Re-formulating color decomposition



$I_{ab} = W \cdot P$

Re-formulating color decomposition



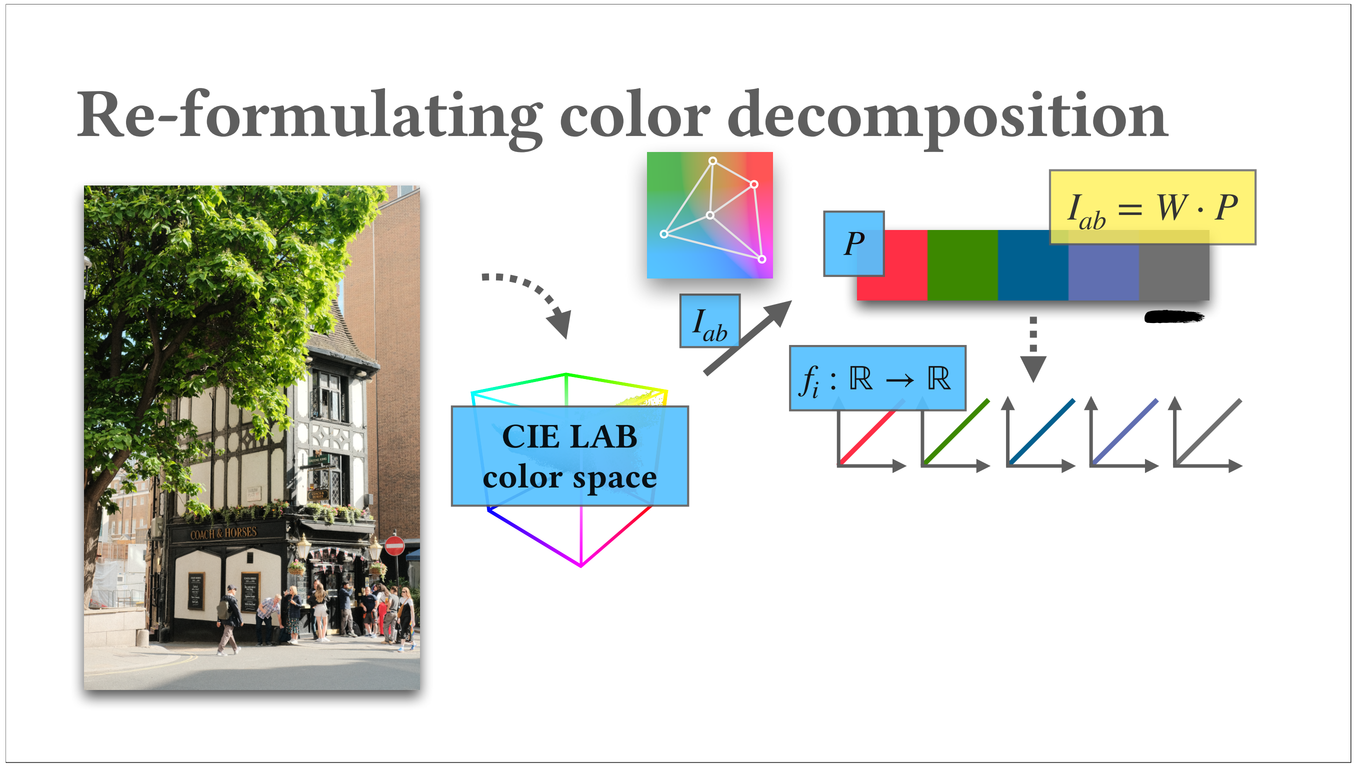
CIE LAB color space

I_{ab}

P

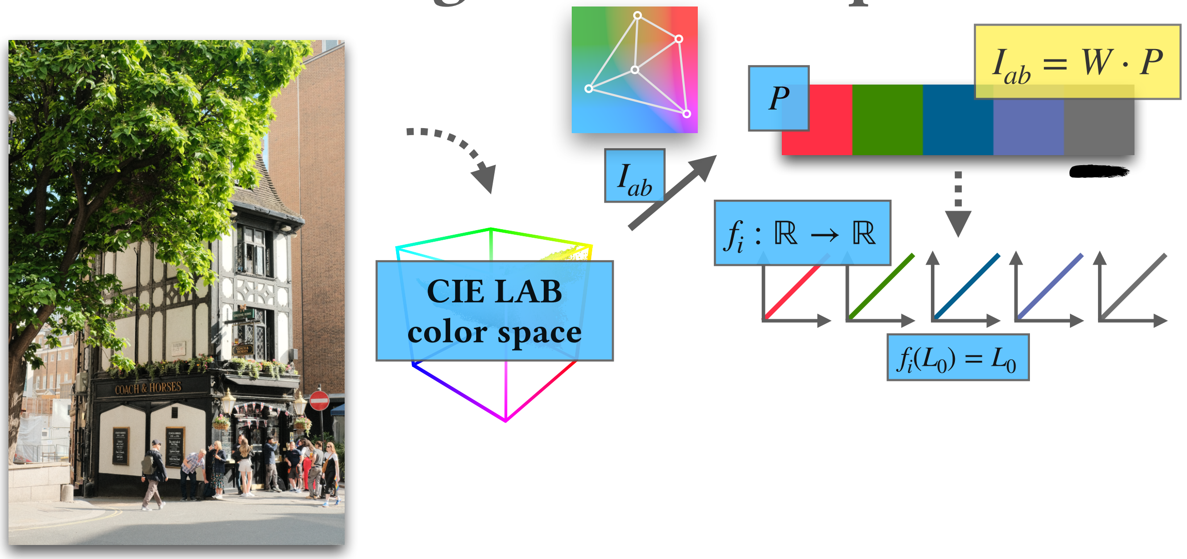
$I_{ab} = W \cdot P$

$f_i : \mathbb{R} \to \mathbb{R}$



Re-formulating color decomposition

Re-formulating color decomposition



CIE LAB color space

I_{ab}

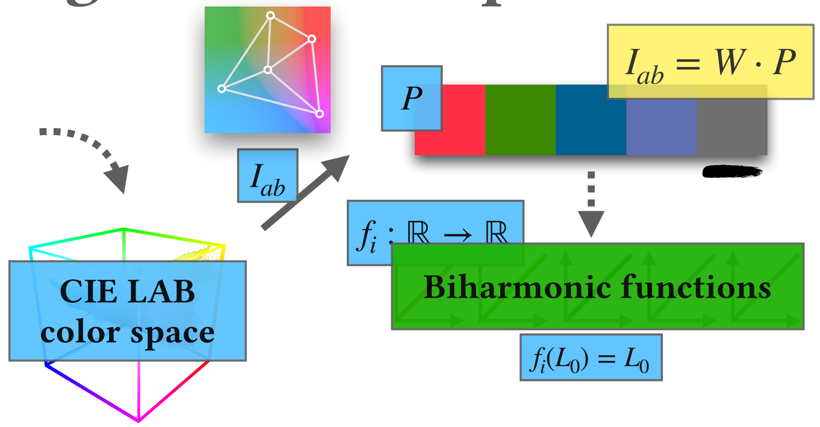
P

$I_{ab} = W \cdot P$

$f_i : \mathbb{R} \to \mathbb{R}$

$f_i(L_0) = L_0$

Re-formulating color decomposition



CIE LAB color space

I_{ab}

P

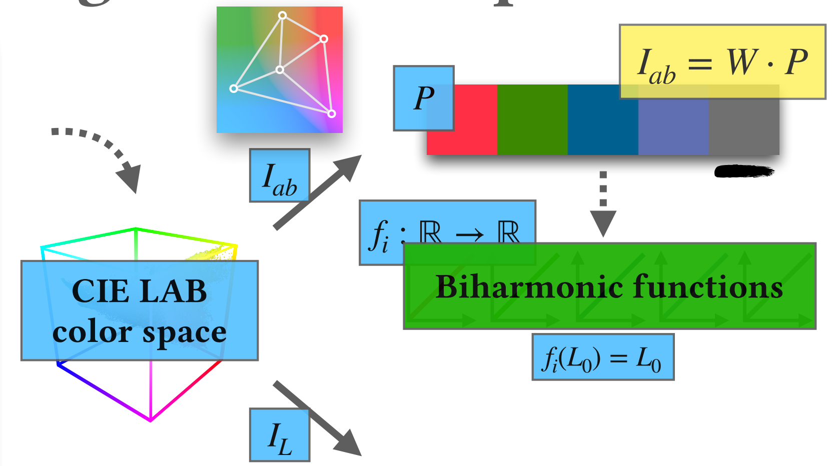
$I_{ab} = W \cdot P$

$f_i : \mathbb{R} \to \mathbb{R}$

Biharmonic functions

$f_i(L_0) = L_0$

Re-formulating color decomposition



$I_{ab} = W \cdot P$

P

I_{ab}

CIE LAB color space

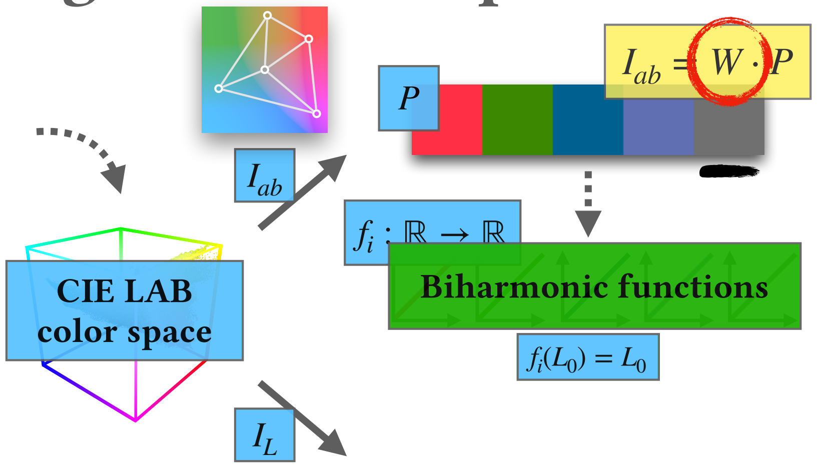
$f_i : \mathbb{R} \to \mathbb{R}$

Biharmonic functions

$f_i(L_0) = L_0$

I_L

Re-formulating color decomposition



$I_{ab} = W \cdot P$

P

I_{ab}

CIE LAB color space

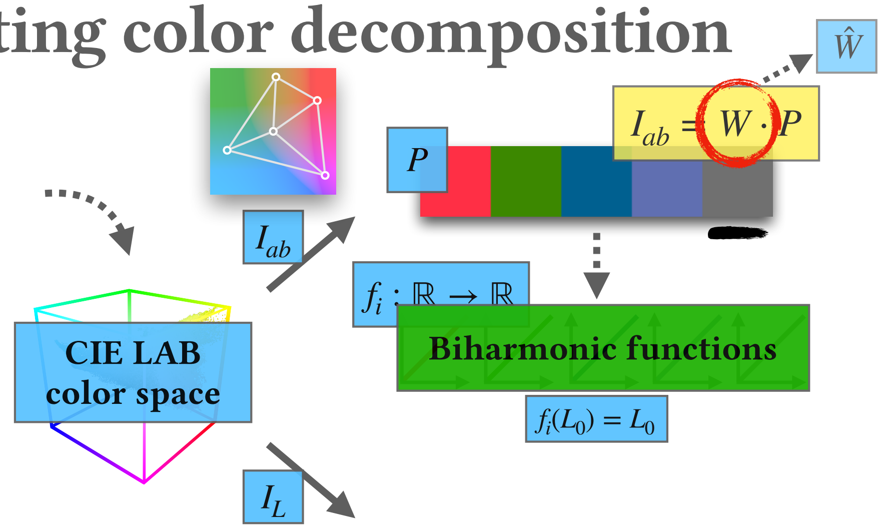
$f_i : \mathbb{R} \to \mathbb{R}$

Biharmonic functions

$f_i(L_0) = L_0$

I_L

Re-formulating color decomposition



$\hat{W}$

$I_{ab} = W \cdot P$

P

I_{ab}

$f_i : \mathbb{R} \to \mathbb{R}$

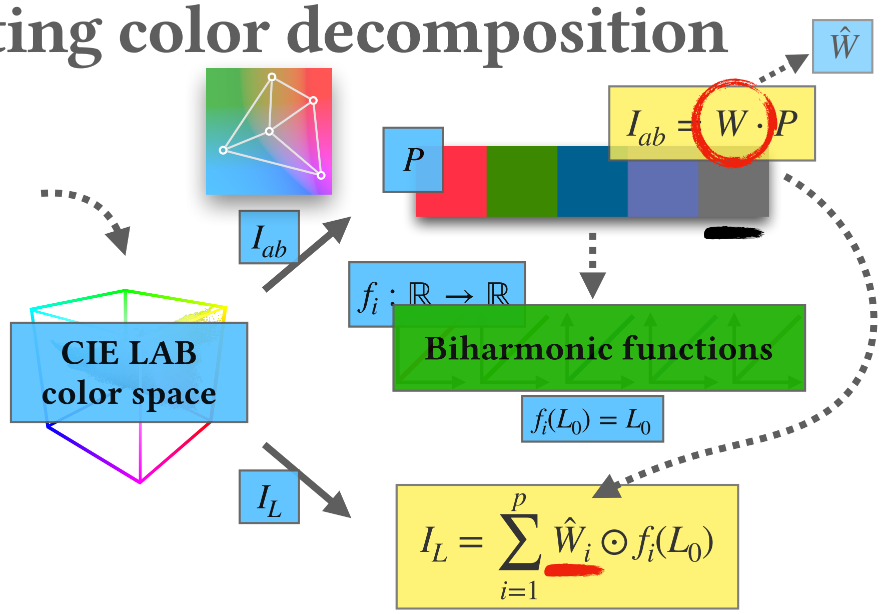
CIE LAB color space

Biharmonic functions

$f_i(L_0) = L_0$

I_L

Re-formulating color decomposition



$\hat{W}$

$I_{ab} = W \cdot P$

P

I_{ab}

CIE LAB color space

$f_i : \mathbb{R} \to \mathbb{R}$

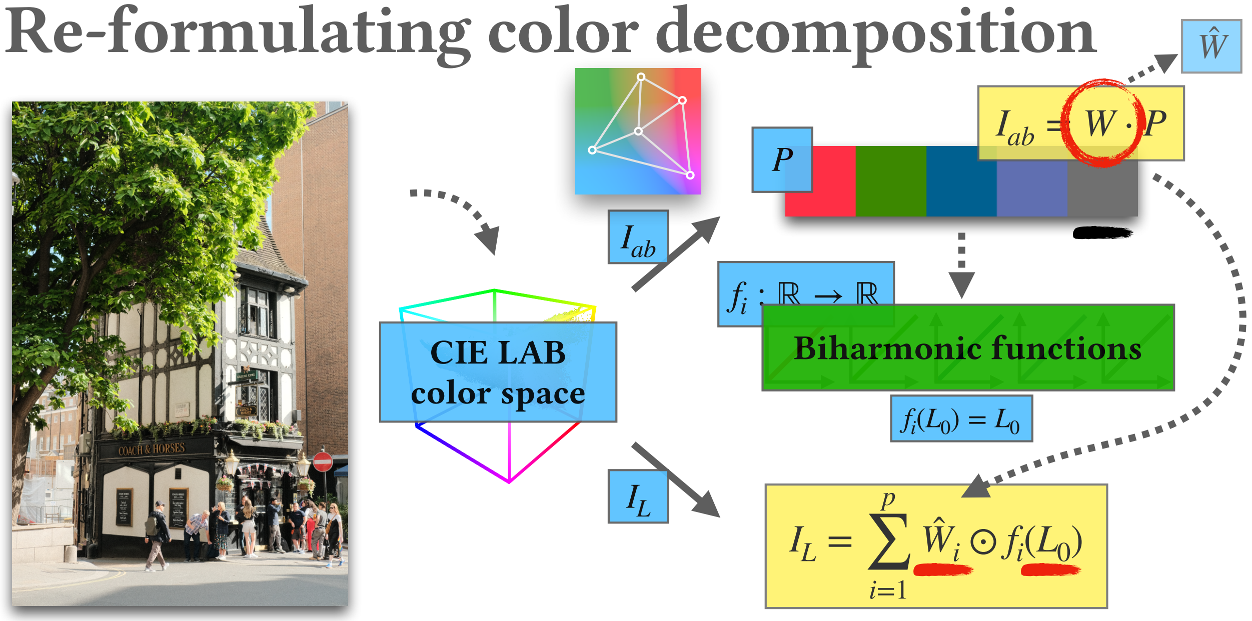
Biharmonic functions

$f_i(L_0) = L_0$

I_L

$$I_L = \sum_{i=1}^{p} \hat{W}_i \odot f_i(L_0)$$

Re-formulating color decomposition



CIE LAB color space

I_{ab}

I_L

P

$$I_{ab} = W \cdot P$$

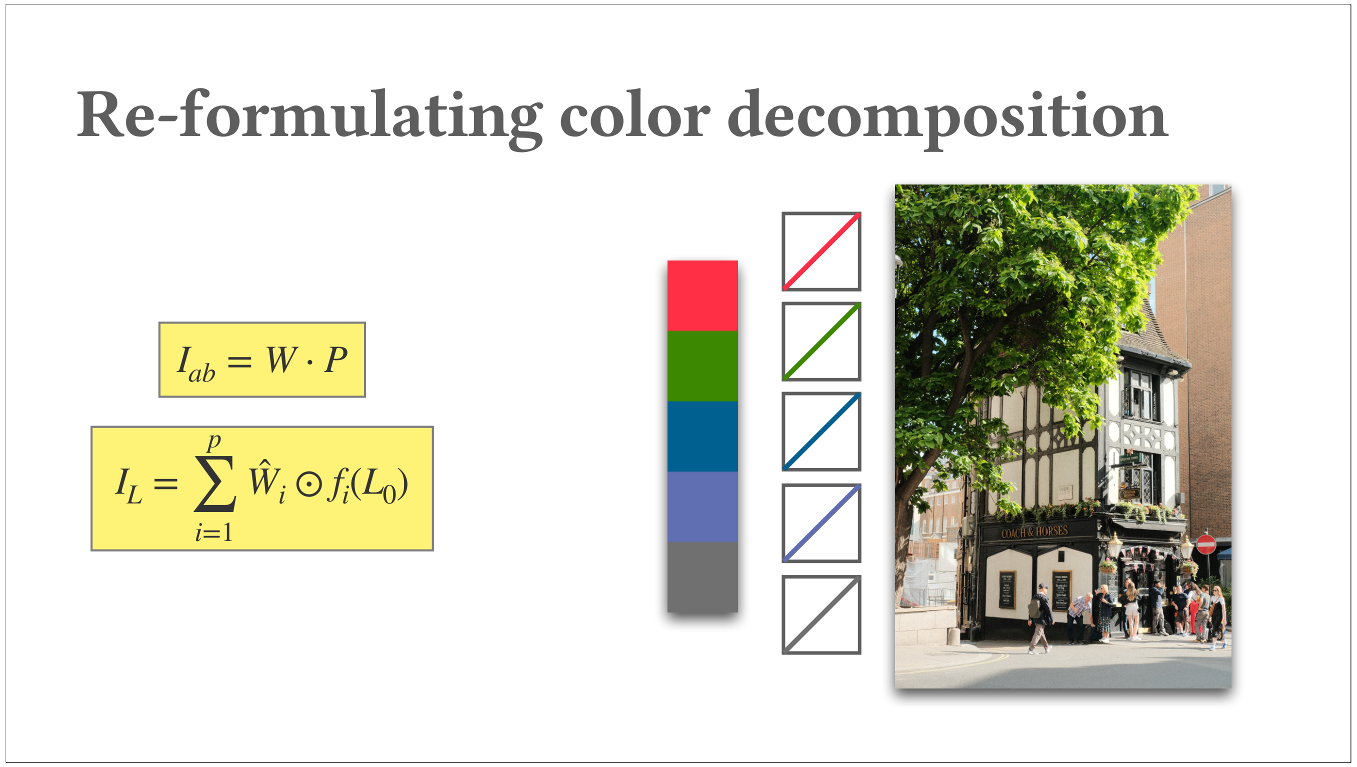
$\hat{W}$

$f_i : \mathbb{R} \to \mathbb{R}$

Biharmonic functions

$f_i(L_0) = L_0$

$$I_L = \sum_{i=1}^{p} \hat{W}_i \odot f_i(L_0)$$



Re-formulating color decomposition

$$I_{ab} = W \cdot P$$

$$I_L = \sum_{i=1}^{p} \hat{W}_i \odot f_i(L_0)$$

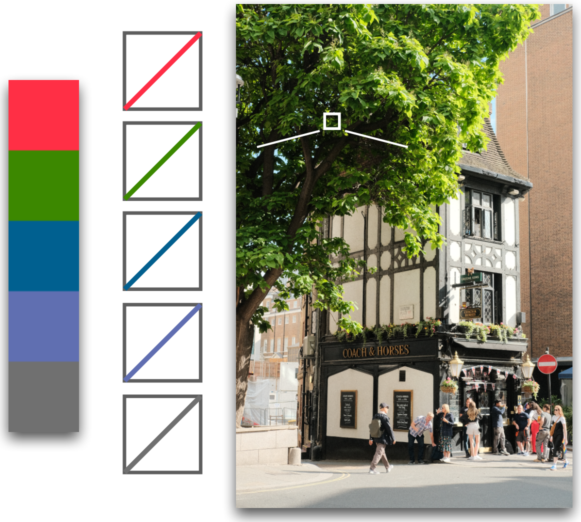
(18s)

Under these formulations, for any given image-space constraint, or palette constraint, we solve for the best palette and a set of tone curve functions to satisfy all these constraints by measuring their sparsest changes.
And by fixing the weights, we can edit the image into new beautiful one.

Re-formulating color decomposition

$$I_{ab} = W \cdot P$$

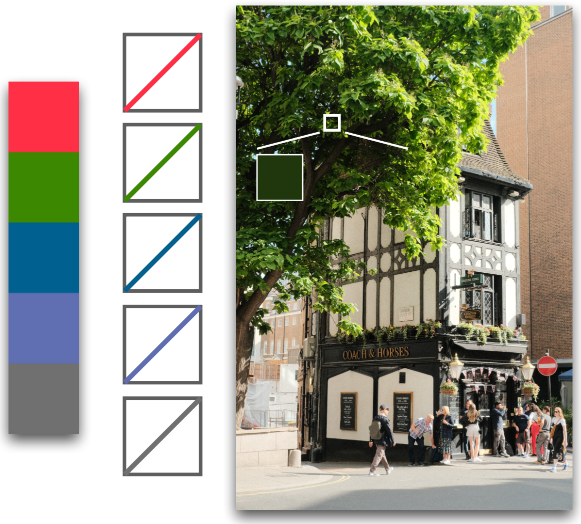
$$I_L = \sum_{i=1}^{p} \hat{W}_i \odot f_i(L_0)$$



Re-formulating color decomposition

$$I_{ab} = W \cdot P$$

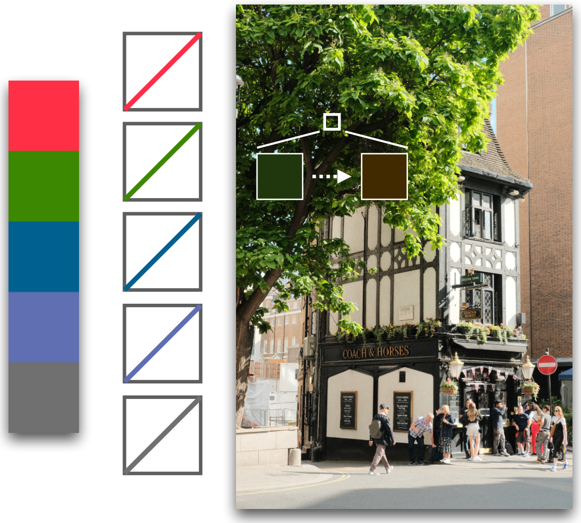
$$I_L = \sum_{i=1}^{p} \hat{W}_i \odot f_i(L_0)$$



Re-formulating color decomposition

$$\boxed{I_{ab} = W \cdot P}$$

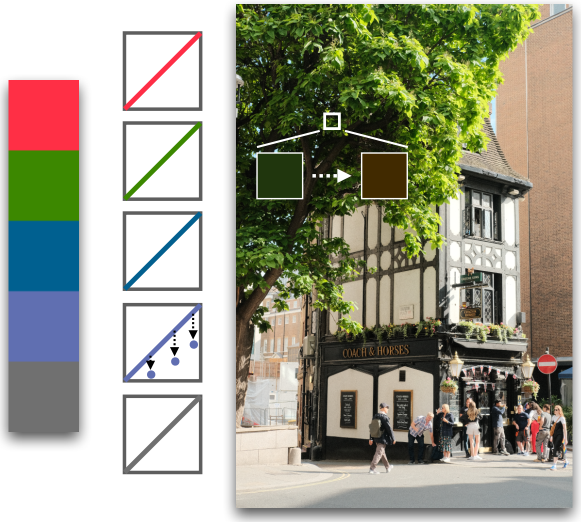
$$\boxed{I_L = \sum_{i=1}^{p} \hat{W}_i \odot f_i(L_0)}$$



Re-formulating color decomposition

$$I_{ab} = W \cdot P$$

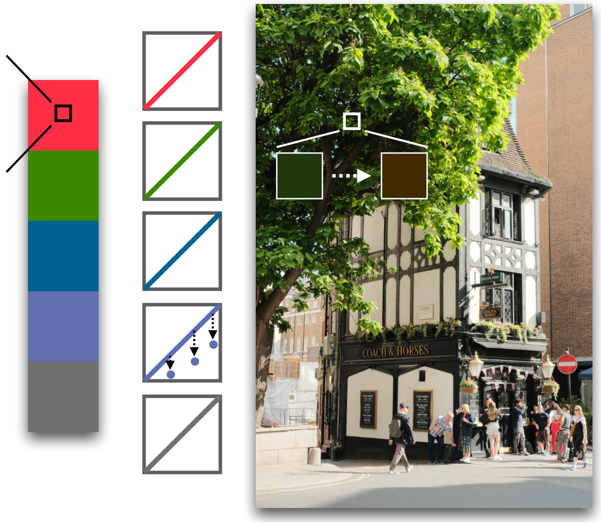
$$I_L = \sum_{i=1}^{p} \hat{W}_i \odot f_i(L_0)$$



Re-formulating color decomposition

$$I_{ab} = W \cdot P$$

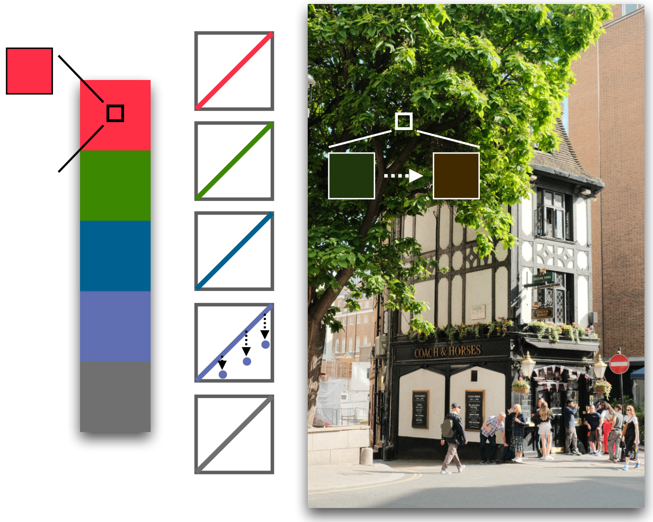
$$I_L = \sum_{i=1}^{p} \hat{W}_i \odot f_i(L_0)$$



Re-formulating color decomposition

$$I_{ab} = W \cdot P$$

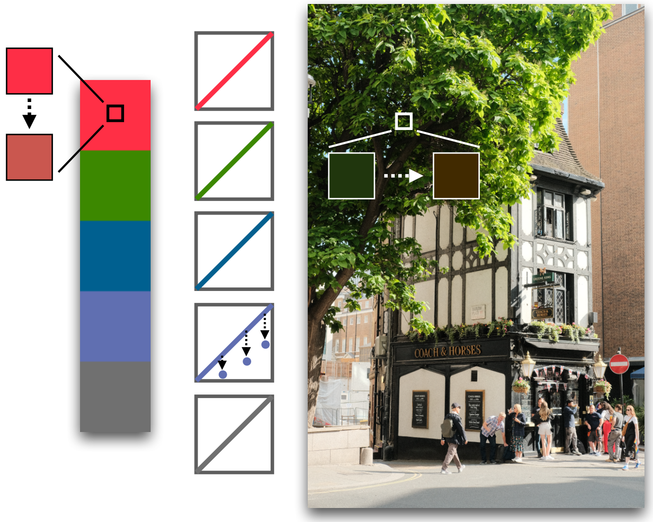
$$I_L = \sum_{i=1}^{p} \hat{W}_i \odot f_i(L_0)$$



Re-formulating color decomposition

$$\boxed{I_{ab} = W \cdot P}$$

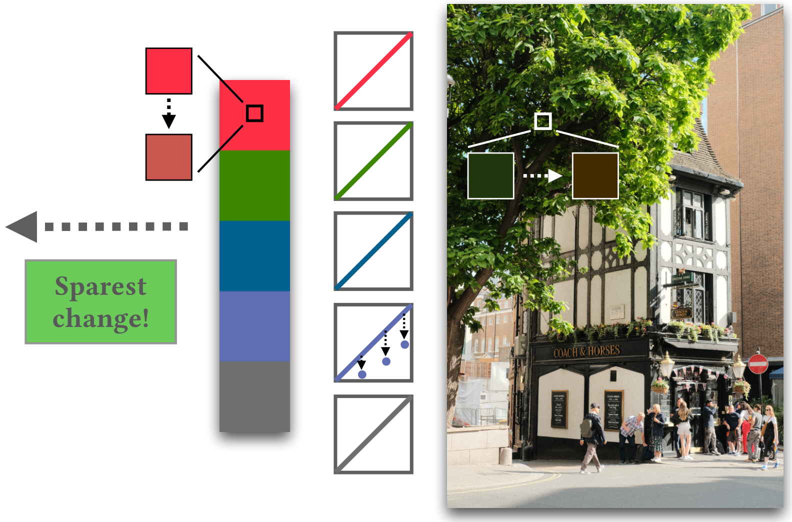
$$\boxed{I_L = \sum_{i=1}^{p} \hat{W}_i \odot f_i(L_0)}$$



Re-formulating color decomposition

$$I_{ab} = W \cdot P^*$$

$$I_L = \sum_{i=1}^{p} \hat{W}_i \odot f_i^*(L_0)$$

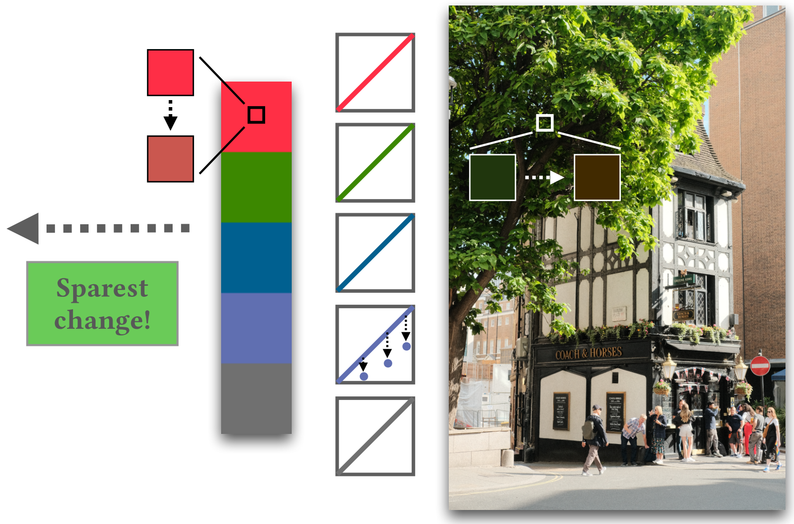


Sparest change!

Re-formulating color decomposition

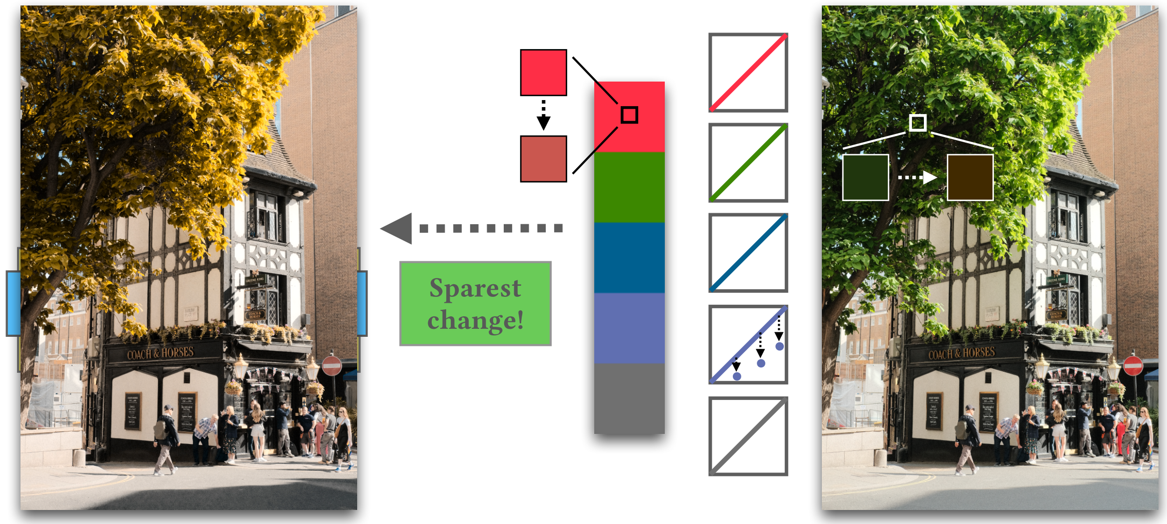
$$I^*_{ab} = W \cdot P^*$$

$$I^*_L = \sum_{i=1}^{p} \hat{W}_i \odot f^*_i(L_0)$$

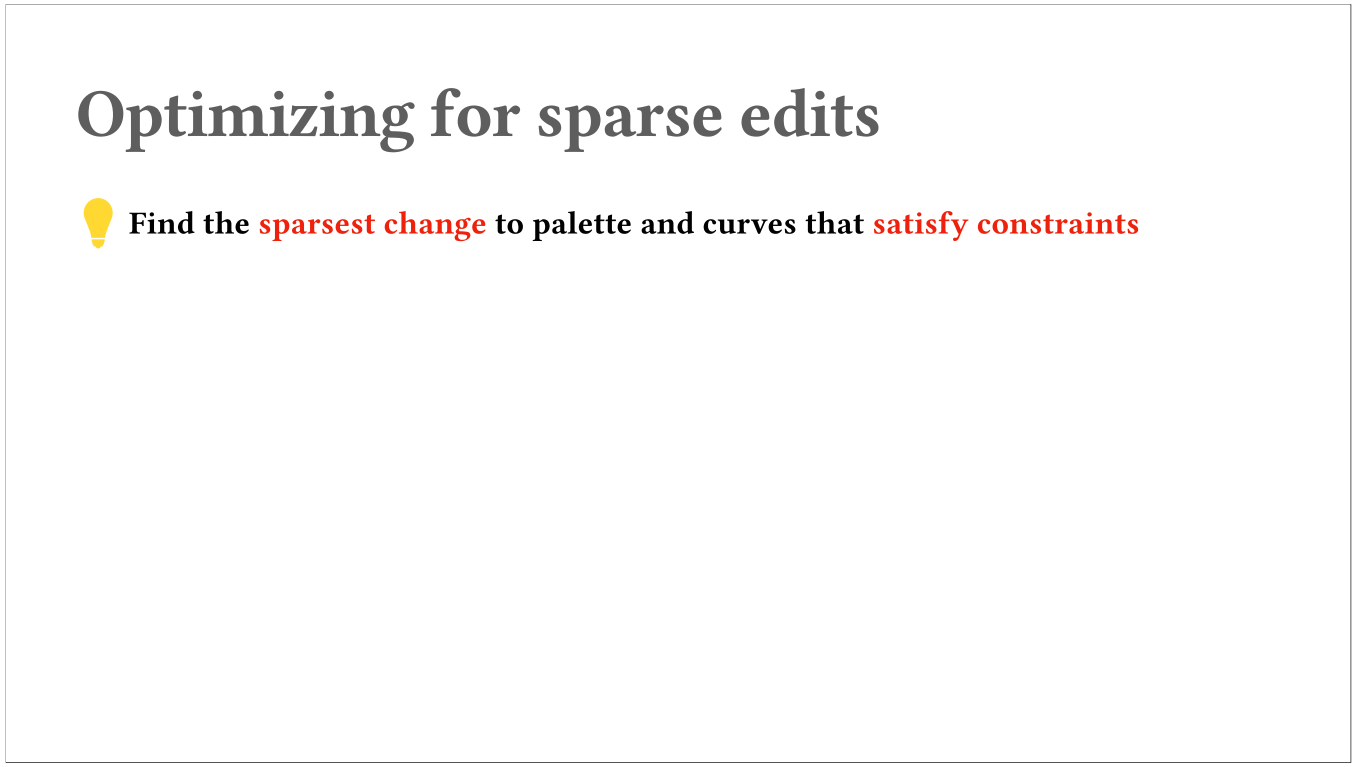


Sparest change!

Re-formulating color decomposition



Sparest change!



Optimizing for sparse edits

💡 **Find the sparsest change to palette and curves that satisfy constraints**

(40s)

So our goal is to find the sparsest change to palette and tone curves that satisfy all constraints.
The sparsity loss can be measured by this equation, which is an L21 sparsity loss of palette and tone curves.
For any given image-space constraint, the ab-portion of it along with any given palette constraint can be formulated as this loss.
Similarly, the lightness portion of the image-space constraint along with given lightness constraints can be formulated as this.
These three complicated-looking equations ……

(To-do: label the equation with colors corresponding to the text color)

Optimizing for sparse edits

💡 Find the sparsest change to palette and curves that satisfy constraints

Optimizing for sparse edits

💡 **Find the sparsest change to palette and curves that satisfy constraints**

$$E_{sp} = \sum_{i=1}^{p} \sqrt{L_i^T B^T B L_i + w_{sp} \cdot \|q_i \cdot \Delta P_{i,*}\|_2^2}$$

Optimizing for sparse edits

Find the sparsest change to palette and curves that satisfy constraints

$L_{2,1}$ norm

$$E_{sp} = \sum_{i=1}^{p} \sqrt{L_i^T B^T B L_i + w_{sp} \cdot \|q_i \cdot \Delta P_{i,*}\|_2^2}$$

Optimizing for sparse edits

- Find the sparsest change to palette and curves that satisfy constraints

$L_{2,1}$ norm

$$E_{sp} = \sum_{i=1}^{p} \sqrt{L_i^T B^T B L_i + w_{sp} \cdot \|q_i \cdot \Delta P_{i,*}\|_2^2}$$

Optimizing for sparse edits

- **Find the sparsest change to palette and curves that satisfy constraints**

$L_{2,1}$ norm

$$E_{sp} = \sum_{i=1}^{p} \sqrt{L_i^T B^T B L_i + w_{sp} \cdot \|q_i \cdot \Delta P_{i,*}\|_2^2}$$

Optimizing for sparse edits

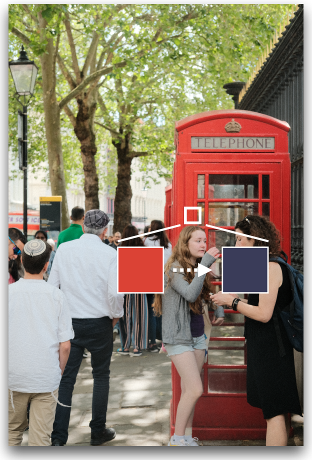
Find the sparsest change to palette and curves that satisfy constraints

L$_{2,1}$ norm

$$E_{sp} = \sum_{i=1}^{p} \sqrt{L_i^T B^T B L_i + w_{sp} \cdot \|q_i \cdot \Delta P_{i,*}\|_2^2}$$

Optimizing for sparse edits

💡 Find the sparsest change to palette and curves that satisfy constraints

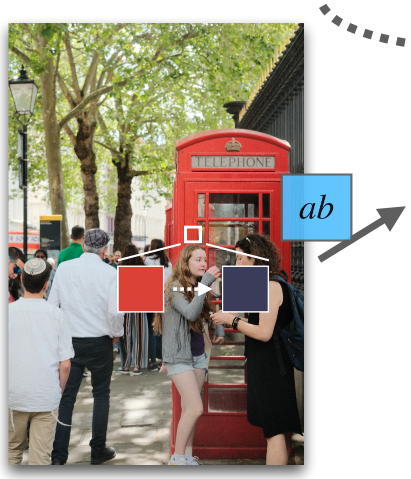


$L_{2,1}$ norm

$$E_{sp} = \sum_{i=1}^{p} \sqrt{L_i^T B^T B L_i + w_{sp} \cdot \|q_i \cdot \Delta P_{i,*}\|_2^2}$$

Optimizing for sparse edits

Find the sparsest change to palette and curves that satisfy constraints



L$_{2,1}$ norm

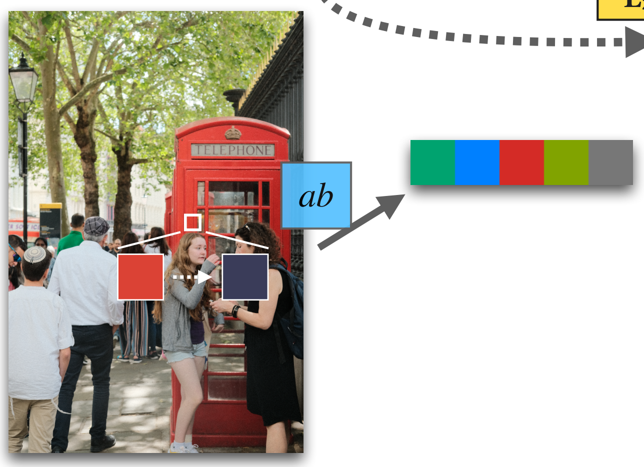
$$E_{sp} = \sum_{i=1}^{p} \sqrt{L_i^T B^T B L_i + w_{sp} \cdot \|q_i \cdot \Delta P_{i,*}\|_2^2}$$

Optimizing for sparse edits

💡 Find the sparsest change to palette and curves that satisfy constraints

L$_{2,1}$ norm

$$E_{sp} = \sum_{i=1}^{p} \sqrt{L_i^T B^T B L_i + w_{sp} \cdot \|q_i \cdot \Delta P_{i,*}\|_2^2}$$

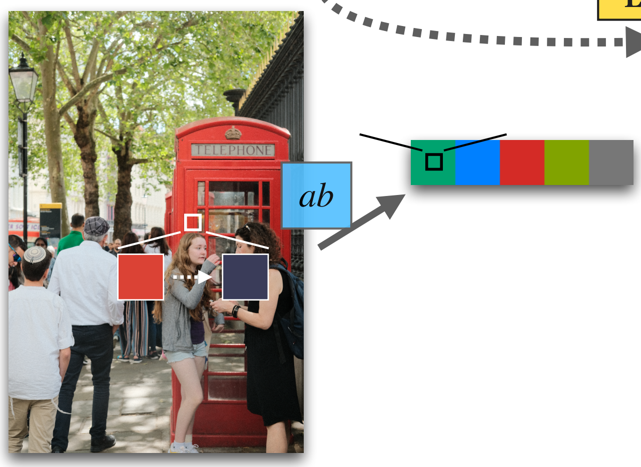


Optimizing for sparse edits

Find the sparsest change to palette and curves that satisfy constraints

L$_{2,1}$ norm

$$E_{sp} = \sum_{i=1}^{p} \sqrt{L_i^T B^T B L_i + w_{sp} \cdot \|q_i \cdot \Delta P_{i,*}\|_2^2}$$

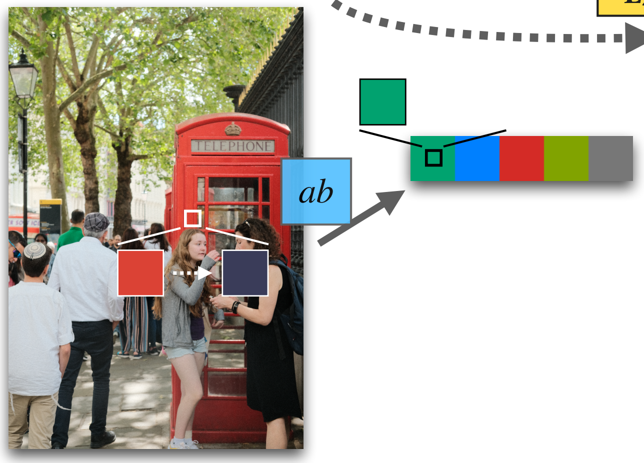


Optimizing for sparse edits

- Find the **sparsest change** to palette and curves that **satisfy constraints**

$L_{2,1}$ norm

$$E_{sp} = \sum_{i=1}^{p} \sqrt{L_i^T B^T B L_i + w_{sp} \cdot \|q_i \cdot \Delta P_{i,*}\|_2^2}$$

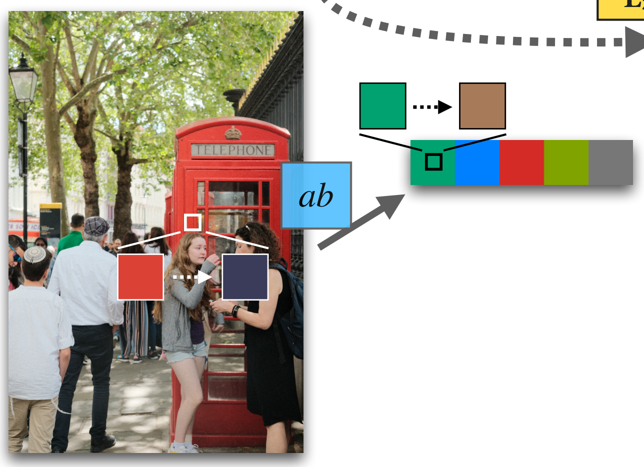


Optimizing for sparse edits

- Find the sparsest change to palette and curves that satisfy constraints

$L_{2,1}$ norm

$$E_{sp} = \sum_{i=1}^{p} \sqrt{L_i^T B^T B L_i + w_{sp} \cdot \|q_i \cdot \Delta P_{i,*}\|_2^2}$$

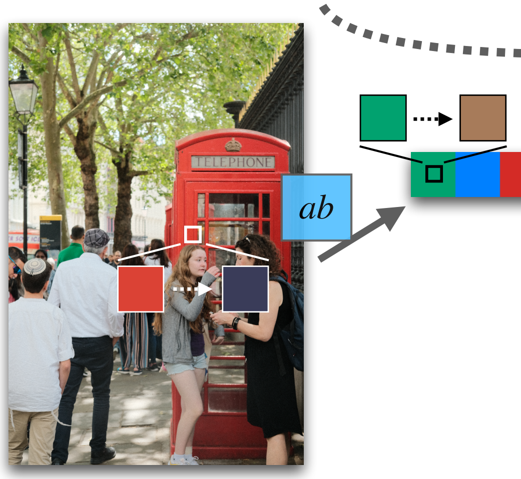


Optimizing for sparse edits

💡 **Find the sparsest change to palette and curves that satisfy constraints**

$L_{2,1}$ norm

$$E_{sp} = \sum_{i=1}^{p} \sqrt{L_i^T B^T B L_i + w_{sp} \cdot \|q_i \cdot \Delta P_{i,*}\|_2^2}$$



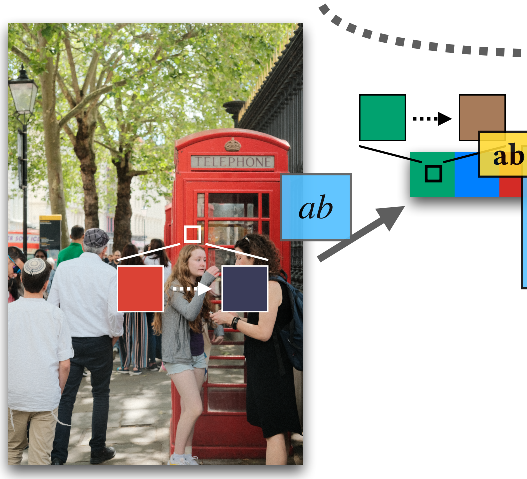
$$E_p = \sum_{j=1}^{c} \left\| \left(\sum_{i=1}^{p} \widetilde{W}_{ij}(P_{i,*} + \Delta P_{i,*}) \right) - \widehat{C}_{*,j} \right\|_2^2 + \sum_{i \in P^c} \left\| (P_{i,*} + \Delta P_{i,*}) - \widehat{P}_{i,*} \right\|_2^2$$

Optimizing for sparse edits

- Find the sparsest change to palette and curves that satisfy constraints

L$_{2,1}$ norm

$$E_{sp} = \sum_{i=1}^{p} \sqrt{L_i^T B^T B L_i + w_{sp} \cdot \|q_i \cdot \Delta P_{i,*}\|_2^2}$$

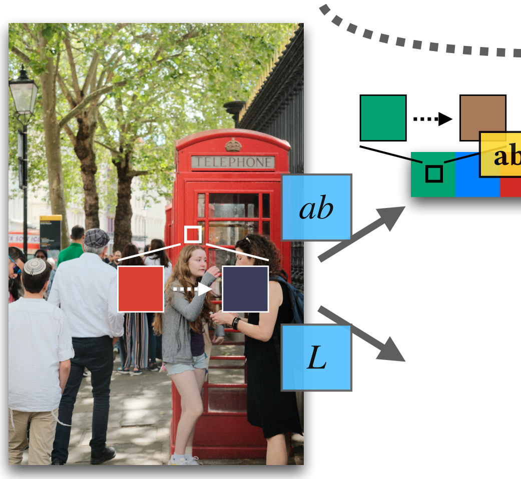


ab portion of the constraints

$$E_p = \sum_{j=1}^{c} \left\| \left(\sum_{i=1}^{p} \widetilde{W}_{ij}(P_{i,*} + \Delta P_{i,*}) \right) - \widehat{C}_{*,j} \right\|_2^2 + \sum_{i \in P^c} \left\| (P_{i,*} + \Delta P_{i,*}) - \widehat{P}_{i,*} \right\|_2^2$$

Optimizing for sparse edits

- Find the sparsest change to palette and curves that satisfy constraints



L

$\mathrm{L}_{2,1}$ norm

$$E_{sp} = \sum_{i=1}^{p} \sqrt{L_i^T B^T B L_i + w_{sp} \cdot \|q_i \cdot \Delta P_{i,*}\|_2^2}$$

ab portion of the constraints

$$E_p = \sum_{j=1}^{c} \left\| \left(\sum_{i=1}^{p} \widetilde{W}_{ij}(P_{i,*} + \Delta P_{i,*}) \right) - \widehat{C}_{*,j} \right\|_2^2 + \sum_{i \in P^c} \left\| (P_{i,*} + \Delta P_{i,*}) - \widehat{P}_{i,*} \right\|_2^2$$

Optimizing for sparse edits

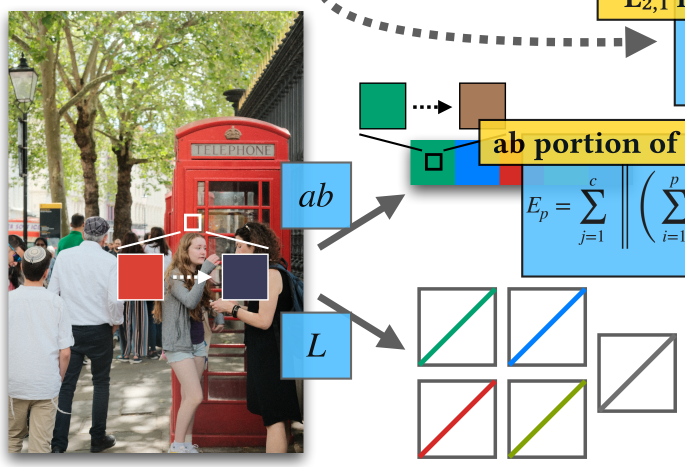
- Find the sparsest change to palette and curves that satisfy constraints

L$_{2,1}$ norm

$$E_{sp} = \sum_{i=1}^{p} \sqrt{L_i^T B^T B L_i + w_{sp} \cdot \|q_i \cdot \Delta P_{i,*}\|_2^2}$$

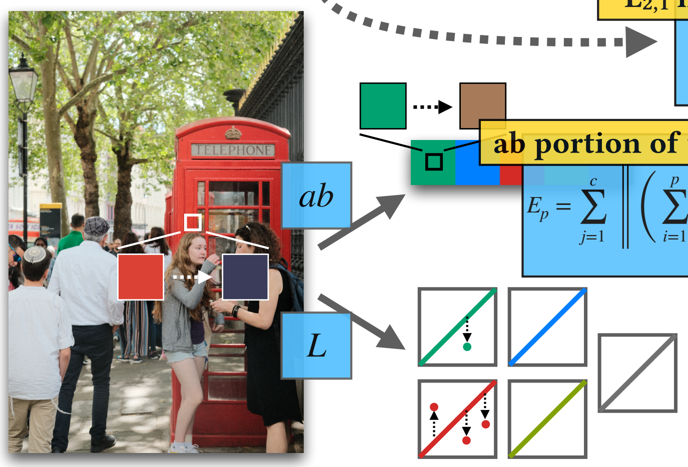
ab portion of the constraints

$$E_p = \sum_{j=1}^{c} \left\| \left(\sum_{i=1}^{p} \widetilde{W}_{ij}(P_{i,*} + \Delta P_{i,*}) \right) - \widehat{C}_{*,j} \right\|_2^2 + \sum_{i \in P^c} \left\| (P_{i,*} + \Delta P_{i,*}) - \widehat{P}_{i,*} \right\|_2^2$$



Optimizing for sparse edits

💡 **Find the sparsest change to palette and curves that satisfy constraints**



L$_{2,1}$ norm

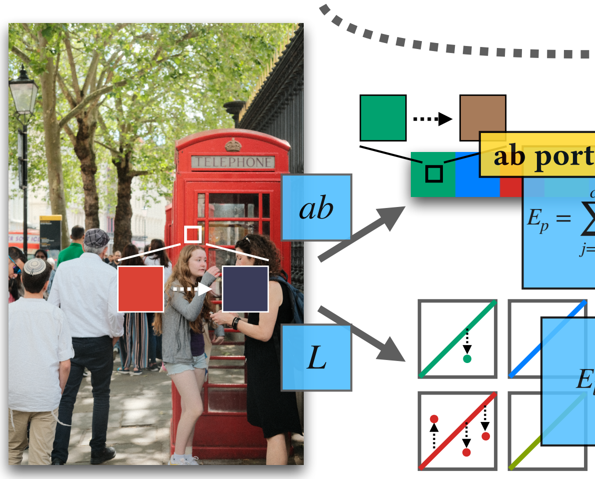
$$E_{sp} = \sum_{i=1}^{p} \sqrt{L_i^T B^T B L_i + w_{sp} \cdot \|q_i \cdot \Delta P_{i,*}\|_2^2}$$

ab portion of the constraints

$$E_p = \sum_{j=1}^{c} \left\| \left(\sum_{i=1}^{p} \widetilde{W}_{ij}(P_{i,*} + \Delta P_{i,*}) \right) - \widehat{C}_{*,j} \right\|_2^2 + \sum_{i \in P^c} \left\| (P_{i,*} + \Delta P_{i,*}) - \widehat{P}_{i,*} \right\|_2^2$$

Optimizing for sparse edits

💡 **Find the sparsest change to palette and curves that satisfy constraints**



L$_{2,1}$ norm

$$E_{sp} = \sum_{i=1}^{p} \sqrt{L_i^T B^T B L_i + w_{sp} \cdot \|q_i \cdot \Delta P_{i,*}\|_2^2}$$

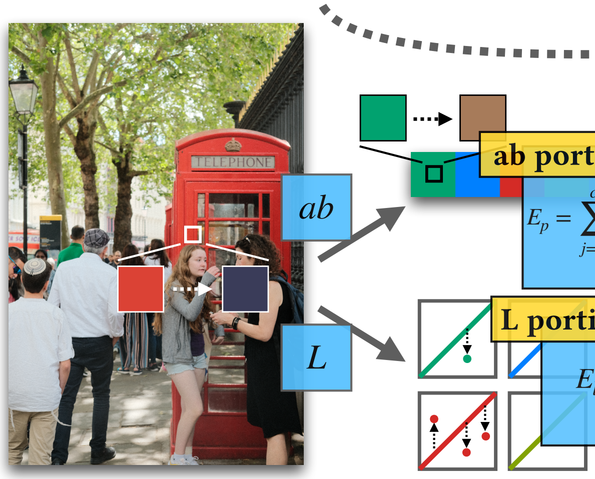
ab portion of the constraints

$$E_p = \sum_{j=1}^{c} \left\| \left(\sum_{i=1}^{p} \widetilde{W}_{ij}(P_{i,*} + \Delta P_{i,*}) \right) - \widehat{C}_{*,j} \right\|_2^2 + \sum_{i \in P^c} \left\| (P_{i,*} + \Delta P_{i,*}) - \widehat{P}_{i,*} \right\|_2^2$$

$$E_l = \sum_{j=1}^{c} \|\widetilde{S}_j \odot \left(\sum_{i=1}^{p} \widetilde{W}_{ij} L_i \right) - \widetilde{C}_j\|_2^2 + \sum_{i=1}^{p} \|\overline{S}_i \odot L_i - \overline{C}_i\|_2^2$$

Optimizing for sparse edits

- Find the sparsest change to palette and curves that satisfy constraints



L2,1 norm

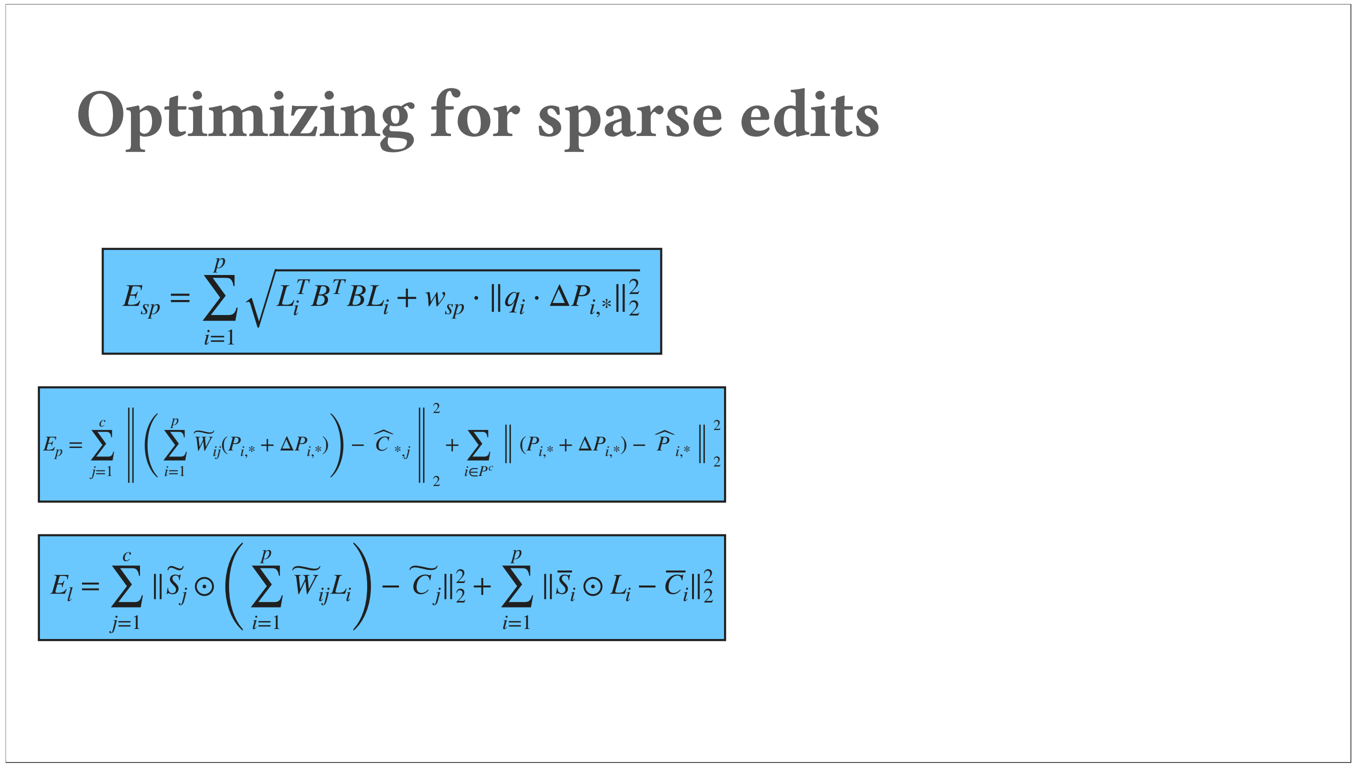
$$E_{sp} = \sum_{i=1}^{p} \sqrt{L_i^T B^T B L_i + w_{sp} \cdot \|q_i \cdot \Delta P_{i,*}\|_2^2}$$

ab portion of the constraints

$$E_p = \sum_{j=1}^{c} \left\| \left(\sum_{i=1}^{p} \widetilde{W}_{ij}(P_{i,*} + \Delta P_{i,*}) \right) - \widehat{C}_{*,j} \right\|_2^2 + \sum_{i \in P^c} \left\| (P_{i,*} + \Delta P_{i,*}) - \widehat{P}_{i,*} \right\|_2^2$$

L portion of the constraints

$$E_l = \sum_{j=1}^{c} \|\widetilde{S}_j \odot \left(\sum_{i=1}^{p} \widetilde{W}_{ij} L_i \right) - \widetilde{C}_j\|_2^2 + \sum_{i=1}^{p} \|\overline{S}_i \odot L_i - \overline{C}_i\|_2^2$$



Optimizing for sparse edits

$$E_{sp} = \sum_{i=1}^{p} \sqrt{L_i^T B^T B L_i + w_{sp} \cdot \|q_i \cdot \Delta P_{i,*}\|_2^2}$$

$$E_p = \sum_{j=1}^{c} \left\| \left(\sum_{i=1}^{p} \widetilde{W}_{ij}(P_{i,*} + \Delta P_{i,*}) \right) - \widehat{C}_{*,j} \right\|_2^2 + \sum_{i \in P^c} \left\| (P_{i,*} + \Delta P_{i,*}) - \widehat{P}_{i,*} \right\|_2^2$$

$$E_l = \sum_{j=1}^{c} \|\widetilde{S}_j \odot \left(\sum_{i=1}^{p} \widetilde{W}_{ij} L_i \right) - \widetilde{C}_j\|_2^2 + \sum_{i=1}^{p} \|\overline{S}_i \odot L_i - \overline{C}_i\|_2^2$$

(15s)

then be put together to be solved and subject to the color gamut constraints and 0-1 curve property.
Traditional SLSQP solver needs roughly 30 secs to solve the problem, which is totally not good for interactive editing experiences.

Optimizing for sparse edits

$$\left.\begin{aligned}
&\boxed{E_{sp} = \sum_{i=1}^{p} \sqrt{L_i^T B^T B L_i + w_{sp} \cdot \|q_i \cdot \Delta P_{i,*}\|_2^2}} \\
&\boxed{E_p = \sum_{j=1}^{c} \left\| \left(\sum_{i=1}^{p} \widetilde{W}_{ij}(P_{i,*} + \Delta P_{i,*}) \right) - \widehat{C}_{*,j} \right\|_2^2 + \sum_{i \in P^c} \left\| (P_{i,*} + \Delta P_{i,*}) - \widehat{P}_{i,*} \right\|_2^2} \\
&\boxed{E_l = \sum_{j=1}^{c} \|\widetilde{S}_j \odot \left(\sum_{i=1}^{p} \widetilde{W}_{ij} L_i \right) - \widetilde{C}_j\|_2^2 + \sum_{i=1}^{p} \|\overline{S}_i \odot L_i - \overline{C}_i\|_2^2}
\end{aligned}\right\}$$

Optimizing for sparse edits

$$\left.\begin{array}{l} \boxed{E_{sp} = \sum_{i=1}^{p} \sqrt{L_i^T B^T B L_i + w_{sp} \cdot \|q_i \cdot \Delta P_{i,*}\|_2^2}} \\ \boxed{E_p = \sum_{j=1}^{c} \left\| \left(\sum_{i=1}^{p} \widetilde{W}_{ij}(P_{i,*} + \Delta P_{i,*}) \right) - \widehat{C}_{*,j} \right\|_2^2 + \sum_{i \in P^c} \left\| (P_{i,*} + \Delta P_{i,*}) - \widehat{P}_{i,*} \right\|_2^2} \\ \boxed{E_l = \sum_{j=1}^{c} \|\widetilde{S}_j \odot \left(\sum_{i=1}^{p} \widetilde{W}_{ij} L_i \right) - \widetilde{C}_j\|_2^2 + \sum_{i=1}^{p} \|\overline{S}_i \odot L_i - \overline{C}_i\|_2^2} \end{array}\right\} \boxed{\{L_i, \Delta P_i\} = \arg\min_{L_i, \Delta P_i} E_{sp} + w_{eq}(E_l + E_p)}$$

Optimizing for sparse edits

$$\left.\begin{array}{c}
\boxed{E_{sp} = \sum_{i=1}^{p} \sqrt{L_i^T B^T B L_i + w_{sp} \cdot \|q_i \cdot \Delta P_{i,*}\|_2^2}} \\
\boxed{E_p = \sum_{j=1}^{c} \left\| \left(\sum_{i=1}^{p} \widetilde{W}_{ij}(P_{i,*} + \Delta P_{i,*}) \right) - \widehat{C}_{*,j} \right\|_2^2 + \sum_{i \in P^c} \left\| (P_{i,*} + \Delta P_{i,*}) - \widehat{P}_{i,*} \right\|_2^2} \\
\boxed{E_l = \sum_{j=1}^{c} \|\widetilde{S}_j \odot \left(\sum_{i=1}^{p} \widetilde{W}_{ij} L_i \right) - \widetilde{C}_j\|_2^2 + \sum_{i=1}^{p} \|\overline{S}_i \odot L_i - \overline{C}_i\|_2^2}
\end{array}\right\}$$

$$\boxed{\{L_i, \Delta P_i\} = \underset{L_i, \Delta P_i}{\arg\min}\ E_{sp} + w_{eq}(E_l + E_p)}$$

subject to

$$\boxed{-128 \leq P_i + \Delta P_i \leq 127}$$

$$\boxed{L_{i,1} = 0, \quad L_{i,s} = 1}$$

Optimizing for sparse edits

$$E_{sp} = \sum_{i=1}^{p} \sqrt{L_i^T B^T B L_i + w_{sp} \cdot \|q_i \cdot \Delta P_{i,*}\|_2^2}$$

$$E_p = \sum_{j=1}^{c} \left\| \left(\sum_{i=1}^{p} \widetilde{W}_{ij}(P_{i,*} + \Delta P_{i,*}) \right) - \widehat{C}_{*,j} \right\|_2^2 + \sum_{i \in P^c} \left\| (P_{i,*} + \Delta P_{i,*}) - \widehat{P}_{i,*} \right\|_2^2$$

$$E_l = \sum_{j=1}^{c} \|\widetilde{S}_j \odot \left(\sum_{i=1}^{p} \widetilde{W}_{ij} L_i \right) - \widetilde{C}_j\|_2^2 + \sum_{i=1}^{p} \|\overline{S}_i \odot L_i - \overline{C}_i\|_2^2$$

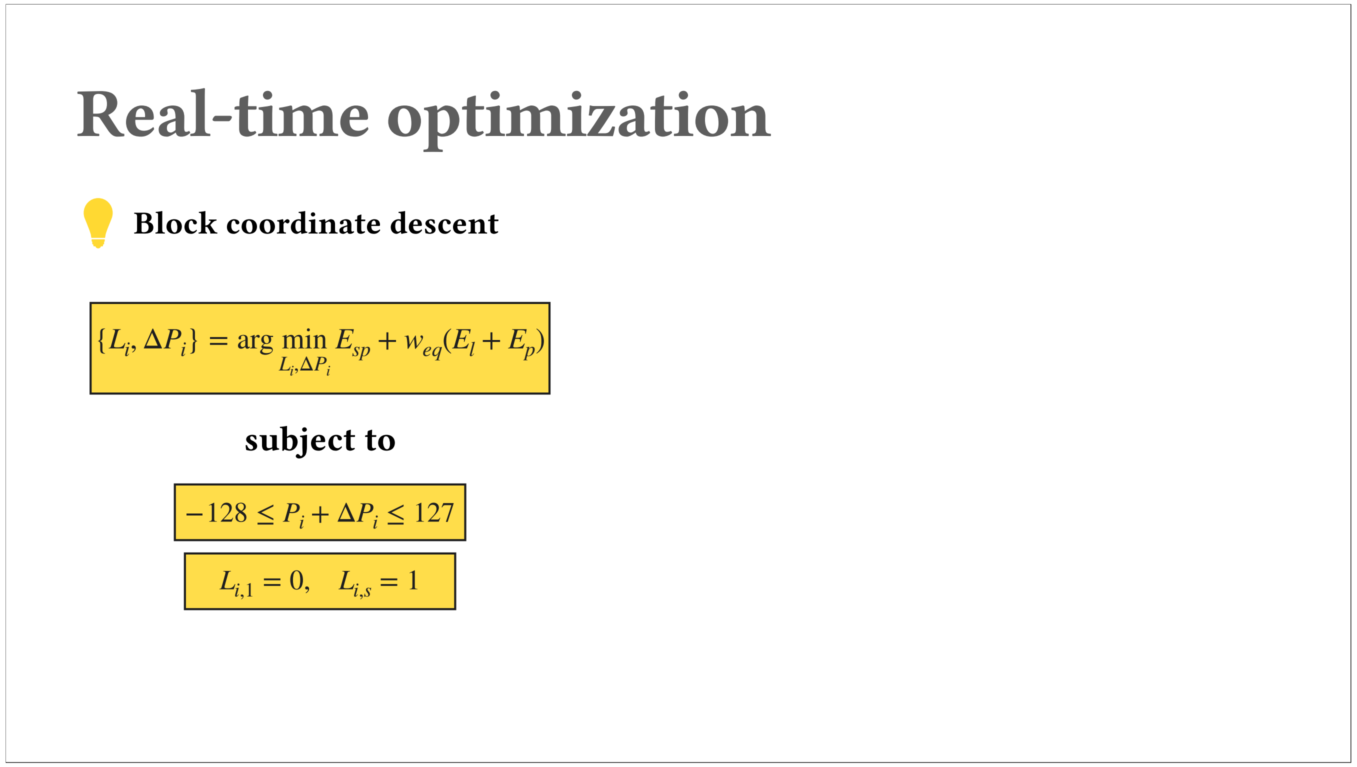
~30 secs
SLSQP solver

$$\{L_i, \Delta P_i\} = \underset{L_i, \Delta P_i}{\arg\min}\, E_{sp} + w_{eq}(E_l + E_p)$$

subject to

$$-128 \leq P_i + \Delta P_i \leq 127$$

$$L_{i,1} = 0, \quad L_{i,s} = 1$$



Real-time optimization

- **Block coordinate descent**

$$\boxed{\{L_i, \Delta P_i\} = \arg\min_{L_i, \Delta P_i} E_{sp} + w_{eq}(E_l + E_p)}$$

subject to

$$\boxed{-128 \leq P_i + \Delta P_i \leq 127}$$

$$\boxed{L_{i,1} = 0, \quad L_{i,s} = 1}$$

(30s)

We design a real-time solver based on block coordinate descent. We break the problem into two subproblems of solving tone curves and palette changes, respectively.
After complicated algebraic derivation (please see our paper for details), the lightness can be solved under a small linear system, and the palette changes can be solved with a small qp problem. We alternatively solve these two subproblems and update the variables alternatively until both lightness and palette changes converge.
This gives us a real-time solver that has roughly 400x speedup.

Real-time optimization

Block coordinate descent

$$\boxed{\{L_i, \Delta P_i\} = \arg\min_{L_i, \Delta P_i} E_{sp} + w_{eq}(E_l + E_p)}$$

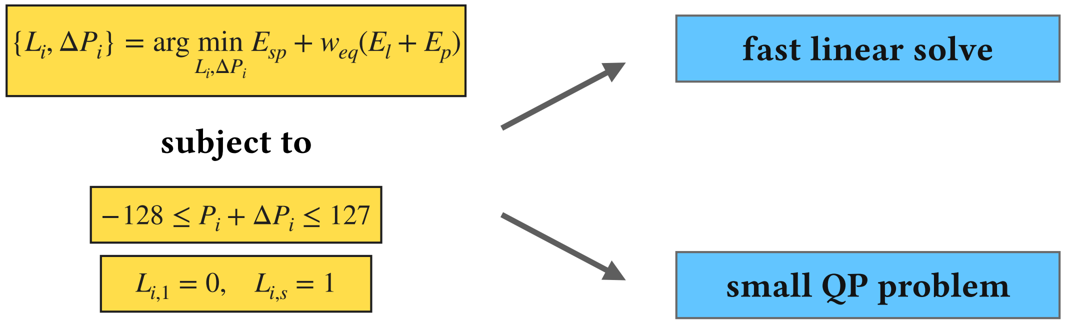
subject to

$$\boxed{-128 \leq P_i + \Delta P_i \leq 127}$$

$$\boxed{L_{i,1} = 0, \quad L_{i,s} = 1}$$

Real-time optimization

Block coordinate descent



$$\{L_i, \Delta P_i\} = \arg\min_{L_i,\Delta P_i} E_{sp} + w_{eq}(E_l + E_p)$$

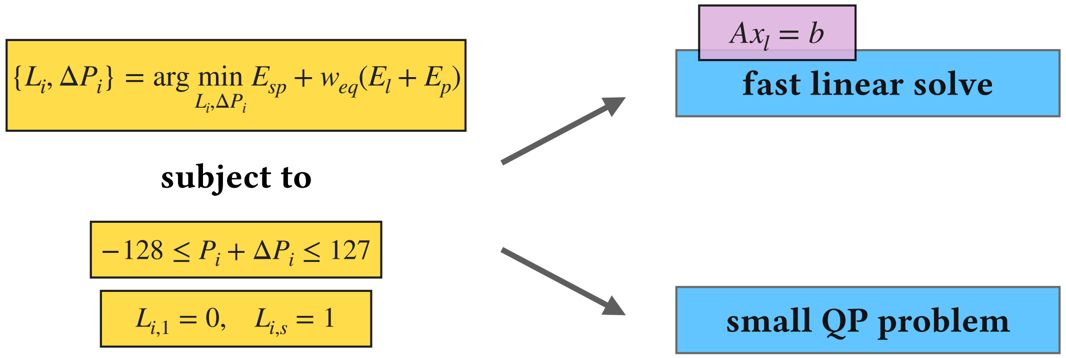
subject to

$$-128 \leq P_i + \Delta P_i \leq 127$$

$$L_{i,1} = 0, \quad L_{i,s} = 1$$

Real-time optimization

Block coordinate descent



$$\{L_i, \Delta P_i\} = \arg\min_{L_i, \Delta P_i} E_{sp} + w_{eq}(E_l + E_p)$$

subject to

$$-128 \leq P_i + \Delta P_i \leq 127$$

$$L_{i,1} = 0, \quad L_{i,s} = 1$$

$Ax_l = b$

fast linear solve

small QP problem

Real-time optimization

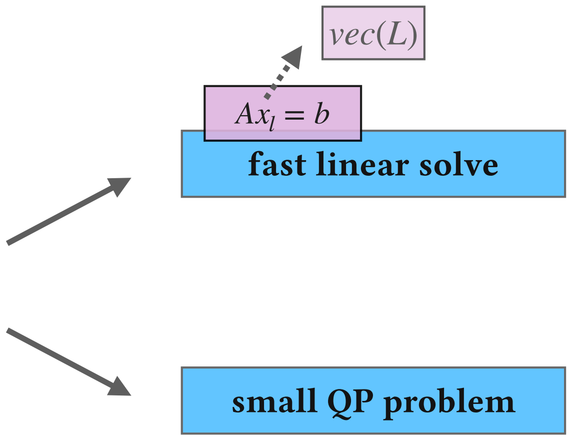
Block coordinate descent

$$\{L_i, \Delta P_i\} = \arg\min_{L_i, \Delta P_i} E_{sp} + w_{eq}(E_l + E_p)$$

subject to

$$-128 \leq P_i + \Delta P_i \leq 127$$

$$L_{i,1} = 0, \quad L_{i,s} = 1$$



Real-time optimization

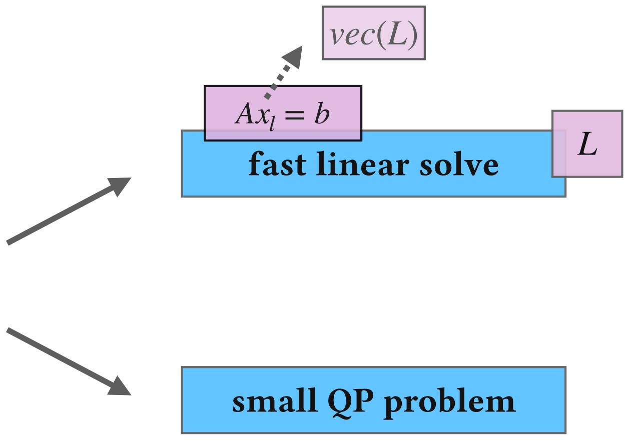
💡 **Block coordinate descent**

$$\boxed{\{L_i, \Delta P_i\} = \underset{L_i, \Delta P_i}{\arg\min}\ E_{sp} + w_{eq}(E_l + E_p)}$$

subject to

$$\boxed{-128 \leq P_i + \Delta P_i \leq 127}$$

$$\boxed{L_{i,1} = 0, \quad L_{i,s} = 1}$$



$vec(L)$

$Ax_l = b$

fast linear solve

L

small QP problem

Real-time optimization

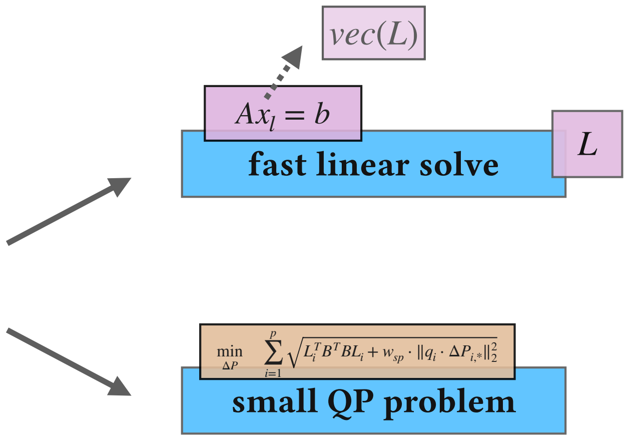
💡 **Block coordinate descent**

$$\{L_i, \Delta P_i\} = \arg\min_{L_i, \Delta P_i} E_{sp} + w_{eq}(E_l + E_p)$$

subject to

$$-128 \le P_i + \Delta P_i \le 127$$

$$L_{i,1} = 0, \quad L_{i,s} = 1$$



$vec(L)$

$Ax_l = b$

fast linear solve

L

$$\min_{\Delta P} \sum_{i=1}^{p} \sqrt{L_i^T B^T B L_i + w_{sp} \cdot \|q_i \cdot \Delta P_{i,*}\|_2^2}$$

small QP problem

Real-time optimization

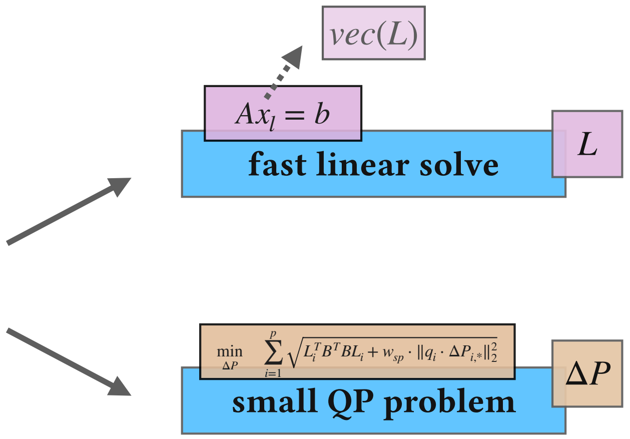
Block coordinate descent

$$\boxed{\{L_i, \Delta P_i\} = \arg\min_{L_i, \Delta P_i} E_{sp} + w_{eq}(E_l + E_p)}$$

subject to

$$\boxed{-128 \leq P_i + \Delta P_i \leq 127}$$

$$\boxed{L_{i,1} = 0, \quad L_{i,s} = 1}$$



$vec(L)$

$Ax_l = b$

fast linear solve

L

$$\min_{\Delta P} \sum_{i=1}^{p} \sqrt{L_i^T B^T B L_i + w_{sp} \cdot \|q_i \cdot \Delta P_{i,*}\|_2^2}$$

small QP problem

ΔP

Real-time optimization

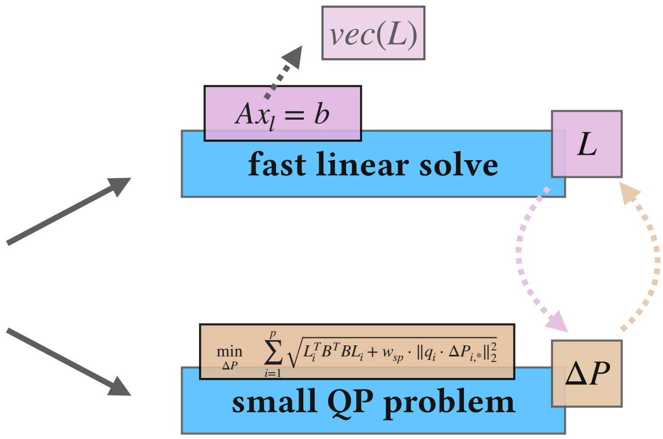
Block coordinate descent

$$\{L_i, \Delta P_i\} = \arg\min_{L_i, \Delta P_i} E_{sp} + w_{eq}(E_l + E_p)$$

subject to

$$-128 \leq P_i + \Delta P_i \leq 127$$

$$L_{i,1} = 0, \quad L_{i,s} = 1$$



$vec(L)$

$Ax_l = b$

fast linear solve

L

$\min_{\Delta P} \sum_{i=1}^{p} \sqrt{L_i^T B^T B L_i + w_{sp} \cdot \|q_i \cdot \Delta P_{i,*}\|_2^2}$

small QP problem

ΔP

Real-time optimization

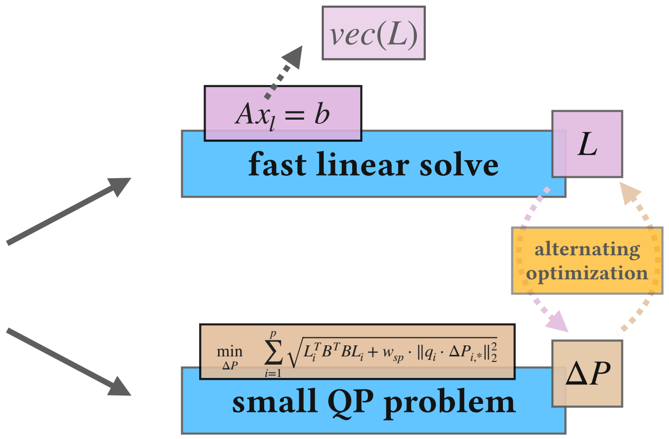
Block coordinate descent

$$\{L_i, \Delta P_i\} = \arg\min_{L_i, \Delta P_i} E_{sp} + w_{eq}(E_l + E_p)$$

subject to

$$-128 \le P_i + \Delta P_i \le 127$$

$$L_{i,1} = 0, \quad L_{i,s} = 1$$



$vec(L)$

$Ax_l = b$

fast linear solve

L

alternating optimization

$$\min_{\Delta P} \sum_{i=1}^{p} \sqrt{L_i^T B^T B L_i + w_{sp} \cdot \|q_i \cdot \Delta P_{i,*}\|_2^2}$$

small QP problem

ΔP

Real-time optimization

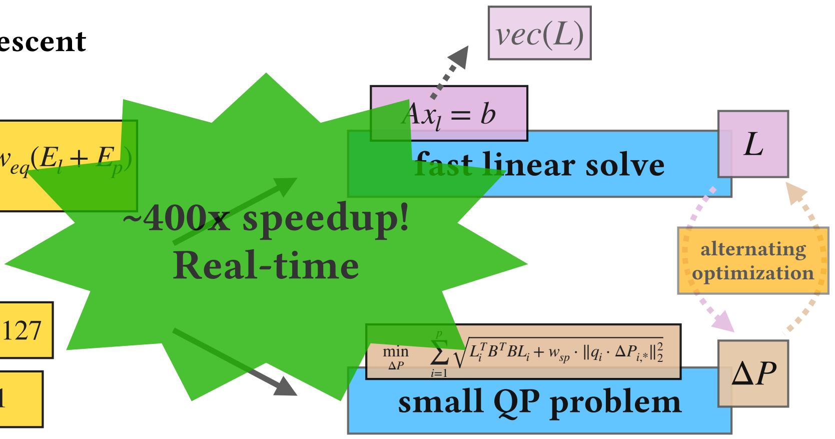
Block coordinate descent

$$\{L_i, \Delta P_i\} = \arg\min_{L_i, \Delta P_i} E_{sp} + w_{eq}(E_l + E_p)$$

subject to

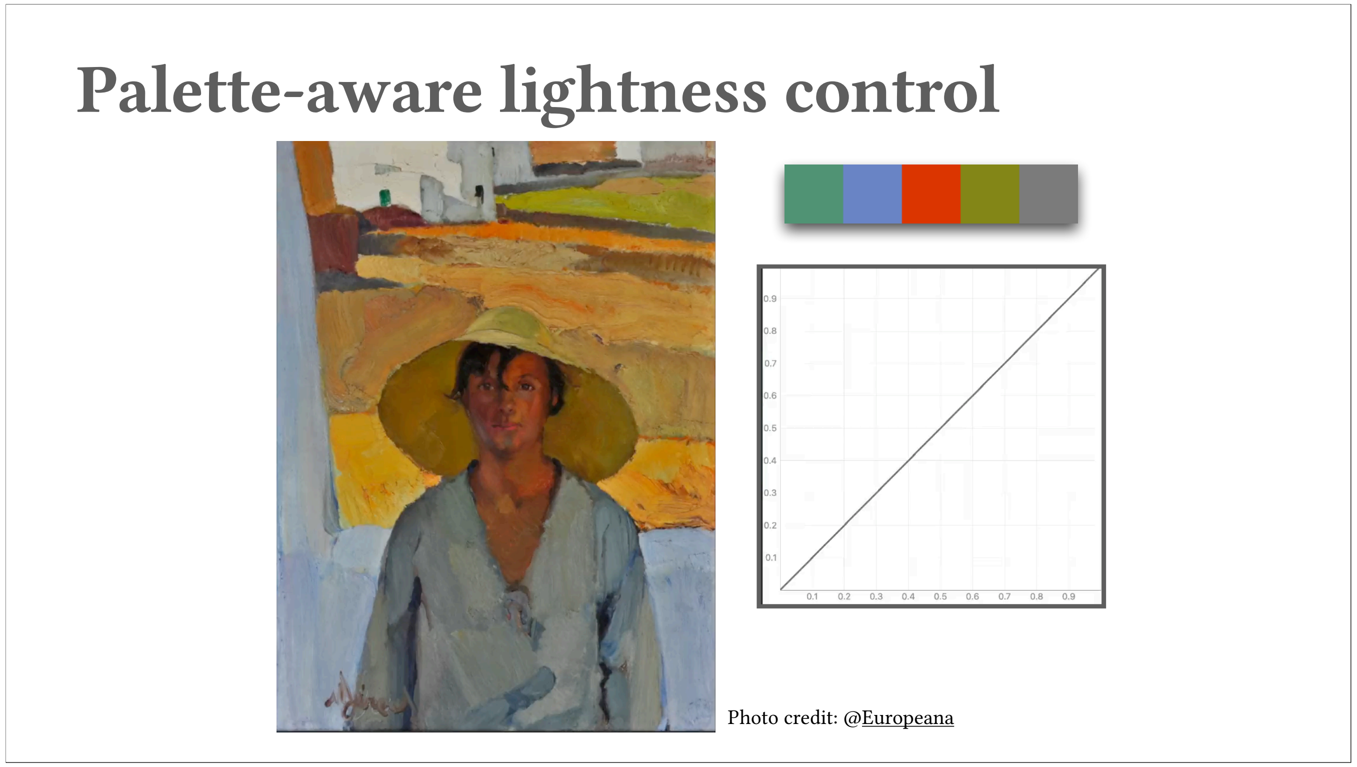
$$-128 \leq P_i + \Delta P_i \leq 127$$

$$L_{i,1} = 0, \quad L_{i,s} = 1$$



~400x speedup!
Real-time

$vec(L)$
$Ax_l = b$
fast linear solve
L
alternating optimization
$\min_{\Delta P} \sum_{i=1}^{p} \sqrt{L_i^T B^T B L_i + w_{sp} \cdot \|q_i \cdot \Delta P_{i,*}\|_2^2}$
small QP problem
ΔP



Palette-aware lightness control

Photo credit: @Europeana

(17s)

Using this solver, we can perform real-time editing and let's see some results! Here, I'll be showing users edit the lightness through per-palette tone curves from the first to the third palette colors. Our approach allows flexible multi-tone lightness editing and most importantly they are color-aware.

Palette-aware lightness control

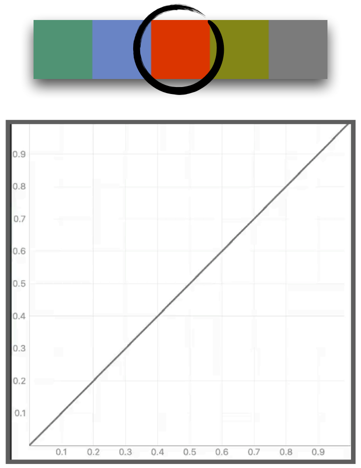
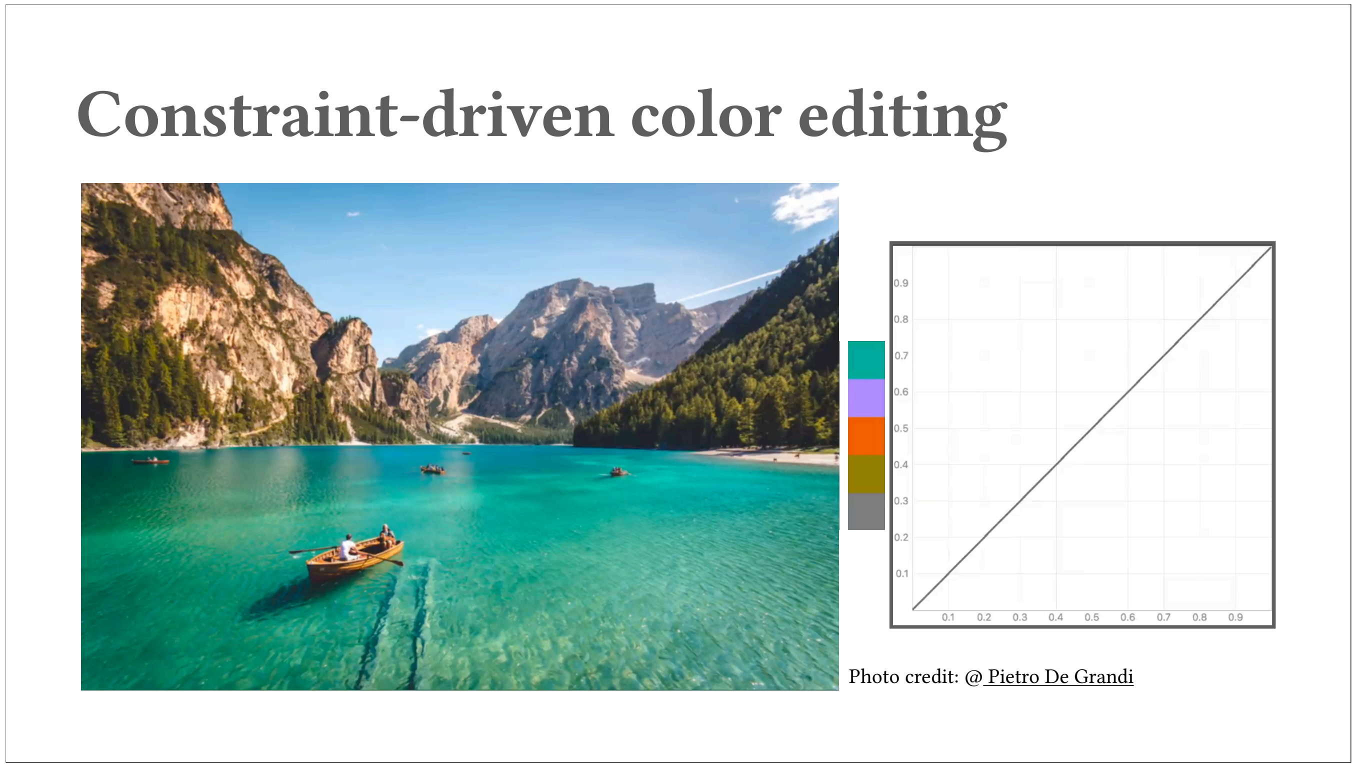


Photo credit: @Europeana



Constraint-driven color editing

Photo credit: @ Pietro De Grandi

(45s)

And here is an example of our whole editing pipeline.
Users first drop image-space constraint and wish to change color of the lake into blue. And users drop another constraint on palette to change the mountain color into orange.
Finally, users choose the tone curve according to the second palette color, and make contrast on the mountain and the sky.
Our optimization solves for all kinds of constraints on-the-fly and in real-time.
And notice that Colorfulcurves can operate on high-resolution images. Its lightness control is simply 1D function mapping and the palette computation is in practice basically independent of image size.

Constraint-driven color editing

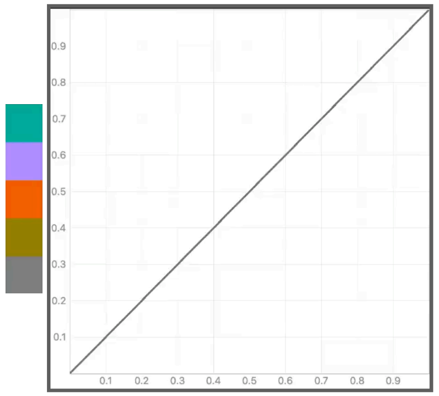
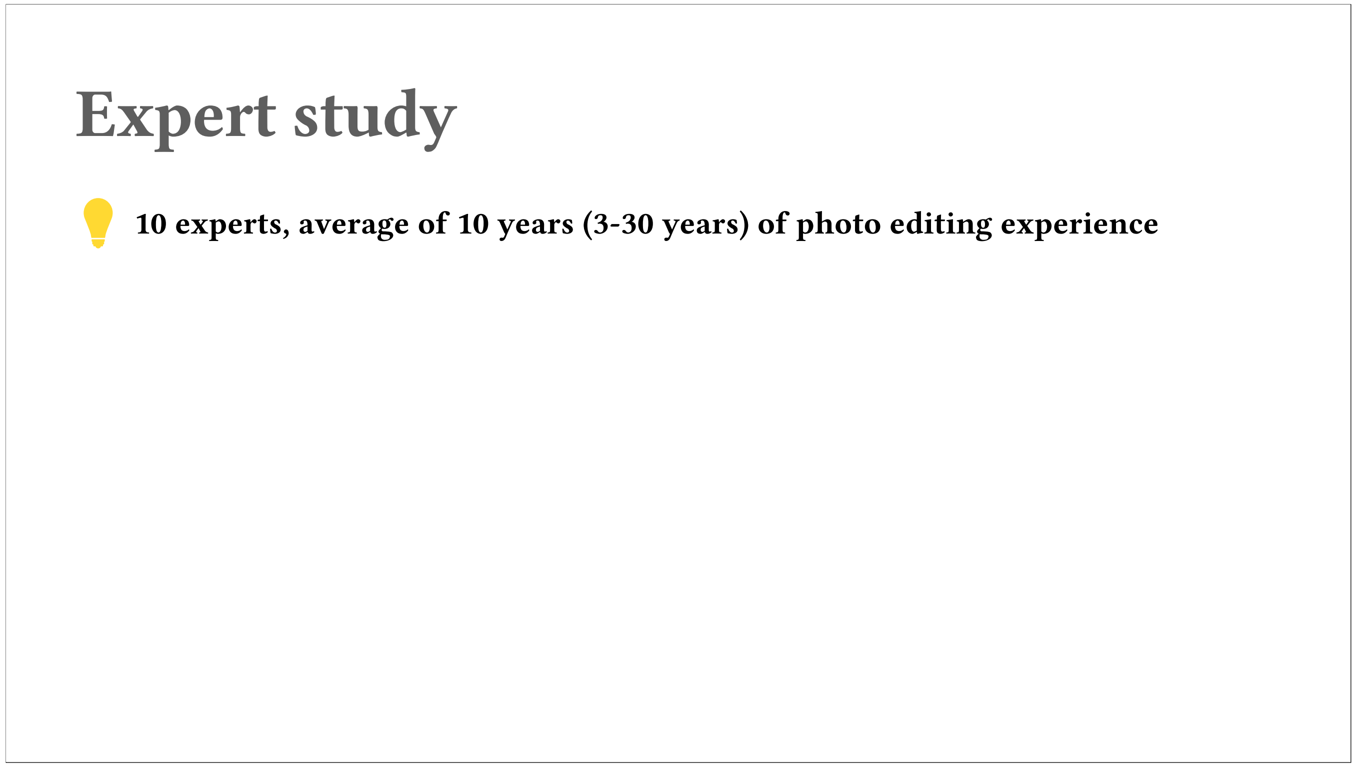


Photo credit: @ Pietro De Grandi



Expert study

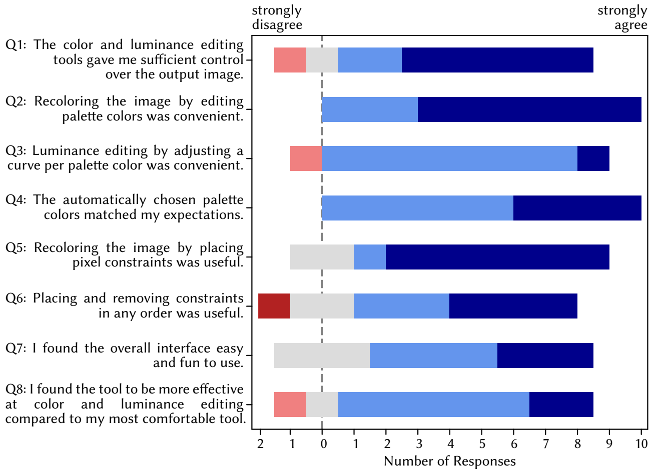
- **10 experts, average of 10 years (3-30 years) of photo editing experience**

(35s)

To evaluate ColorfulCurves, we performed expert study from 10 professional photographers who have average 10 year professional editing experiences.
I won’t show much about the study here so please check our paper for the detail, but here is the glance of the likert plot of the study by comparing ColorfulCurves to experts’ preferable software, where it shows significantly positive feedback from the experts. They also quoted how accurate and intuitive ColorfulCurves is compared to software such as LR or PS.

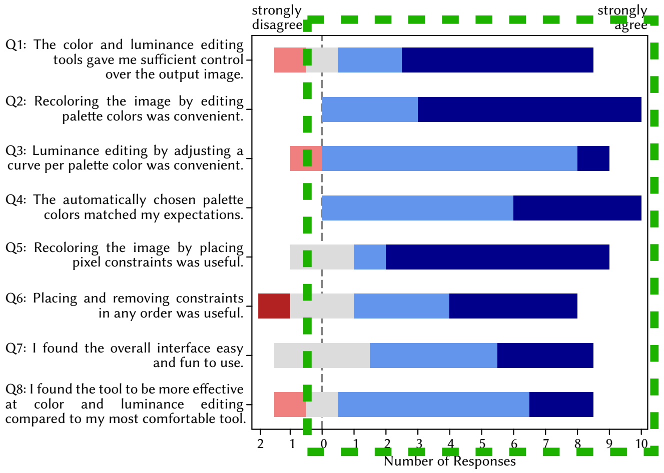
Expert study

- **10 experts, average of 10 years (3-30 years) of photo editing experience**



Expert study

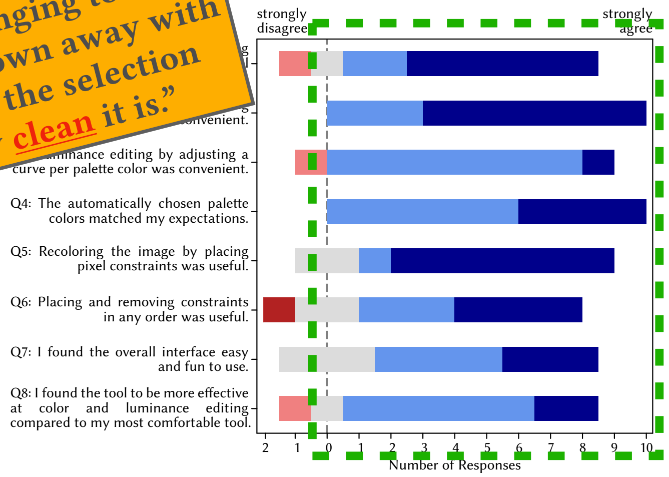
- **10 experts, average of 10 years (3-30 years) of photo editing experience**



Expert study

- **10 experts, average of 1 s (3-30 years) of photo editing experience**

“[T]he color changing tool is amazing. I was blown away with the accuracy of the selection tool and how clean it is.”

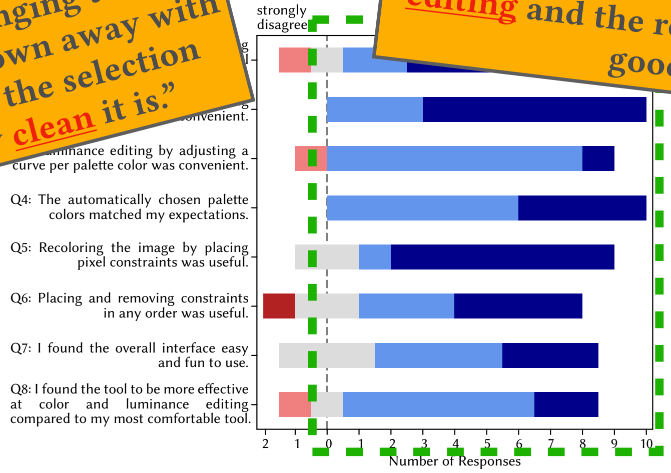


Expert study

- 10 experts, average of s (3-30 year

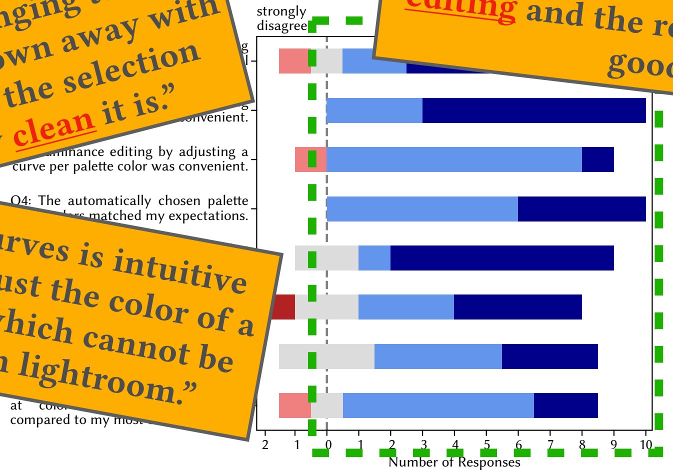
“[T]he color changing tool is amazing. I was blown away with the **accuracy** of the selection tool and how **clean** it is.”

“[T]o push the challenge further, I choose a portrait to do a **skin tone editing** and the results are really good.”



Expert study

- 10 experts, average of 1 s (3-30 year



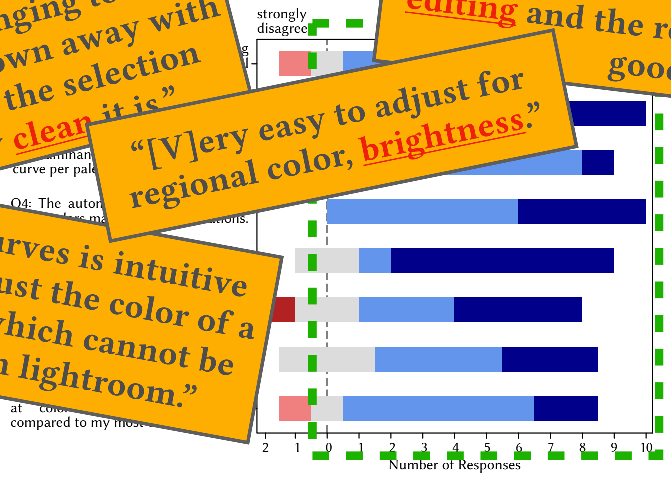
“[T]he color changing tool is amazing. I was blown away with the **accuracy** of the selection l and how **clean** it is.”

“[C]olorfulCurves is intuitive and easy to adjust the color of a **specific area**, which cannot be achieved with lightroom.”

“[T]o push the challenge further, I choose a portrait to do a **skin tone editing** and the results are really good.”

Expert study

- **10 experts, average of 1[illegible]s (3-30 year**



“[T]he color changing tool is amazing. I was blown away with the **accuracy** of the selection [illegible]l and how **clean** it is.”

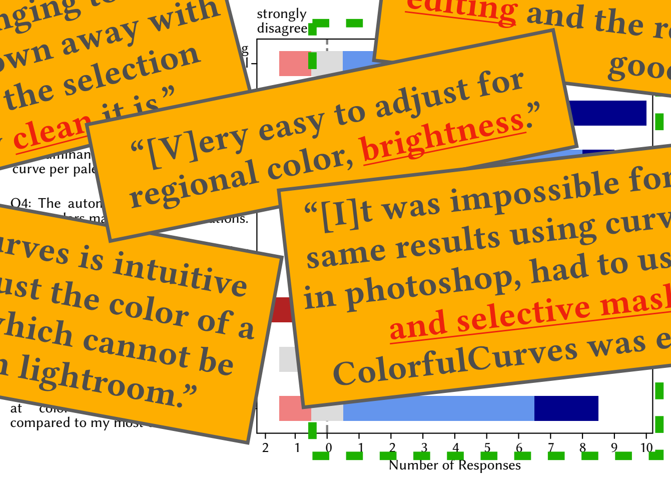
“[T]o push the challenge further, I choose a portrait to do a **skin tone editing** and the results are really good.”

“[V]ery easy to adjust for regional color, **brightness**.”

“[C]olorfulCurves is intuitive and easy to adjust the color of a **specific area**, which cannot be achieved with lightroom.”

Expert study

- 10 experts, average of [illegible]s (3-30 year



“[T]he color changing tool is amazing. I was blown away with the **accuracy** of the selection [illegible] and how **clean** it is.”

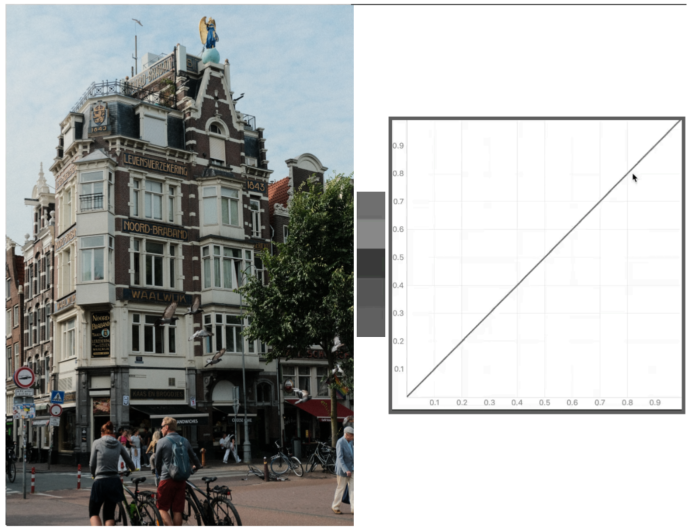
“[V]ery easy to adjust for regional color, **brightness**.”

“[T]o push the challenge further, I choose a portrait to do a **skin tone editing** and the results are really good.”

“[C]olorfulCurves is intuitive and easy to adjust the color of a **specific area**, which cannot be achieved with lightroom.”

“[I]t was impossible for me to achieve same results using curves adjustments in photoshop, had to use **several curves and selective masking**. Using ColorfulCurves was easy and quick.”

Applications

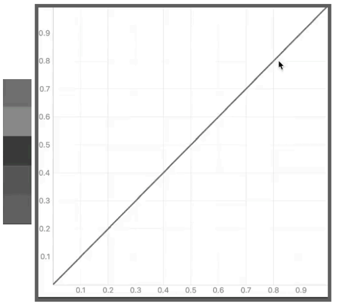


(45s)

Our formulation can also be used in applications such as interactive greyscale editing. Given a color image, after computing its color mixing formulation, we can treat its palette as grey in hue-chroma space. Since tone curves are built upon palette colors, users can interactively control the contrast and exposure.
Users can then also re-assign colors back to the palette to create new color image.
Another application is to control lightness through depth map. Instead of just being palette-aware, the weight can also be computed in 1D depth space to be depth-aware. Tone curves then built upon selected depths. Users can decrease the lightness of certain depth without manually masking.

Applications

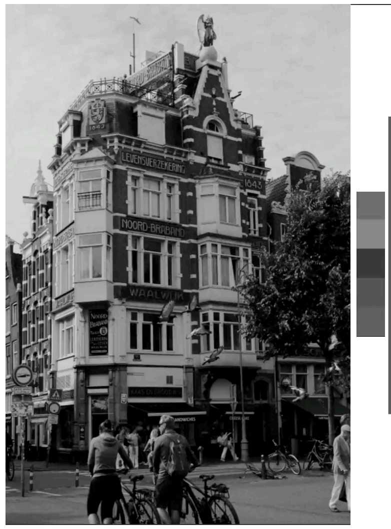
$I_{ab} = W \cdot P$

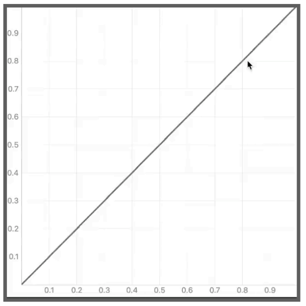


Applications

(0, 0)

$$I_{ab} = W \cdot \cancel{P}$$



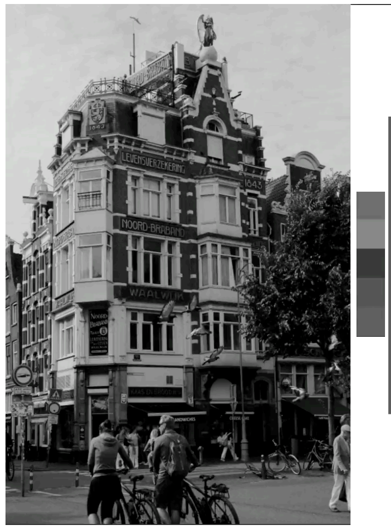


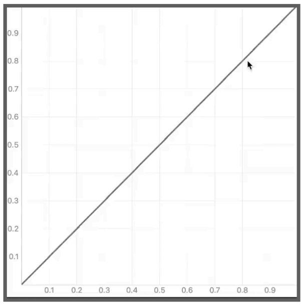
Applications

(0, 0)

$$I_{ab} = W \cdot \cancel{P}$$

$$I_L = \sum_{i=1}^{p} \hat{W}_i \odot f_i(L_0)$$

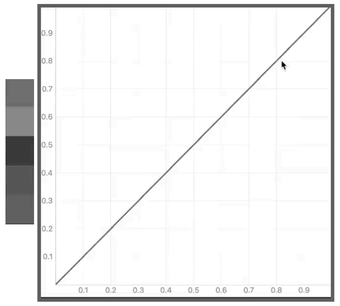




Applications

$$I_{ab} = W \cdot \overset{(0,\,0)}{\cancel{P}}$$

$$I_L = \sum_{i=1}^{p} \hat{W}_i \odot f_i(L_0)$$

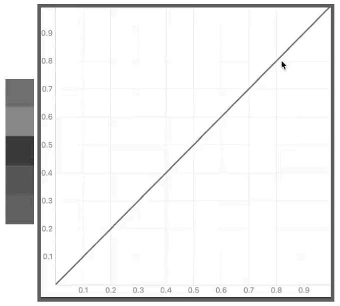


Applications

(0, 0)

$$I_{ab} = W \cdot \cancel{P}$$

$$I_L = \sum_{i=1}^{p} \hat{W}_i \odot f_i(L_0)$$

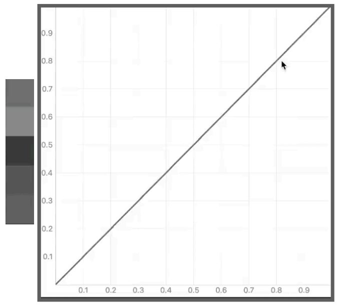


Applications

(0, 0)

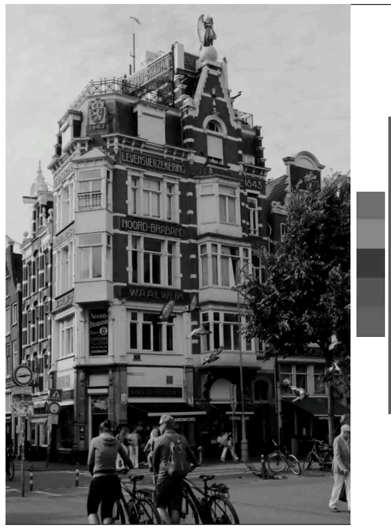
$$I_{ab} = W \cdot \cancel{P}$$

$$I_L = \sum_{i=1}^{p} \hat{W}_i \odot f_i(L_0)$$



[Miangoleh et al. 2021]

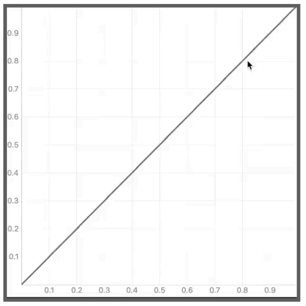
Applications



(0, 0)

$$I_{ab} = W \cdot \cancel{P}$$

$$I_L = \sum_{i=1}^{p} \hat{W}_i \odot f_i(L_0)$$



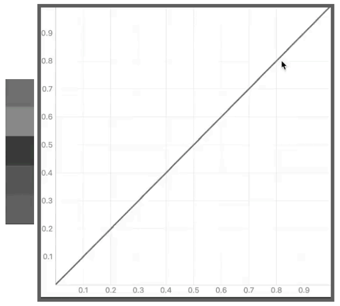
[Miangoleh et al. 2021]

Applications

(0, 0)

$$I_{ab} = W \cdot \cancel{P}$$

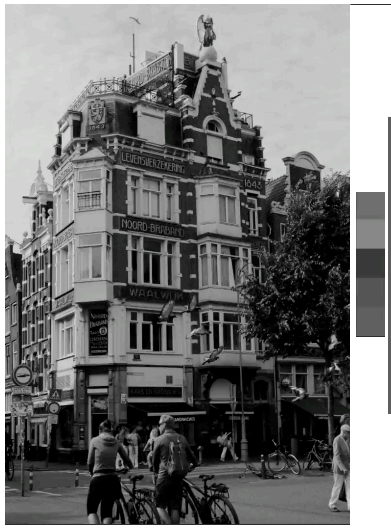
$$I_L = \sum_{i=1}^{P} \hat{W}_i \odot f_i(L_0)$$



Decrease the lightness

[Miangoleh et al. 2021]

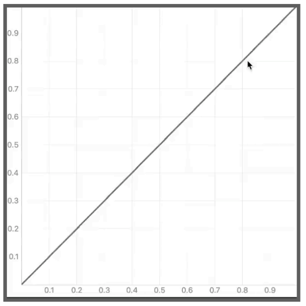
Applications



(0, 0)

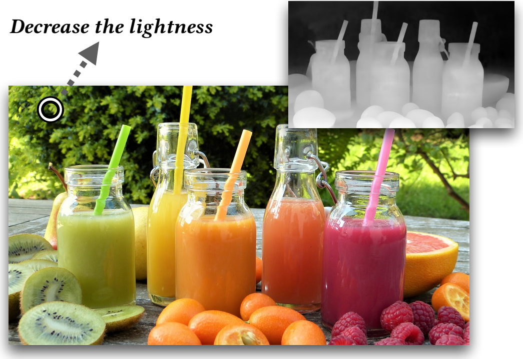
$$I_{ab} = W \cdot \cancel{P}$$

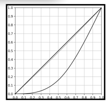
$$I_L = \sum_{i=1}^{p} \hat{W}_i \odot f_i(L_0)$$



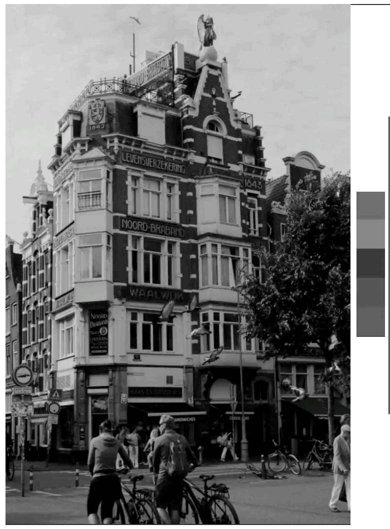
Decrease the lightness

[Miangoleh et al. 2021]





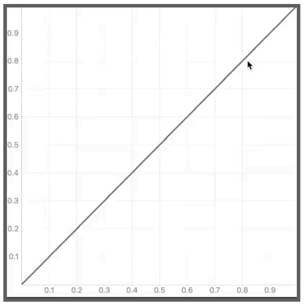
Applications



(0, 0)

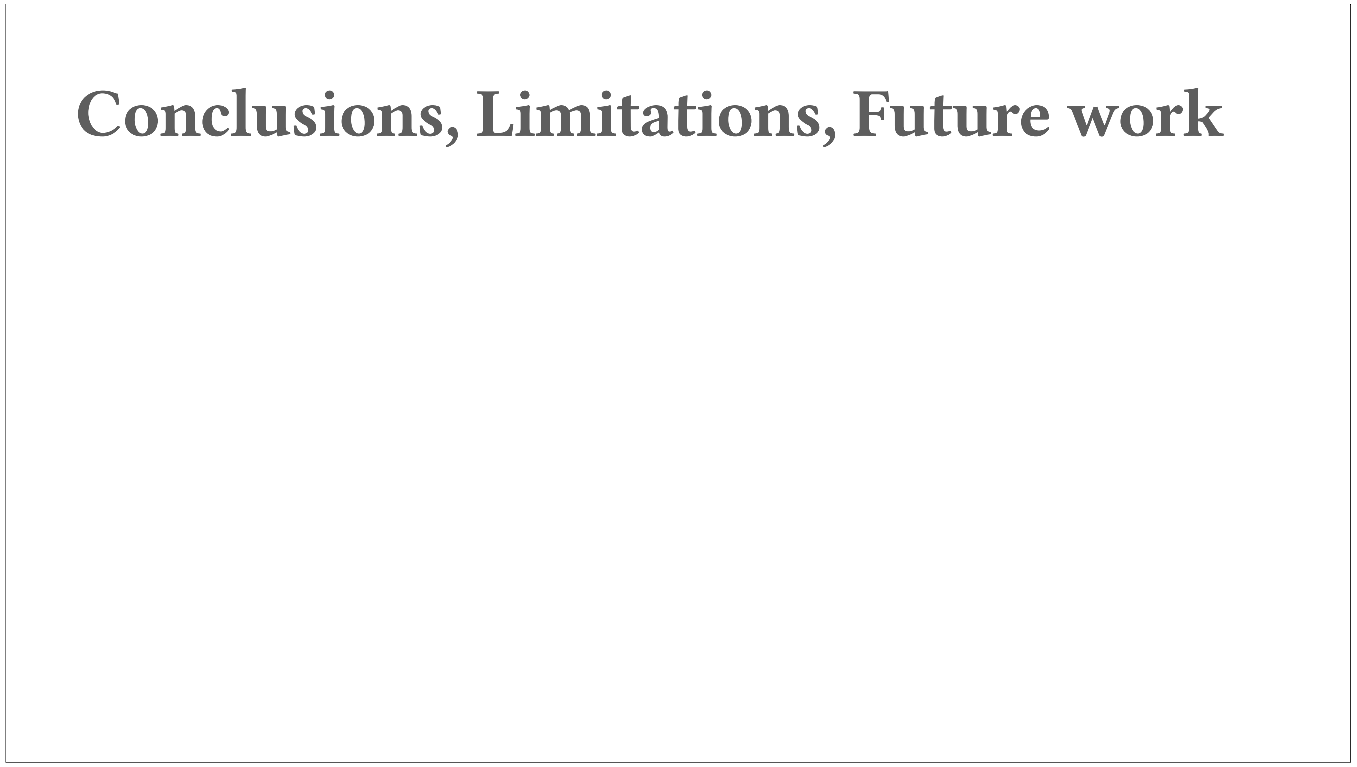
$$I_{ab} = W \cdot \cancel{P}$$

$$I_L = \sum_{i=1}^{p} \hat{W}_i \odot f_i(L_0)$$



[Miangoleh et al. 2021]

Decrease the lightness



Conclusions, Limitations, Future work

(60s)

So, let’s wrap this up.
In short, we propose a new formulation to unite palette-based and tone curve editing, which provides color-aware lightness control.
Also, we solve the common problem of the needs to directly change a pixel color in palette-based editing framework by a constraint-driven sparse optimization, and we propose a real-time solution.

For the limitation, we cannot perform semantic color editing in situation like two objects share the same color. We cannot individually edit it.

For future work, it would be interesting to see how our color-aware lightness editing can be applied to video domain. And also we would like to explore text-guided professional photo editing, like using CLIP loss to guide the parameters in our formulation.
Finally, since we perform optimization separately for lightness and palette, there might be some out-of-gamut colors. Right now our method simply clips these colors into gamut. Though in experiments, we did not find artifacts about this simple clipping, but it would be interesting to see how gamut clipping can be deformed according to different made edits.

Conclusions, Limitations, Future work

✓ Unite palette-based and tone curve editing to enable color-aware lightness control

Conclusions, Limitations, Future work

- ✓ **Unite palette-based and tone curve editing to enable color-aware lightness control**
- ✓ **Real-time editing under constraint-driven sparse optimization**

Conclusions, Limitations, Future work

✓ Unite palette-based and tone curve editing to enable color-aware lightness control

✓ Real-time editing under constraint-driven sparse optimization

✗ Semantic-aware color editing

Conclusions, Limitations, Future work

- ✓ **Unite palette-based and tone curve editing to enable color-aware lightness control**
- ✓ **Real-time editing under constraint-driven sparse optimization**

- ✗ **Semantic-aware color editing**

- 💡 **Extend to video domain**

Conclusions, Limitations, Future work

✓ Unite palette-based and tone curve editing to enable color-aware lightness control

✓ Real-time editing under constraint-driven sparse optimization

✗ Semantic-aware color editing

💡 Extend to video domain

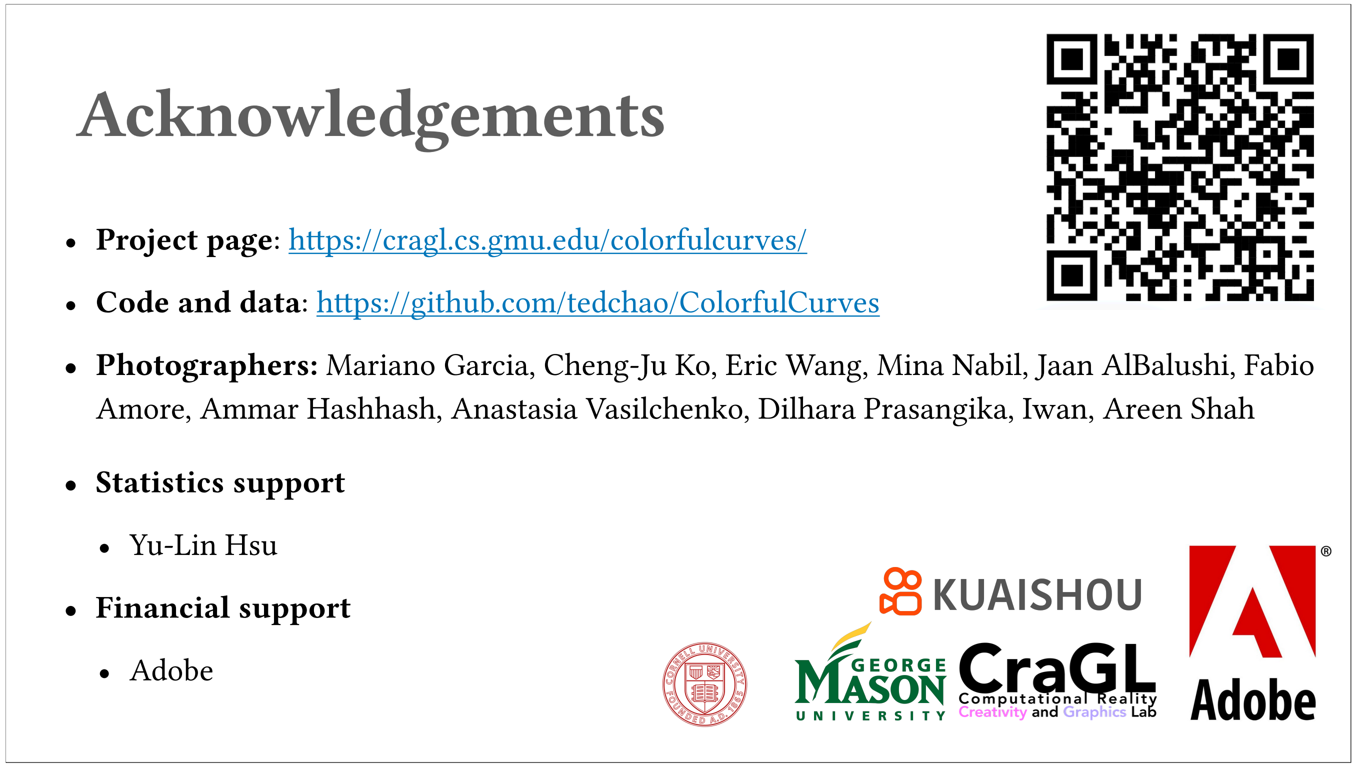
💡 Text-guided professional photo editing

Conclusions, Limitations, Future work

- ✓ Unite palette-based and tone curve editing to enable color-aware lightness control
- ✓ Real-time editing under constraint-driven sparse optimization

- ✗ Semantic-aware color editing

- 💡 Extend to video domain
- 💡 Text-guided professional photo editing
- 💡 Dynamic gamut deformation



Acknowledgements

- **Project page**: https://cragl.cs.gmu.edu/colorfulcurves/
- **Code and data**: https://github.com/tedchao/ColorfulCurves
- **Photographers:** Mariano Garcia, Cheng-Ju Ko, Eric Wang, Mina Nabil, Jaan AlBalushi, Fabio Amore, Ammar Hashhash, Anastasia Vasilchenko, Dilhara Prasangika, Iwan, Areen Shah
- **Statistics support**
  - Yu-Lin Hsu
- **Financial support**
  - Adobe

(15s)

Thank you. Our paper, code, and data have already been uploaded to our webpage. Here is the QR code to the webpage.
And we thank all the photographers, Yu-Lin for statistics support on post-evaluation, and Adobe for financial support.

Thank you.