Pigmento: Pigment-Based Image Analysis and Editing

Jianchao Tan George Mason University

Stephen DiVerdi Adobe Research

Jingwan Lu Adobe Research

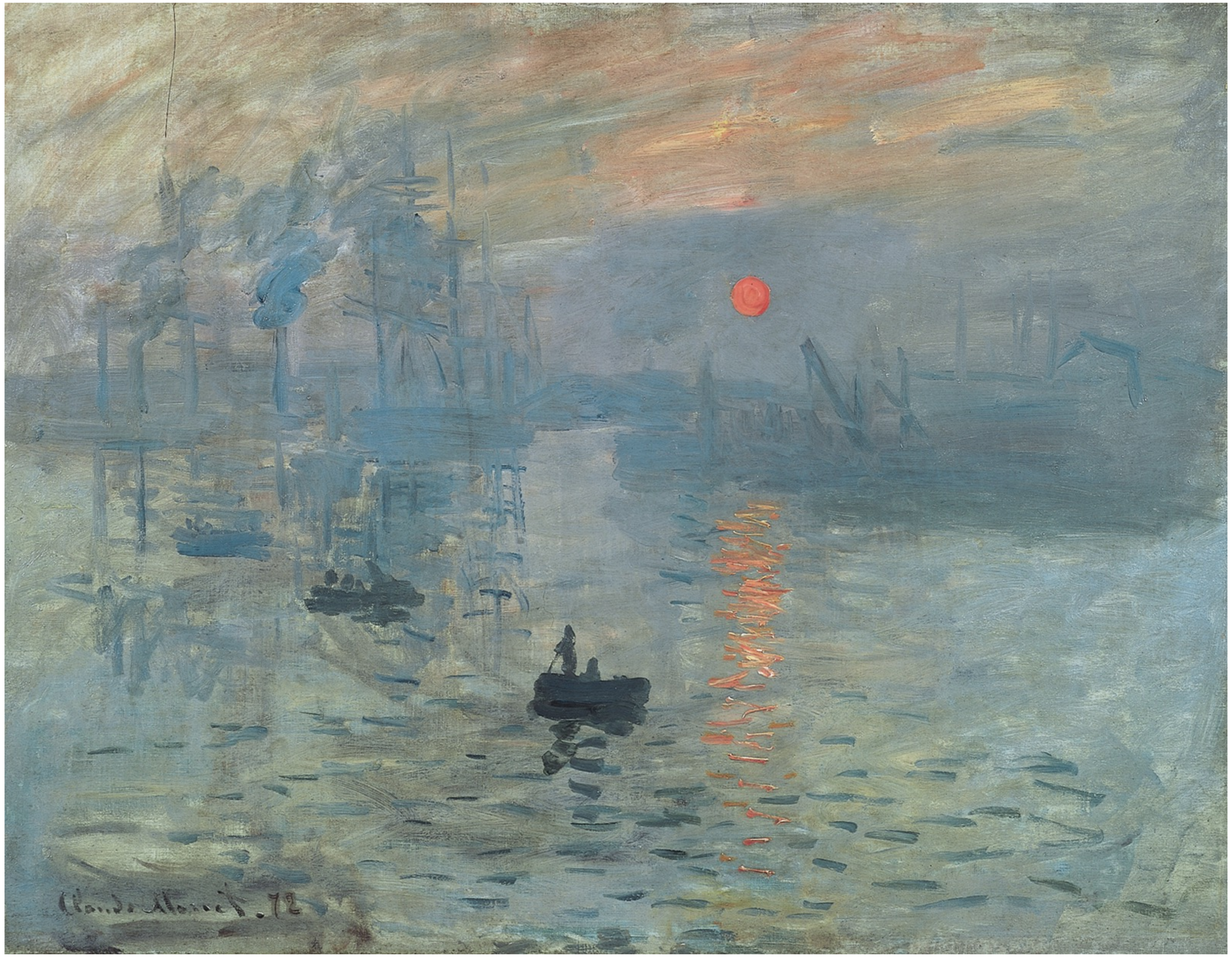
Yotam Gingold George Mason University

GEORGE MASON UNIVERSITY

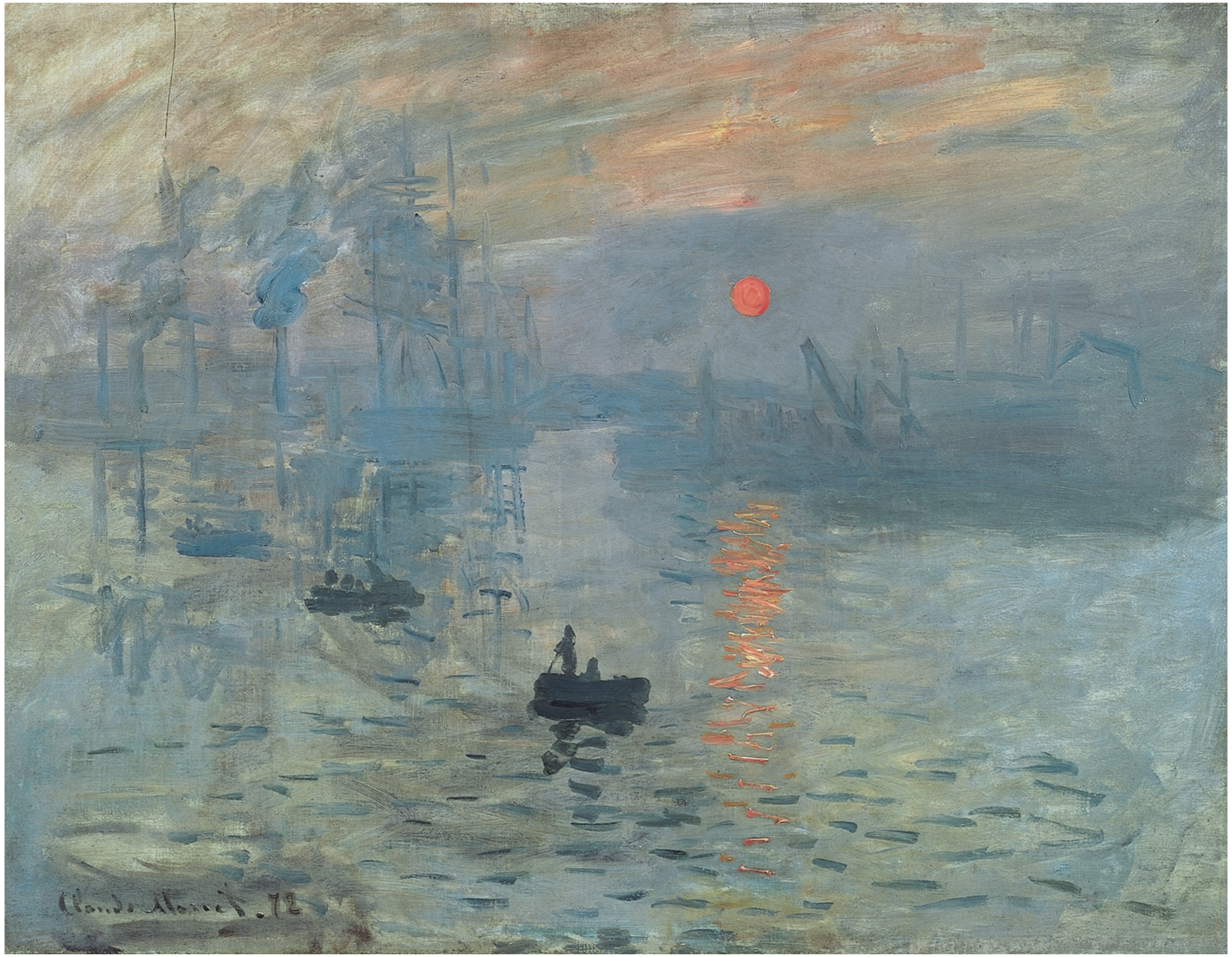
CraGL Creativity and Graphics Lab

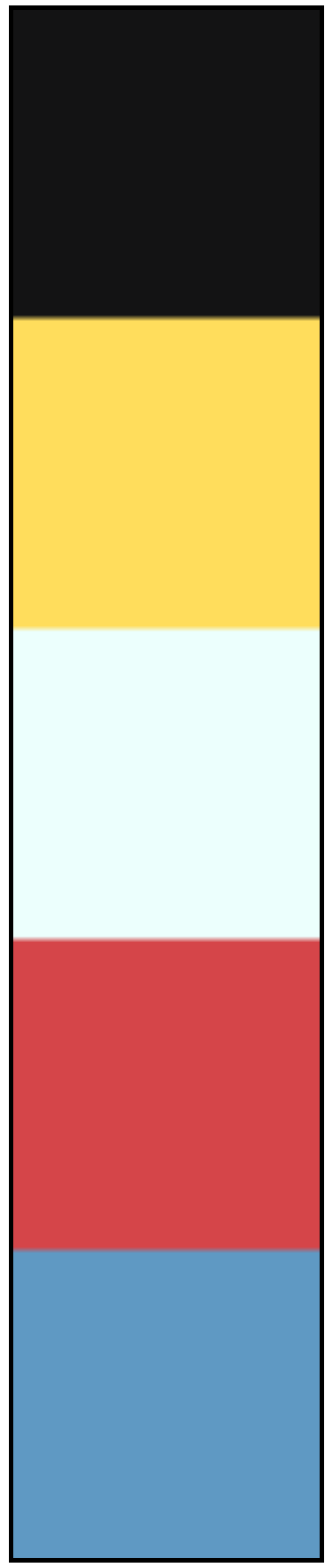
Adobe

Background: Physical Painting



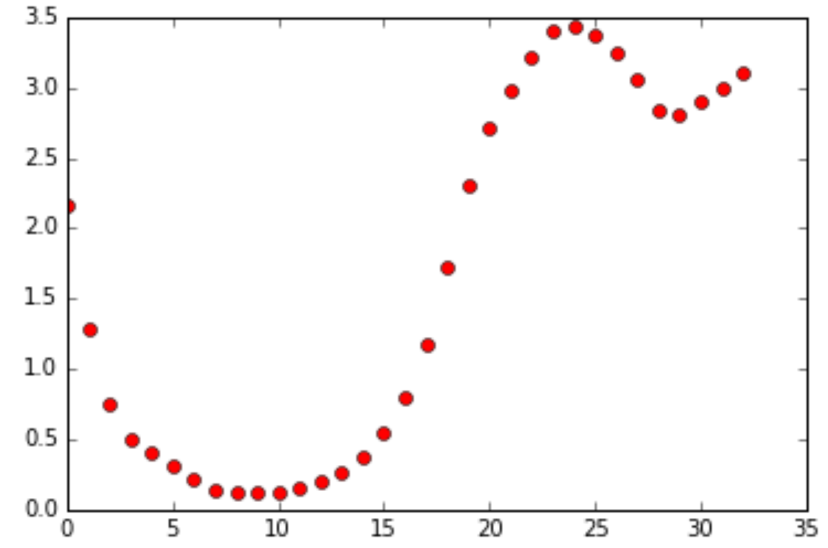
Background: Physical Painting





Background: Kubelka-Munk Model

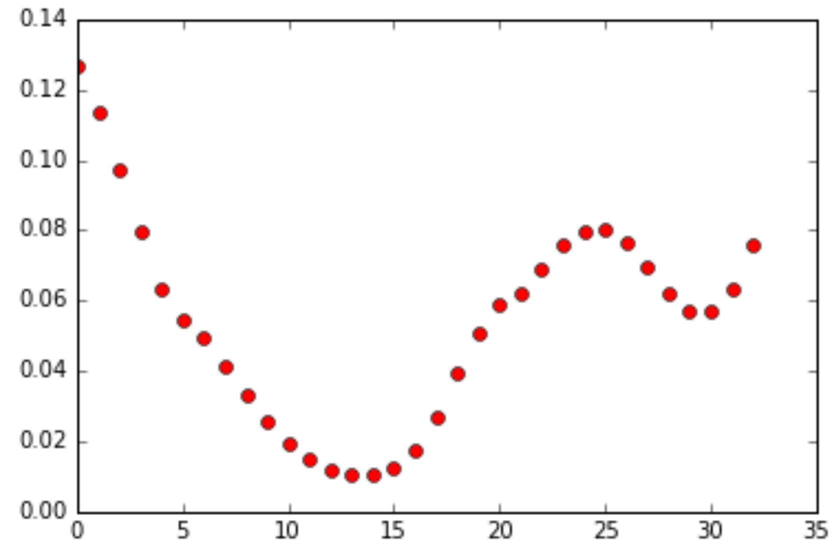
Cyan pigment ground truth data.
33 wavelength, from 380 to 700 nm, every 10 nm.



Absorption Curve (a)

Thickness (t)

Substrate Reflectance (ξ)



Scattering Curve (s)

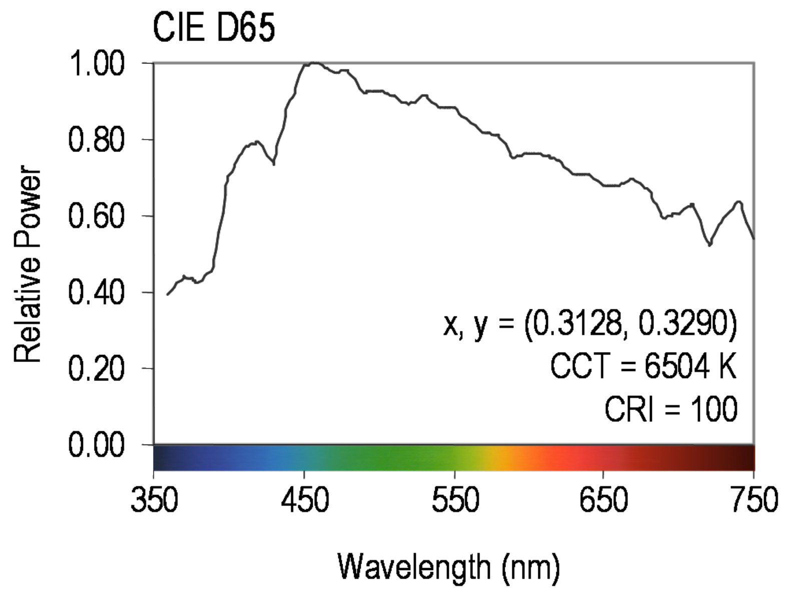
RGB

Background: Kubelka-Munk Model

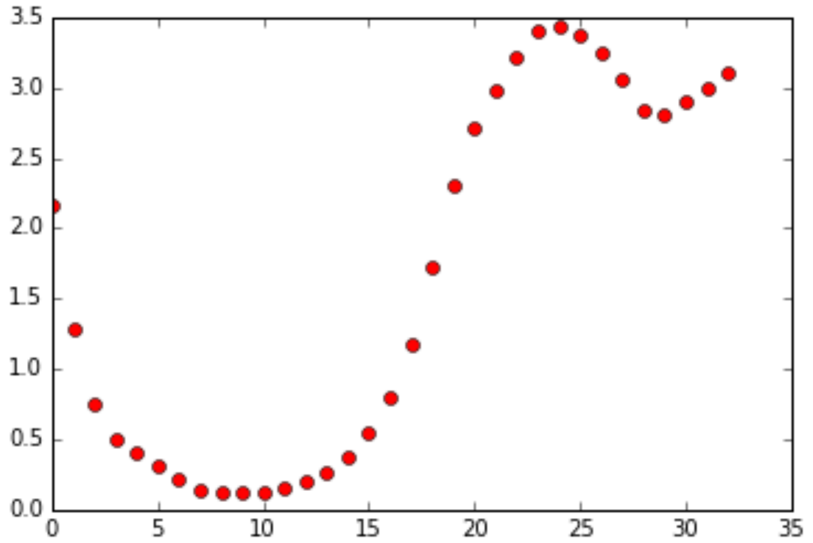
Cyan pigment ground truth data.
33 wavelength, from 380 to 700 nm, every 10 nm.

Background: Kubelka-Munk Model

Cyan pigment ground truth data.
33 wavelength, from 380 to 700 nm, every 10 nm.



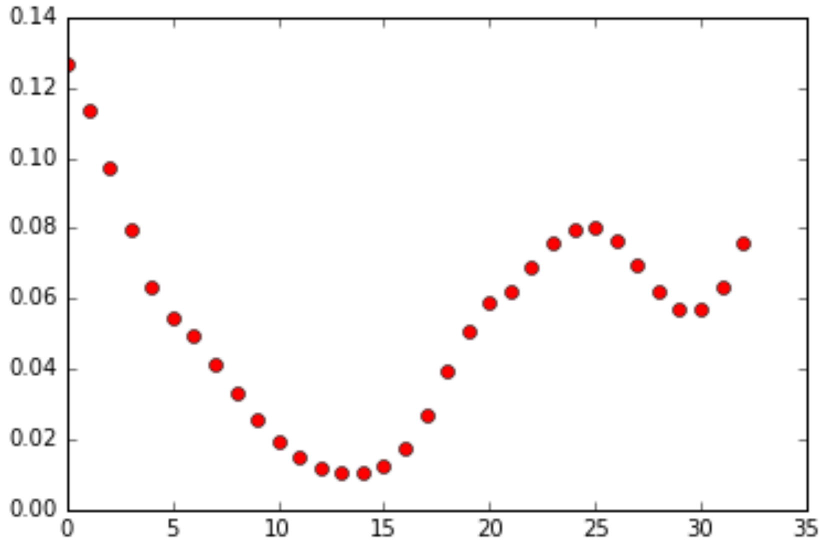
Illuminant



Absorption Curve (a)

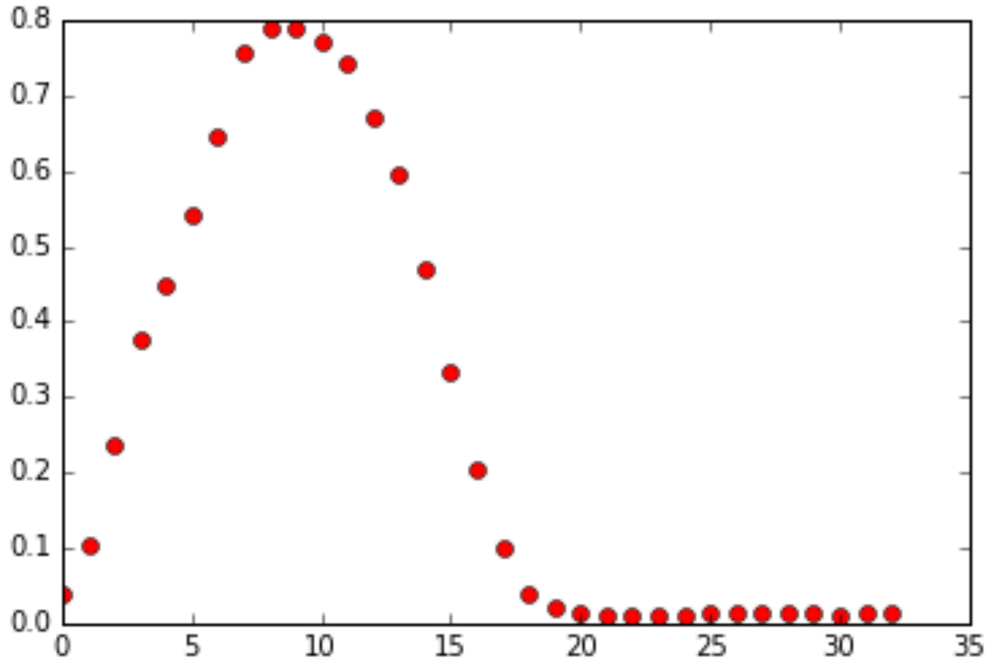
Thickness (t)

Substrate Reflectance (ξ)



Scattering Curve (s)

Kubelka-Munk model (KM)

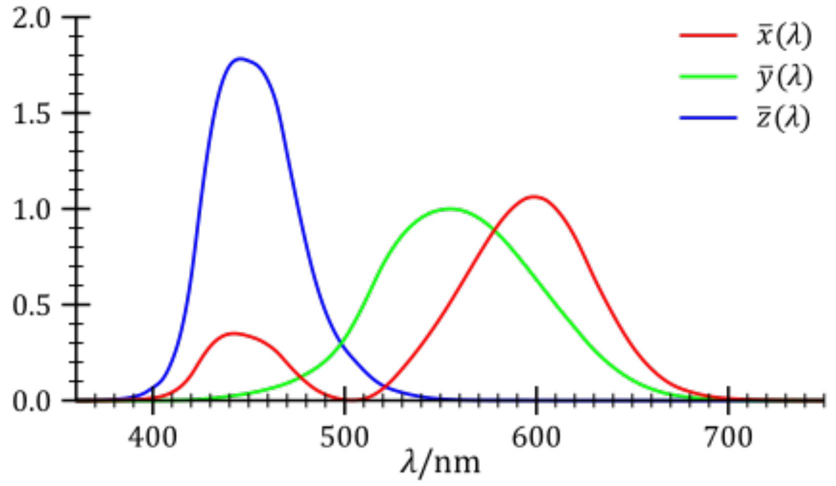


Reflectance Curve (r)

XYZ to RGB Transform

Gamma Correction

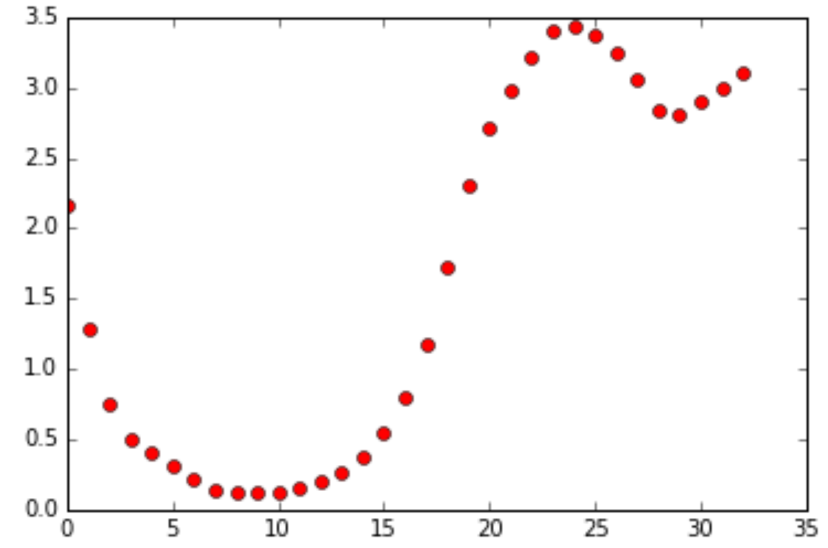
RGB



Color Matching Function

Background: Kubelka-Munk Model

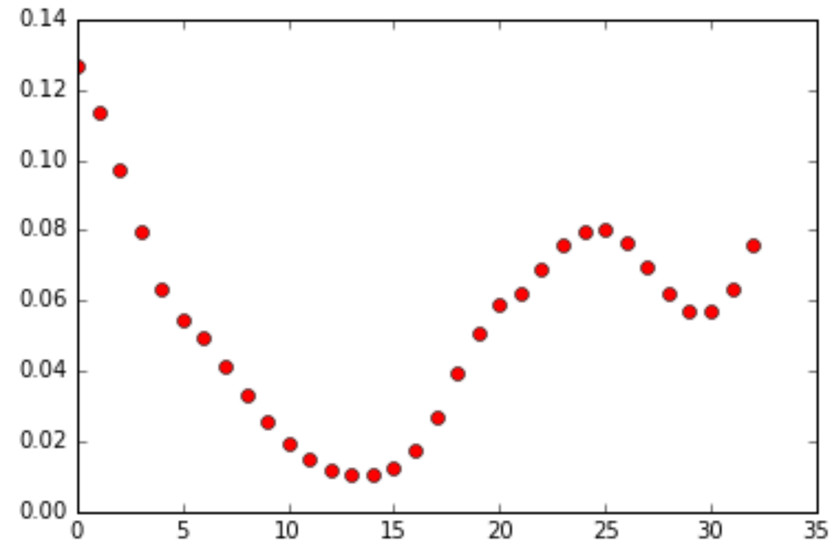
Cyan pigment ground truth data.
33 wavelength, from 380 to 700 nm, every 10 nm.



Absorption Curve (a)

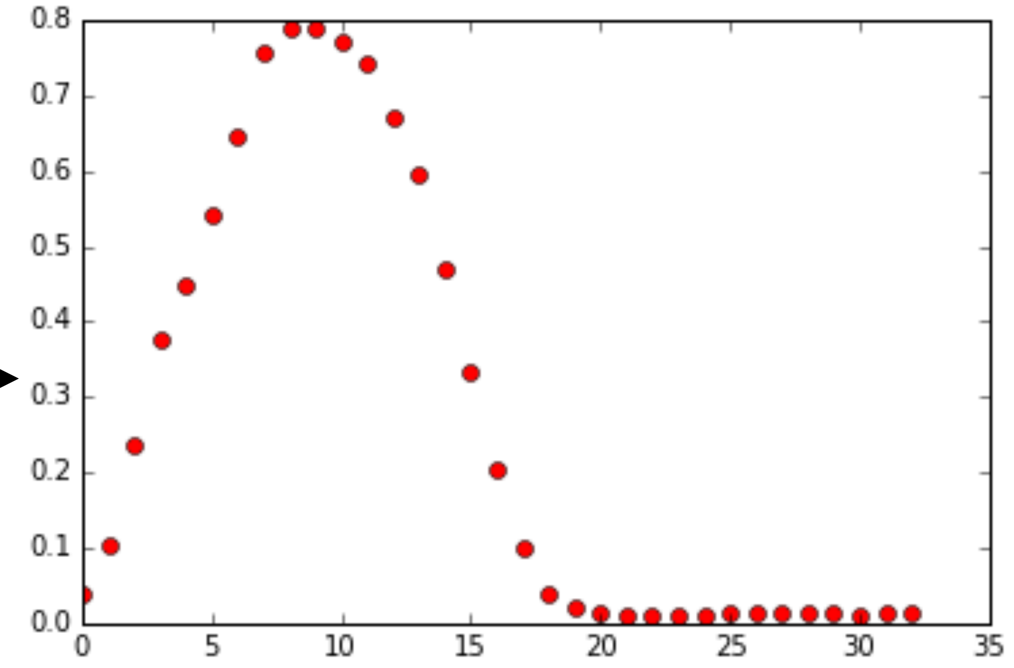
Thickness (t)

Substrate Reflectance (ξ)

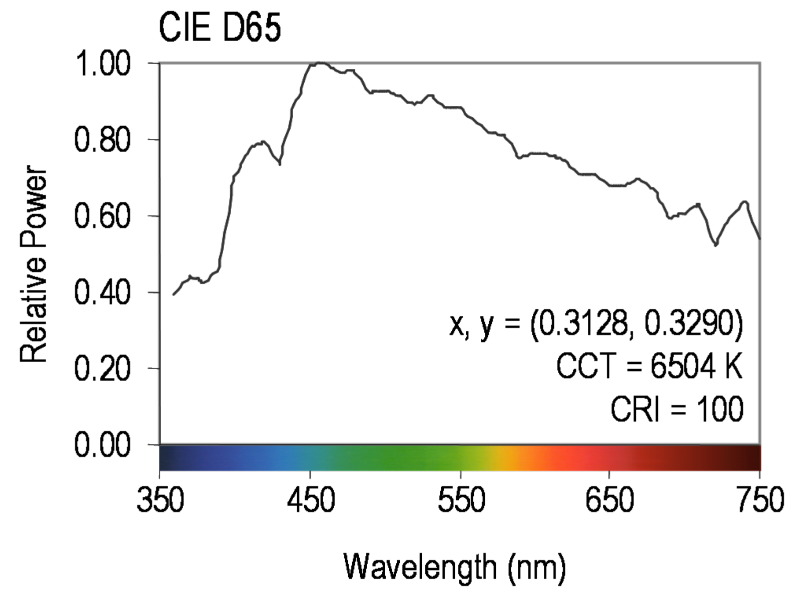


Scattering Curve (s)

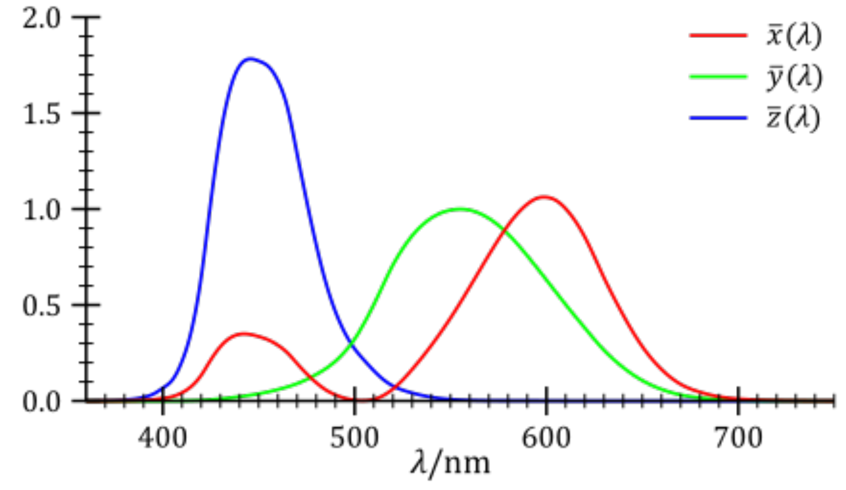
Kubelka-Munk model (KM)



Reflectance Curve (r)



Illuminant



Color Matching Function

XYZ to RGB Transform

Gamma Correction

RGB

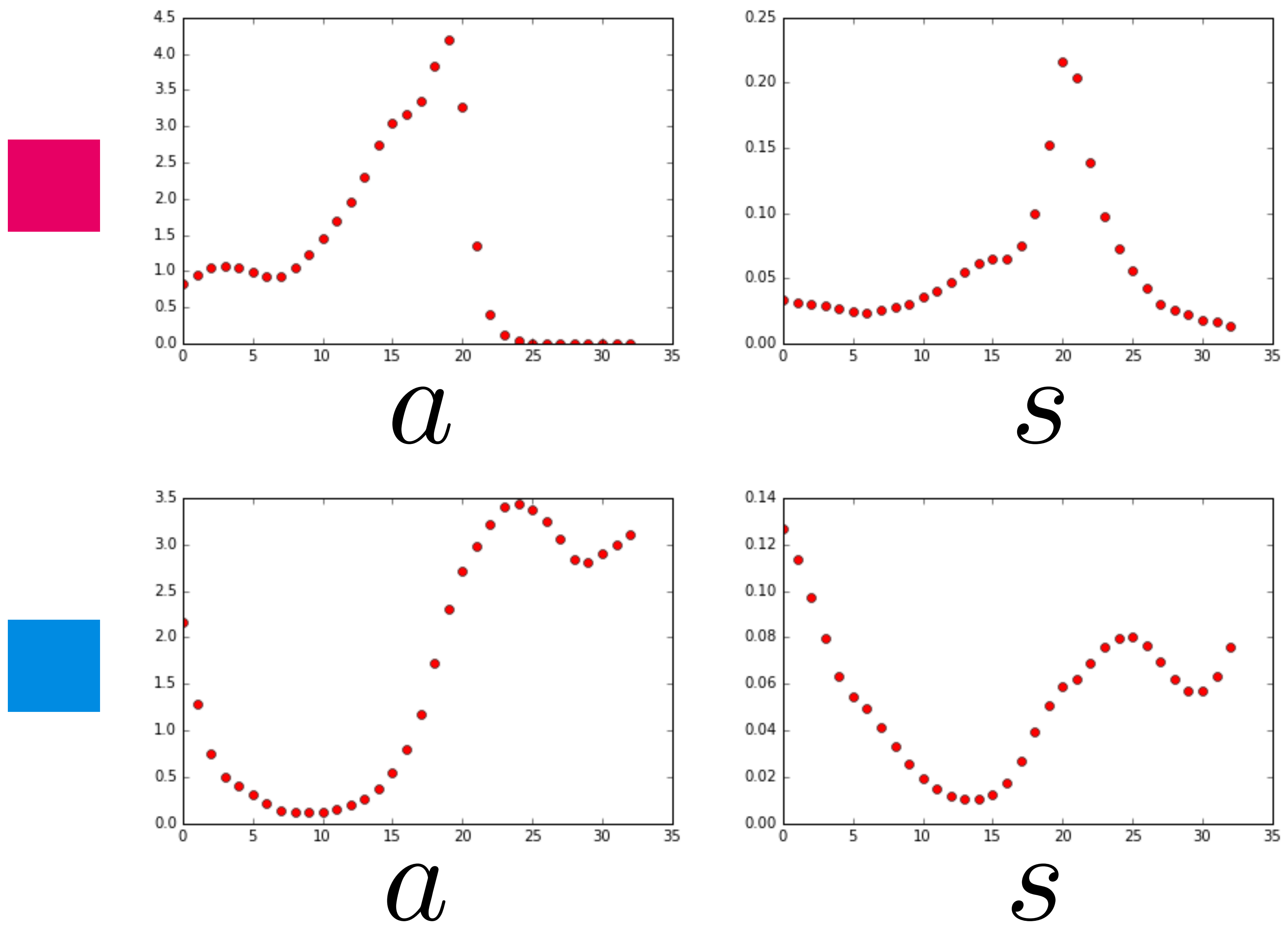
KM model $r = km(a, s, t, \xi)$

Background: Kubelka-Munk Model

Cyan pigment ground truth data.
33 wavelength, from 380 to 700 nm, every 10 nm.

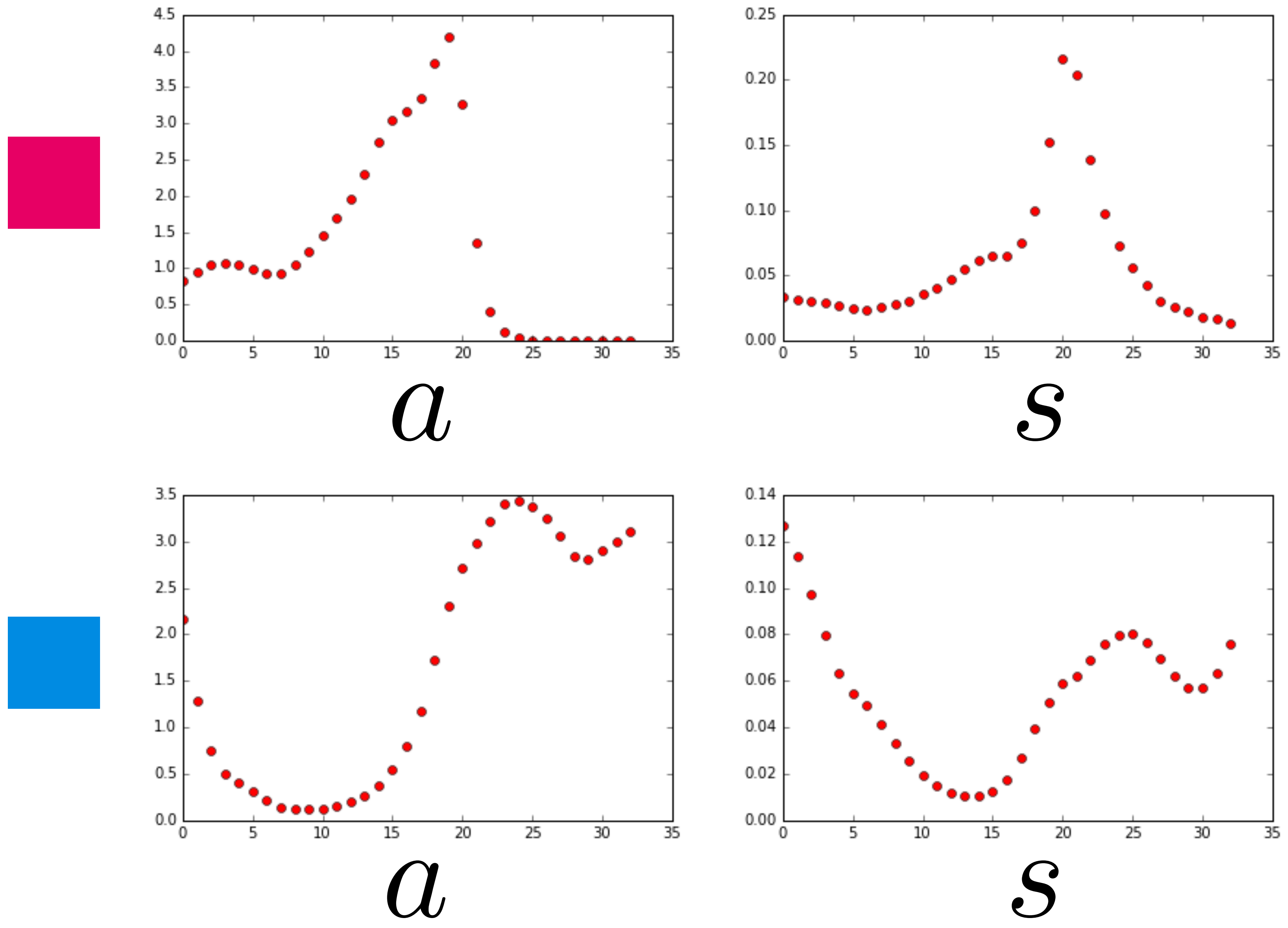
Background: Mixing multispectral pigments

$$I_{RGB} = \phi(km(a, s, t = 1, \xi = 1))$$



Background: Mixing multispectral pigments

$$I_{RGB} = \phi(km(a, s, t = 1, \xi = 1))$$

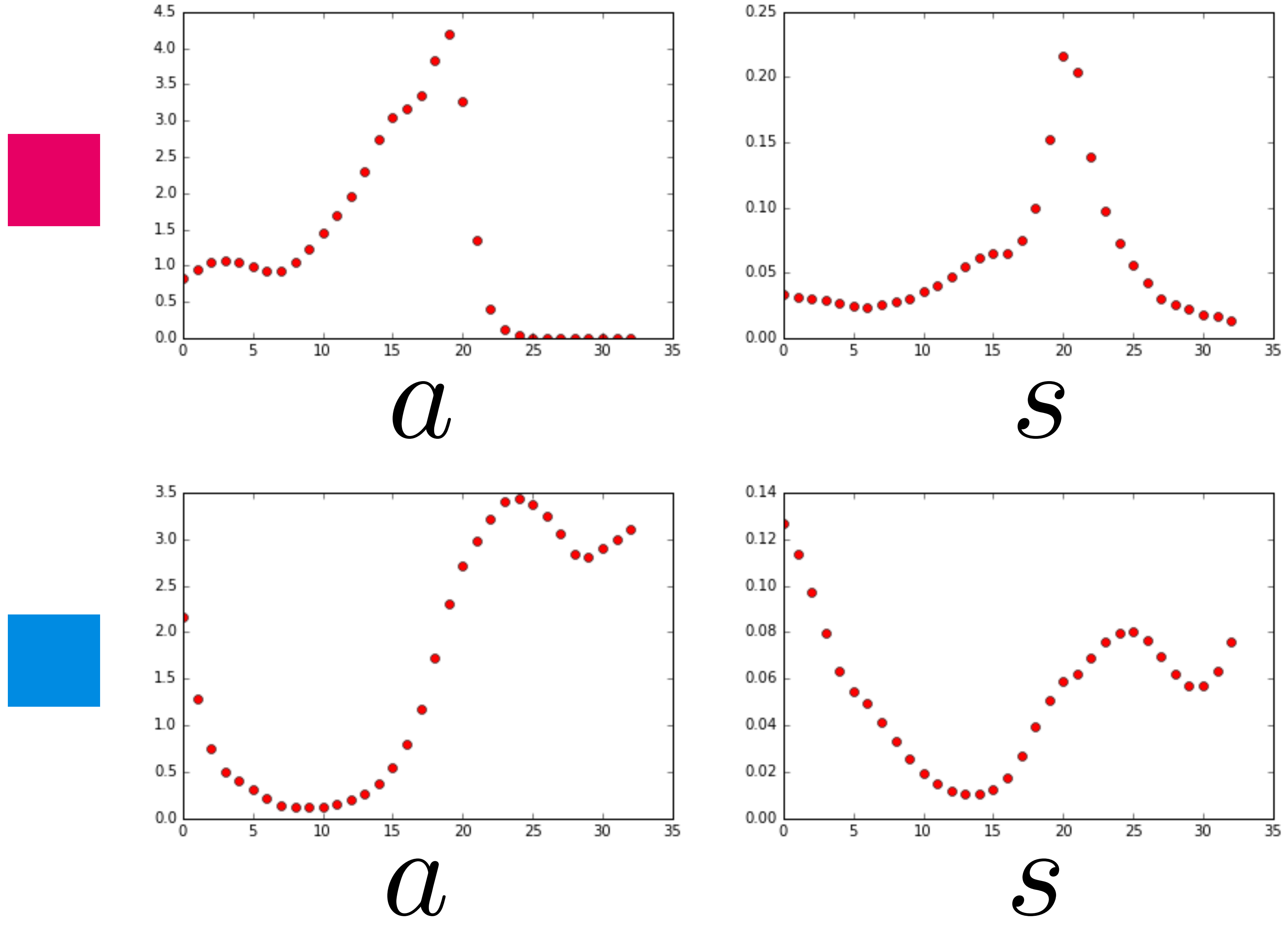


$$a_{mix} = \sum w_i a_i$$

$$s_{mix} = \sum w_i s_i$$

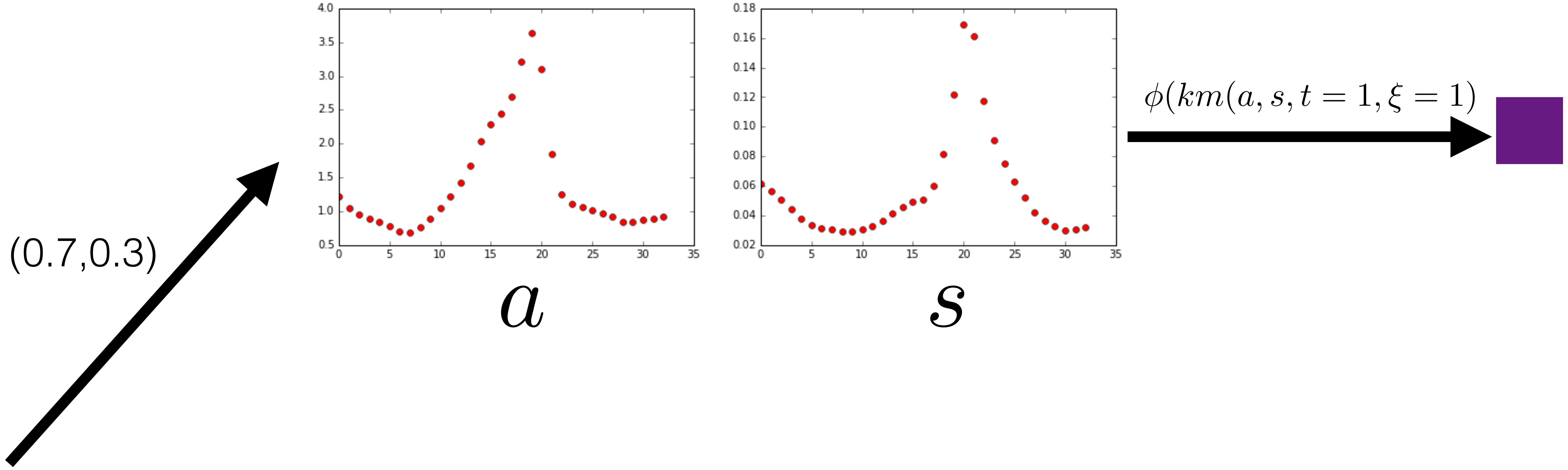
Background: Mixing multispectral pigments

$$I_{RGB} = \phi(km(a, s, t = 1, \xi = 1))$$



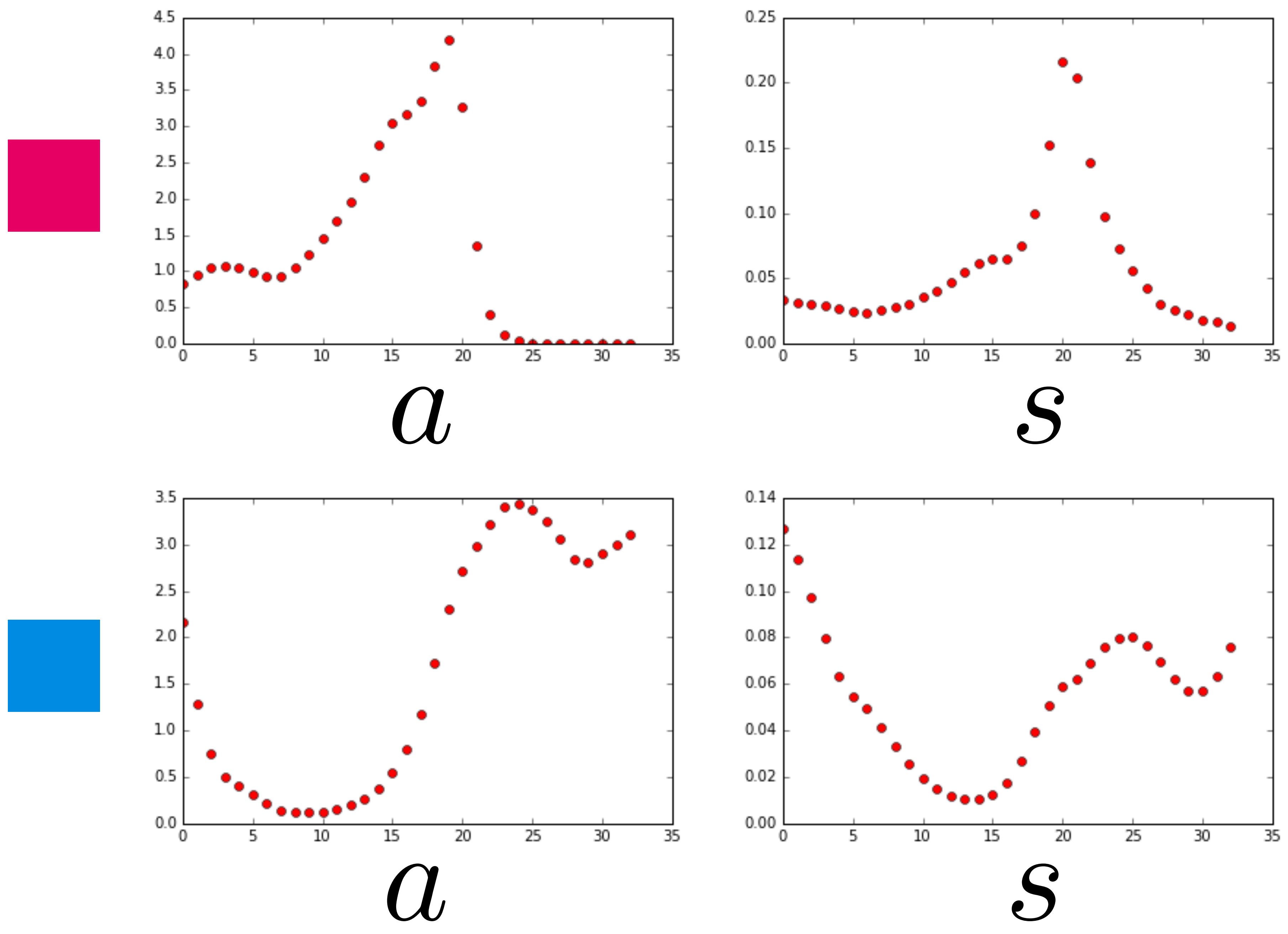
$$a_{mix} = \sum w_i a_i$$

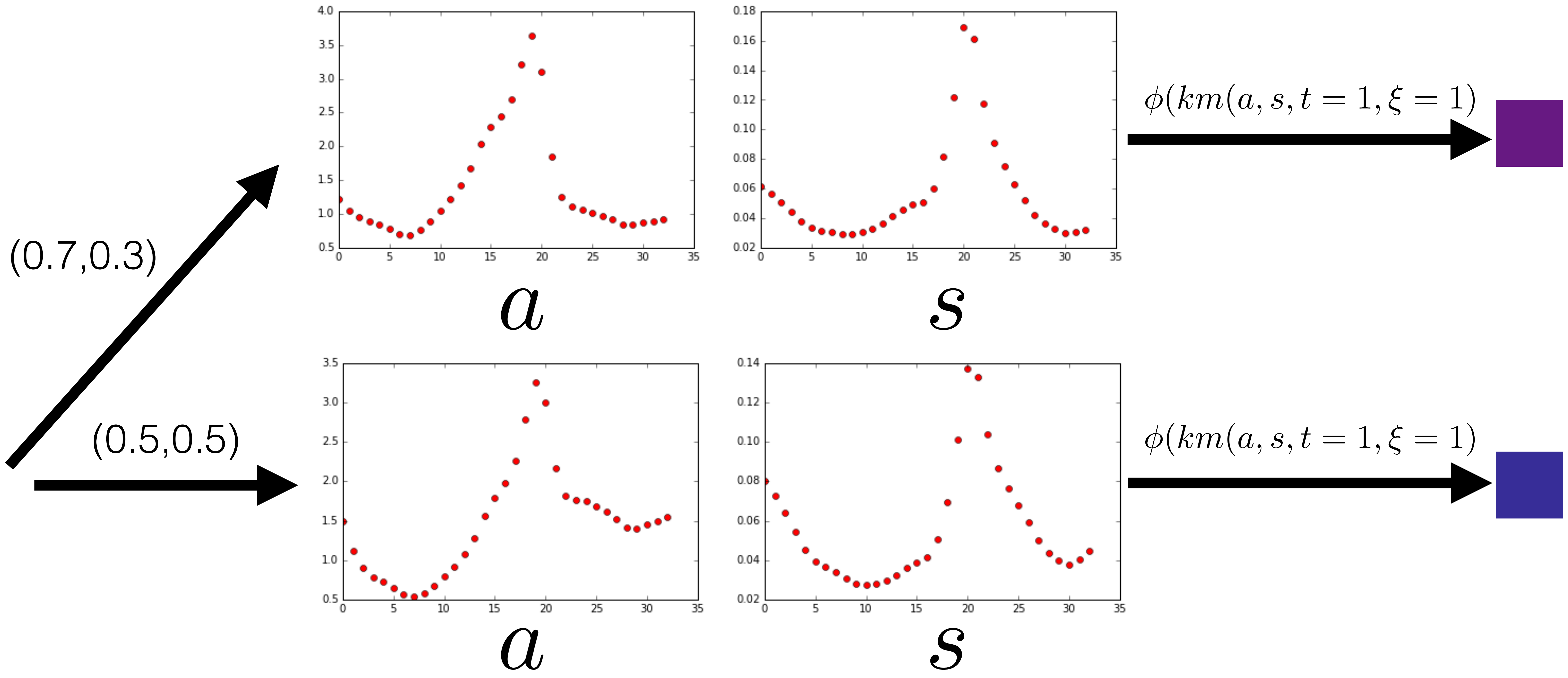
$$s_{mix} = \sum w_i s_i$$



Background: Mixing multispectral pigments

$$I_{RGB} = \phi(km(a, s, t = 1, \xi = 1))$$



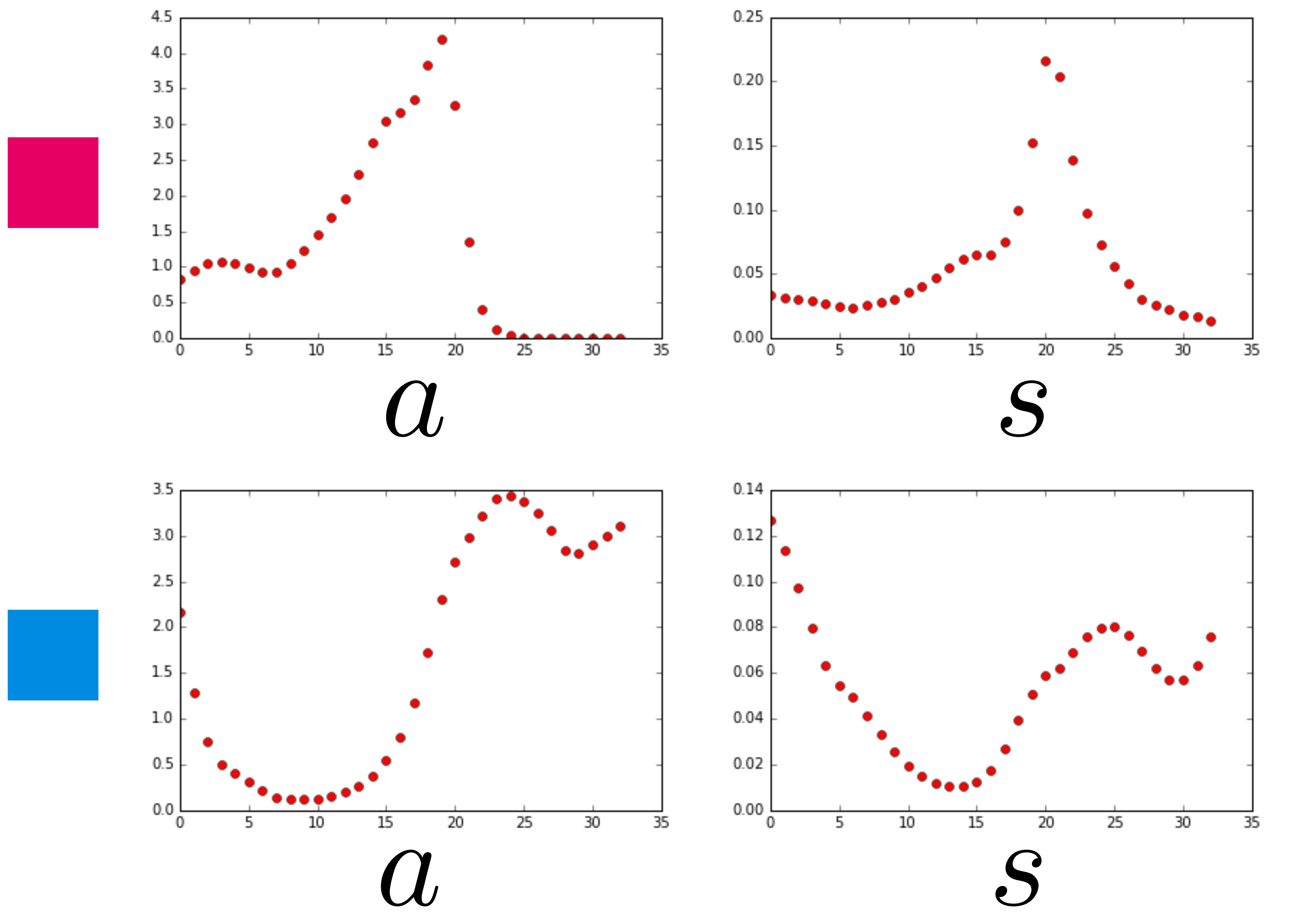


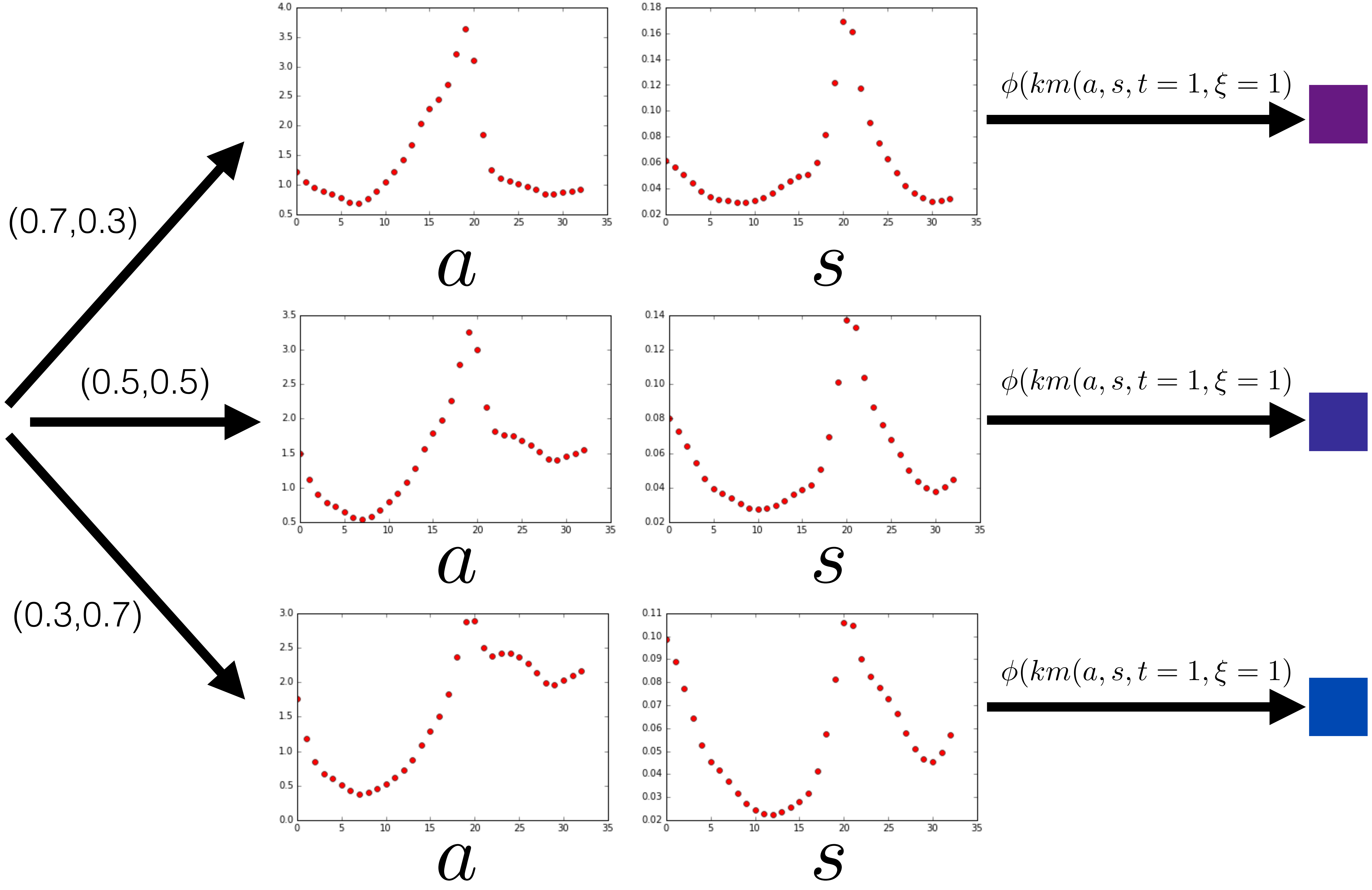
$$a_{mix} = \sum w_i a_i$$

$$s_{mix} = \sum w_i s_i$$

Background: Mixing multispectral pigments

$$I_{RGB} = \phi(km(a, s, t = 1, \xi = 1))$$

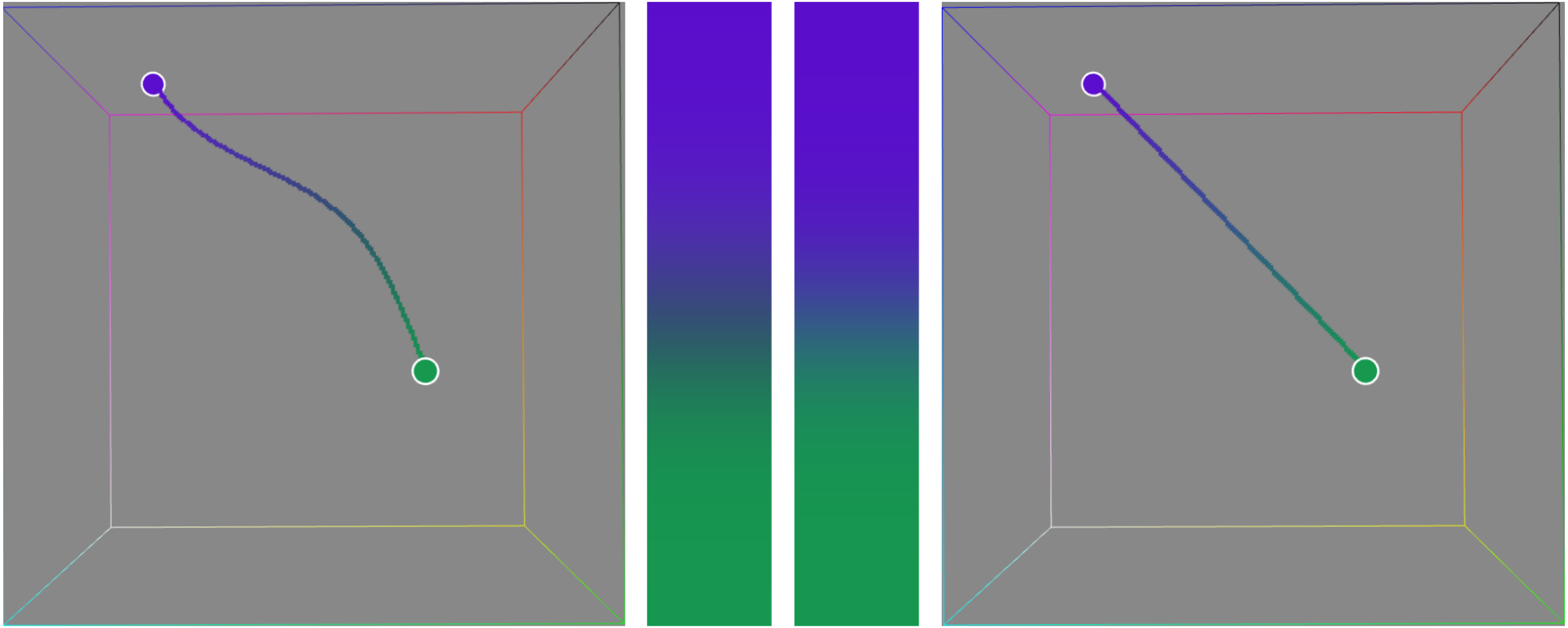




$$a_{mix} = \sum w_i a_i$$

$$s_{mix} = \sum w_i s_i$$

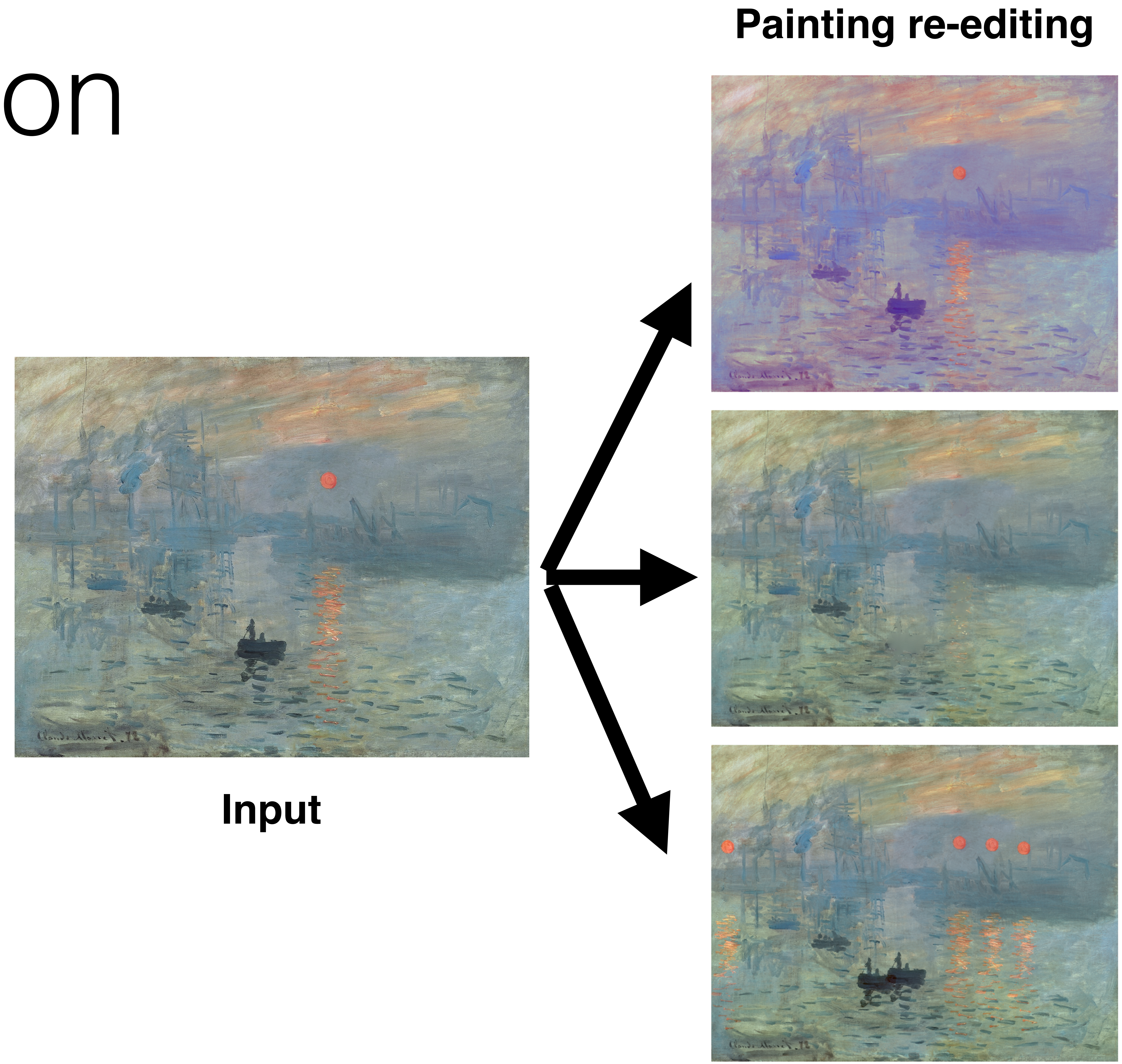
Background: Kubelka-Munk Mixing Model



Multispectral KM Mixing

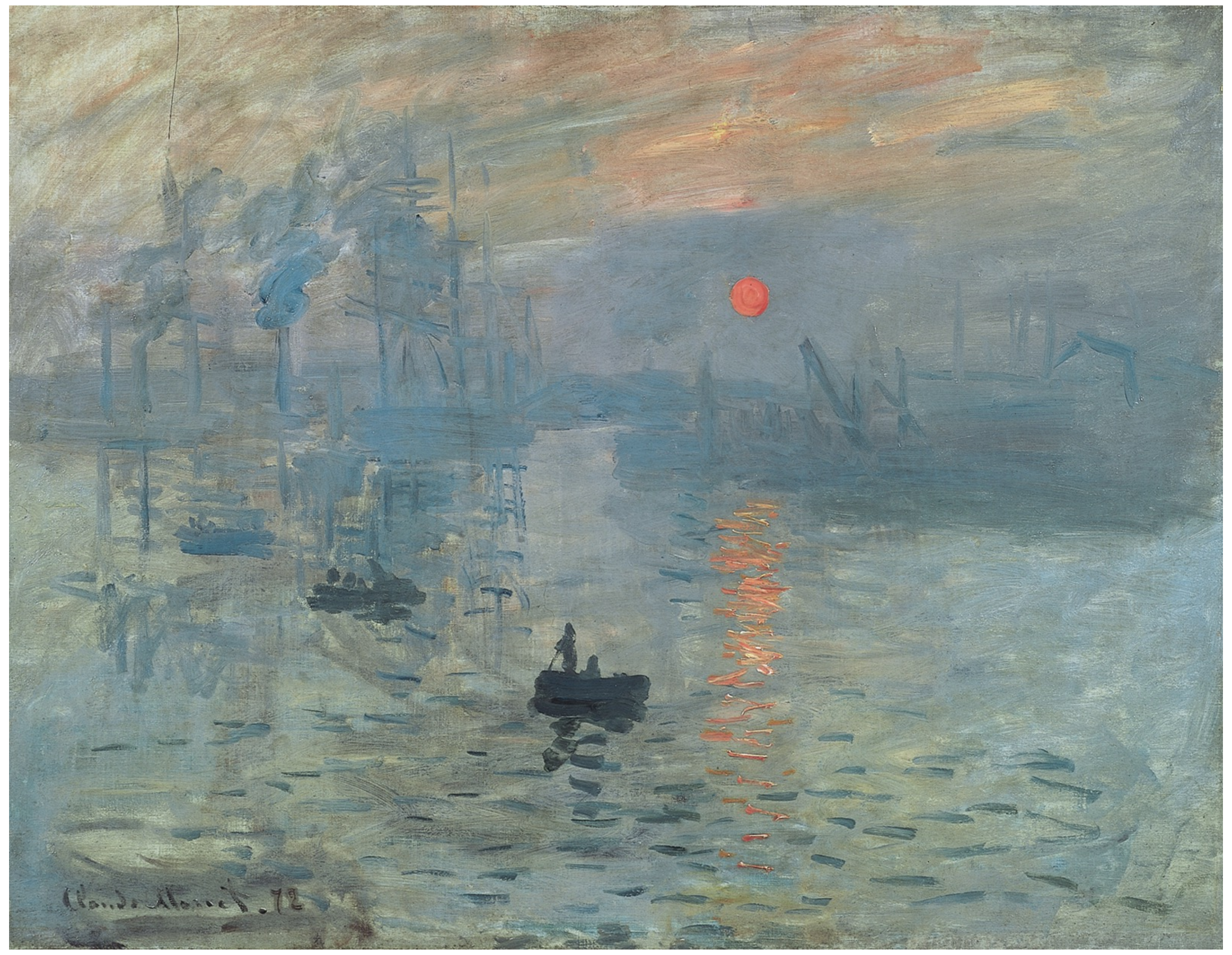
Linear RGB Mixing

Motivation



Motivation

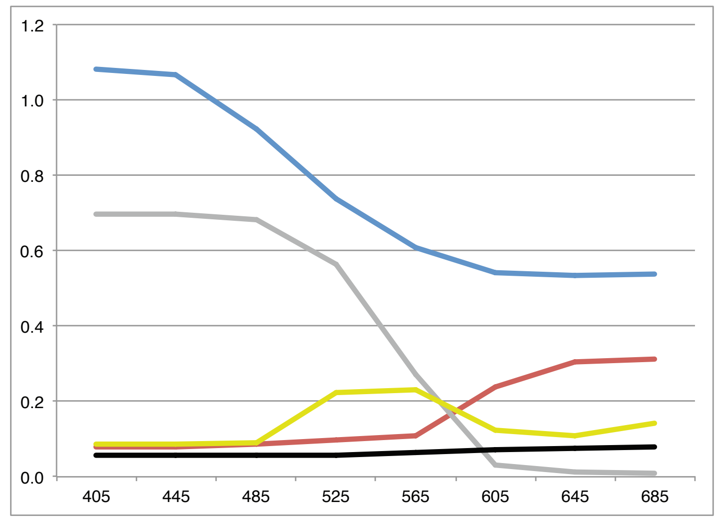
Primary pigments **Mixing weights** **Painting re-editing**

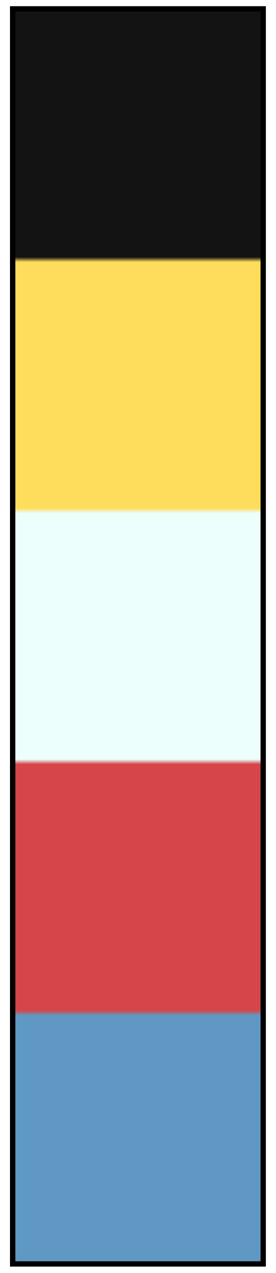


Input

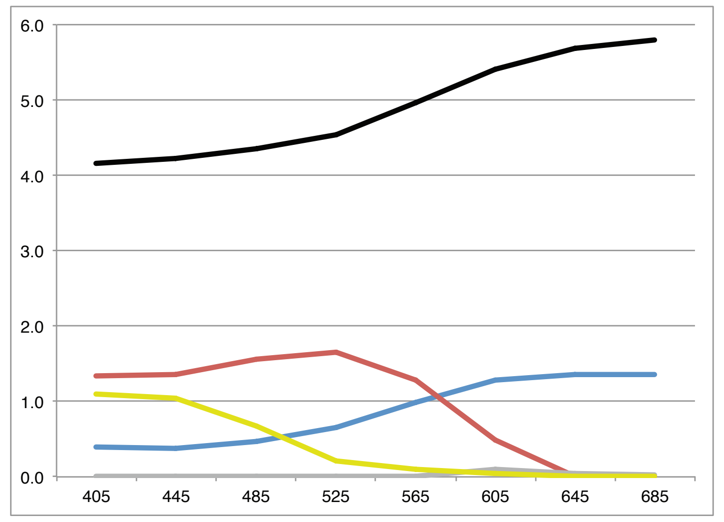
Decomposition

Scattering

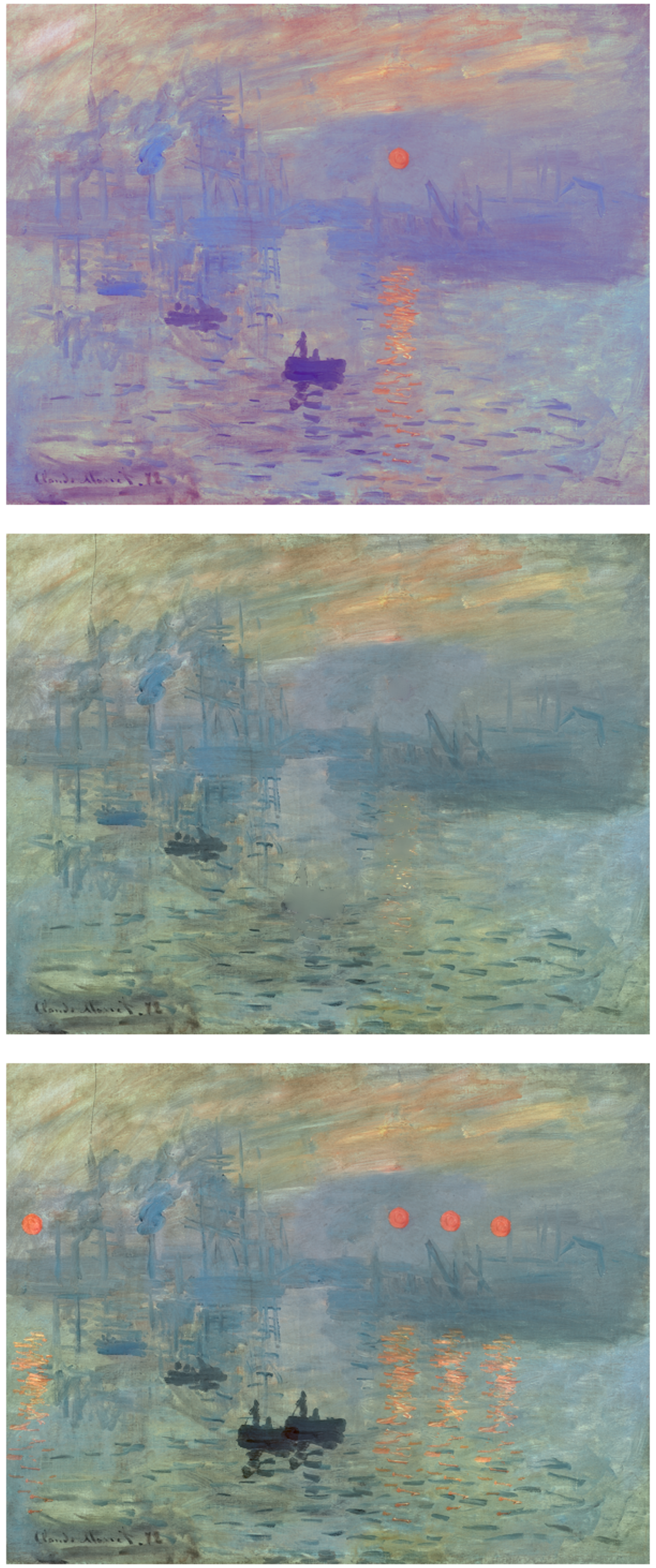




Absorption



×



Related Work

- Digital palette based editing.
    - Chang et al. 2015; Tan et al. 2016; Lin et al. 2017; Zhang et al. 2017, Aksoy et al. 2017.

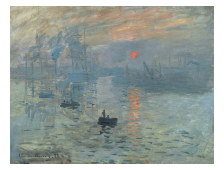
Decomposing Images into Layers via RGB-space Geometry (Tan et al. 2016)

Related Work

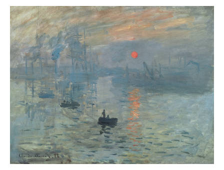
- Kubelka-Munk model based editing.
  - Curtis et al. 1997; IMPaSTo (Baxter et al. 2004); Okumura et al. 2005; Zhao et al. 2008; RealPigment (Lu et al. 2014); Abed et al. 2014; Tan et al. 2015; Aharoni-Mack et al. 2017

Pigment-Based Recoloring of Watercolor Paintings (Aharoni-Mack et al. 2017)

Problem Statement

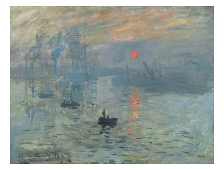
Input: Image pixels' RGB colors: **I**. 

Problem Statement

Input: Image pixels' RGB colors: **I**. 

Output: Primary multispectral pigments: **H**=[**A**|**S**]. Their per-pixel mixing weights: **W**.

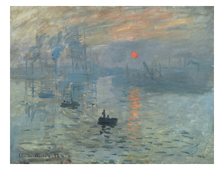
Problem Statement

Input: Image pixels' RGB colors: **I**. 

Output: Primary multispectral pigments: **H**=[**A**|**S**]. Their per-pixel mixing weights: **W**.

$$\mathbf{I} = \phi(km(\mathbf{WH}, t, \xi))$$

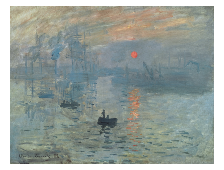
Problem Statement

Input: Image pixels' RGB colors: **I**. 

Output: Primary multispectral pigments: **H**=[**A**|**S**].  Their per-pixel mixing weights: **W**. 

$$\mathbf{I} = \phi(km(\mathbf{WH}, t, \xi = 1))$$

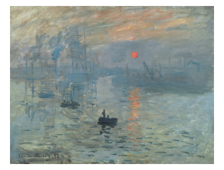
Problem Statement

Input: Image pixels' RGB colors: **I**. 

Output: Primary multispectral pigments: **H**=[**A**|**S**]. Their per-pixel mixing weights: **W**.

$$\mathbf{I} = \phi(km(\mathbf{WH}, t = 1, \xi = 1))$$

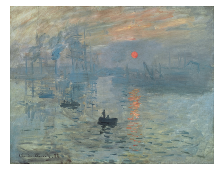
Problem Statement

Input: Image pixels' RGB colors: **I**. 

Output: Primary multispectral pigments: **H**=[**A**|**S**]. Their per-pixel mixing weights: **W**.

$$\mathbf{I} = \phi(km(\mathbf{WH}))$$

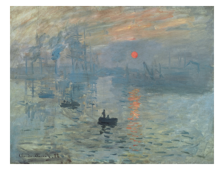
Problem Statement

Input: Image pixels' RGB colors: **I**. 

Output: Primary multispectral pigments: **H**=[**A**|**S**].  Their per-pixel mixing weights: **W**. 

$$||\mathbf{I} - \phi(km(\mathbf{WH}))||^2$$

Problem Statement

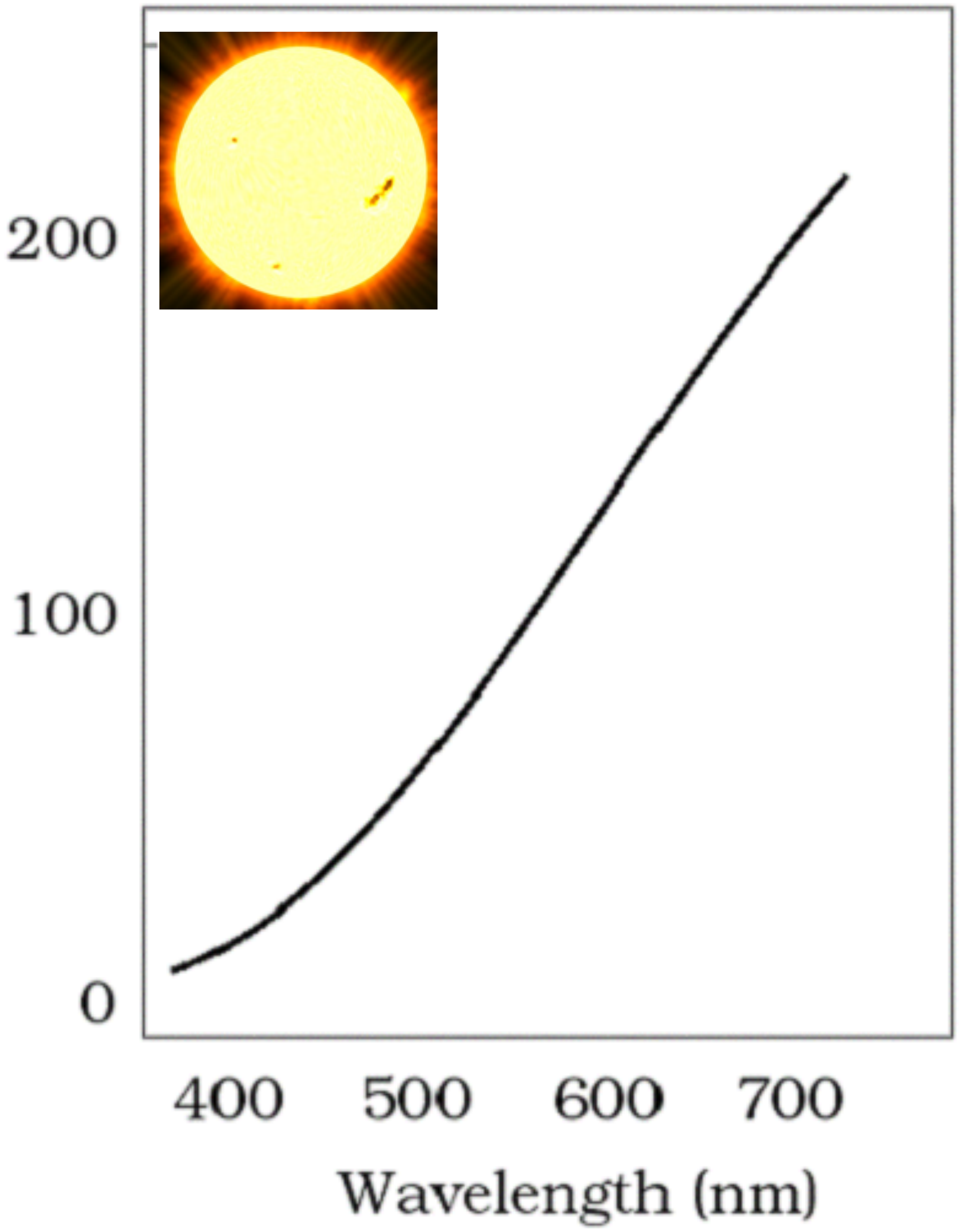
Input: Image pixels' RGB colors: **I**. 

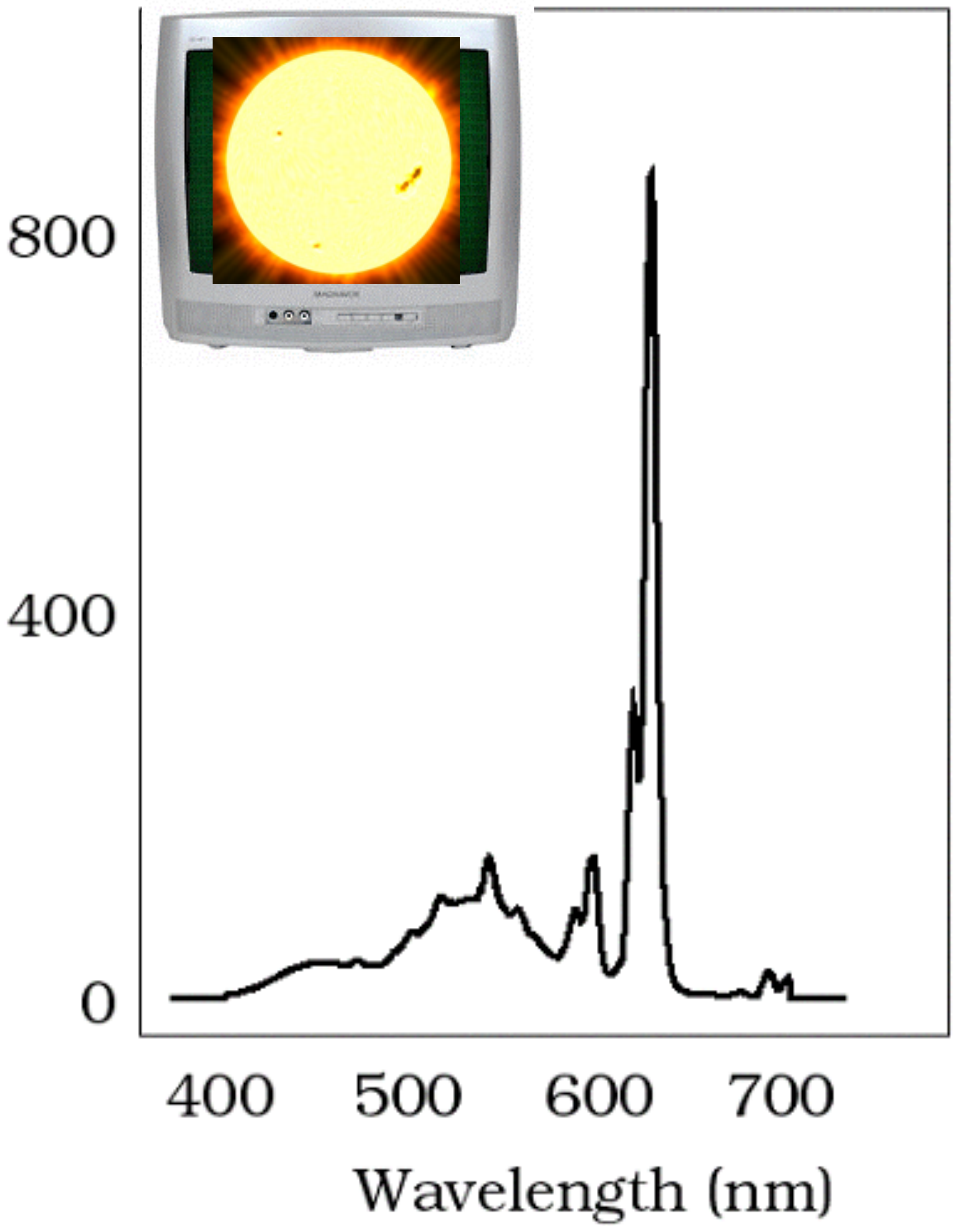
Output: Primary multispectral pigments: **H**=[**A**|**S**]. Their per-pixel mixing weights: **W**.

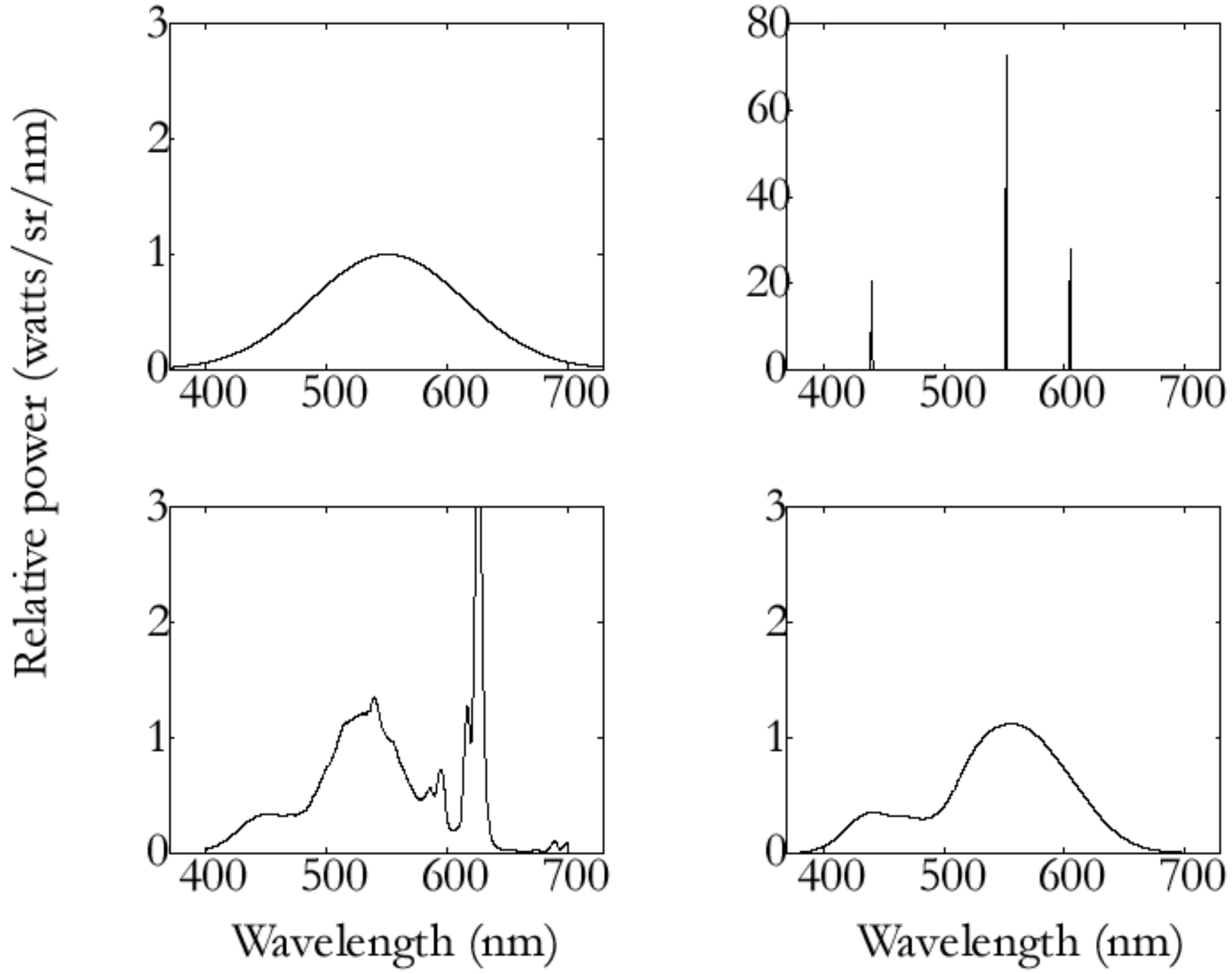
$$||\mathbf{I} - \phi(km(\mathbf{WH}))||^2$$

It is under-constrained, and there are two additional challenges!

Challenge 1: Metamerism





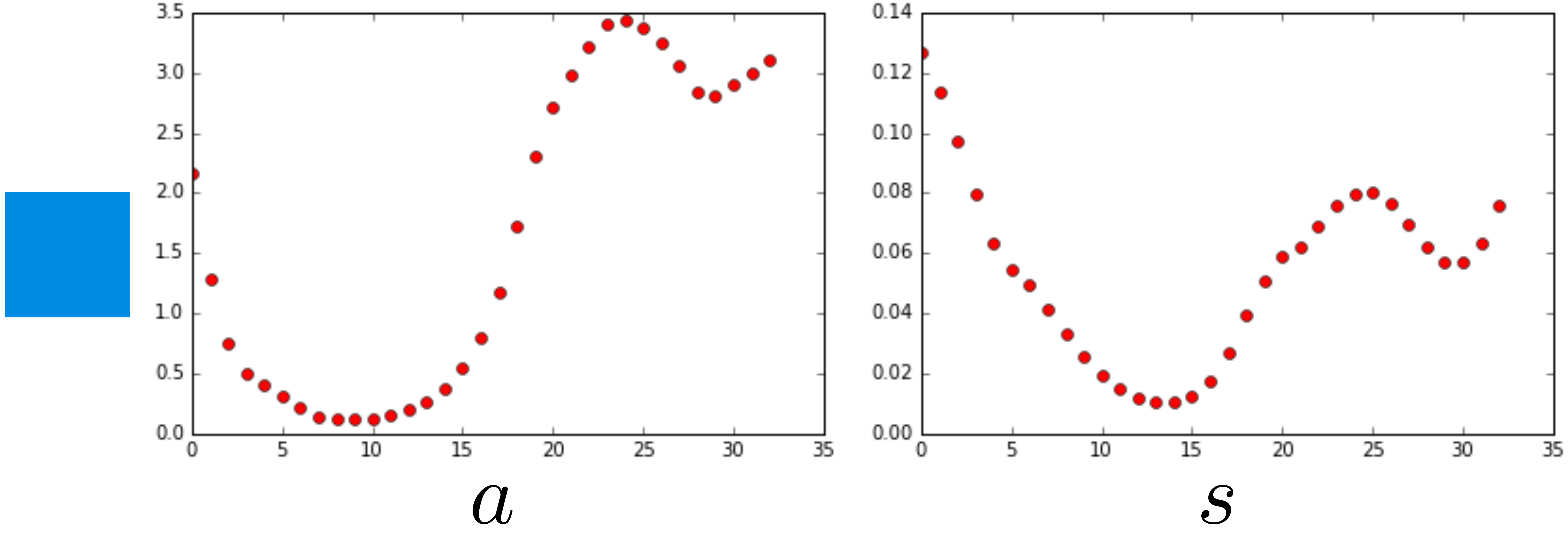


Brian Wandell

Solution: Smoothness Regularization

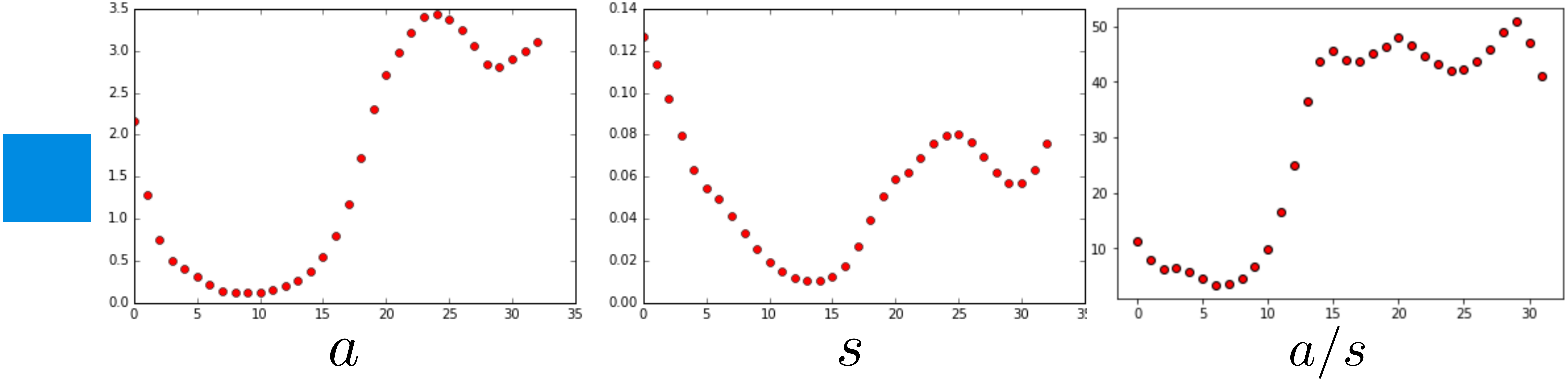
Solution: Smoothness Regularization

Absorption and Scattering curve of each primary pigment should be smooth.



Solution: Smoothness Regularization

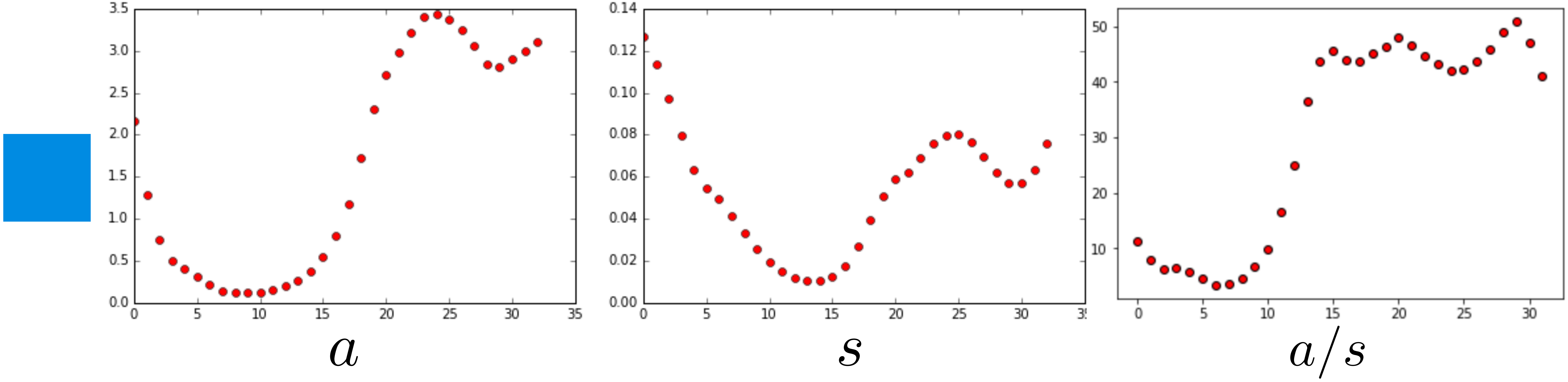
Absorption and Scattering curve of each primary pigment should be smooth.



Absorption and Scattering's division curve should also be smooth.

Solution: Smoothness Regularization

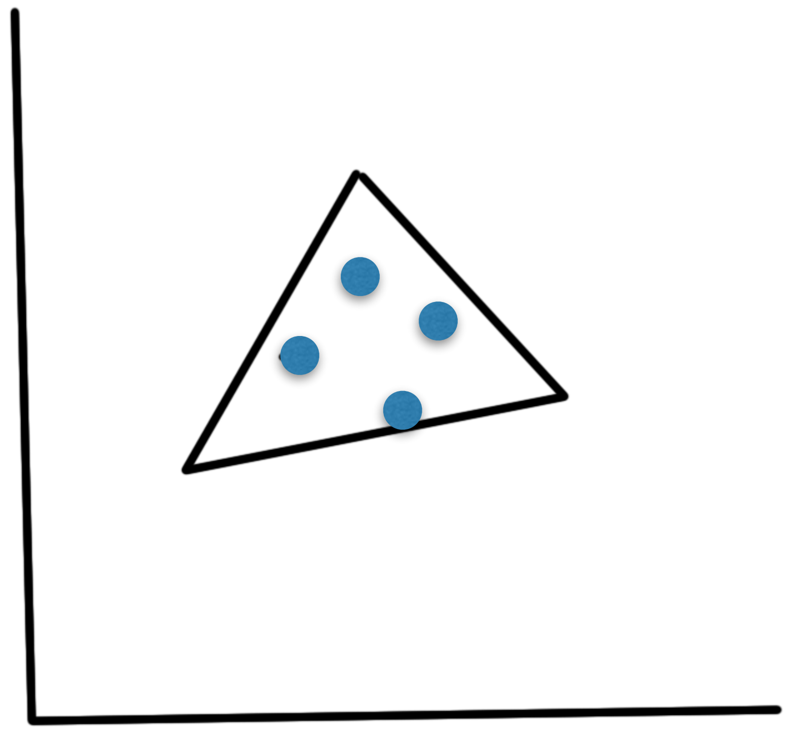
Absorption and Scattering curve of each primary pigment should be smooth.



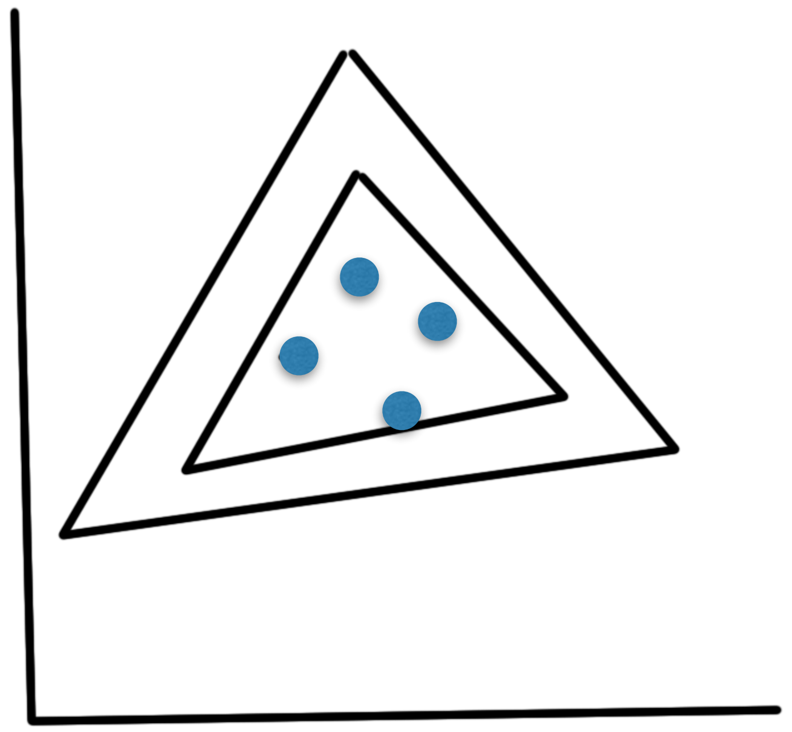
Absorption and Scattering's division curve should also be smooth.

Useful for Metamerism problem!

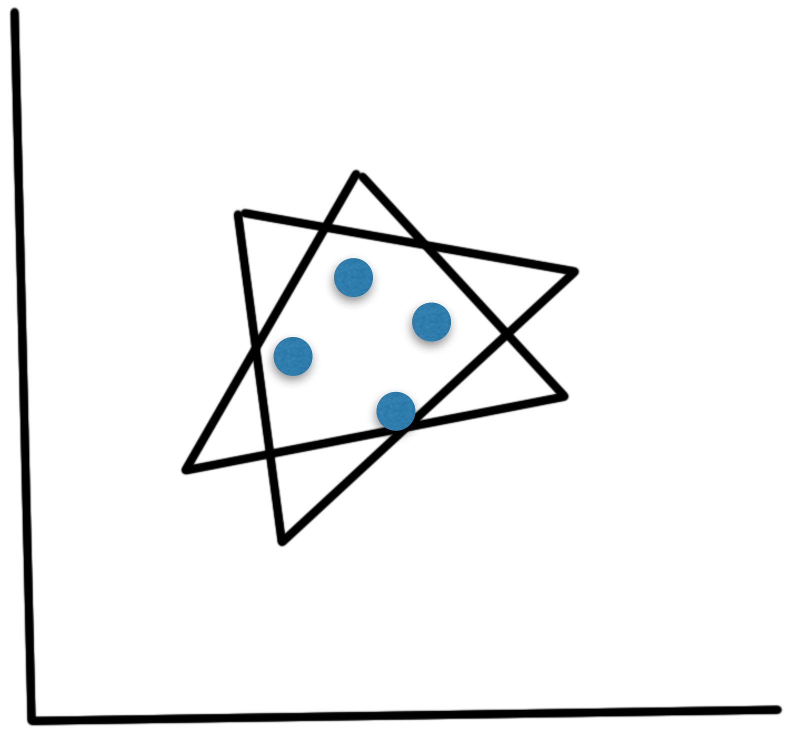
Challenge 2: Solution Space



Gamut H for
4 color points

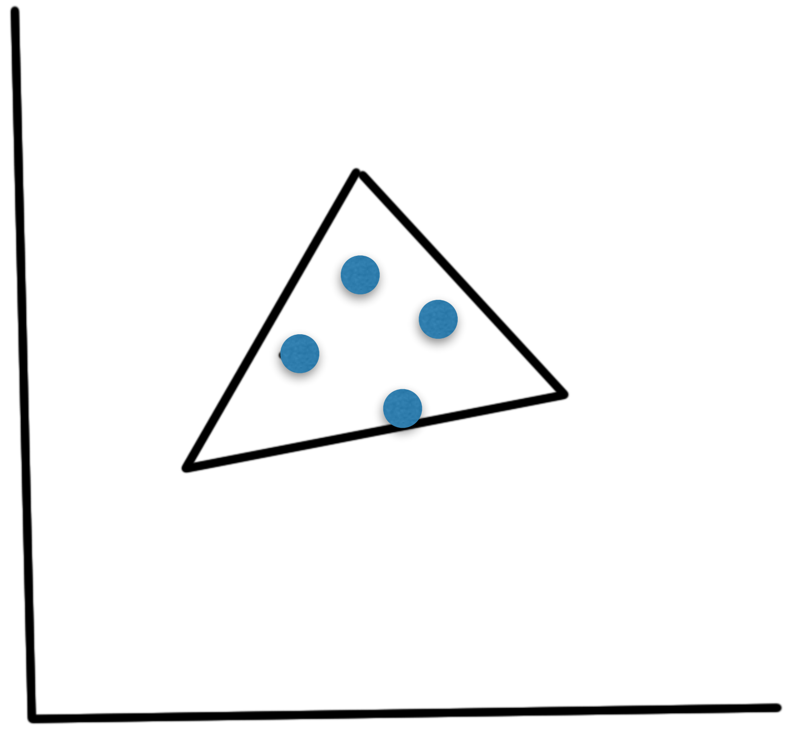


Gamut H1 by
scaling H

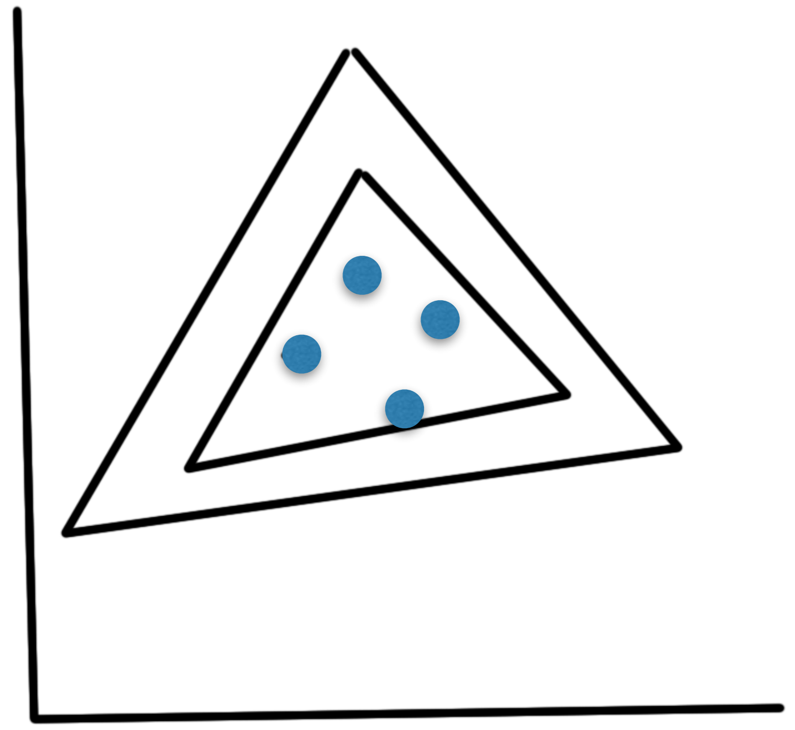


Gamut H2 by
rotating H

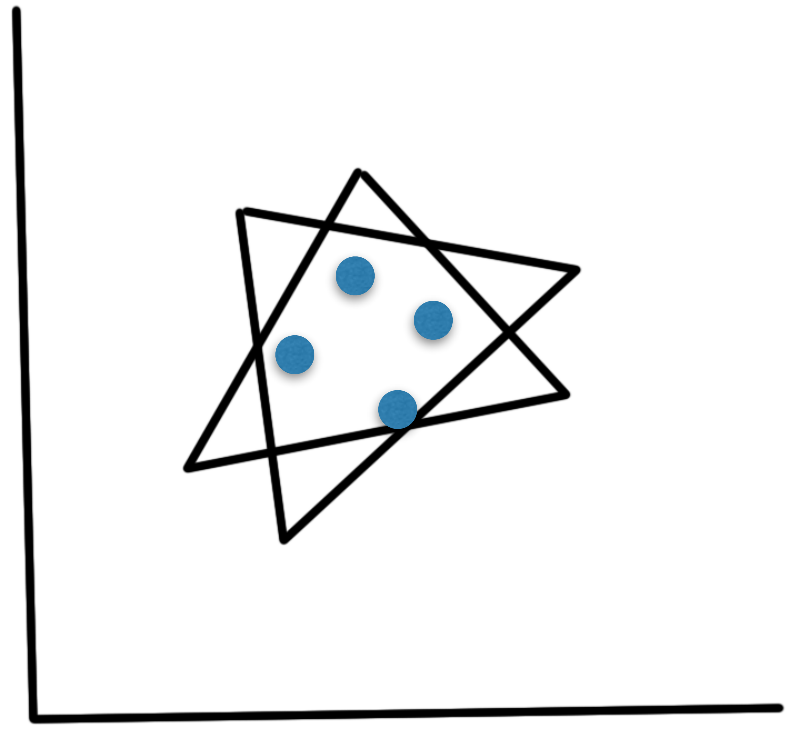
Challenge 2: Solution Space



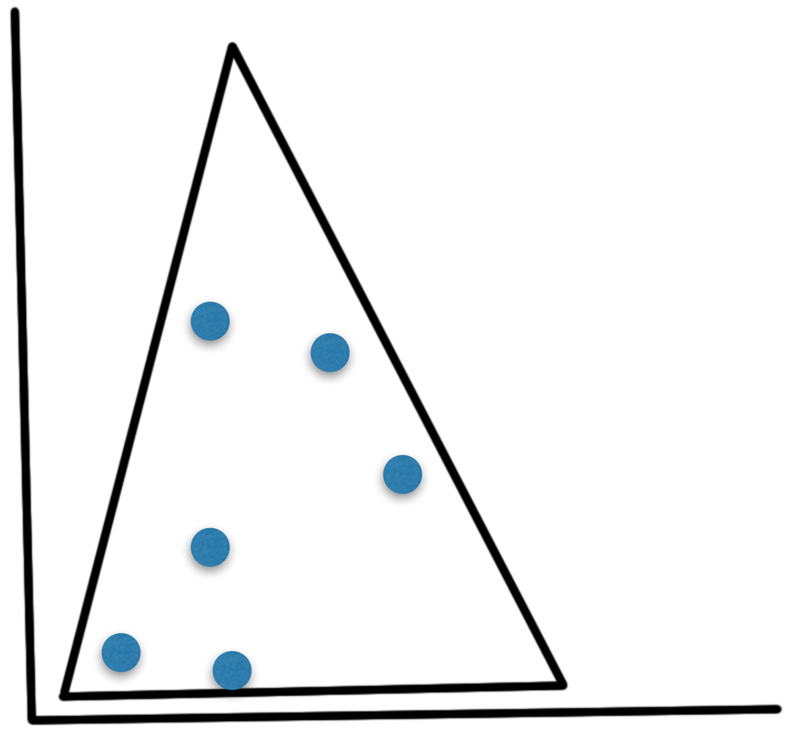
Gamut H for
4 color points



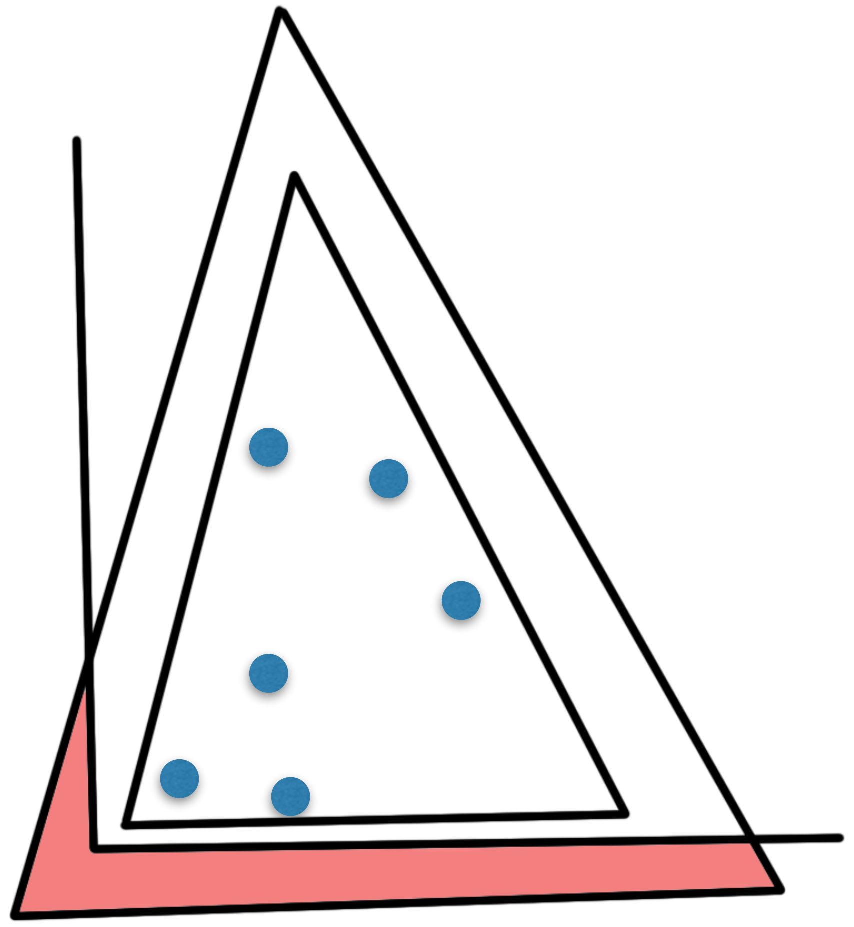
Gamut H1 by
scaling H



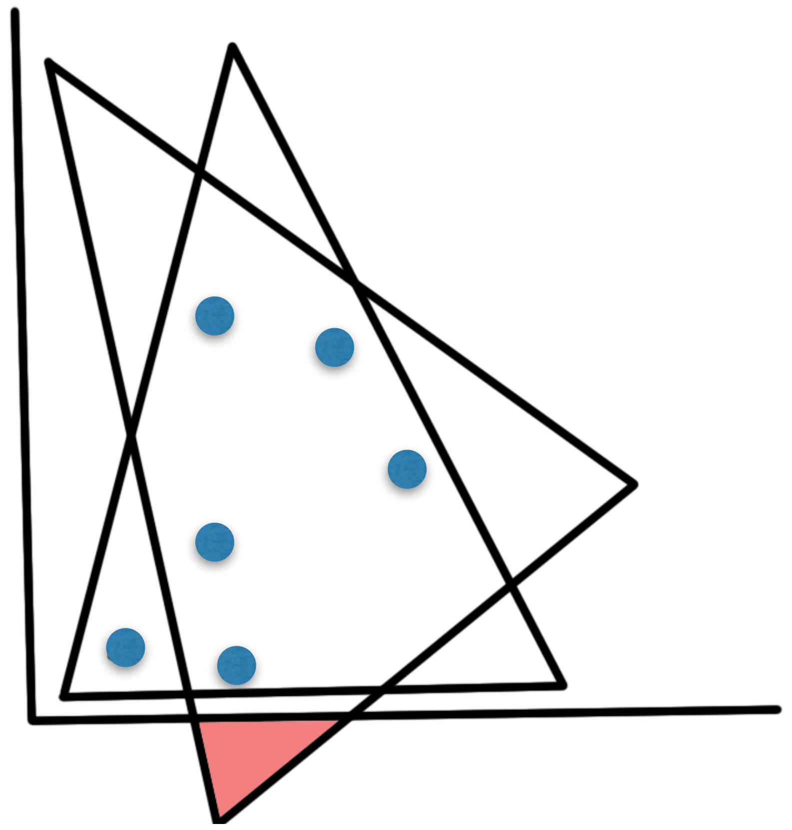
Gamut H2 by
rotating H



Gamut Q for
more points

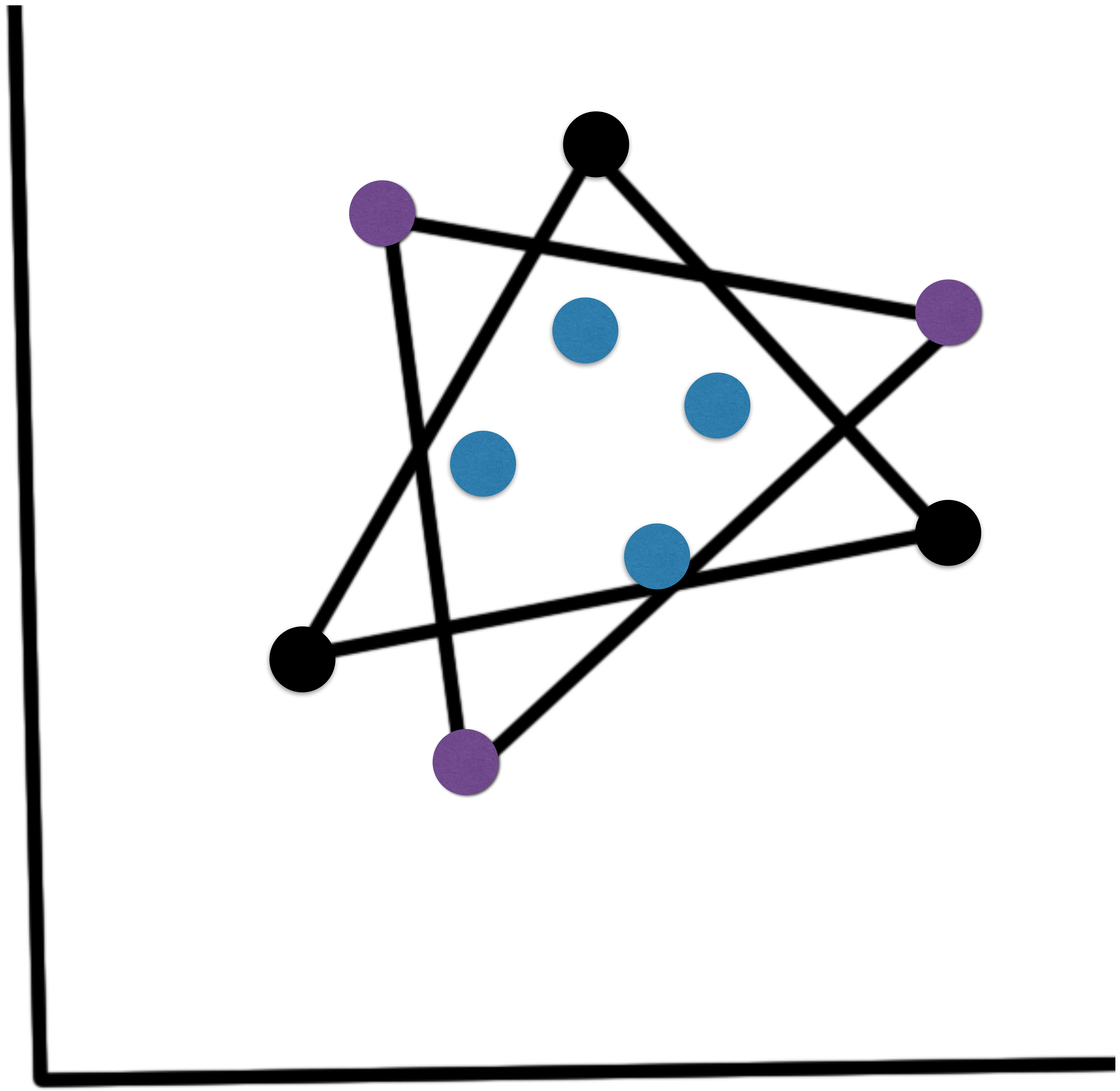


Gamut Q1 by
scaling Q

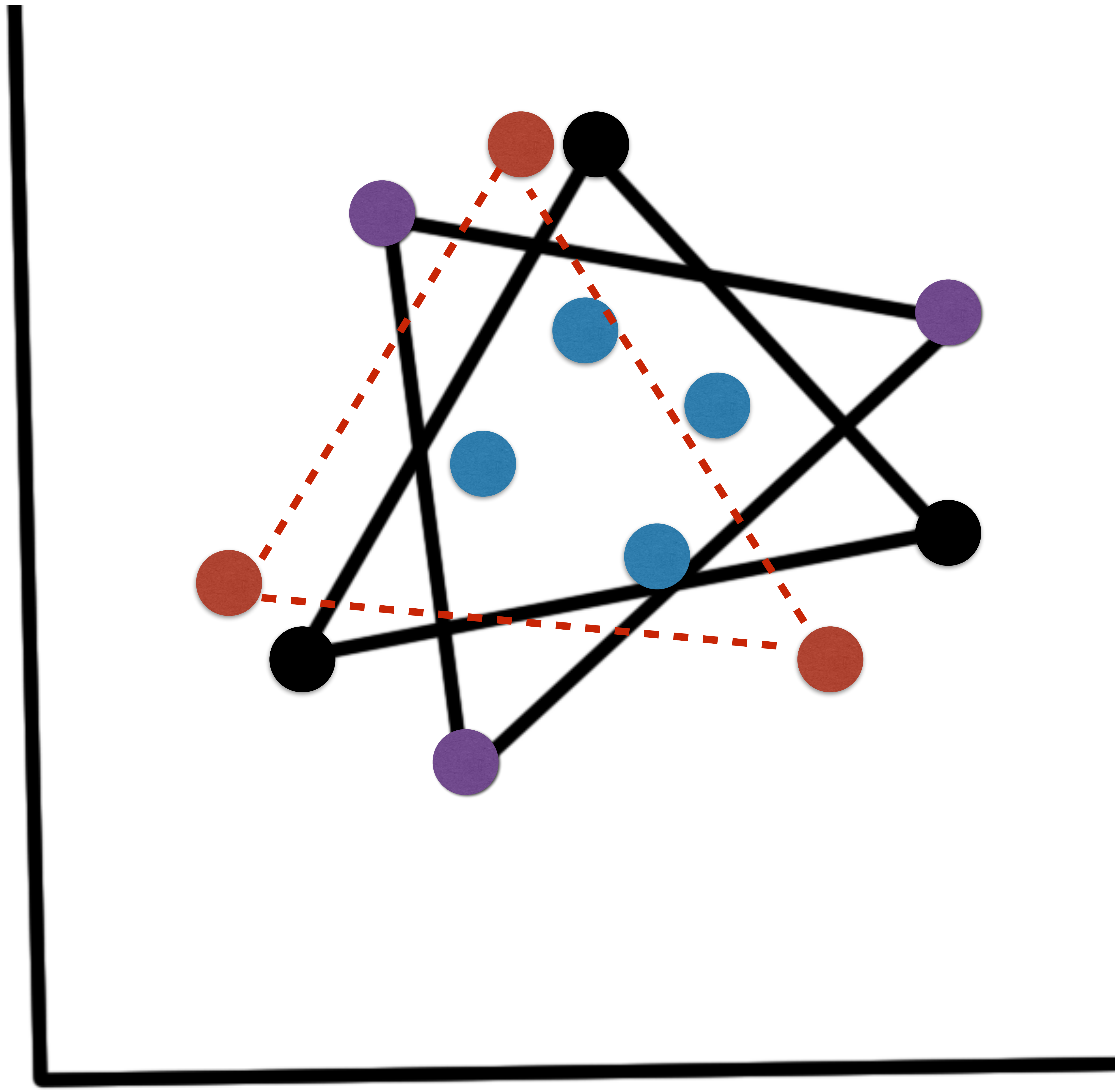


Gamut Q2 by
rotating Q

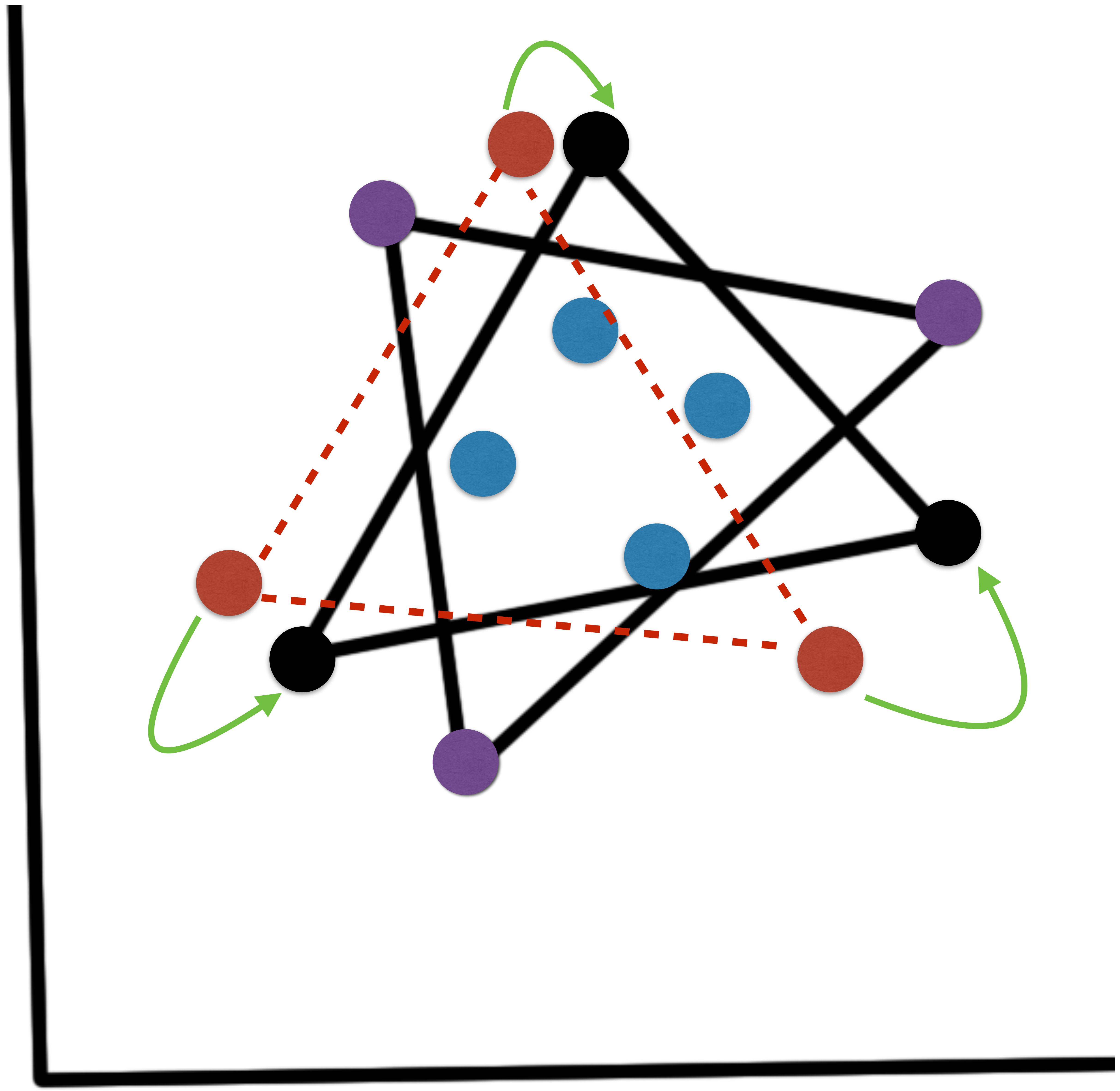
Good Initial values



Good Initial values



Good Initial values



Divide into two subproblems

Directly solving this problem is hard.

Divide into two subproblems

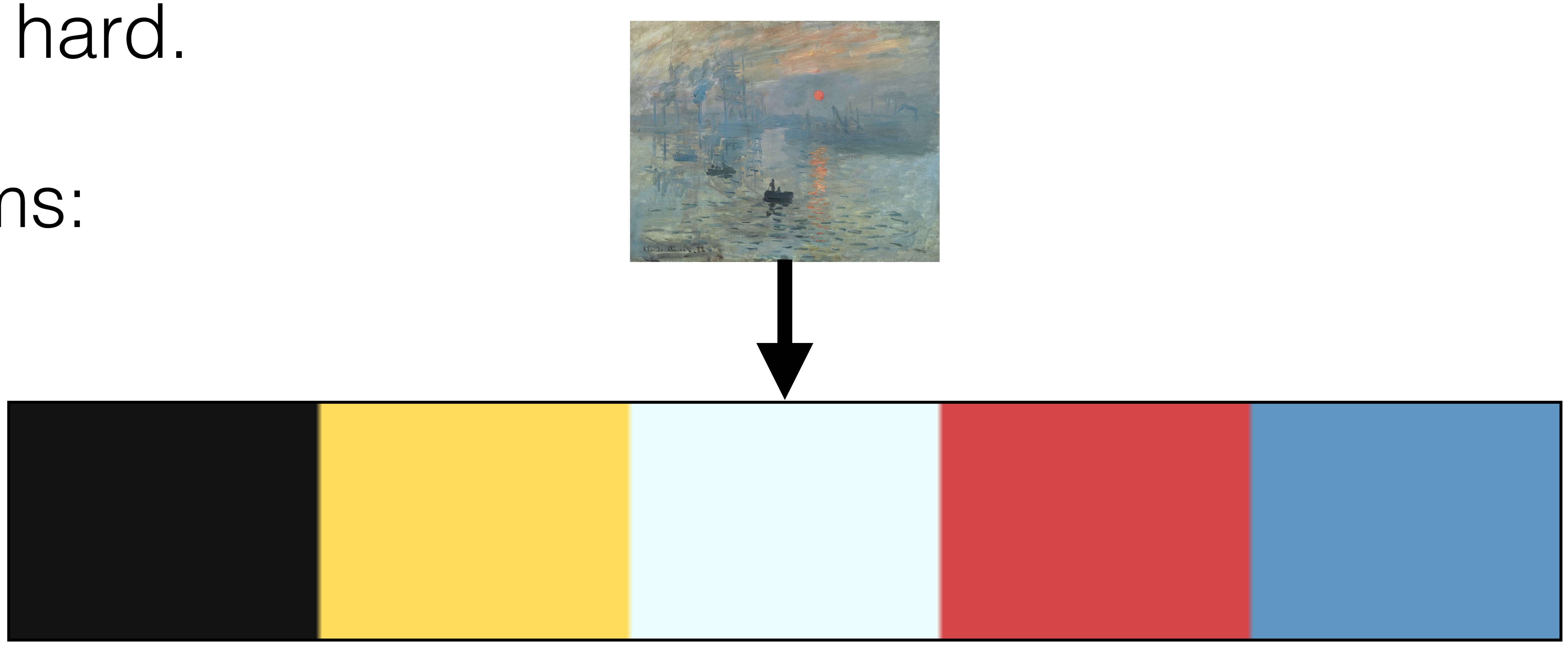
Directly solving this problem is hard.

We divide it into two subproblems:

Divide into two subproblems

Directly solving this problem is hard.

We divide it into two subproblems:

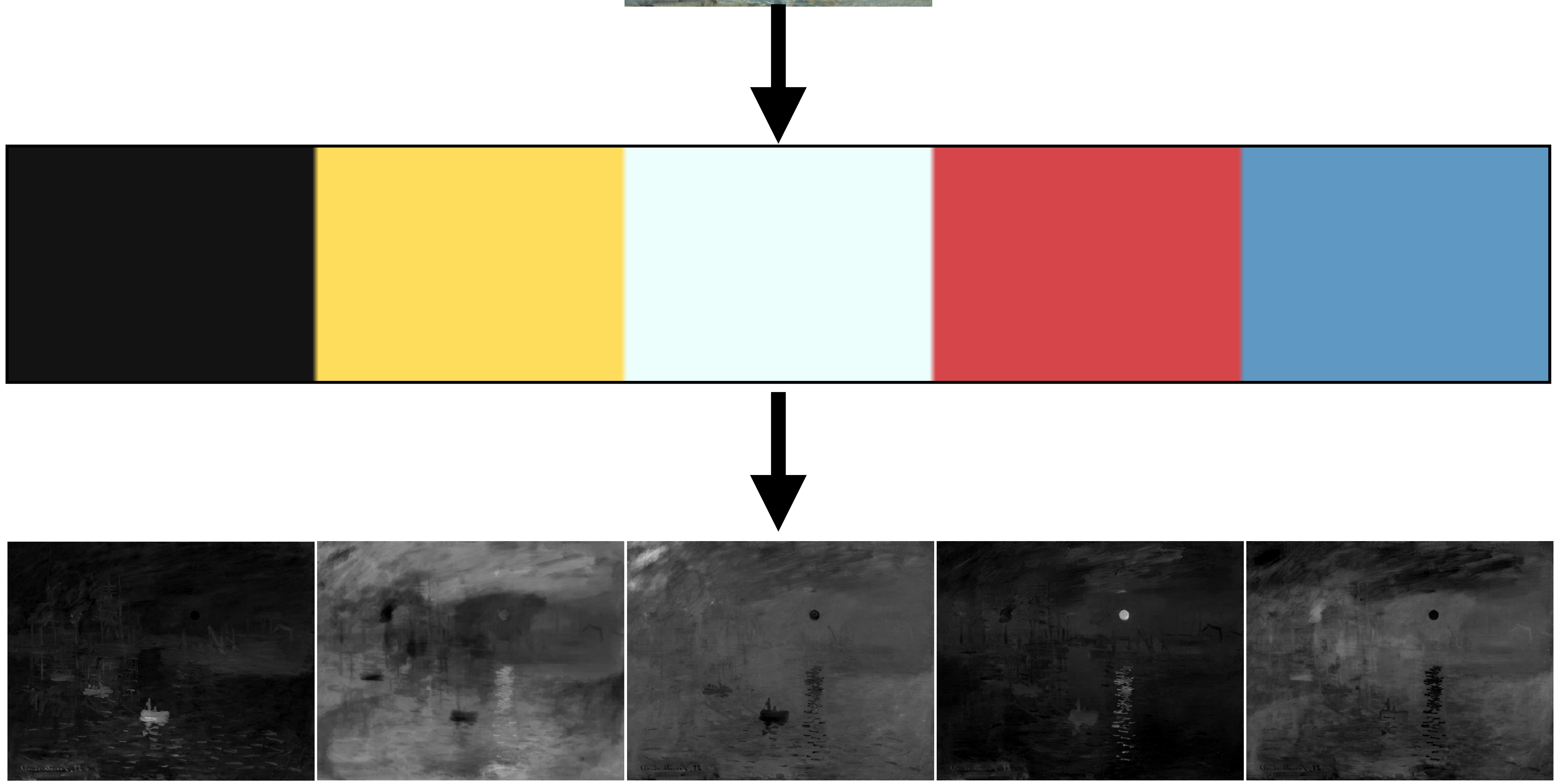


1. Primary pigments extraction

Divide into two subproblems

Directly solving this problem is hard.

We divide it into two subproblems:

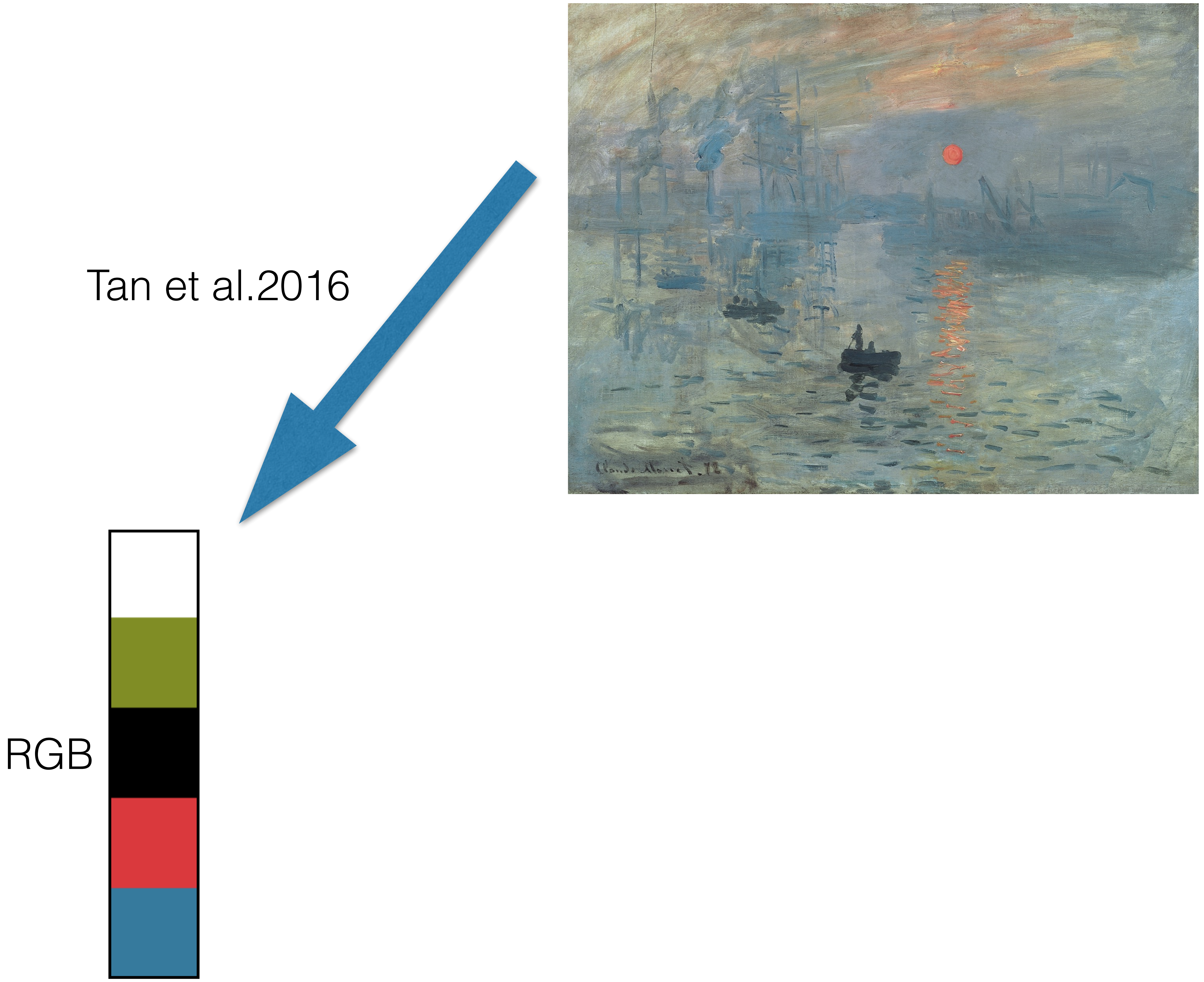


1. Primary pigments extraction

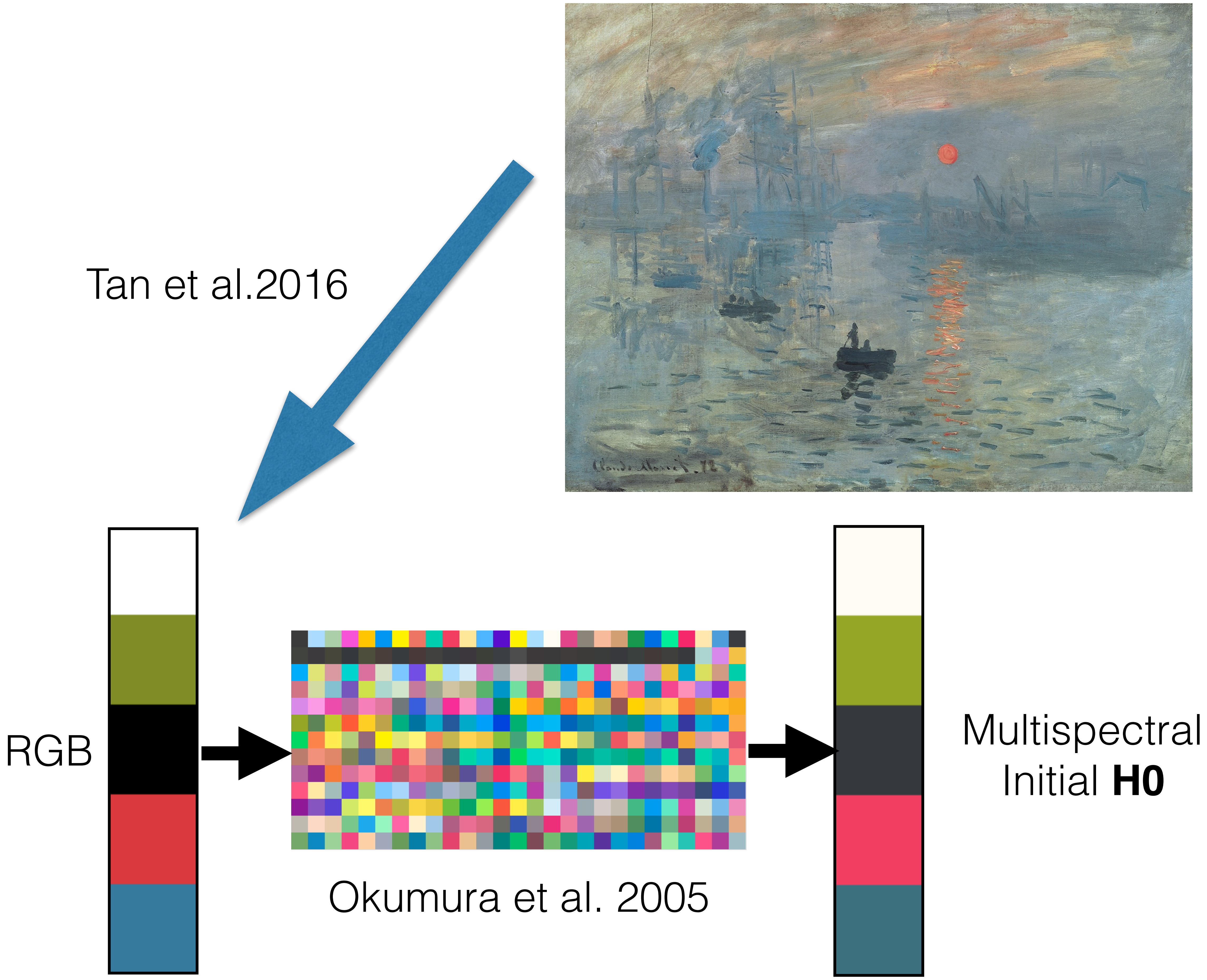
2. Mixing weights extraction

Pigments Extraction

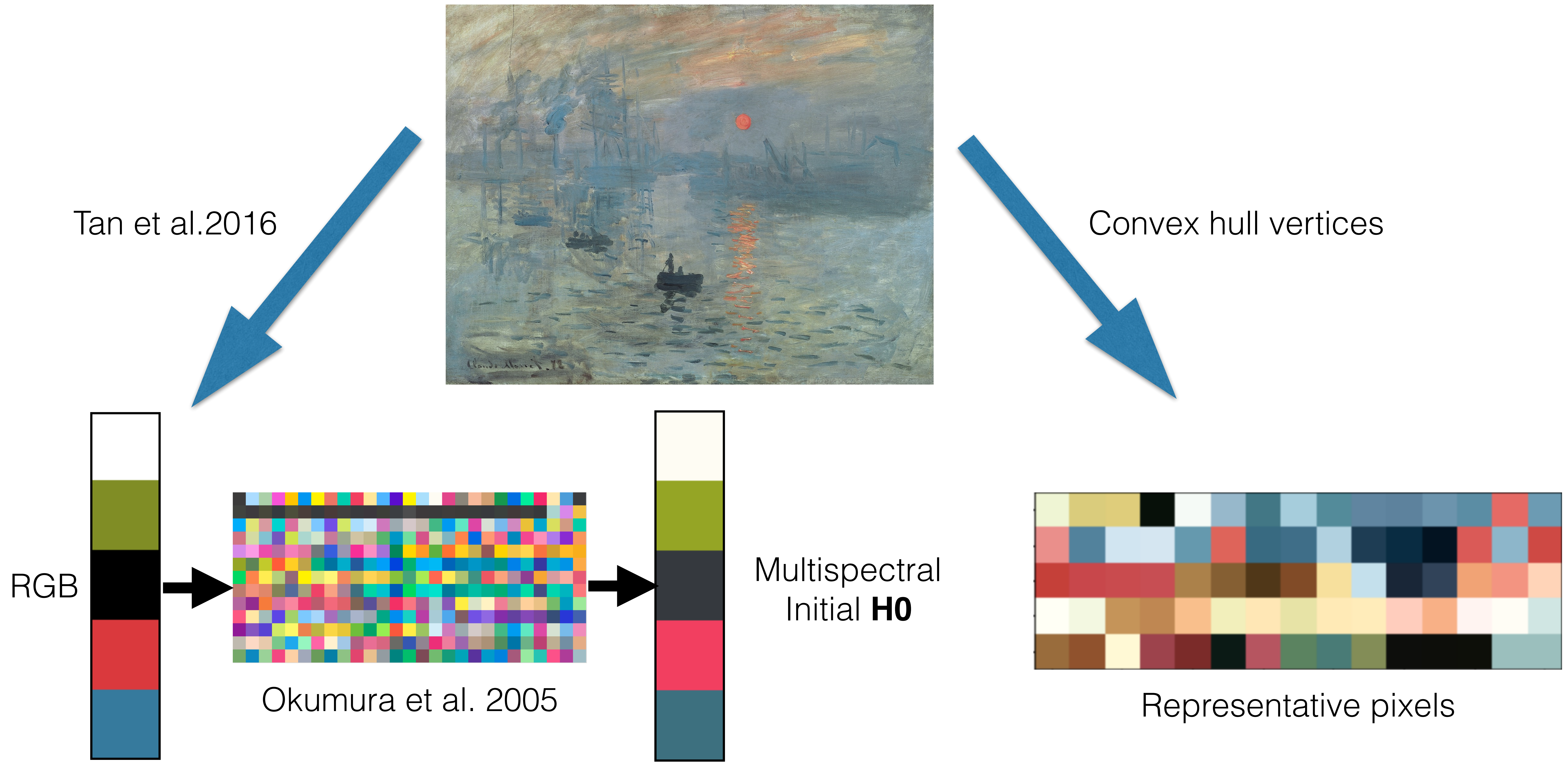
Pigments Extraction



Pigments Extraction



Pigments Extraction

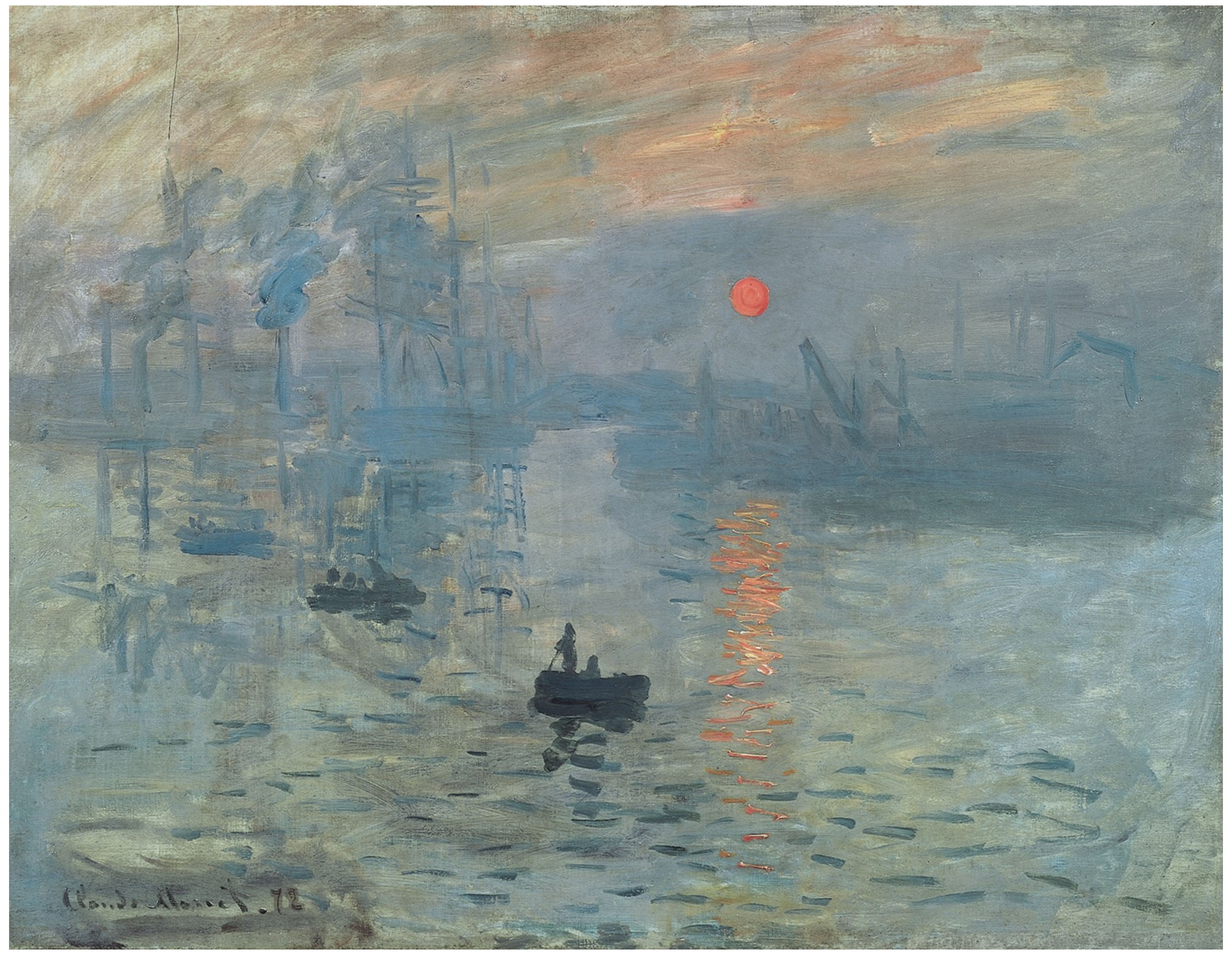


Pigments Extraction

Pigments Extraction

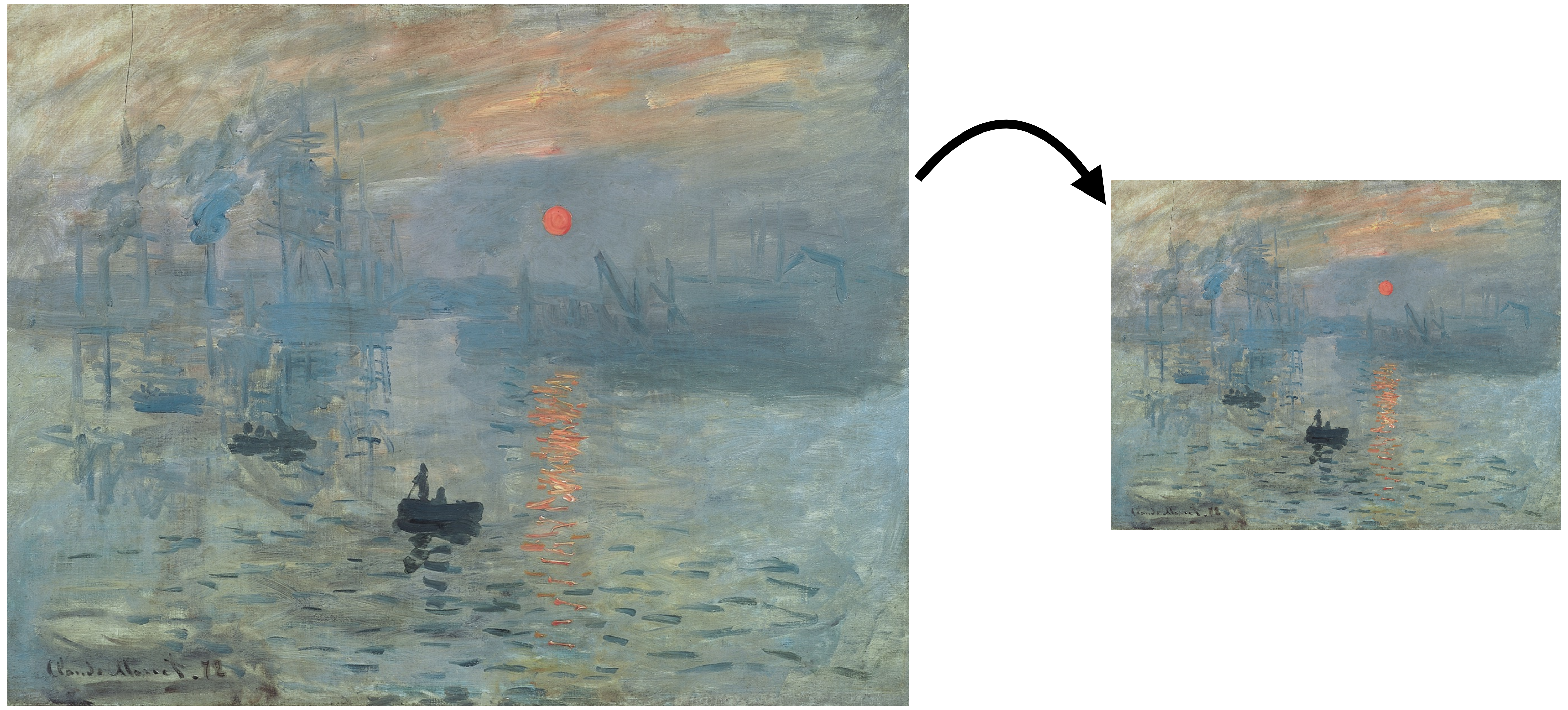
Mixing Weights Extraction

Given primary pigments, find per-pixel mixing weights.



Mixing Weights Extraction

Given primary pigments, find per-pixel mixing weights.

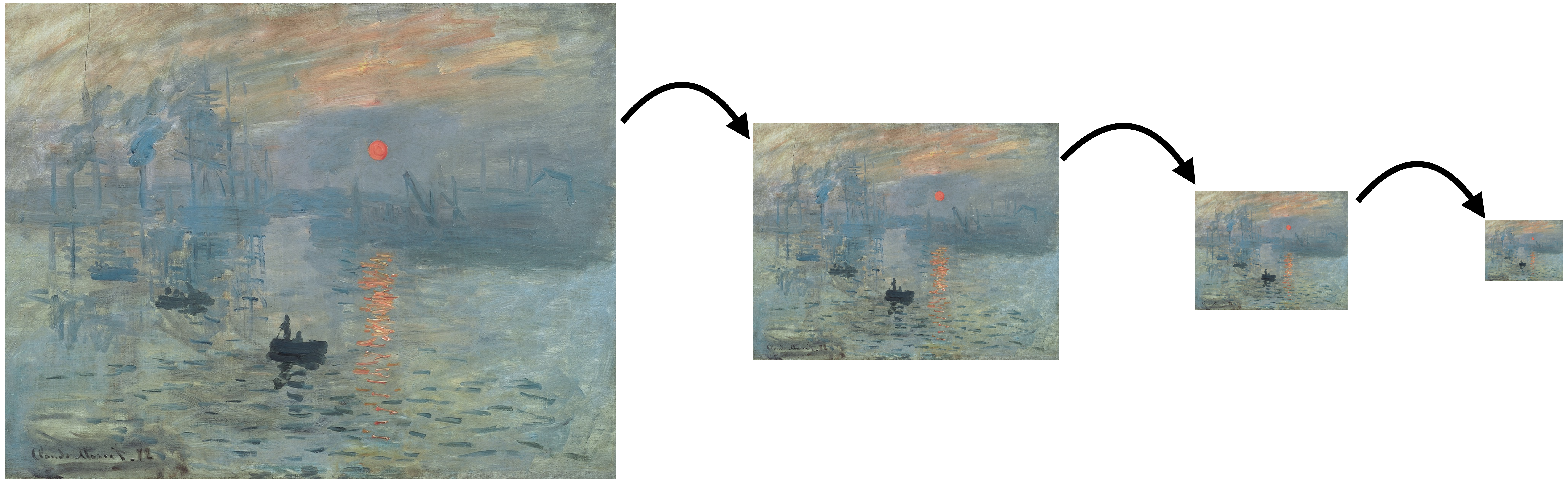


Mixing Weights Extraction

Given primary pigments, find per-pixel mixing weights.

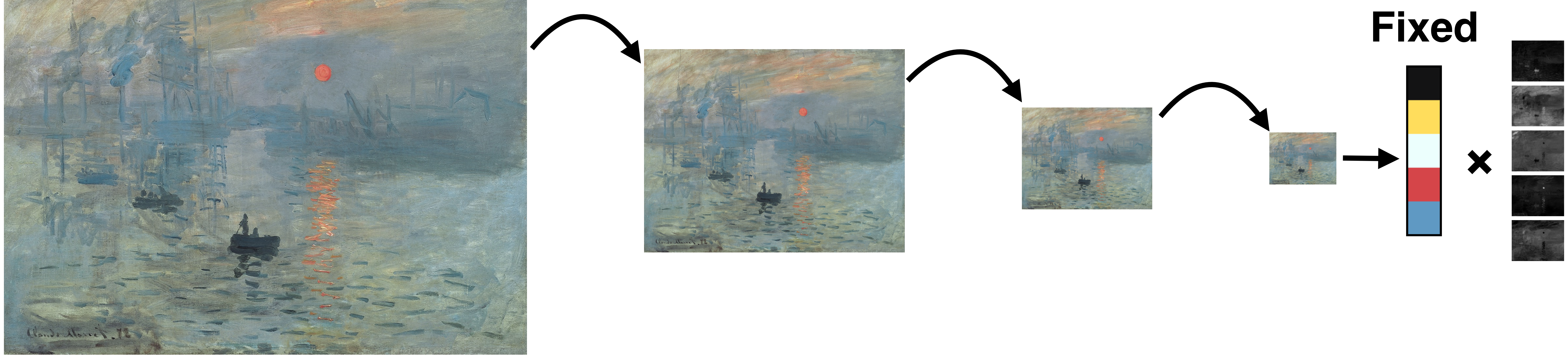
Mixing Weights Extraction

Given primary pigments, find per-pixel mixing weights.



Mixing Weights Extraction

Given primary pigments, find per-pixel mixing weights.



Mixing Weights Extraction

Given primary pigments, find per-pixel mixing weights.

Smoothness: Each primary pigment's mixing weights map is spatially smooth

Mixing Weights Extraction

Given primary pigments, find per-pixel mixing weights.

Smoothness: Each primary pigment's mixing weights map is spatially smooth

Sparsity: Each pixel's color is a mixing of smallest subset of primary pigments

Mixing Weights Extraction

Given primary pigments, find per-pixel mixing weights.

Smoothness: Each primary pigment's mixing weights map is spatially smooth

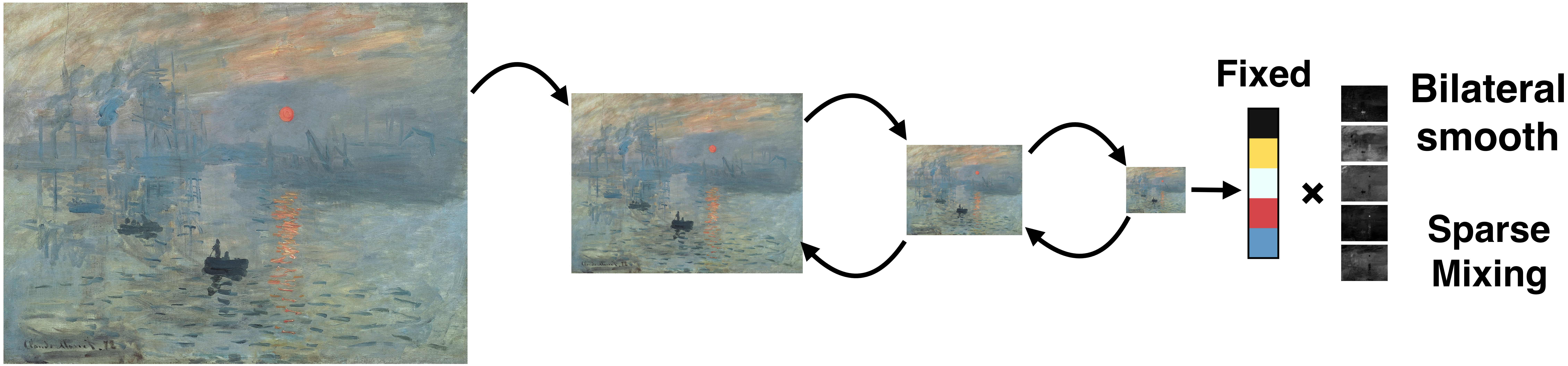
Sparsity: Each pixel's color is a mixing of smallest subset of primary pigments

Mixing Weights Extraction

Given primary pigments, find per-pixel mixing weights.

Smoothness: Each primary pigment's mixing weights map is spatially smooth

Sparsity: Each pixel's color is a mixing of smallest subset of primary pigments

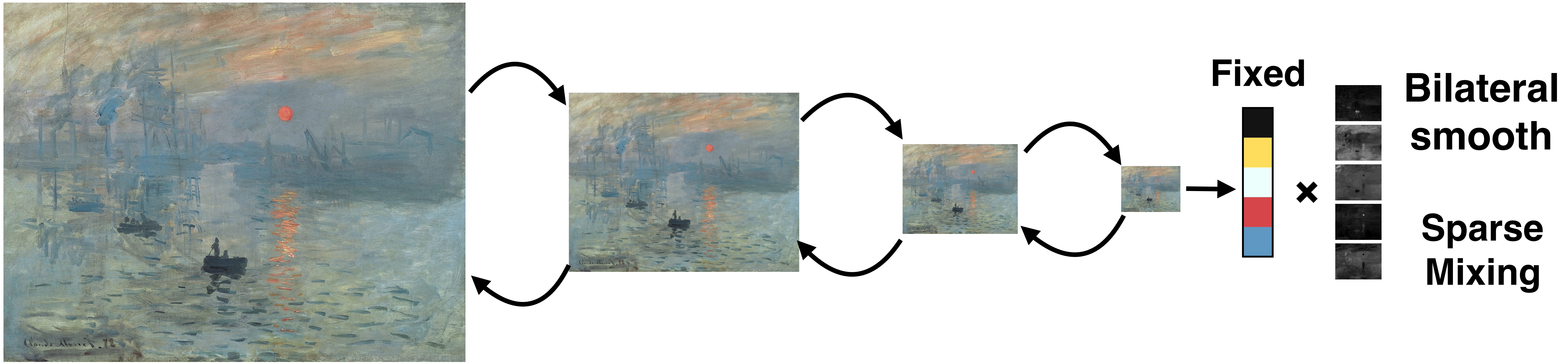


Mixing Weights Extraction

Given primary pigments, find per-pixel mixing weights.

Smoothness: Each primary pigment's mixing weights map is spatially smooth

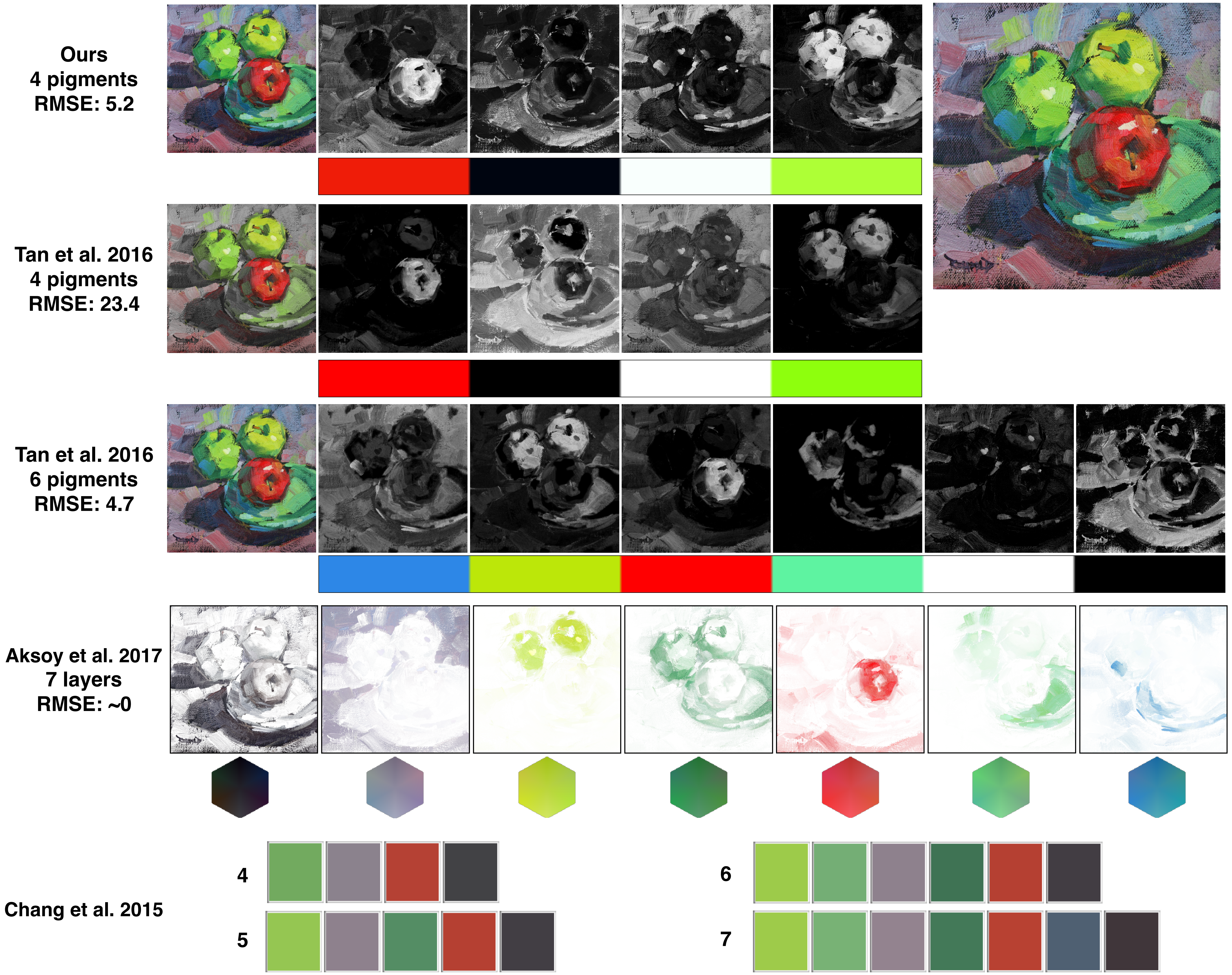
Sparsity: Each pixel's color is a mixing of smallest subset of primary pigments



Our results


soleil
autumn
Impasto_flower2
landscape4
Original
Recovery
10x error
Mixing Weights

Compare to results from other models



Recoloring comparison

Original

blue pigment -> green (ours)

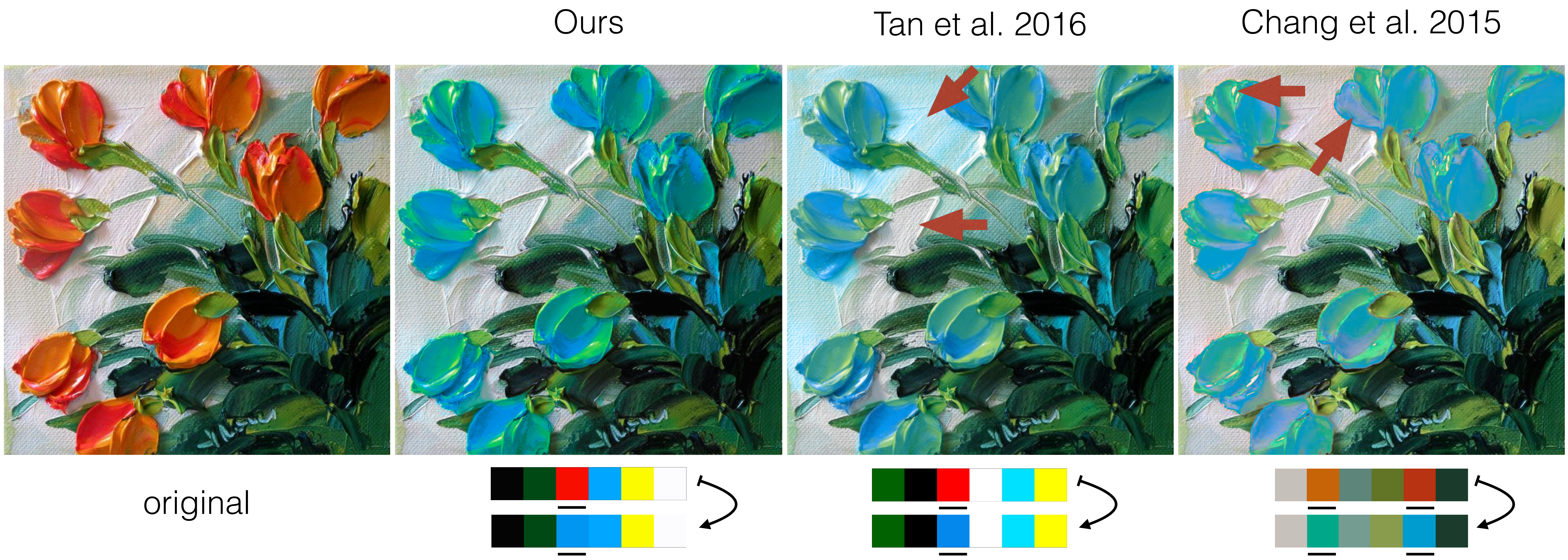
blue RGB -> green (Tan2016)

Original

red pigment -> blue (ours)

red RGB -> blue (Tan2016)

Recoloring comparison



Applications

Recoloring by modifying pigment weights

Original

Reduce yellow

Add more yellow

Original

Reduce red

Add more red

Modify weights of black/white pigment

Original

Increase the mixing weight of black pigment

Decrease brightness

Increase all weights

Increase the mixing weight of white pigment

Increase brightness

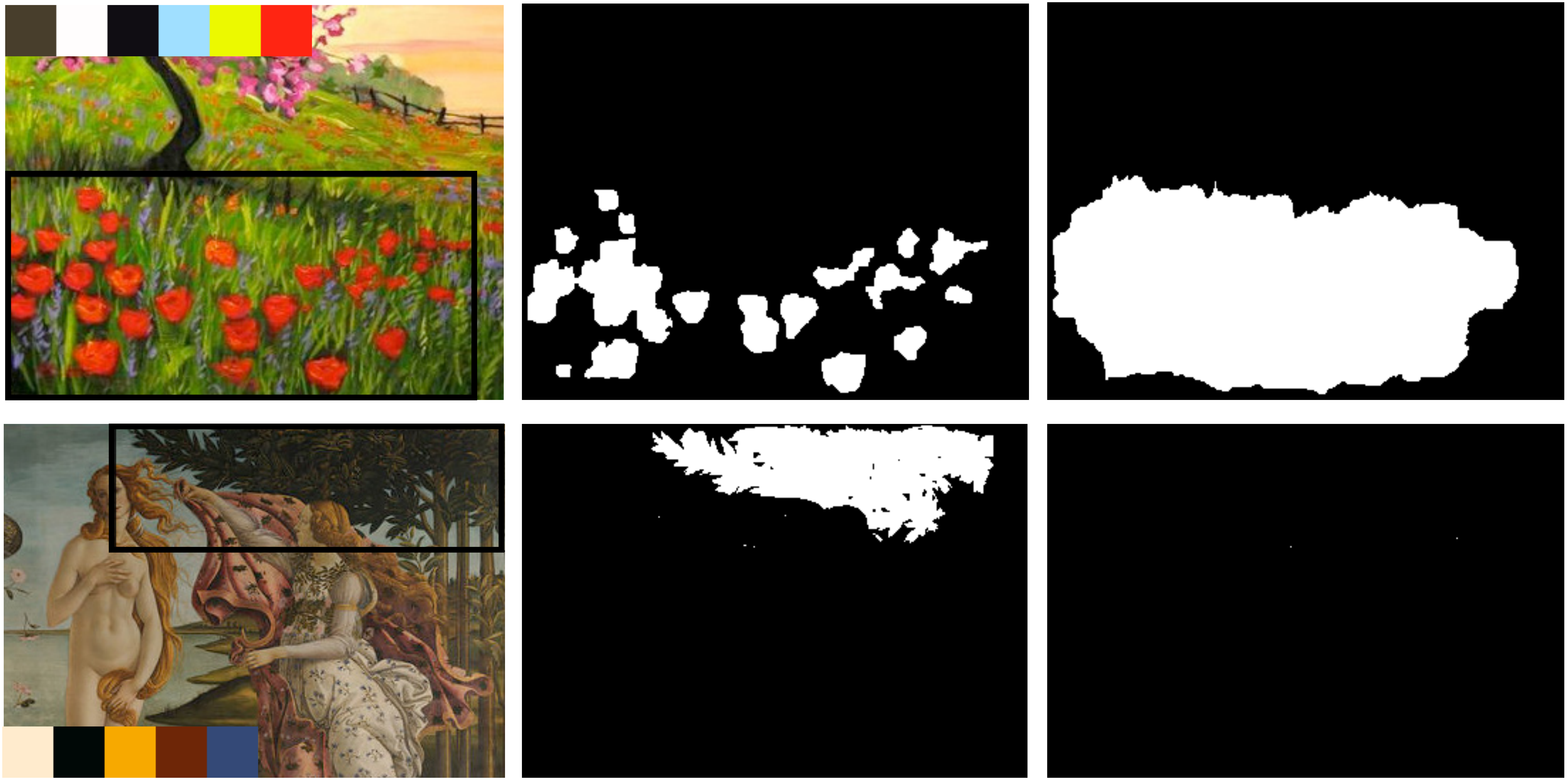
Modify pigment scattering parameters

Original

Increase scattering

Decrease scattering

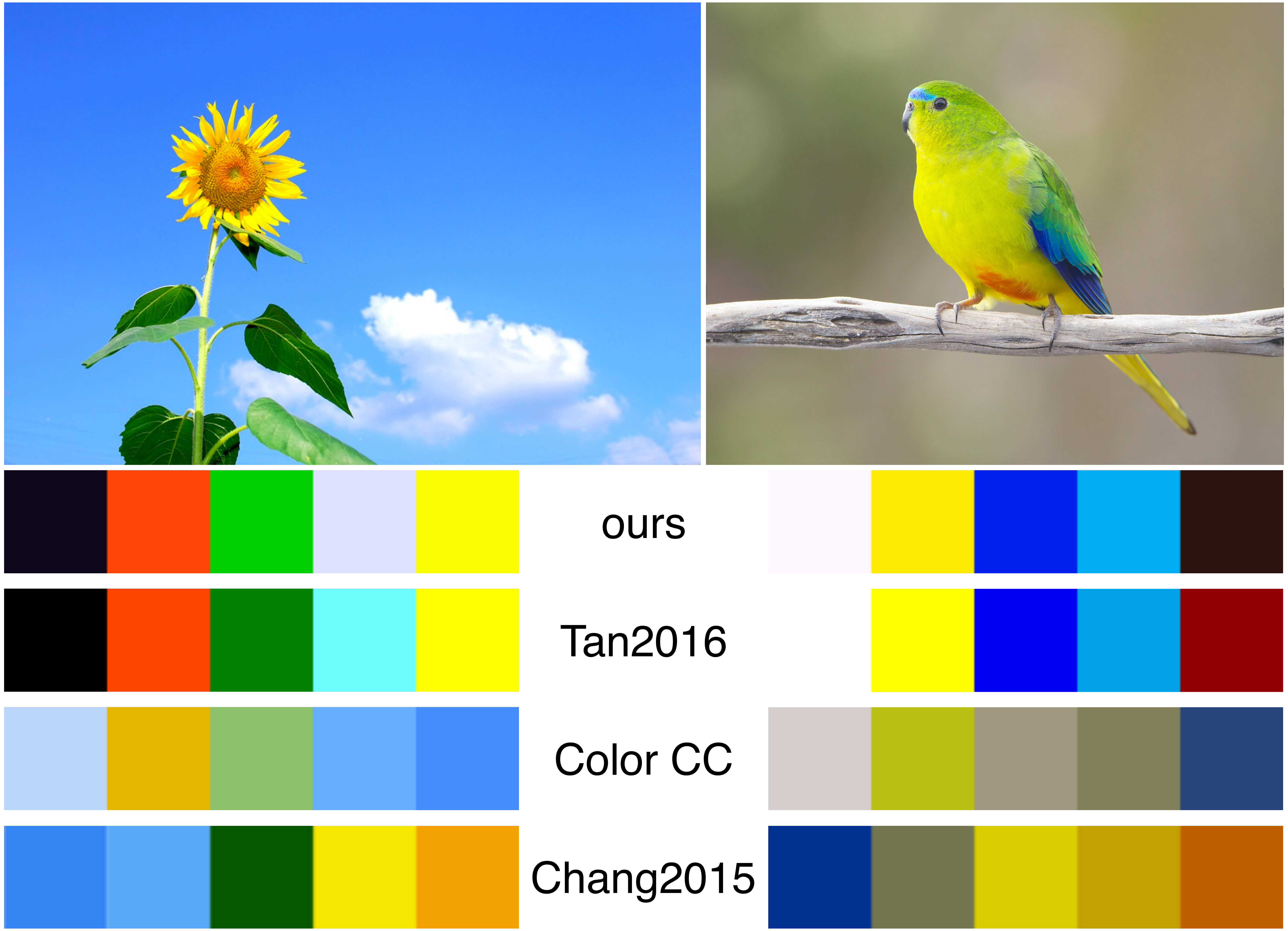
Mask Selection



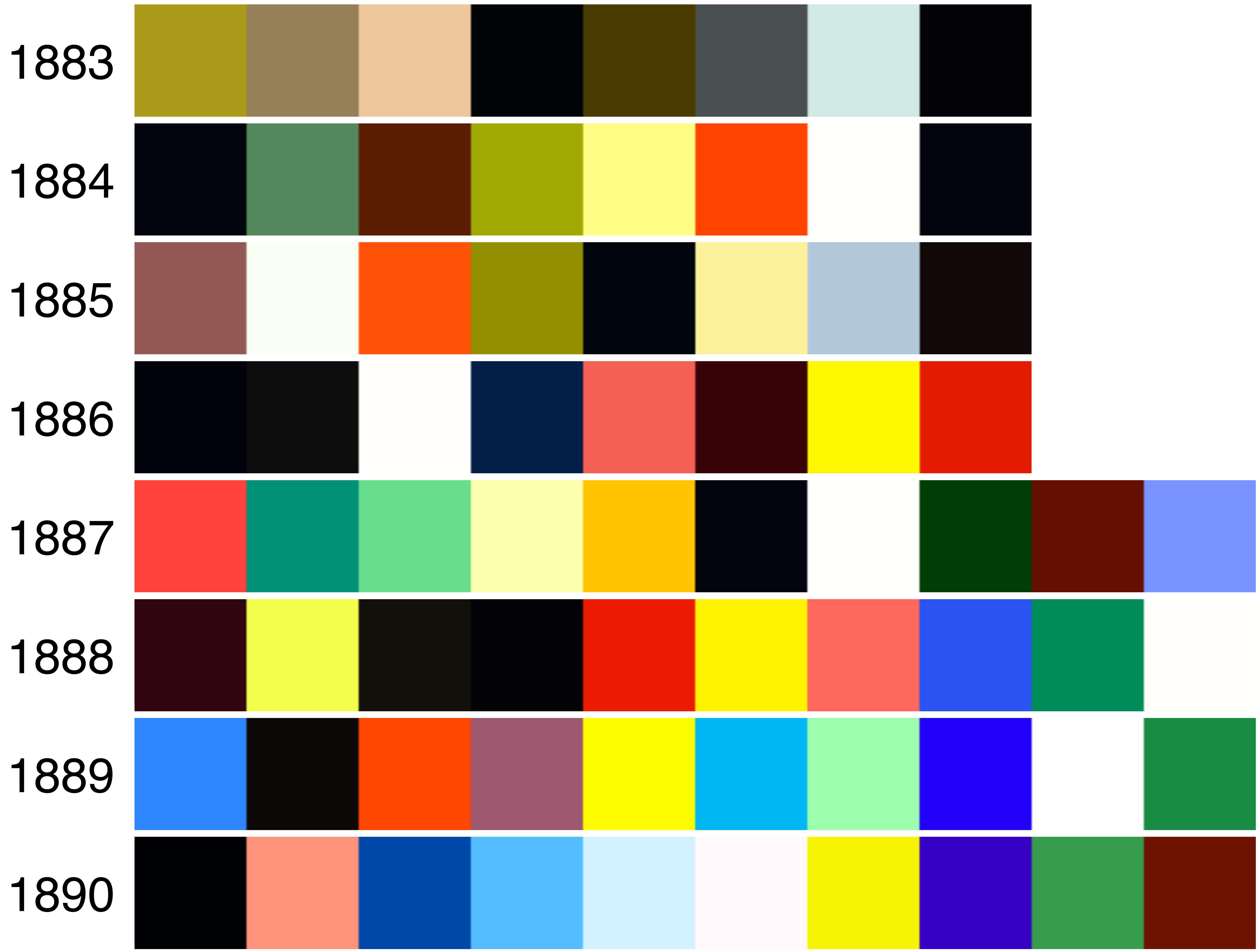
Rectangle Input | Grabcut on KM layer | Grabcut on RGB

Copy-Paste in pigment space

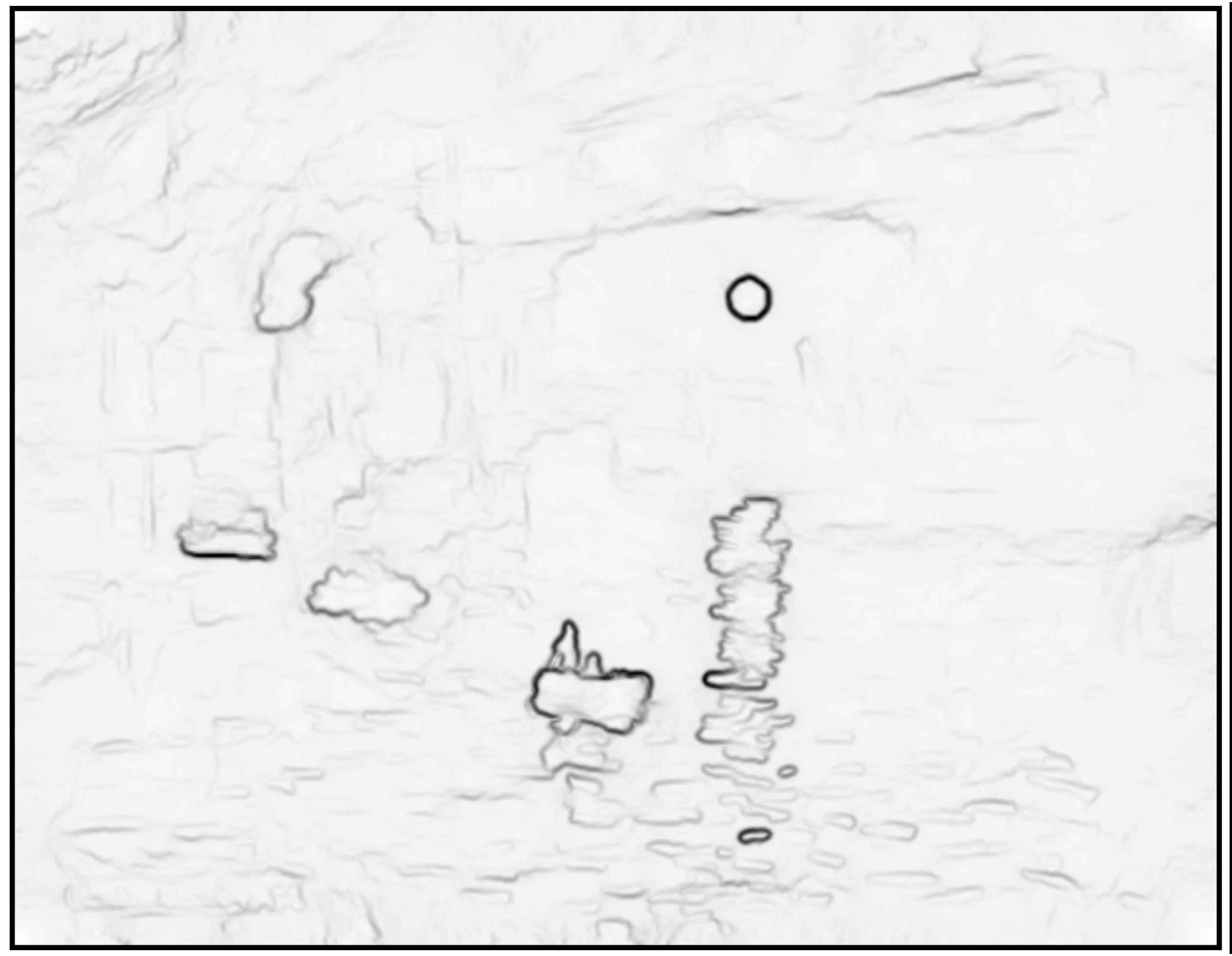
Palette Summarization - Photos

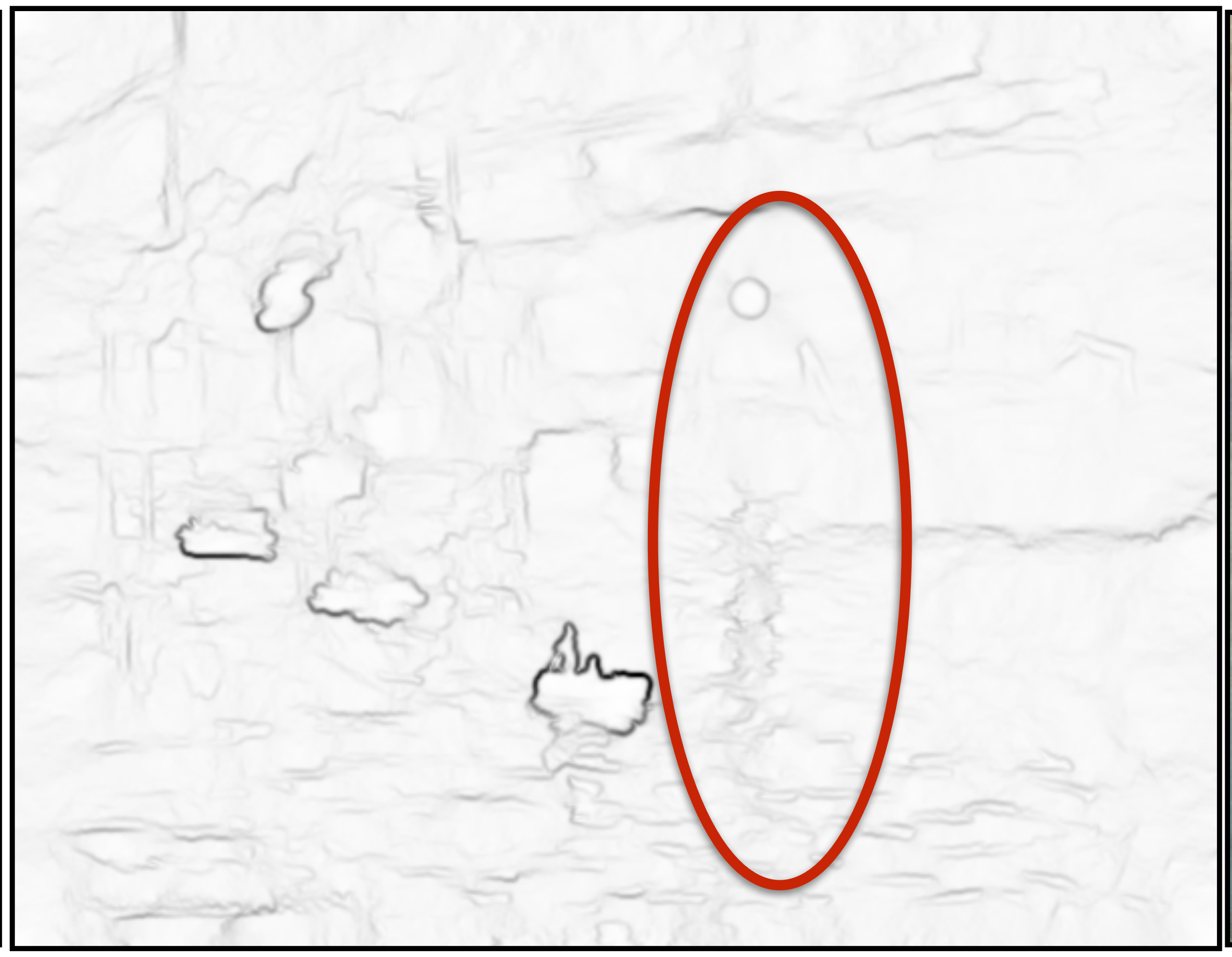


Palette Summarization - Collections

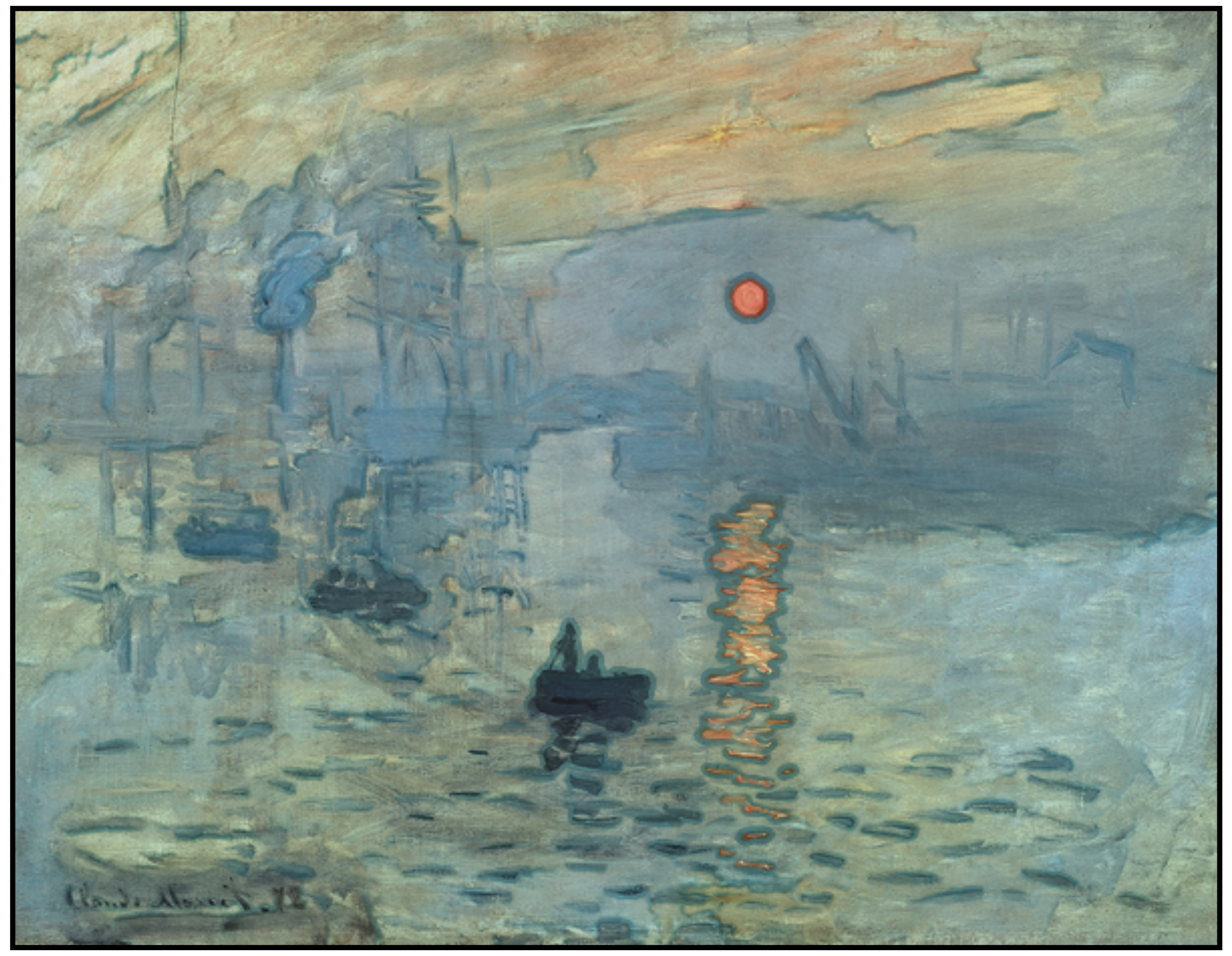


Edge detection and enhancement


on weights map


on RGB


original


Enhancement

Conclusion

Conclusion

- Provide an efficient optimization framework to extract multispectral pigments and their per-pixel mixing weights from given RGB painting image.

Conclusion

- Provide an efficient optimization framework to extract multispectral pigments and their per-pixel mixing weights from given RGB painting image.
- Enable many paint-like edits of the painting, which are beyond RGB space.

Conclusion

- Provide an efficient optimization framework to extract multispectral pigments and their per-pixel mixing weights from given RGB painting image.
- Enable many paint-like edits of the painting, which are beyond RGB space.
- Our discussion of gamut problem and several regularization terms used in our optimization are useful in other similar problems.

Limitation and future work

Limitation and future work

- Using prior acyclic pigment database as initial value may cause overfitting problem.

Limitation and future work

- Using prior acyclic pigment database as initial value may cause overfitting problem.
    - We do not have other datasets (e.g. watercolor pigment) to verify it.

Limitation and future work

- Using prior acyclic pigment database as initial value may cause overfitting problem.
    - We do not have other datasets (e.g. watercolor pigment) to verify it.
- We assume constant paint thickness to simplify optimization.

Limitation and future work

- Using prior acyclic pigment database as initial value may cause overfitting problem.
    - We do not have other datasets (e.g. watercolor pigment) to verify it.
- We assume constant paint thickness to simplify optimization.
- We may want to estimate pigment layers instead of just mixtures, then layer order is needed.

Limitation and future work

- Using prior acyclic pigment database as initial value may cause overfitting problem.
    - We do not have other datasets (e.g. watercolor pigment) to verify it.
- We assume constant paint thickness to simplify optimization.
- We may want to estimate pigment layers instead of just mixtures, then layer order is needed.
- Use our decomposition results to help extract brushstroke-level structure from painting images.

Limitation and future work

- Using prior acyclic pigment database as initial value may cause overfitting problem.
    - We do not have other datasets (e.g. watercolor pigment) to verify it.
- We assume constant paint thickness to simplify optimization.
- We may want to estimate pigment layers instead of just mixtures, then layer order is needed.
- Use our decomposition results to help extract brushstroke-level structure from painting images.

.

Thank You!

- **Contact Information:**

  - Jianchao Tan: jtan8@gmu.edu
  - Stephen DiVerdi: diverdi@adobe.com
  - Jingwan Lu: jlu@adobe.com
  - Yotam Gingold: ygingold@gmu.edu

- **Project Website:** https://cragl.cs.gmu.edu/pigmento/
- **Our exposure in I-Programmer website:** https://www.i-programmer.info/news/144-graphics-and-games/10990-pigments-beyond-rgb.html
- **Artists:**
  - MontMarteArt, Jan Ironside, Graham Gercken, Nel Jansen, Cathleen Rehfeld, Patty Baker, John Larriva, Pamela Gatens, Mark Adam Webster, Patti Mollica, Jan Ironside.

- **Sponsors:**
  - United States National Science Foundation, Adobe Research.

Extra Slides

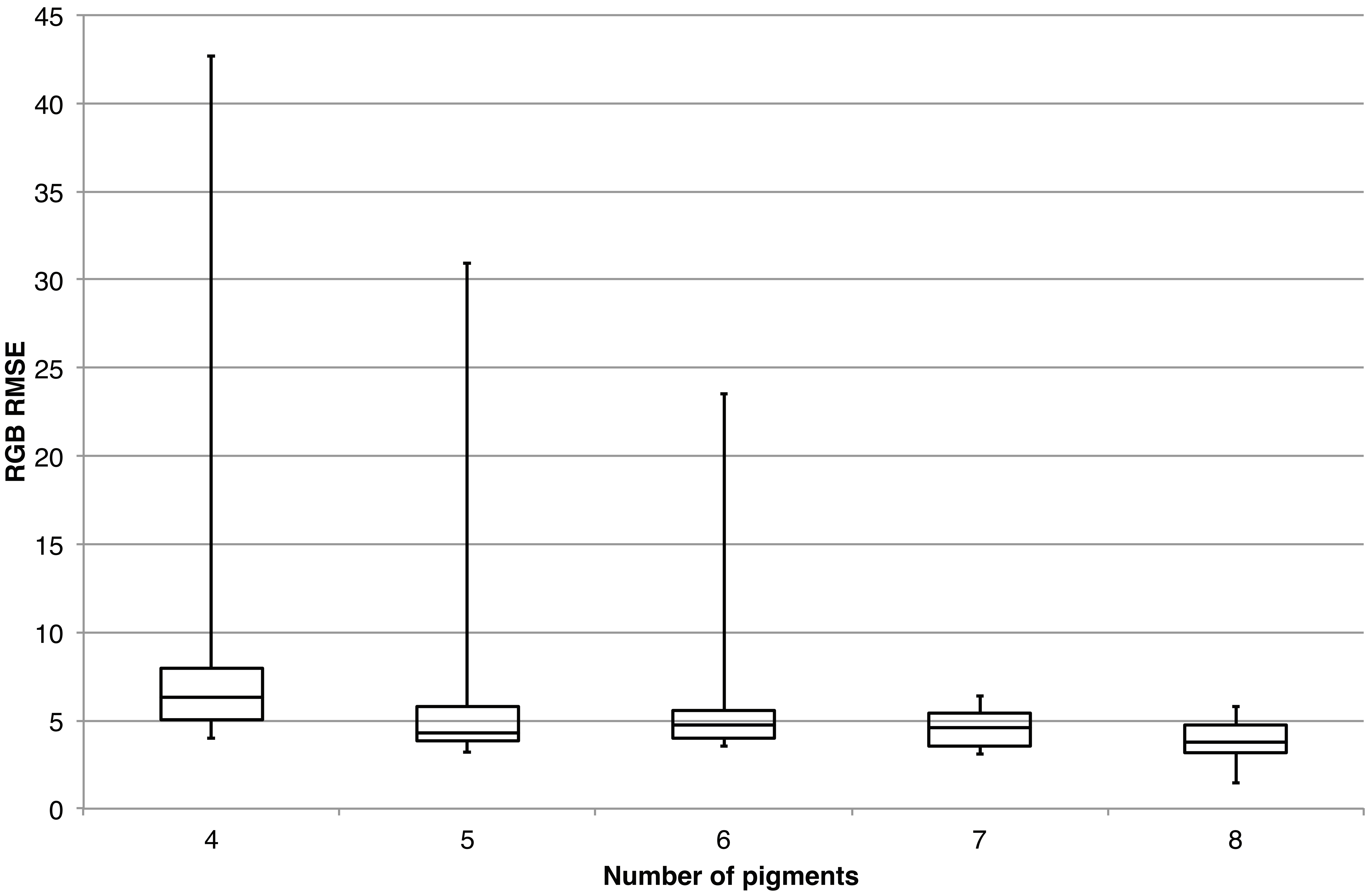
Performance Information

Examples	Image size	Pigments number	CPU	KM primary pigments extraction Time (sec)	KM mixing weights extraction Time (sec)	KM original image reconstruction RMSE (0-255)
soleil	600*467	6	core i7	35	155	1.9
autumn	600*458	5	xeon	16	225	6.0
four_colors_2	600*598	4	core i7	9	211	5.2
Impasto_flower2	595*600	6	xeon	44	615	5.1
Landscape4	600*479	5	xeon	26	256	4.7
Portrait2	600*441	6	xeon	29	243	4.4
tree	600*492	4	core i7	14	151	4.0

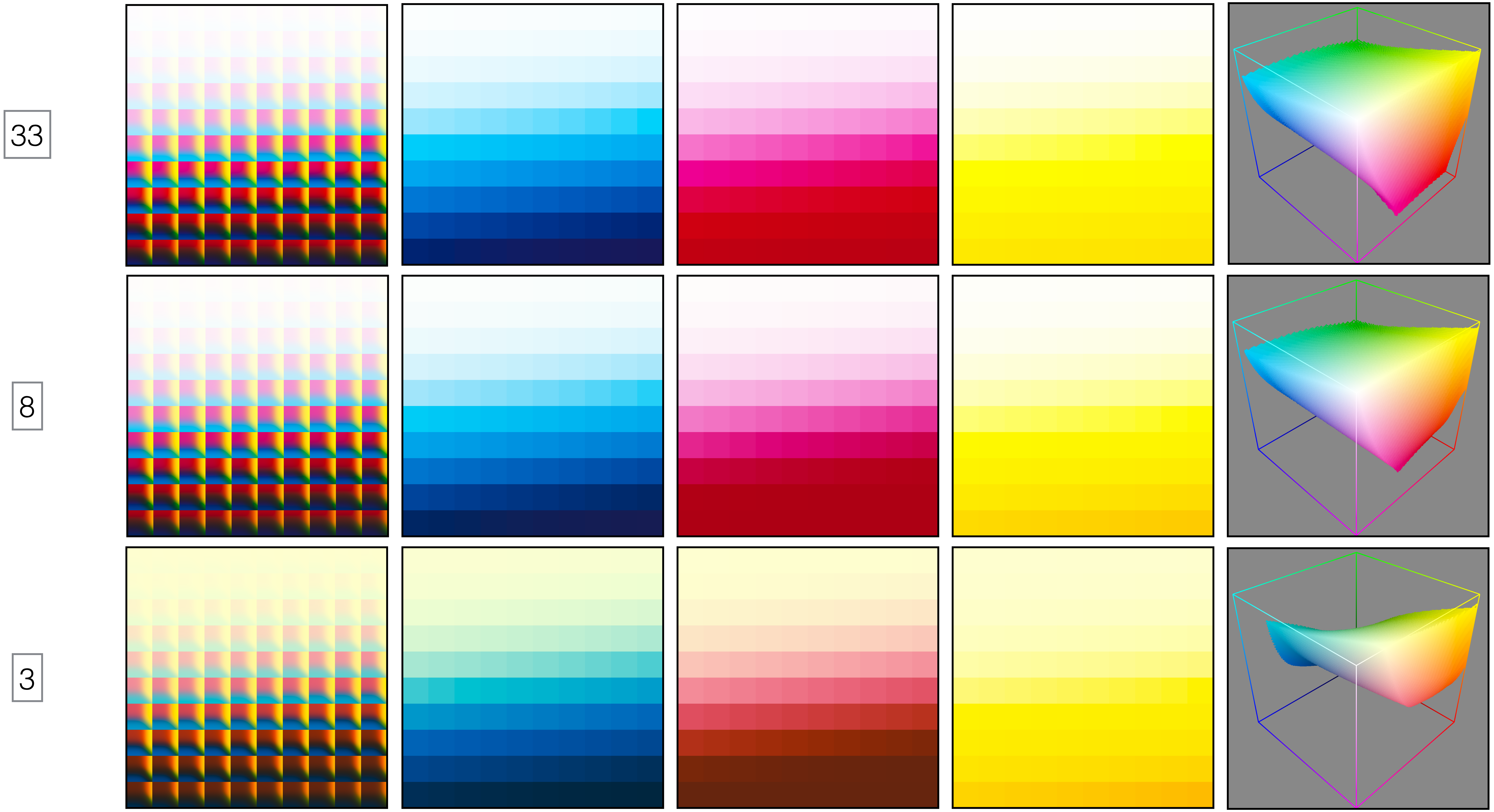
Pigment smoothness and thickness

Pigment smoothness and thickness

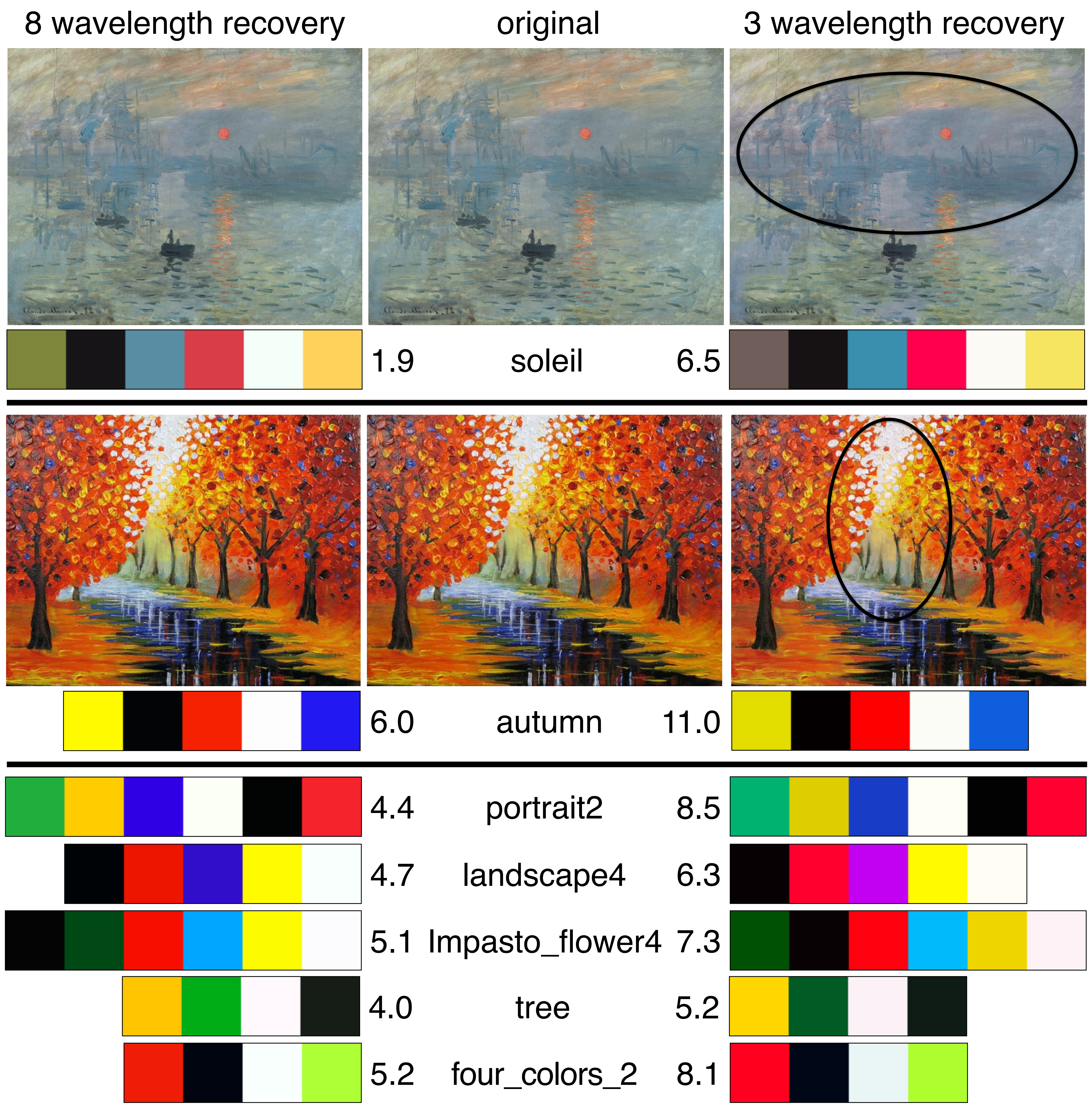
Pigment number influence



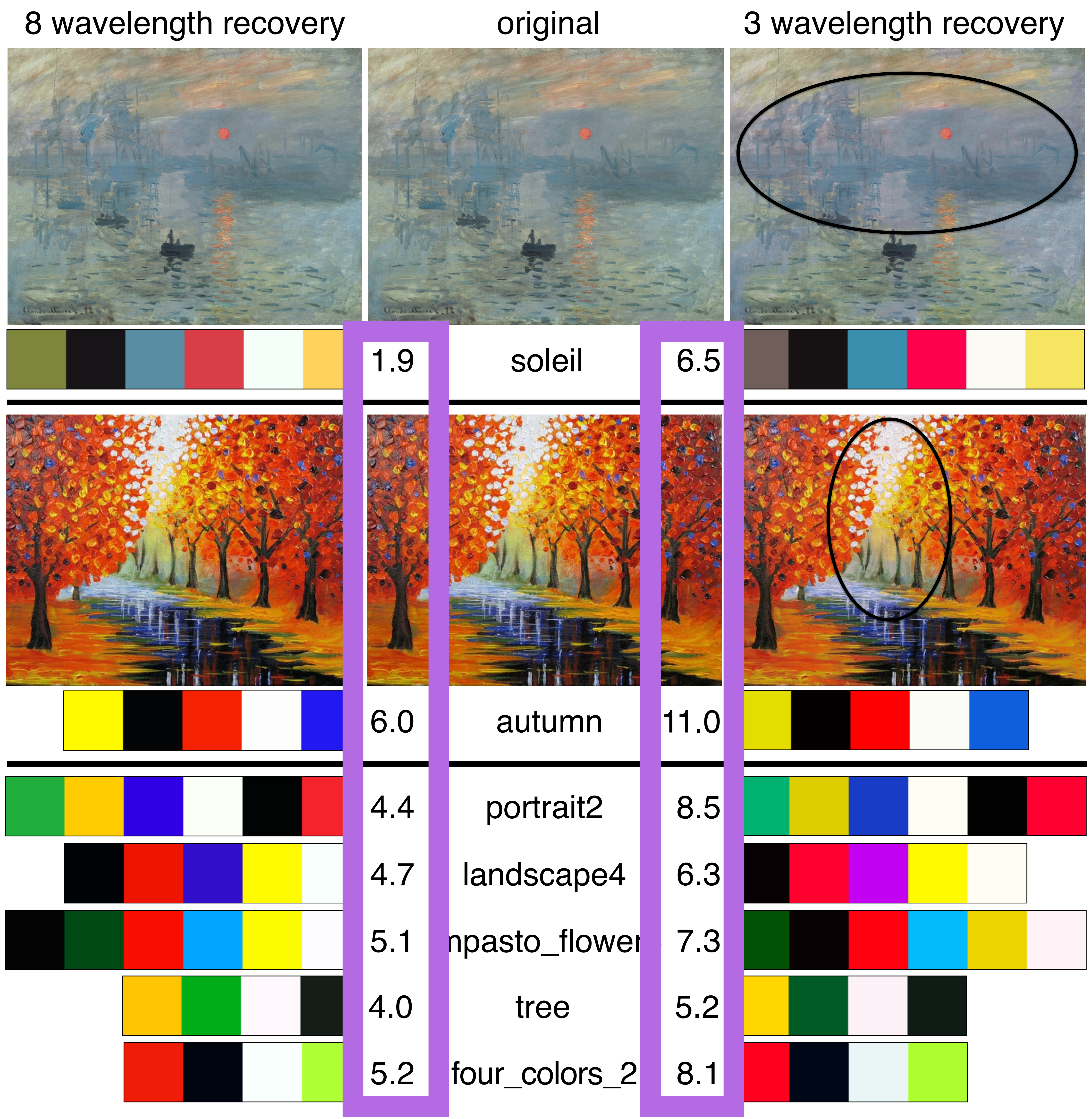
Wavelength influence



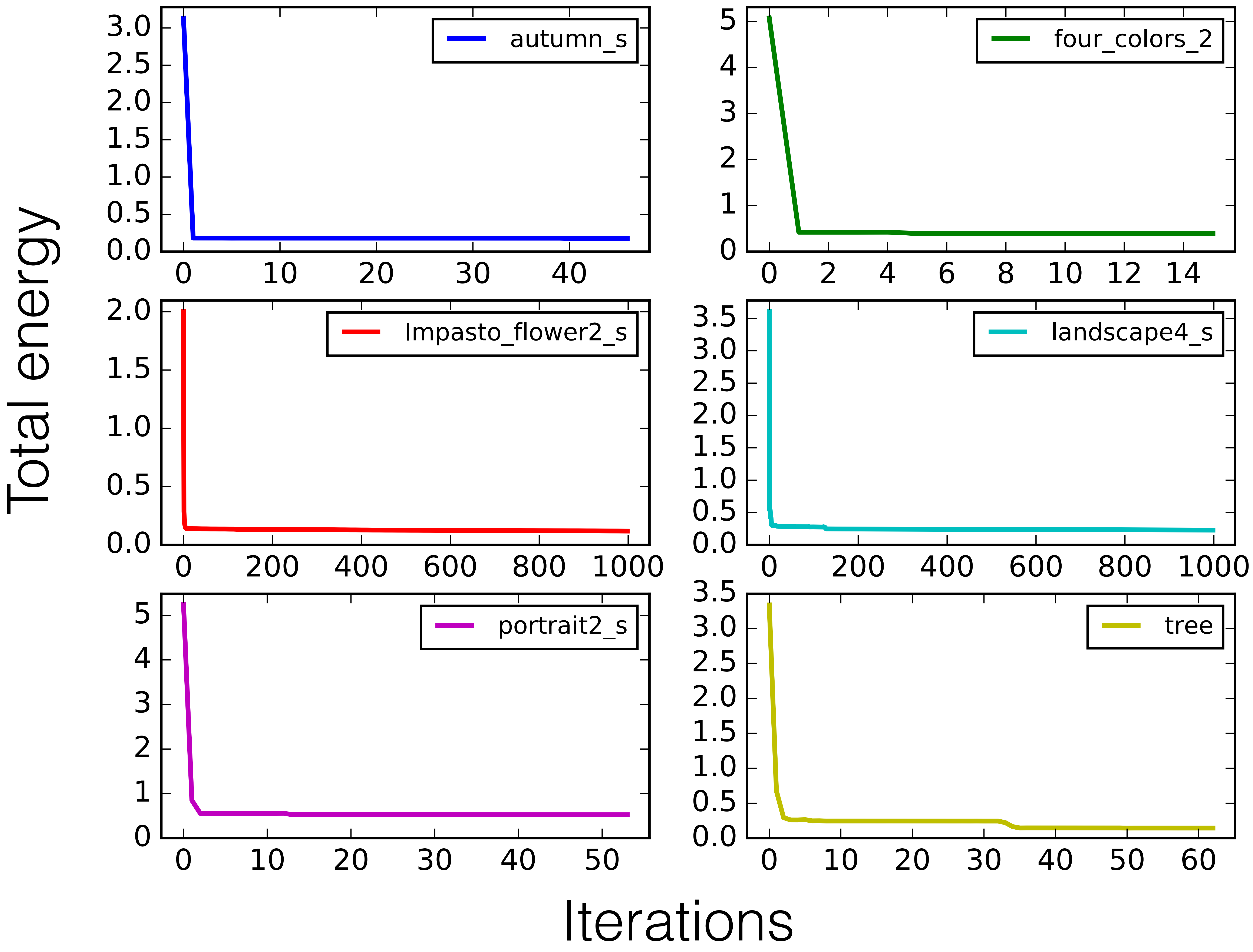
Wavelength influence



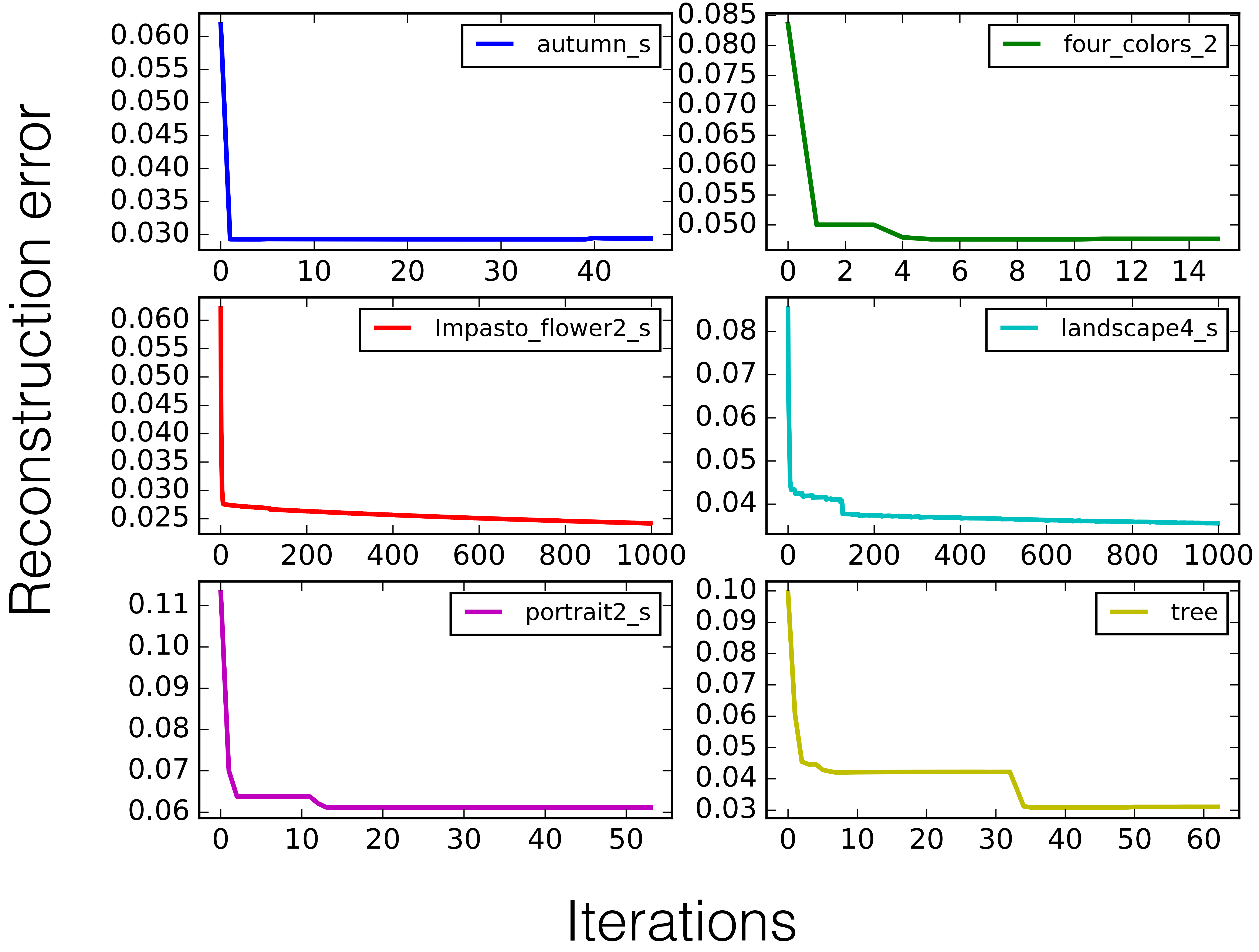
Wavelength influence



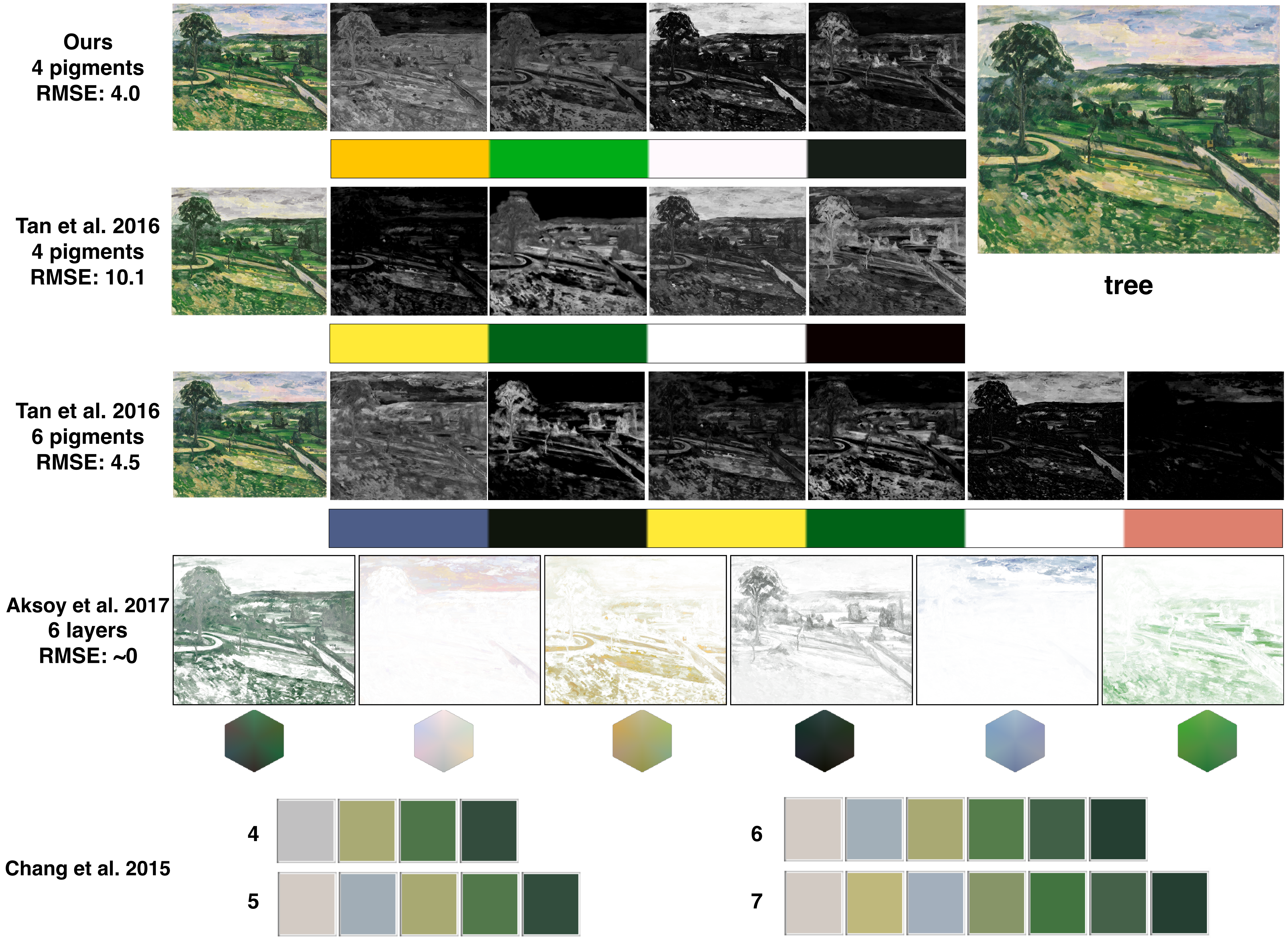
Primary pigment estimation convergence



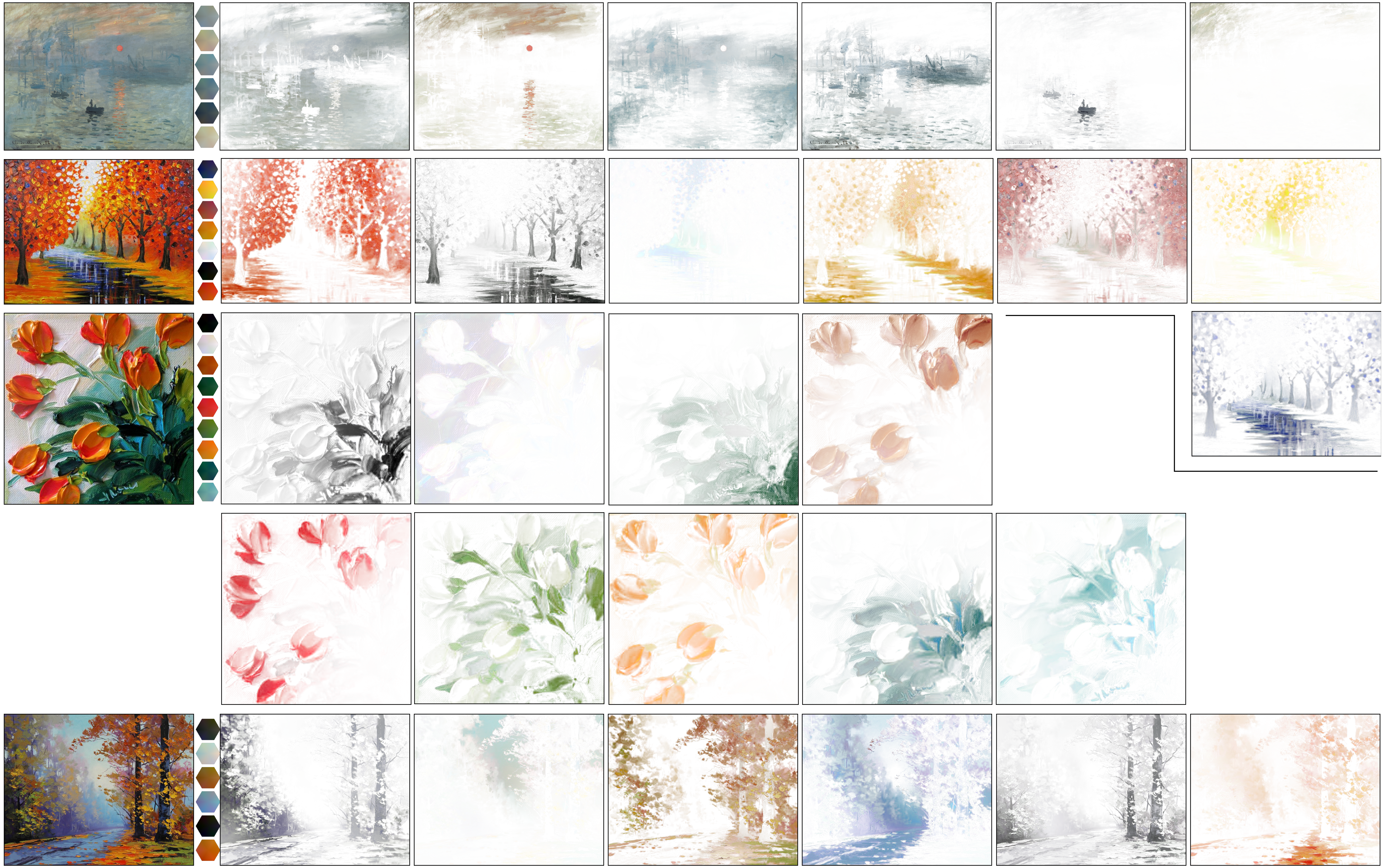
Primary pigment estimation convergence



Compare to results from other models



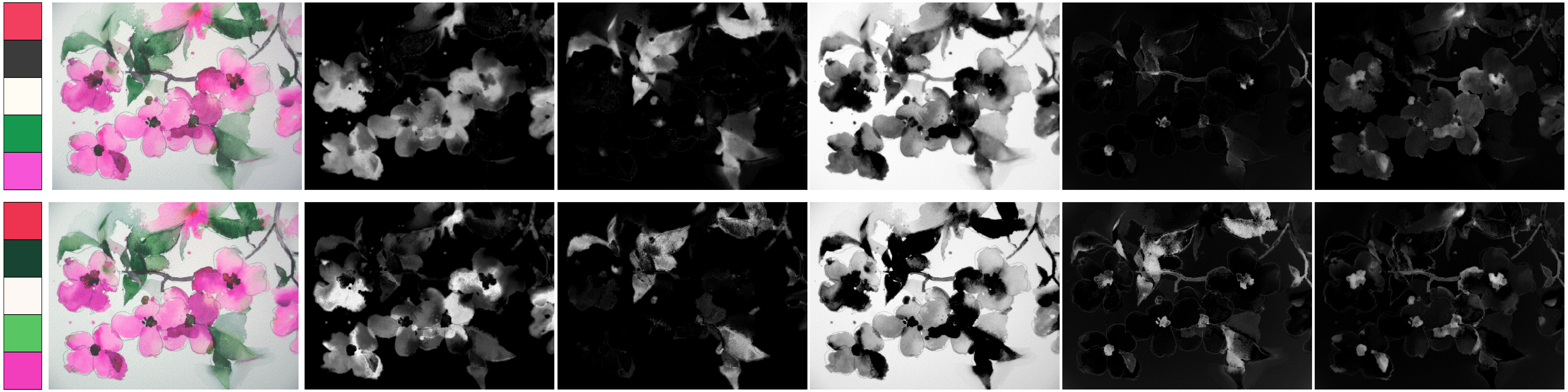
Aksoy et al. 2017 results



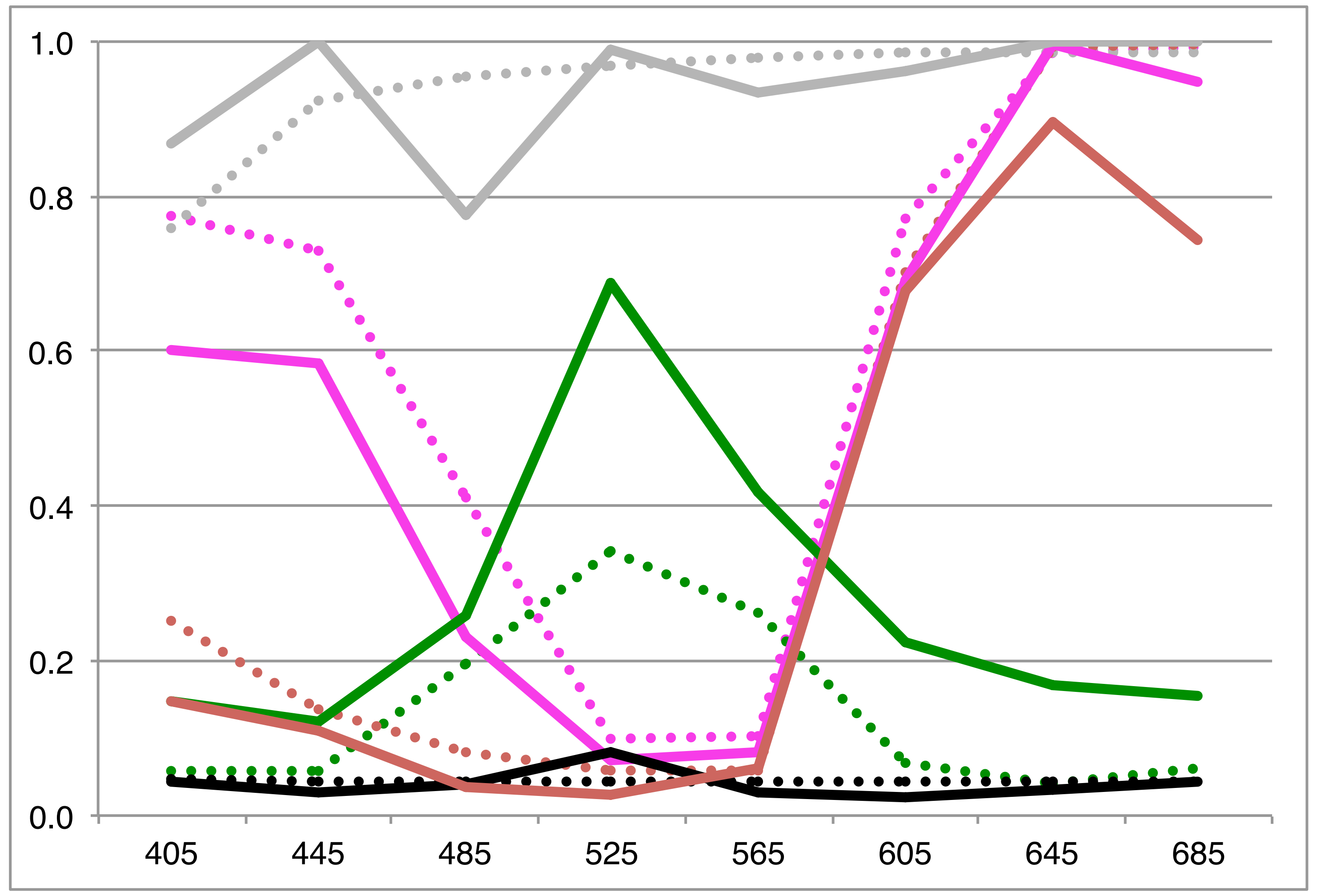
Ground Truth Test

Ground truth

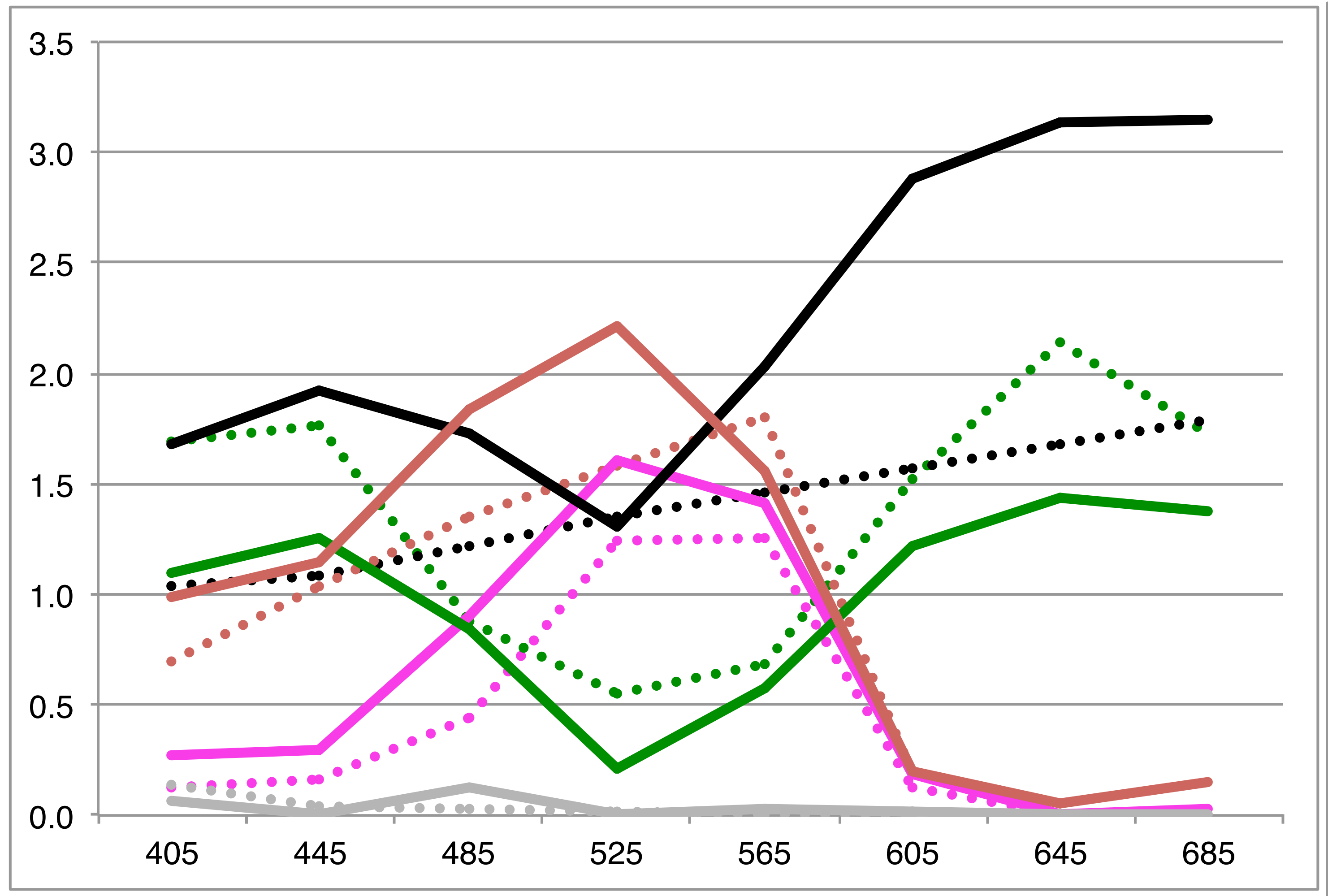
Recovered



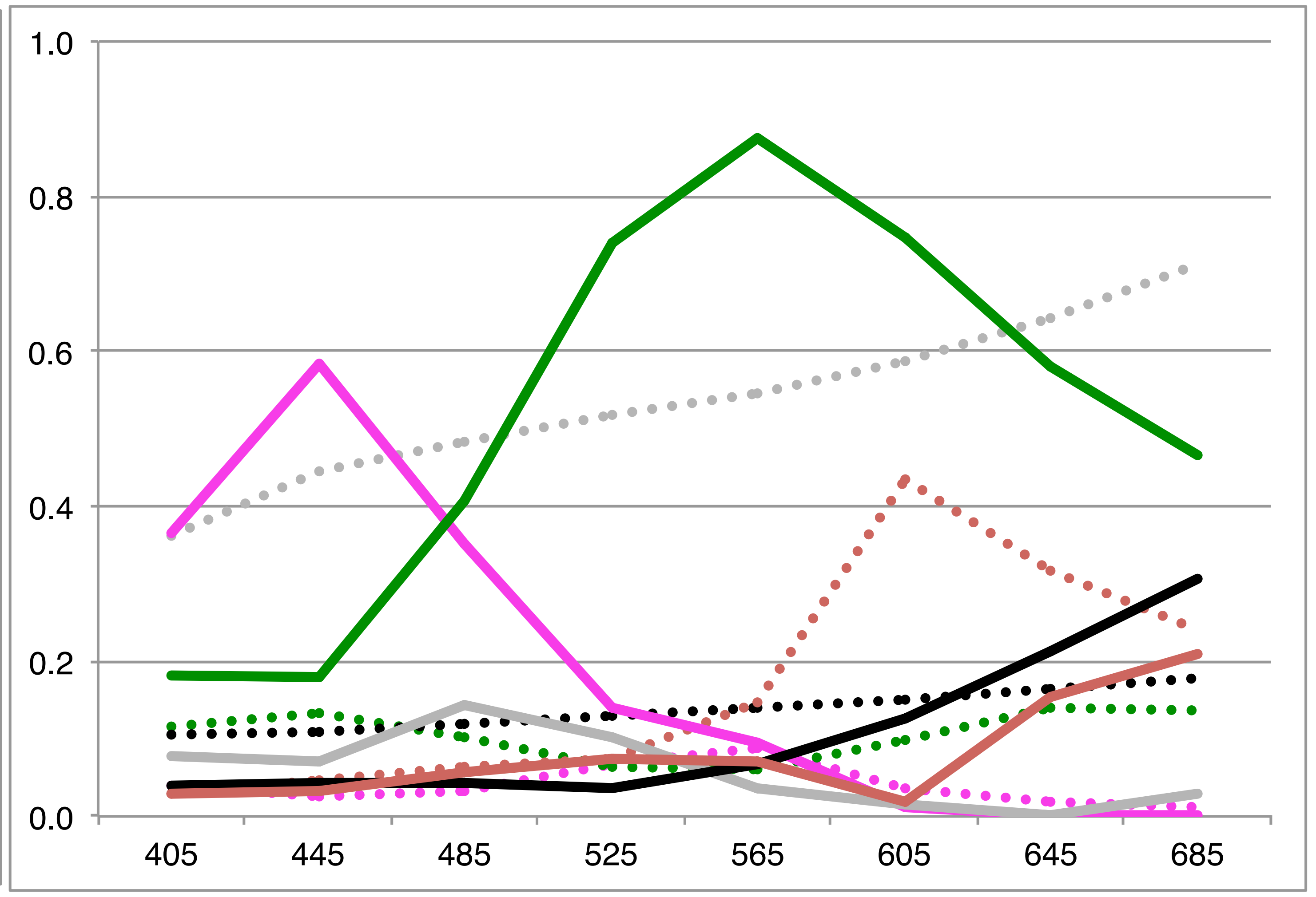
Reflectance



Absorption



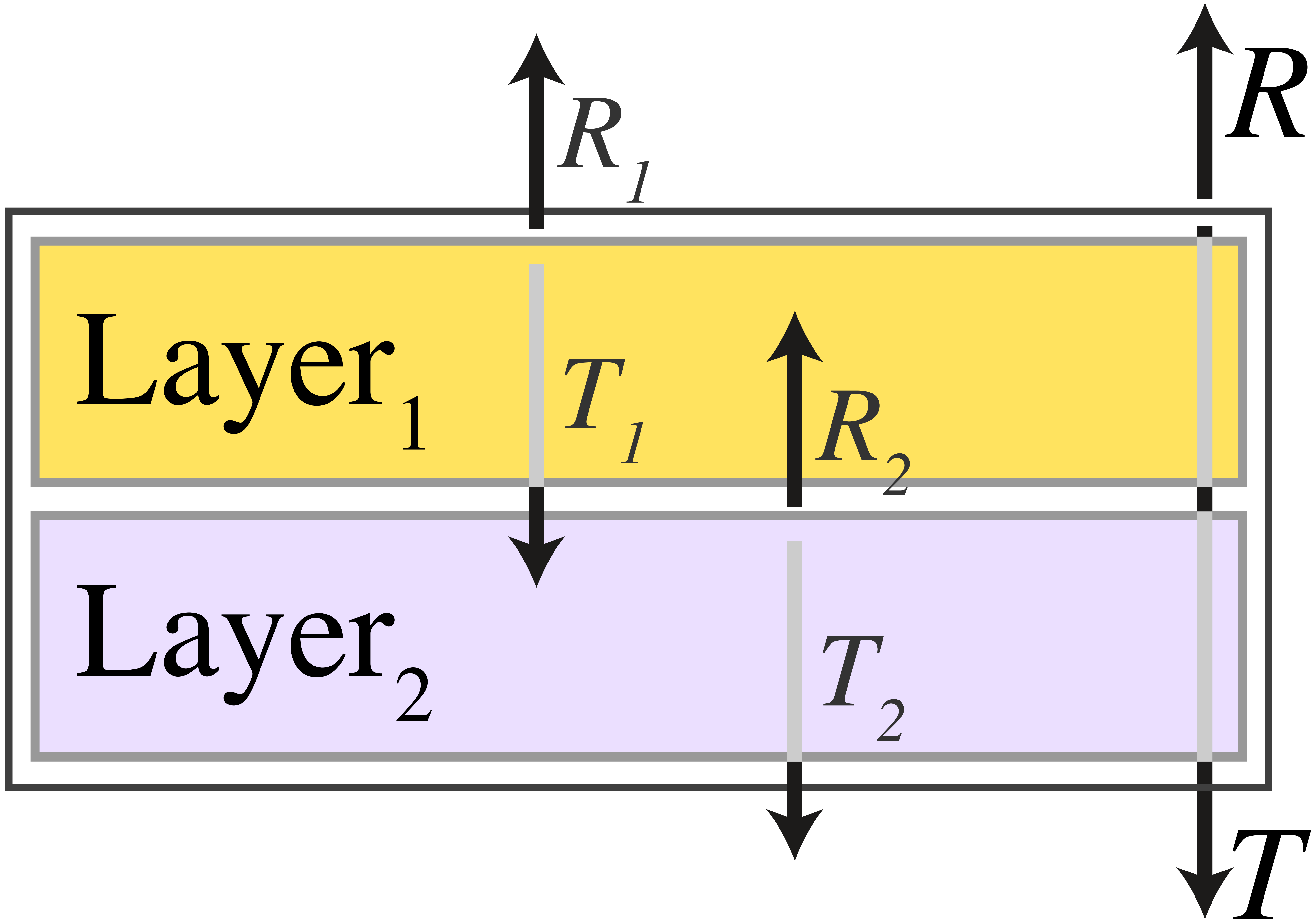
Scattering



Ground truth test information

Experiments	RMSE for recovering pigments parameters H (A / S)	RMSE for recovering pigments Reflectance R	RMSE for weights recovering using recovered pigments	RMSE for weights recovering using ground truth pigments	RMSE for image recovering using recovered pigments	RMSE for image recovering using ground truth pigments
Exp1	6.2 / 1.2	0.3	29	15.2	4.8	5.9
Exp2	1.4 / 0.9	0.3	19.8	11.8	6.8	4.3
Exp3	4.5 / 0.5	0.7	63	21.4	6.7	5.9
Exp4	7.1 / 1.2	0.6	42.3	14.1	8.5	6
Exp5	1.0 / 0.7	0.3	16.6	10.4	5.8	5.2
Mean	4.0 / 0.9	0.4	34.14	14.58	6.52	5.46
Std	2.7 / 0.3	0.2	18.97	4.25	1.37	0.72

Kubelka-Munk Layer Model



$$R = R_1 + \frac{T_1^2 R_2}{1 - R_1 R_2}$$

R is Reflectance
T is Transmittance