Intrinsic Images and Adversarial Losses

D.A. Forsyth, UIUC

Intrinsic images

- (Originally) Maps of an image that explain pixel values
  - Intrinsic properties:
    - independent of viewing; “object” or “world” properties
  - Extrinsic properties:
    - depend on viewing circumstances

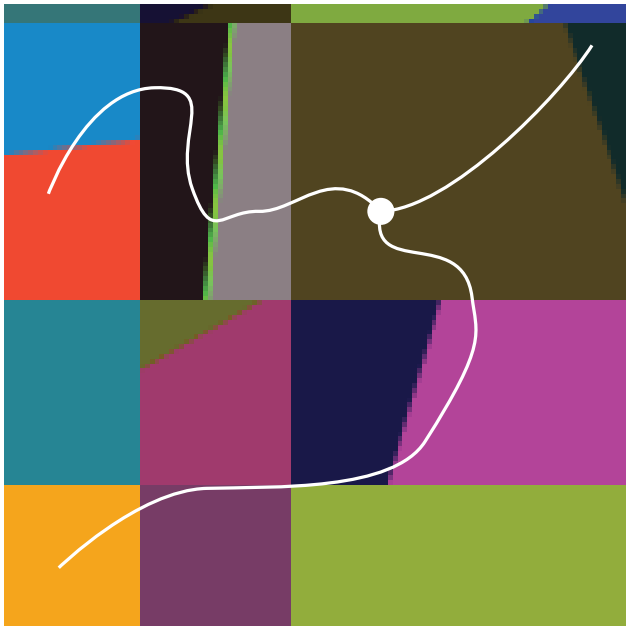
- (Later) Albedo/Shading maps
  - I=A x S
  - Albedo (A) is a natural intrinsic
  - Shading (S) is a natural extrinsic

No ground truth decompositions

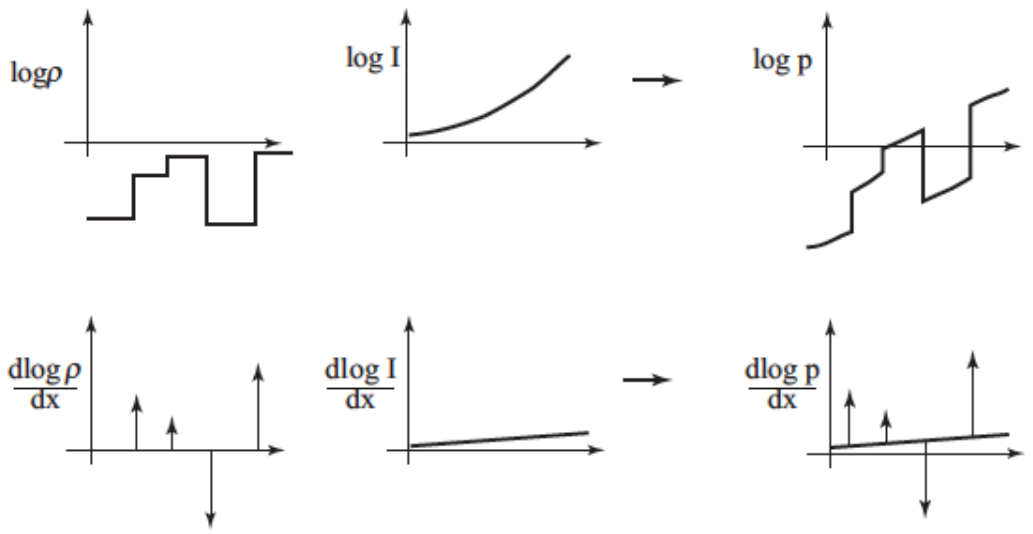
- And there never will be
  - rendering is do-able (but hard)
  - modelling is hopeless
- Q: how do you train an image decomposition method when you don't know the right answer?
- Retinex provides clues - spatial statistics are the key

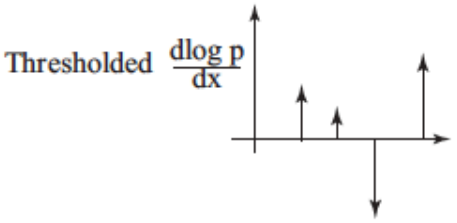
Albedo/shading and Retinex

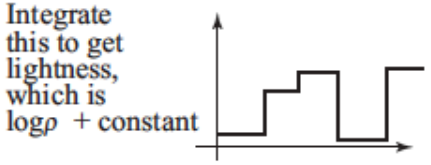
- Spatial reasoning, Land (59, 59, 77); Land +McCann 71:
  - Surface color changes either quickly or not at all
  - Light color changes slowly
  - Retinex
    - large family of algorithms
    - quite hard to know what Retinex does (Brainard+Wandell, 86)



Computer vision versions of Retinex





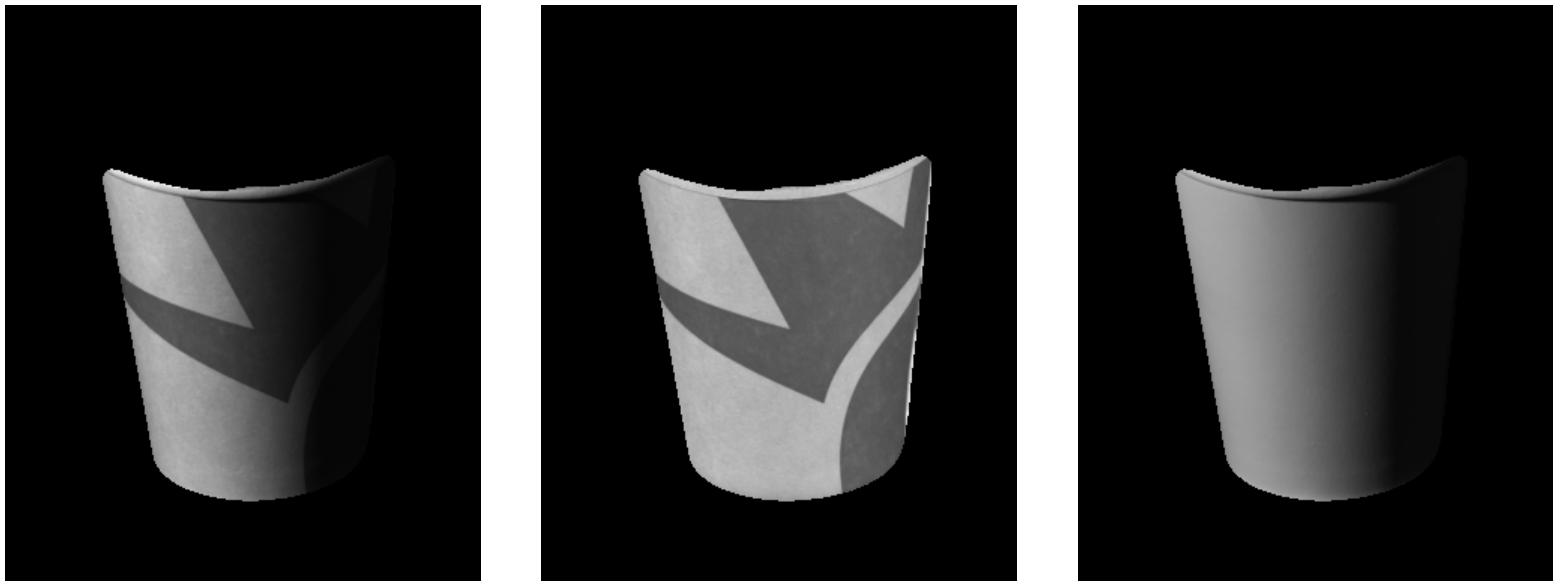


Horn, 73; 74
Brelstaff+Blake, 87;
multiple variants

Real data is hard to collect

- spraypaint, multiple images, etc…

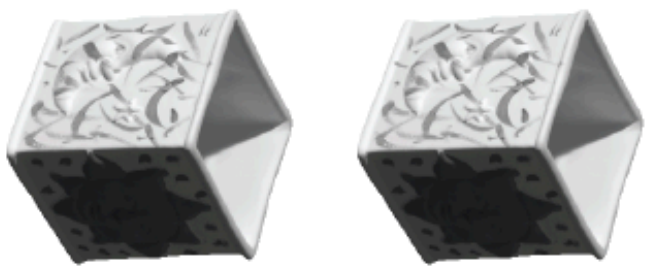
Images from dataset of Gosse et al. 09

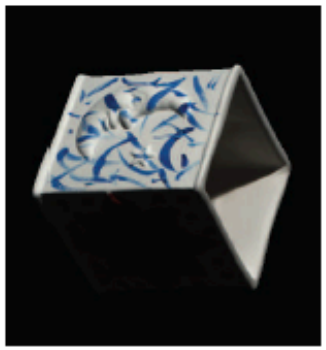


Retinex is really quite good

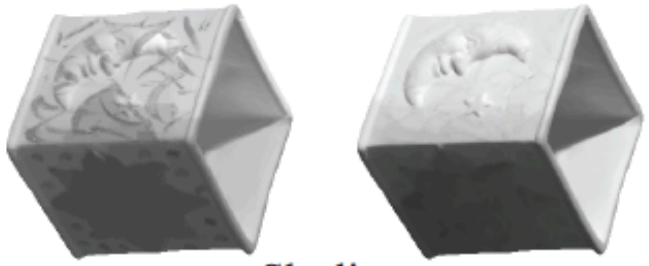
Implementation of Retinex
due to Kevin Karsch

Ground truth
images from dataset of Gosse et al. 09

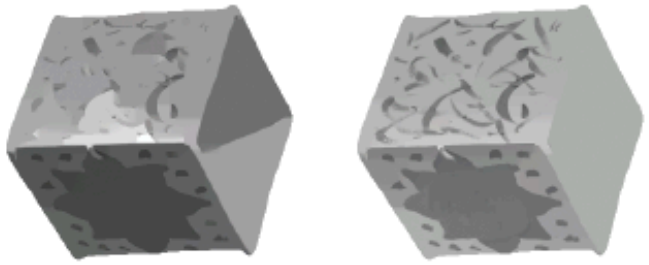


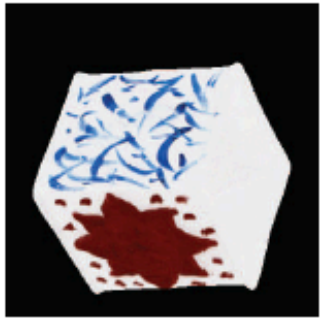


Image



Shading





Albedo

Human judgements are easier

Code (Github repository) | Pre-computed decompositions (release 0, 4.5M)

Intrinsic Images in the Wild | Publications | Browse | Search | Log i

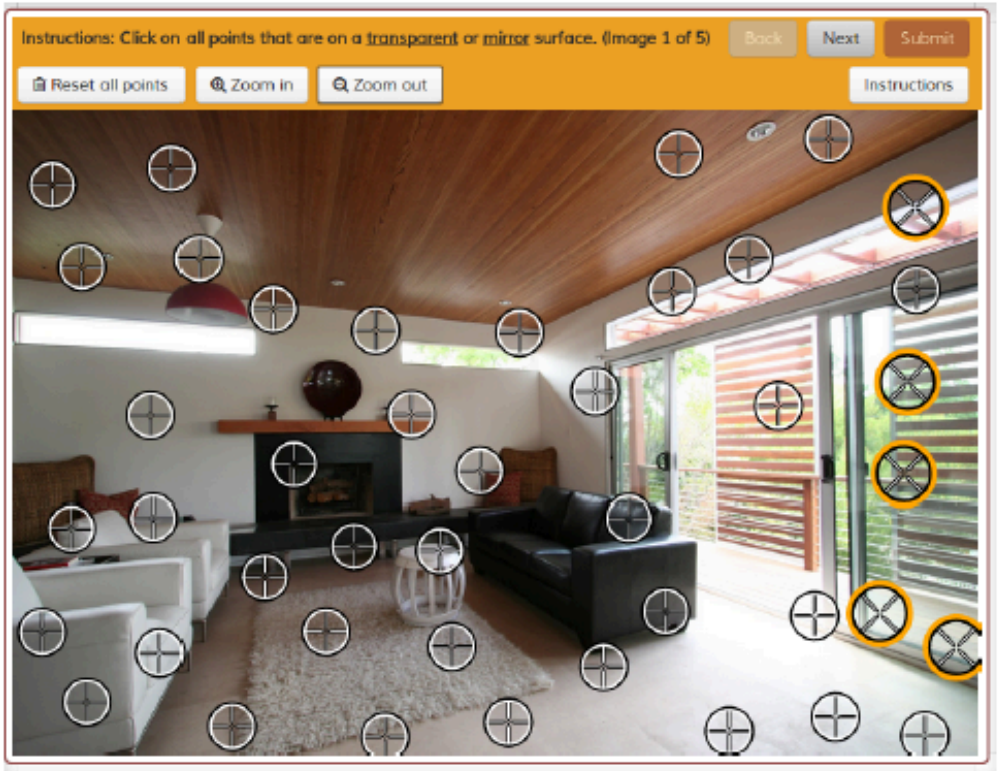
MTurk Tasks

Bell, Bala, Snavely, 2014

We include previews of our instructions, tutorials, and tasks that were shown to online workers.

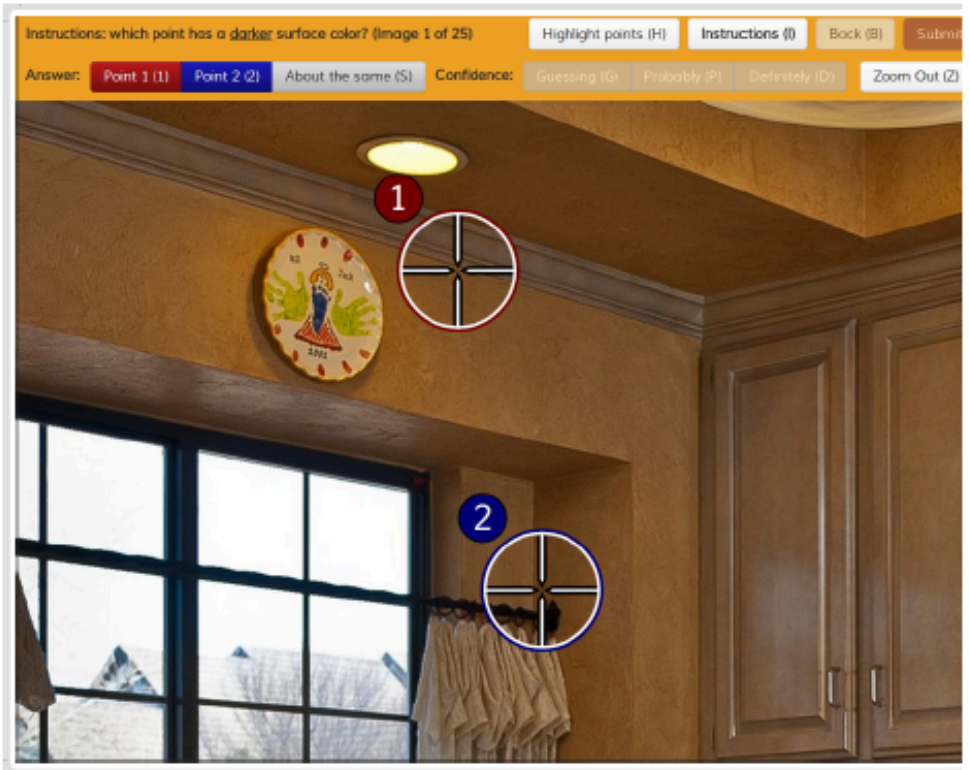
Flag transparent/mirror points

Preview: Intructions | Tutorial | Task



Compare surface reflectance

Preview: Intructions | Tutorial | Task



This gives an evaluation task

- WHDR=Weighted Human Disagreement Ratio
  - compute lightness from intrinsic image representation at points
  - predict
    - A lighter than B
    - B lighter than A
    - Lightness match
  - compute weighted estimate of accuracy
    - weights low where human judgements are uncertain, high otherwise
- There are issues, but allows evaluation
  - and competition

Modern strategies - Optimization

- Apply the priors that
  - albedo is piecewise constant
  - there are “few” albedo values
  - albedo and shading explain image
- Solve
  - eg Bell 14, Nestmeyer 17, Bi 15

Modern strategies - Regression

- Regression of ground truth against image
  - use training set from WHDR data (Narihira et al 2015)
    - and perhaps rendered data
  - surprisingly, rendered data is very helpful
    - Li et al 18; Bi et al 18; Fan et al 18; etc
- Surprising because
  - Albedo in renderings isn't like albedo in the world
  - Illumination in renderings *really* isn't like illumination in the world

Recent history

TABLE 1

*Summary comparison to recent high performing supervised (above) and unsupervised (below) methods, all evaluated on the standard IIW test set; sources indicated. We distinguish between training with IIW and threshold selection using IIW. WHDR values computed for Retinex use the most favorable scaling, using the rescaling experiments of [12]. For our method, we report the held-out threshold value of WHDR. We report two figures for [13], because we found two distinct figures in the literature. Key: *: method uses IIW training data to set scale or threshold ONLY. +: [14] build models of albedo and shading from CGI, but do not use them for direct supervision. a: [15] use patches of registered images from MegaDepth.*

Class	Method	Source	IIW labels	CGI labels	Flattening	Test WHDR
N	*Zhao *et al.* '12 [16]	[12]	N	N	N	26.4
	*Shen and Yeo '11 [17]	[12]	N	N	N	26.1
	Yu and Smith '19 [15]	ibid	N	N	N	21.4 (a)
	Retinex (rescaled; color/gray)	[12]	N	N	N	19.5*/18.69*
	*Bell *et al* '14 [11]	[12]	N	N	Y	18.6
	Liu *et al* '20 [14]	ibid	N	Y+	N	18.69
	Bi *et al* '15 [13]	ibid	N	N	Y	18.1
	Bi *et al* '15 [13]	[18]	N	N	Y	17.69
S	Liu *et al* '20 [14]	ibid	N	Y+	N	18.69
G	Shi *et al.* '17 [19]	[18]	N	Y	N	54.44
	Zhou *et al* '15 [20]	[18]	Y	N	Y	19.95
	*Narihira *et al* [12]	ibid	N	N	N	18.1
	Bi *et al* '18 [18]	ibid	N	Y	Y	17.18
	Zhou *et al* '15 [21]	ibid	Y	N	Y	15.7
	Li and Snavely '18 [1]	ibid	Y	Y	Y	14.8
	Fan *et al* '18 [22]	ibid	Y	N	Y	14.45

WHDR is tricky - I

From Fan 18

Methods	WHDR (mean)
Baseline (const shading)	51.37
Baseline (const reflectance)	36.54
Shen *et al.* 2011 [17]	36.90
Retinex (color) [11]	26.89
Retinex (gray) [11]	26.84
Garces *et al.* 2012 [9]	25.46
Zhao *et al.* 2012 [20]	23.20
L_1 flattening [3]	20.94
Bell *et al.* 2014 [2]	20.64
Zhou *et al.* 2015 [21]	19.95
Nestmeyer *et al.* 2017 (CNN) [16]	19.49
Zoran *et al.* 2015* [22]	17.85
Nestmeyer *et al.* 2017 [16]	17.69
Bi *et al.* 2015 [3]	17.67
Ours w/o D-Filter	15.40
Ours w/o joint training	14.52
Ours	**14.45**

Table 1. Quantitative results on the IIW benchmark. All the results are evaluated on the test split of [15], except for the one marked with * which is evaluated on their own test split and is not directly comparable with other methods.

Narihira et al 15

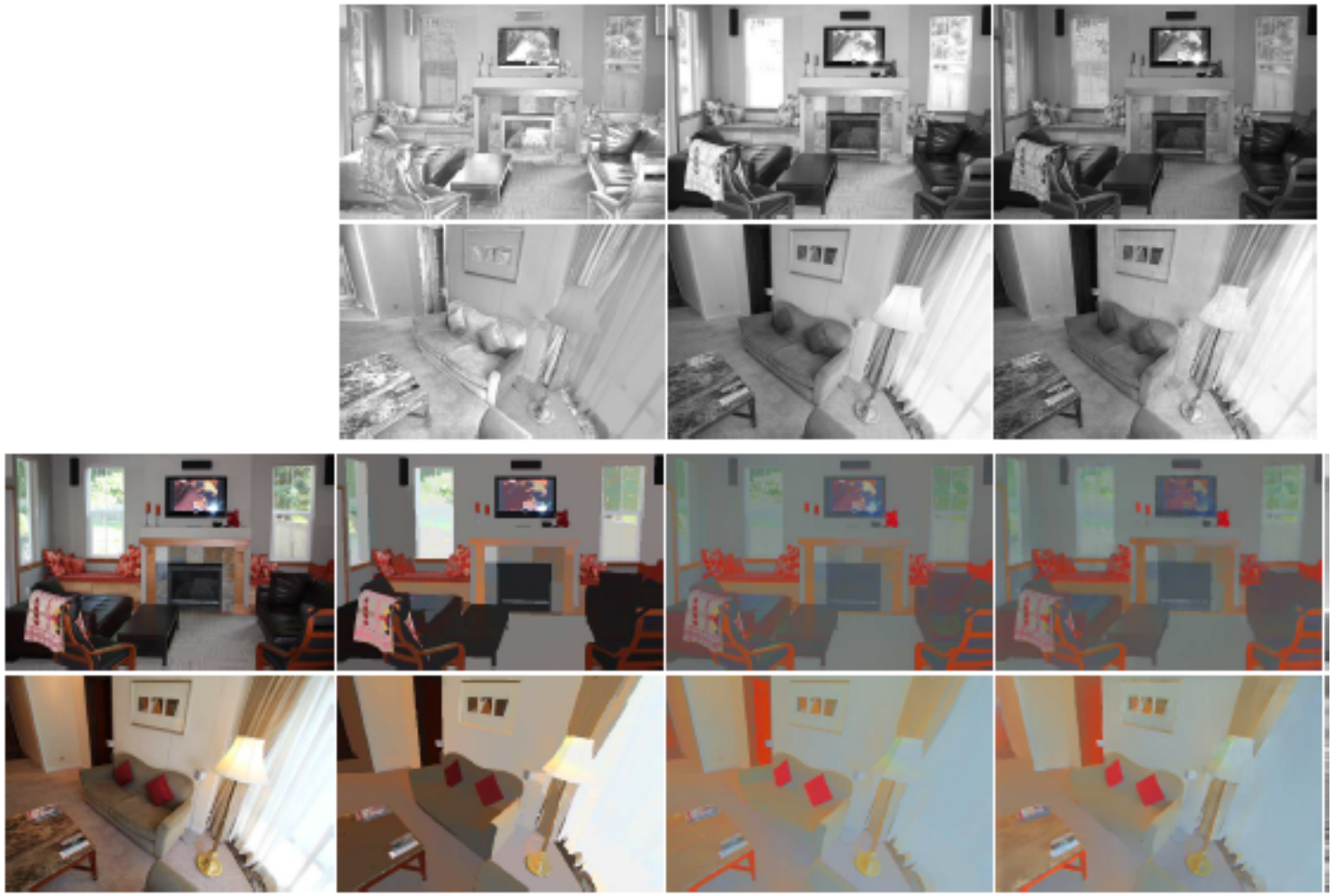
	WHDR (%)	Error Rate (%)
Ours (HSC)	20.9	24.5
Ours (CNN)	18.3	22.3
Ours (CNN-ImageNet)	**18.1**	**22.0**
CRF [4] (rescaled)	18.6	22.3
Retinex-Color [10] (rescaled)	19.5	23.3
Retinex-Gray [10] (rescaled)	19.8	23.8
Shen and Yeo [22] (rescaled)	23.2	26.1
Zhao *et al.* [26] (rescaled)	22.8	26.4
CRF [4]	20.6	25.6
Retinex-Color [10]	26.9	32.4
Retinex-Gray [10]	26.8	32.3
Shen and Yeo [22]	32.5	35.1
Zhao *et al.* [26]	23.8	28.2

Table 1. **Intrinsic Images in the Wild benchmark results.** For each algorithm, we display the weighted human disagreement rate (WHDR, lower is better), as well as the error rate on classifying the sign of lightness change between pairs of points labeled in the ground-truth. We include our own re-evaluation of competing methods, which closely matches the performance reported in [4]. In addition, we report performance of a rescaled version of competing methods, which specifically optimizes their output for the pairwise classification task. Our algorithm is on par with the CRF approach developed by [4] for state-of-the-art performance. We refer the reader to [4] for comparison to an expanded set of prior work.

WHDR is tricky - II

- Predict by
  - f(m1, m2) >t -> 1 is lighter
  - -t<f(m1, m2)<t -> same
  - f(m1, m2)<-t -> 2 is lighter
- Issues:
  - choice of f
    - m1 - m2
    - log(m1/m2)-1
  - choice of m
    - lightness potential
    - predicted albedo
  - choice of threshold
    - interacts with scale

WHDR is tricky - III



Fan 18 - current SOTA WHDR of 14.45%

WHDR is tricky - IV

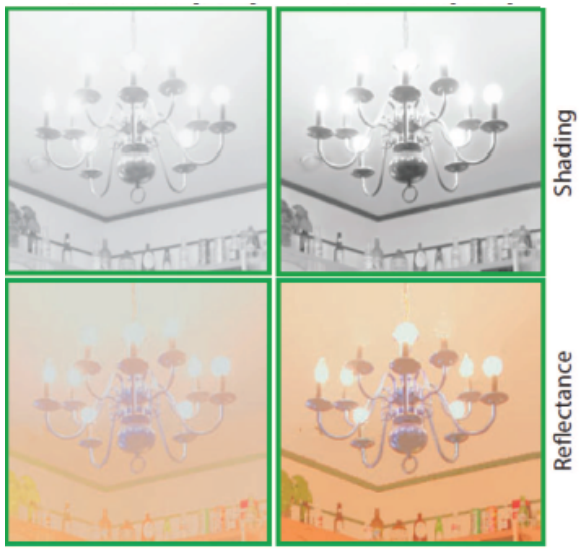
Bi et al, 2018 - this image WHDR 6.61%

WHDR: 75.70%
Shi et al. [2017]

WHDR: 36.03%
Narihira et al. [2015]

WHDR: 11.48%
Zhou et al. [2015]

- Note:
  - odd colors
  - “colored paper” effect
  - “indecision”

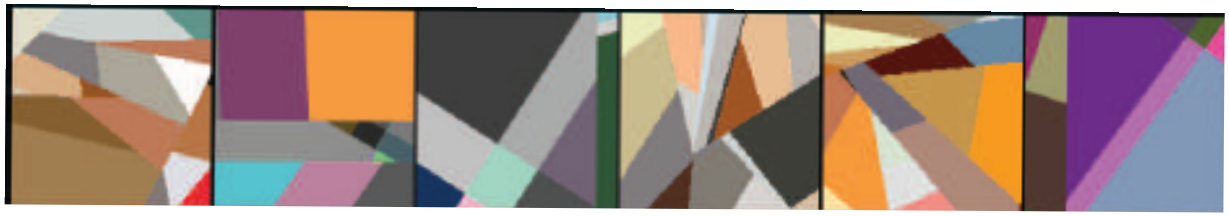


WHDR: 7.35%
Nestmeyer et al. [2017]

WHDR: 6.61%
Bi et al 2018

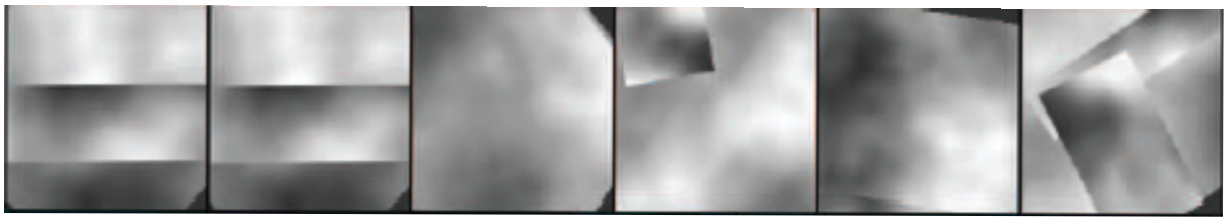
Spatial models

Albedo



*

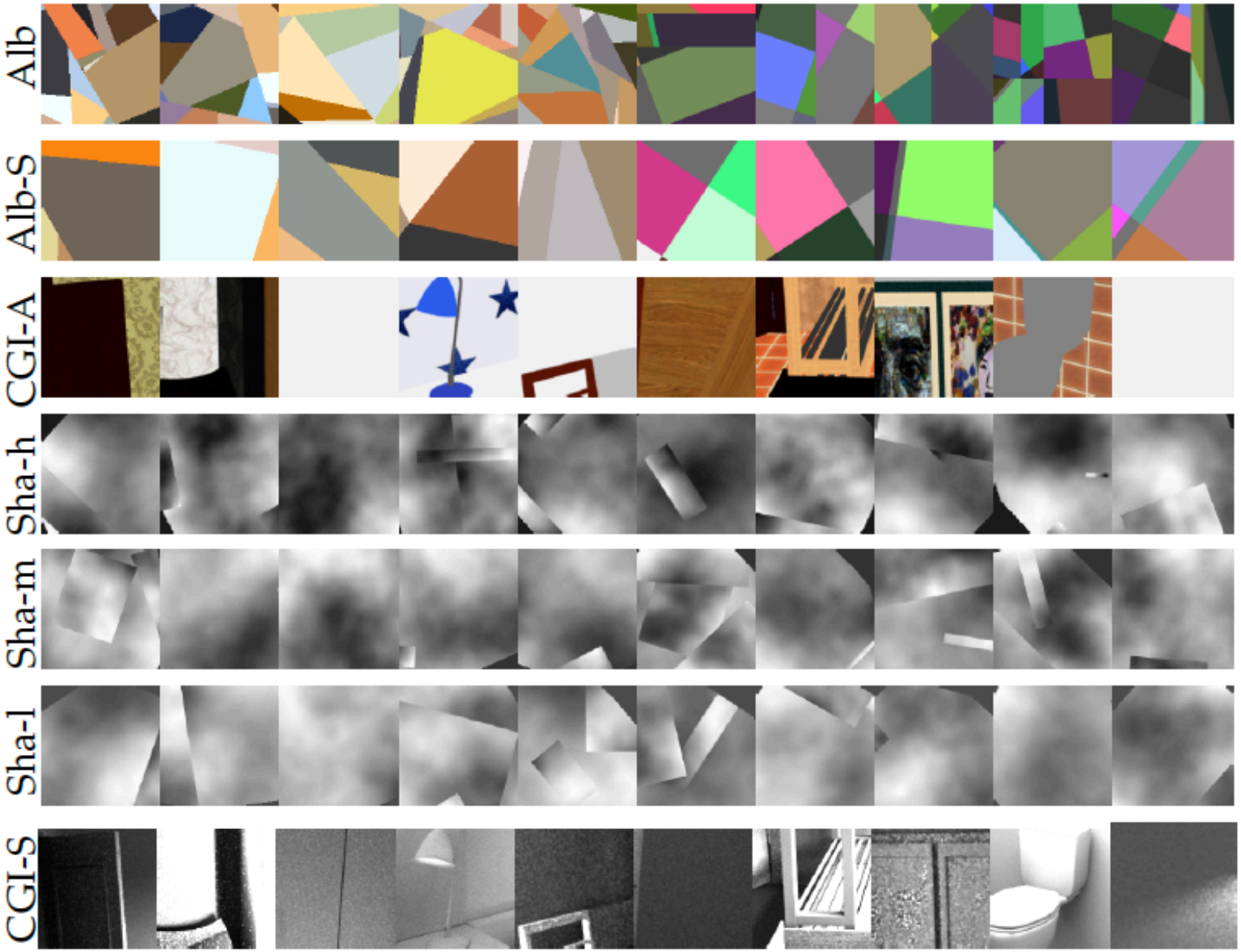
Shading



=

Image

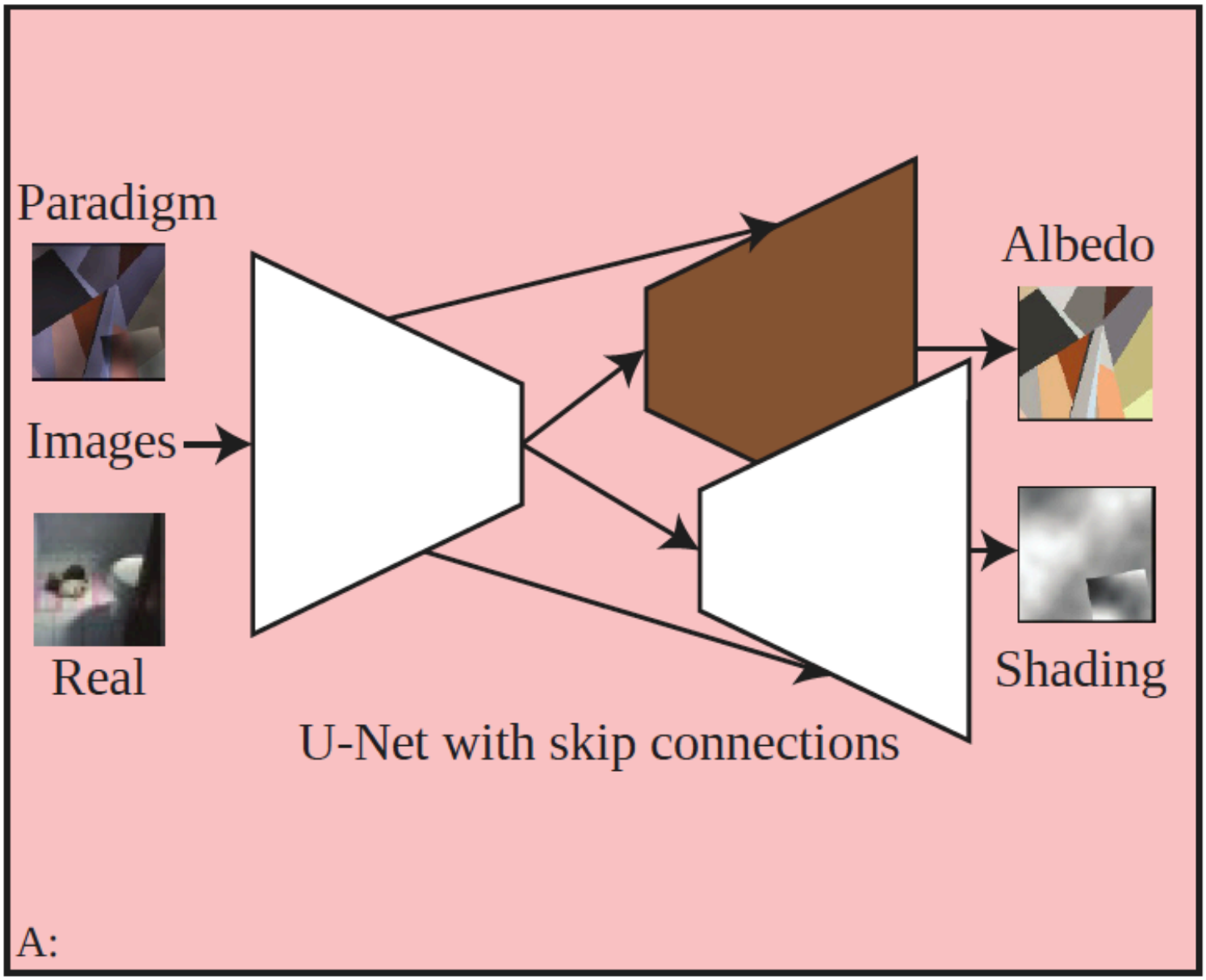
Various options



Choosing paradigms

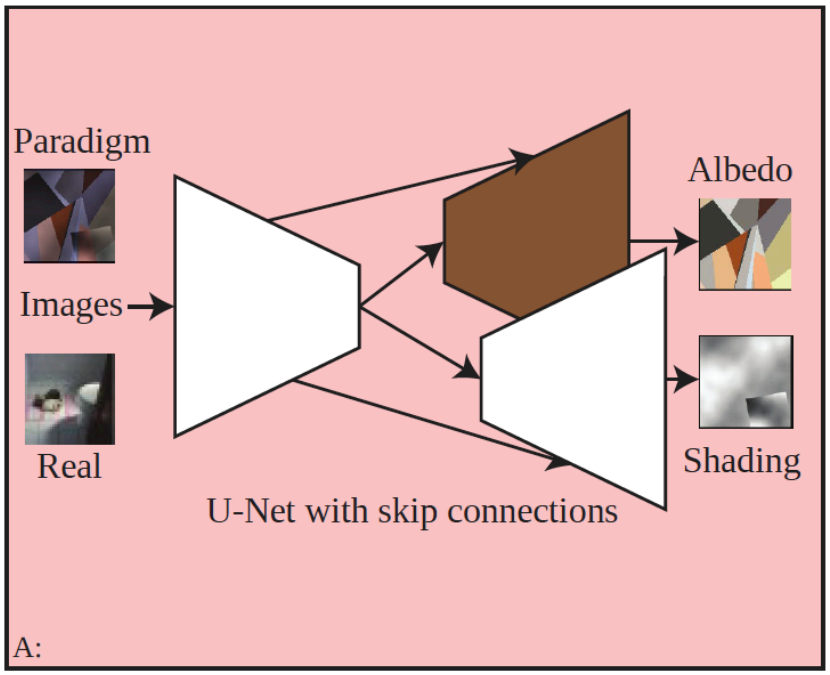
- Albedo paradigm captures:
  - albedos piecewise constant
  - reasonable color distribution
  - many edges; no orientation bias; some vertices with degree>3
- Shading paradigm captures:
  - mostly smooth, but some sharp edges
  - some dark/light spots
  - uniform color
- Samples from a spatial model
  - chosen by best guess; doesn't seem to matter much

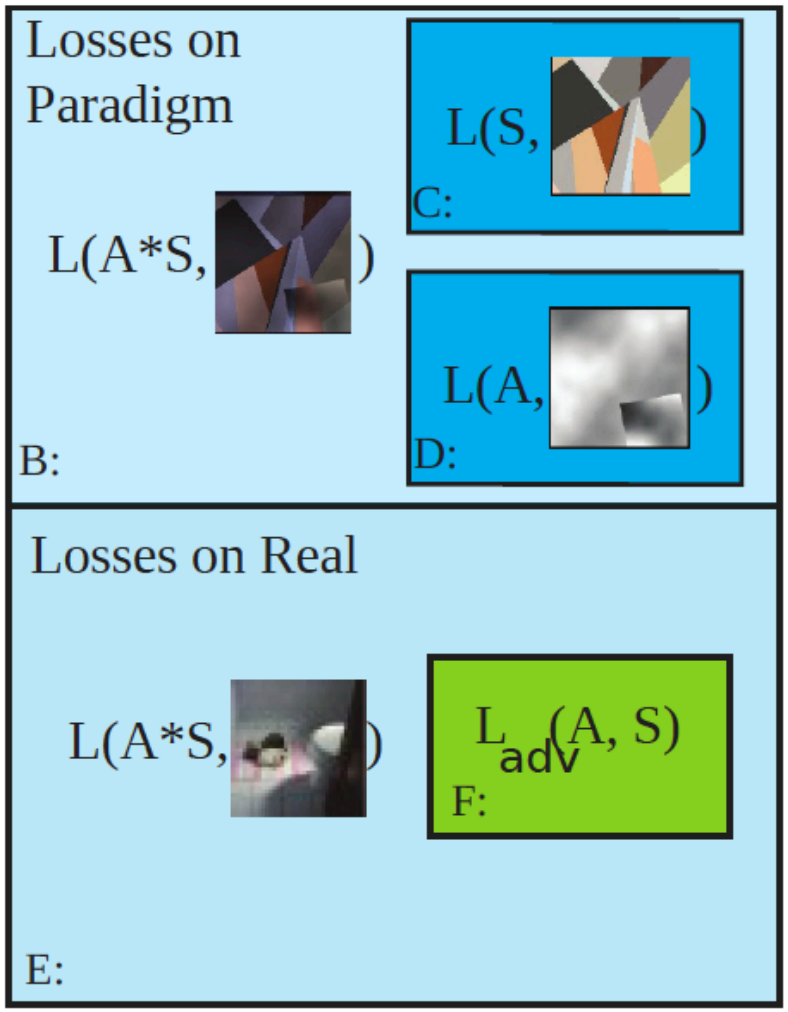
A regression network



Easy losses

- Paradigms should be correctly decomposed
  - with small residual
- Composing decomposed images
  - should have small residual





Training constraints

- Real images should
  - have albedo that locally “looks like” paradigms
  - have shading that locally “looks like” paradigms
  - have small residual

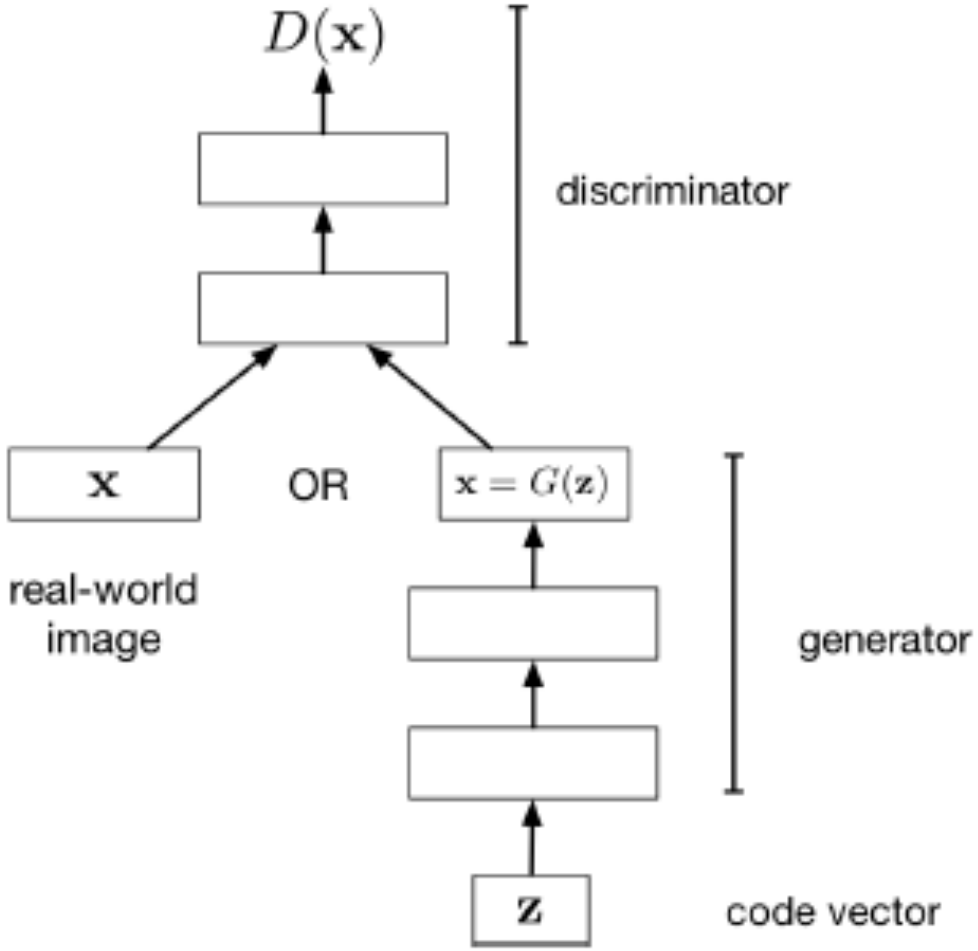
Locally = PatchGAN like trick

Side topic - Adversarial losses

- Issue:
  - we are making pictures should have a strong structure
    - albedo piecewise constant, etc.
    - but we don't know how to write a loss that imposes that structure
- Strategy:
  - build a classifier that tries to tell the difference between
    - true examples
    - examples we made
  - use that classifier as a loss

A GAN

Generative Adversarial Network



Grosse slides

- Let D denote the discriminator's predicted probability of being data
- Discriminator's cost function: cross-entropy loss for task of classifying real vs. fake images

$$\mathcal{J}_D = \mathbb{E}_{\mathbf{x}\sim\mathcal{D}}[-\log D(\mathbf{x})] + \mathbb{E}_{\mathbf{z}}[-\log(1 - D(G(\mathbf{z})))]$$

Notice: we want the discriminator to make a 1 for real data, 0 for fake data

- One possible cost function for the generator: the opposite of the discriminator's

$$\begin{aligned}\mathcal{J}_G &= -\mathcal{J}_D \\ &= \text{const} + \mathbb{E}_{\mathbf{z}}[\log(1 - D(G(\mathbf{z})))]\end{aligned}$$

- This is called the minimax formulation, since the generator and discriminator are playing a zero-sum game against each other:

$$\max_G \min_D \mathcal{J}_D$$

Solution (if exists, which is uncertain; and if can be found, ditto) is known as a saddle point. It has strong properties, but not much worth talking about, as we don't know if it is there or whether we have found it.

Grosse slides

Quote from the original paper on GANs:

> *"The generative model can be thought of as analogous to a team of counterfeiters, trying to produce fake currency and use it without detection, while the discriminative model is analogous to the police, trying to detect the counterfeit currency. Competition in this game drives both teams to improve their methods until the counterfeits are indistinguishable from the genuine articles."*

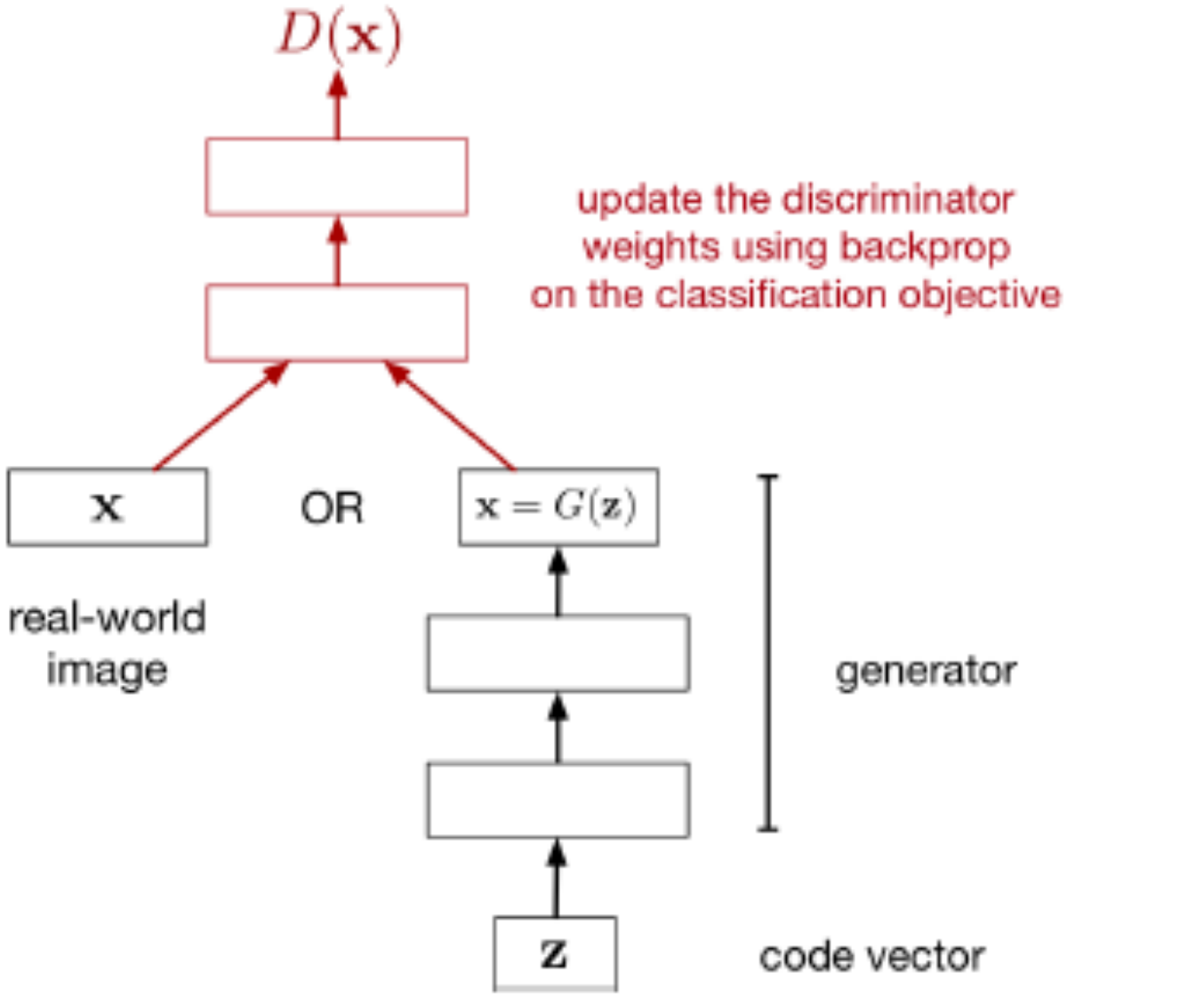
-Goodfellow et. al., "Generative Adversarial Networks" (2014)

Thakar slides

Important, general issue

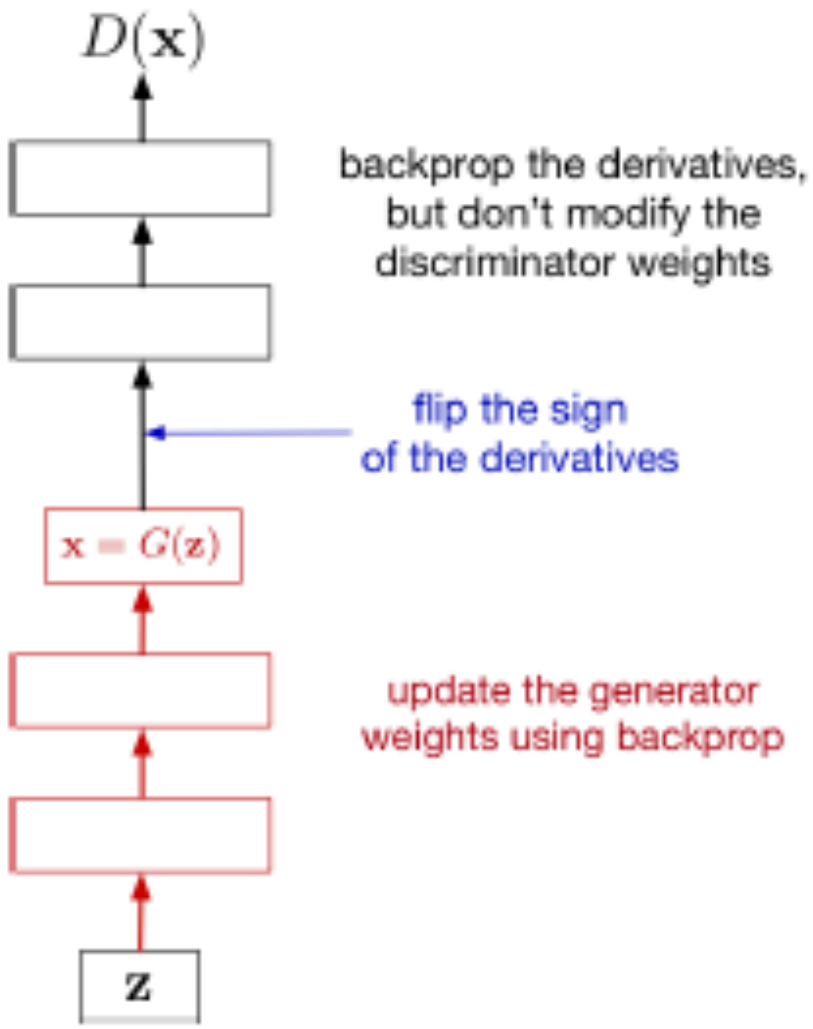
- If either generator or discriminator “wins” -> problem
- Discriminator “wins”
  - it may not be able to tell the generator how to fix examples
  - discriminators classify, rather than supply gradient
- Generator “wins”
  - likely the discriminator is too stupid to be useful
- Very little theory to guide on this point

Updating the discriminator:



Grosse slides

Updating the generator:



Grosse slides

One must be careful about losses…

- We introduced the minimax cost function for the generator:

$$\mathcal{J}_G = \mathbb{E}_{\mathbf{z}}[\log(1 - D(G(\mathbf{z})))]$$

- One problem with this is saturation.
- Recall from our lecture on classification: when the prediction is really wrong,
  - "Logistic + squared error" gets a weak gradient signal
  - "Logistic + cross-entropy" gets a strong gradient signal
- Here, if the generated sample is really bad, the discriminator's prediction is close to 0, and the generator's cost is flat.

Grosse slides

One must be careful about losses…

- Original minimax cost:

$$\mathcal{J}_G = \mathbb{E}_{\mathbf{z}}[\log(1 - D(G(\mathbf{z})))]$$

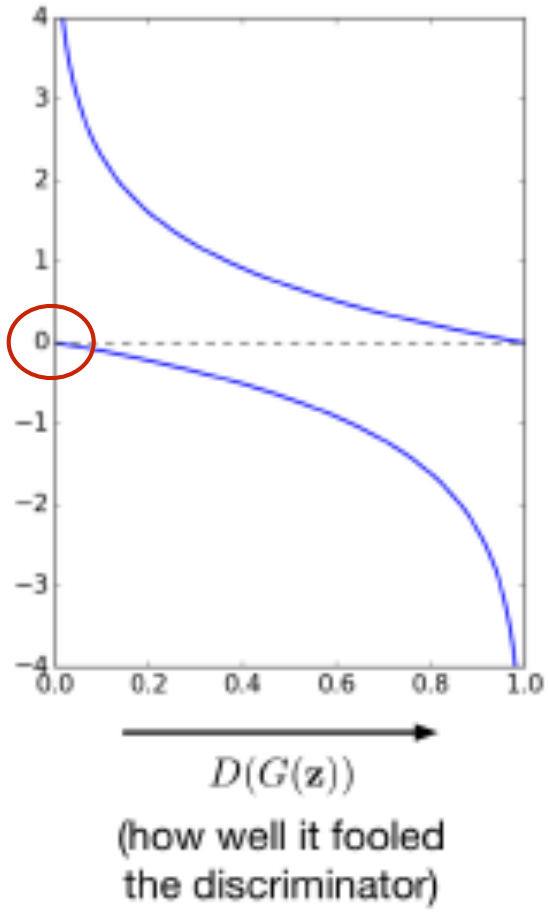
- Modified generator cost:

$$\mathcal{J}_G = \mathbb{E}_{\mathbf{z}}[-\log D(G(\mathbf{z}))]$$

- This fixes the saturation problem.

modified cost

minimax cost



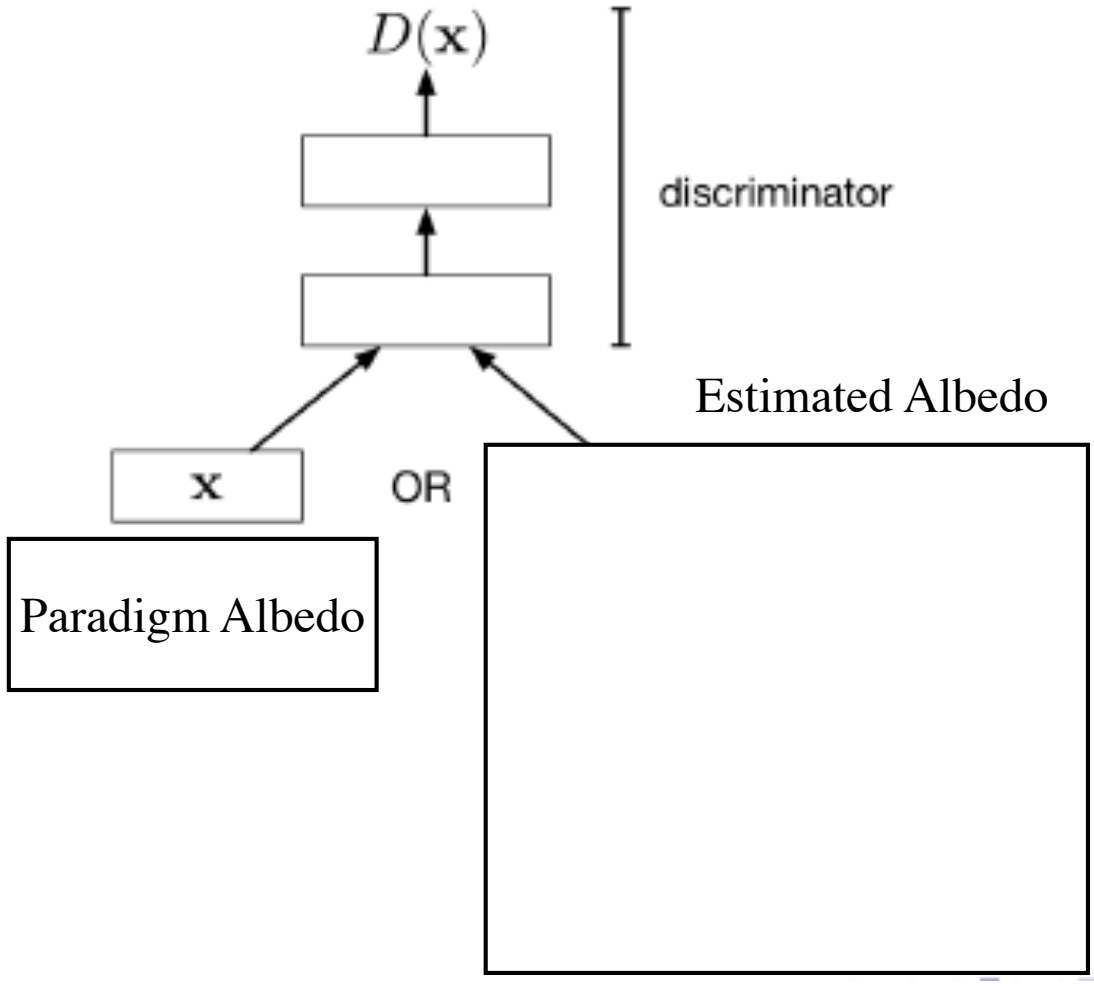
Grosse slides

Alternative losses

- Hinge:
  - Discriminator makes D(im)
    - want
      - real images -> -1
      - fake -> 1
  - Discriminator loss: $$\sum_{\text{fakes and real}} \max(0, 1 - y_i D(I_i))$$
    - where y_i=-1 for real, y_i=1 for fake
  - Generator loss:
    - $$\sum_{\text{fakes}} D(I_i)$$

Adversarial loss

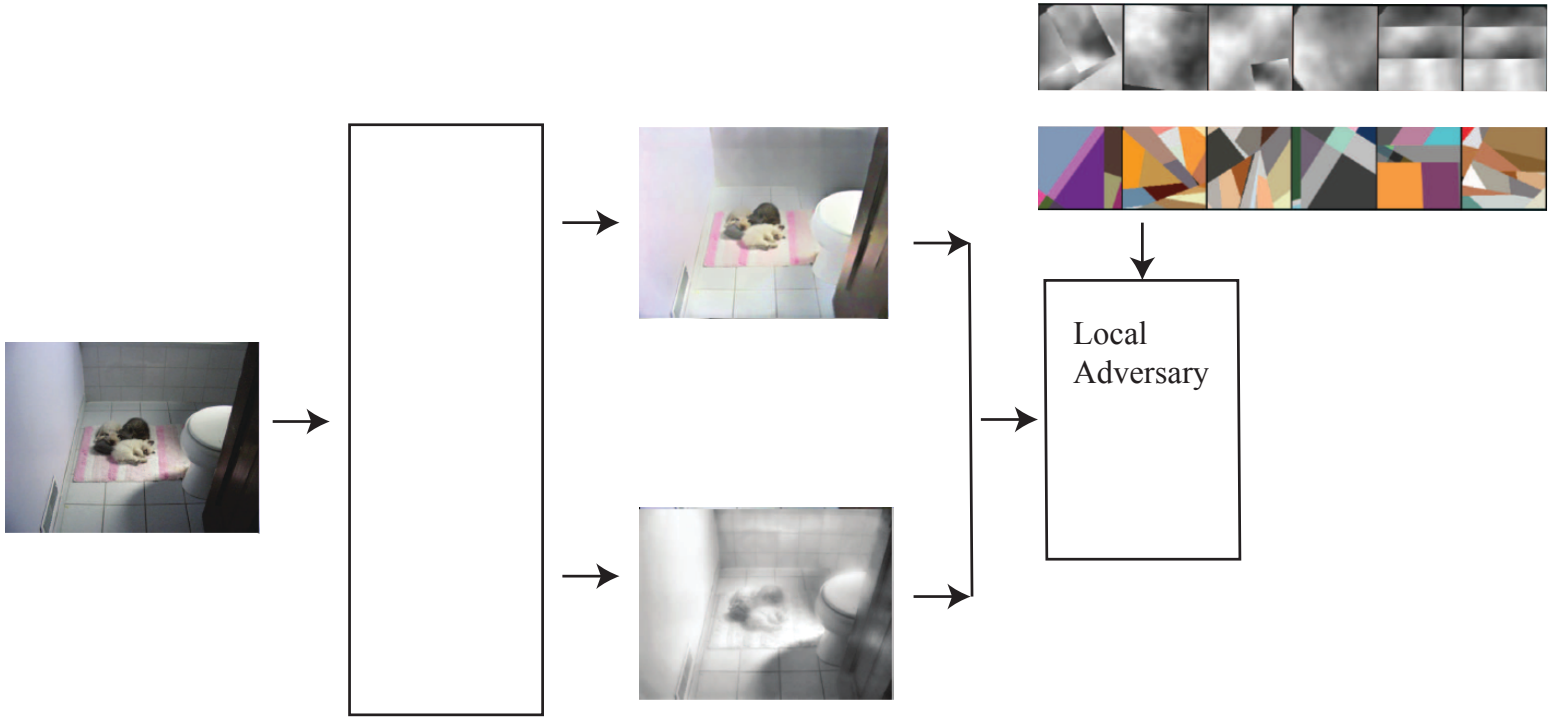
Adversarial loss



Grosse slides

Training constraints

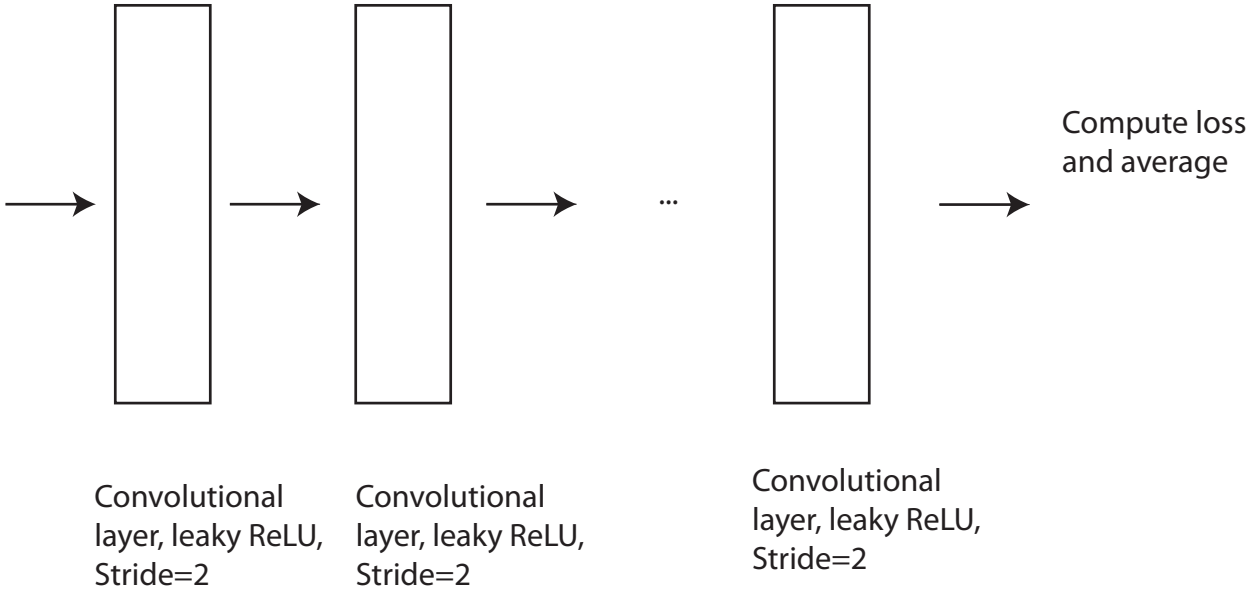
- Real images should
  - have albedo that locally "looks like" paradigms
  - have shading that locally "looks like" paradigms
  - have small residual



Locally = PatchGAN like trick

PatchGAN trick

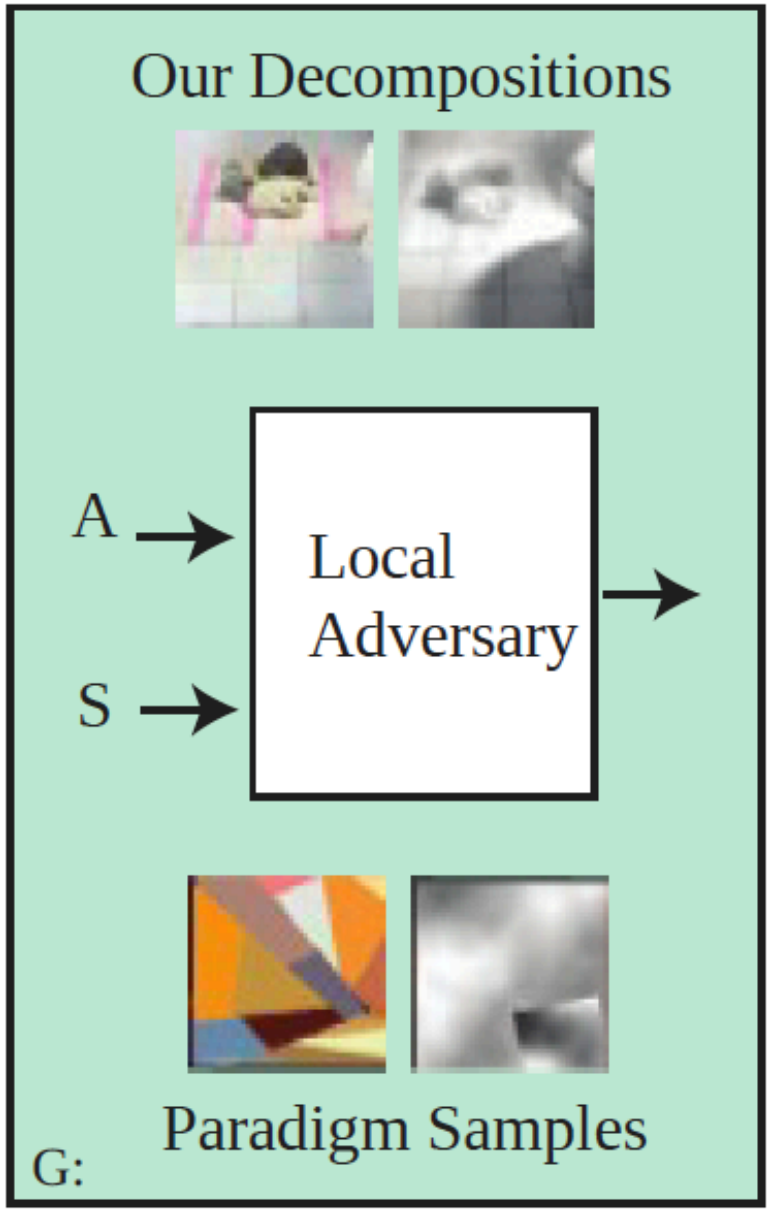
- Gen. albedos look like examples only at short scales
  - Discriminator should NOT see the whole example or it will win easily
- Trick



Adversarial Smoothing

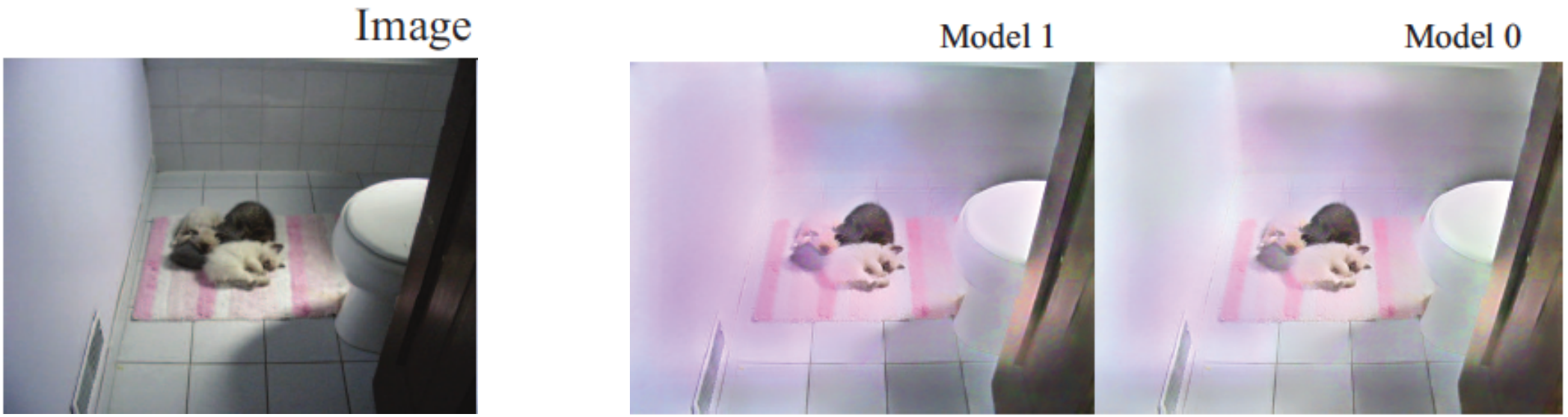
- Repeat:
  - Adjust adversary to distinguish between paradigms and network outputs
  - Adjust network outputs to fool adversary

- Origins in GAN's (Goodfellow et al 15), BUT
  - adversary sees paradigms, network outputs only locally
    - paradigms are short scale models
      - adjust discriminator so that output is mean of per-tile losses



Adversarial Smoothing

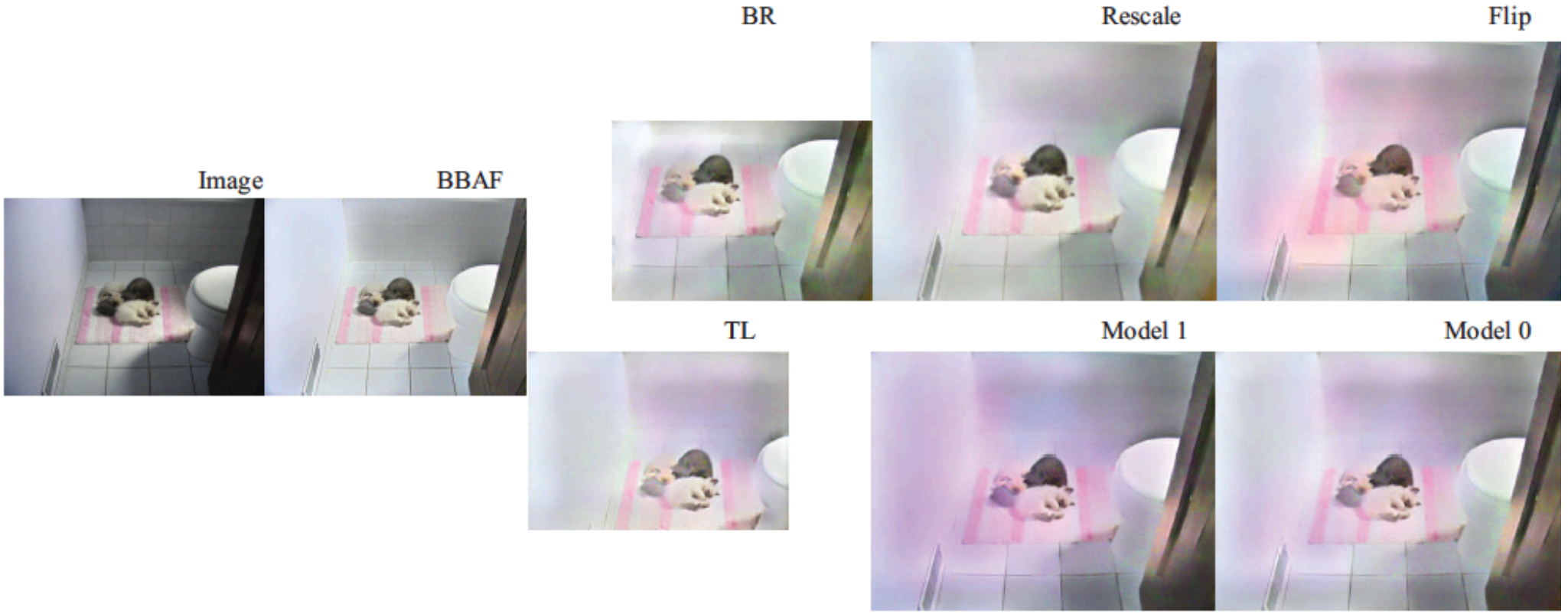
- BUT:
  - GAN “theory” doesn’t apply
  - no reason to believe that distributions can match
    - there may not be a saddle point
    - so this isn’t really a loss, and doesn’t really converge!
- Stopping training at different points -> different albedos!



Inference

- Network is trained on 128 x 128 tiles of image
- We want equivariance properties from albedo, shading
  - eg translate, rotate, scale image
    - albedo for translated (etc) image should be translated albedo
    - shading for translated (etc) image should be translated shading
- This doesn’t come naturally

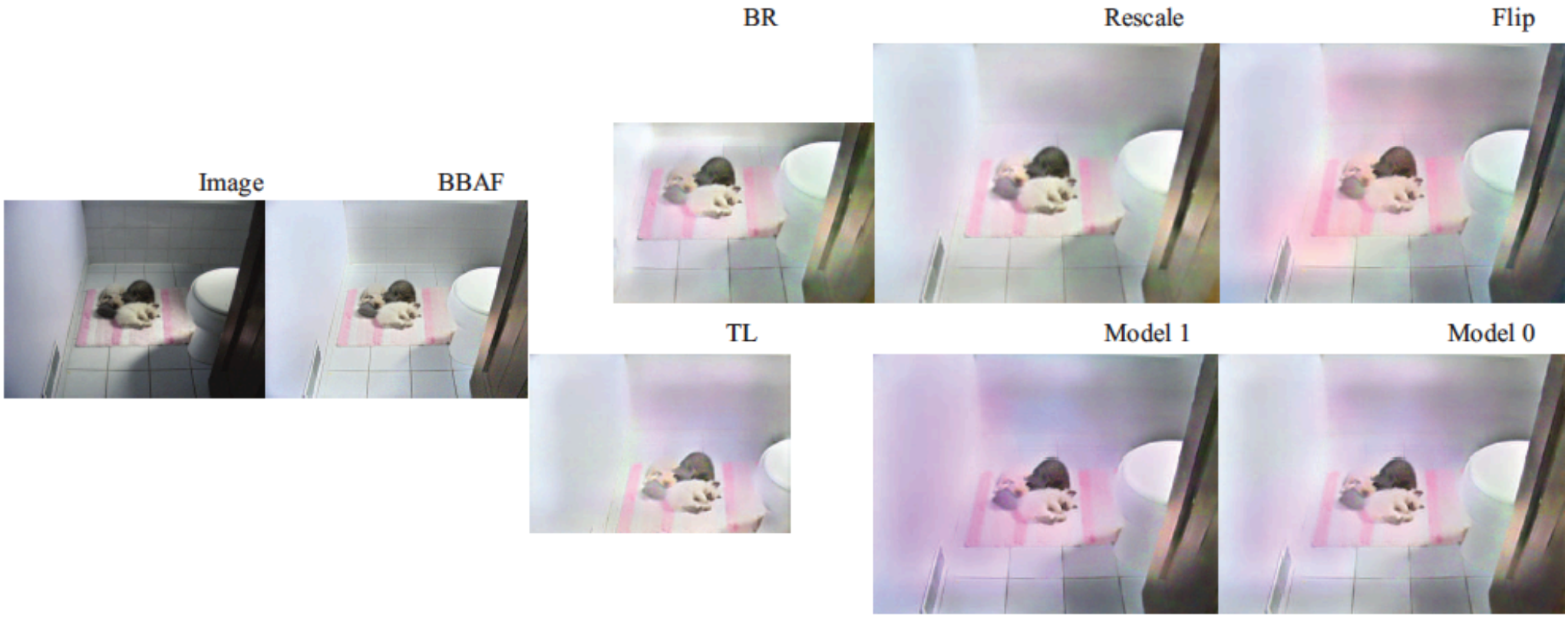
Equivariance must be imposed



Imposing equivariance

- Translation:
  - cover image with many, shifted, overlapping tiles
  - for each, recover albedo, shading
    - albedo at pixel is weighted average of all overlapping tiles
- Scale:
  - rescale image up, down
    - for each, recover albedo/shading using translation averaging
    - then rescale back
  - average results
- Rotation
  - average estimates from above over 8 flips

Averaging very strongly suppresses error



Results

TABLE 1

*Summary comparison to recent high performing supervised (above) and unsupervised (below) methods, all evaluated on the standard IIW test set; sources indicated. We distinguish between training with IIW and threshold selection using IIW. WHDR values computed for Retinex use the most favorable scaling, using the rescaling experiments of [12]. For our method, we report the held-out threshold value of WHDR. We report two figures for [13], because we found two distinct figures in the literature. Key: *: method uses IIW training data to set scale or threshold ONLY. +: [14] build models of albedo and shading from CGI, but do not use them for direct supervision. a: [15] use patches of registered images from MegaDepth.*

Class	Method	Source	IIW labels	CGI labels	Flattening	Test WHDR
N	*Zhao *et al.* '12 [16]	[12]	N	N	N	26.4
	*Shen and Yeo '11 [17]	[12]	N	N	N	26.1
	Yu and Smith '19 [15]	ibid	N	N	N	21.4 (a)
	Retinex (rescaled; color/gray)	[12]	N	N	N	19.5*/18.69*
	*Bell *et al* '14 [11]	[12]	N	N	Y	18.6
	Liu *et al* '20 [14]	ibid	N	Y+	N	18.69
	Bi *et al* '15 [13]	ibid	N	N	Y	18.1
	Bi *et al* '15 [13]	[18]	N	N	Y	17.69
PS	Liu *et al* '20 [14]	ibid	N	Y+	N	18.69
PS	Our best		N	N	N	16.86*
G	Shi *et al.* '17 [19]	[18]	N	Y	N	54.44
	Zhou *et al* '15 [20]	[18]	Y	N	Y	19.95
	*Narihira *et al* [12]	ibid	N	N	N	18.1
	Bi *et al* '18 [18]	ibid	N	Y	Y	17.18
	Zhou *et al* '15 [21]	ibid	Y	N	Y	15.7
	Li and Snavely '18 [1]	ibid	Y	Y	Y	14.8
	Fan *et al* '18 [22]	ibid	Y	N	Y	14.45

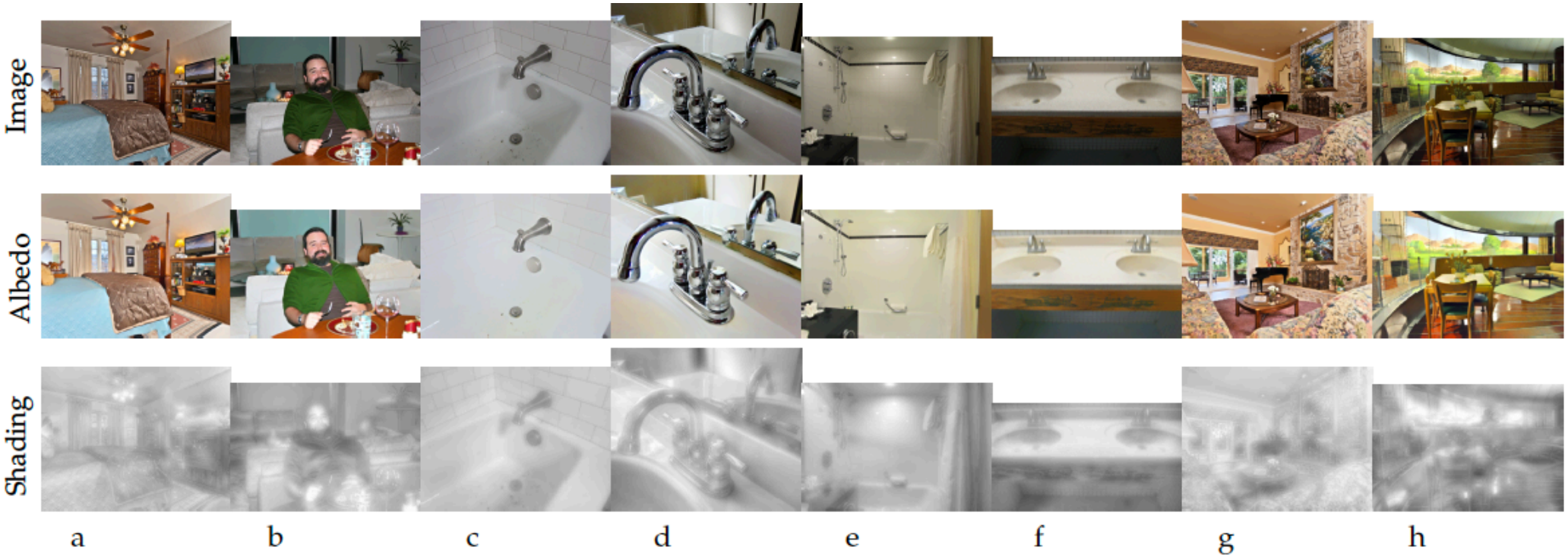


Fig. 3. *Albedo and shading estimates for a subset of IIW images, curated for qualitative effects. Note: strong suppression of shading on material folds (a, b); strong suppression of smooth shadows and glint (c, d, e, f); suppression of reflected smooth shadows (d - in mirror); error at sharp shadow boundaries (f, g - ceiling); apparent flattening (c, e, h). Shading fields are mostly smooth, but have some higher contrast edges.*

Indecisiveness remains (aargh!)

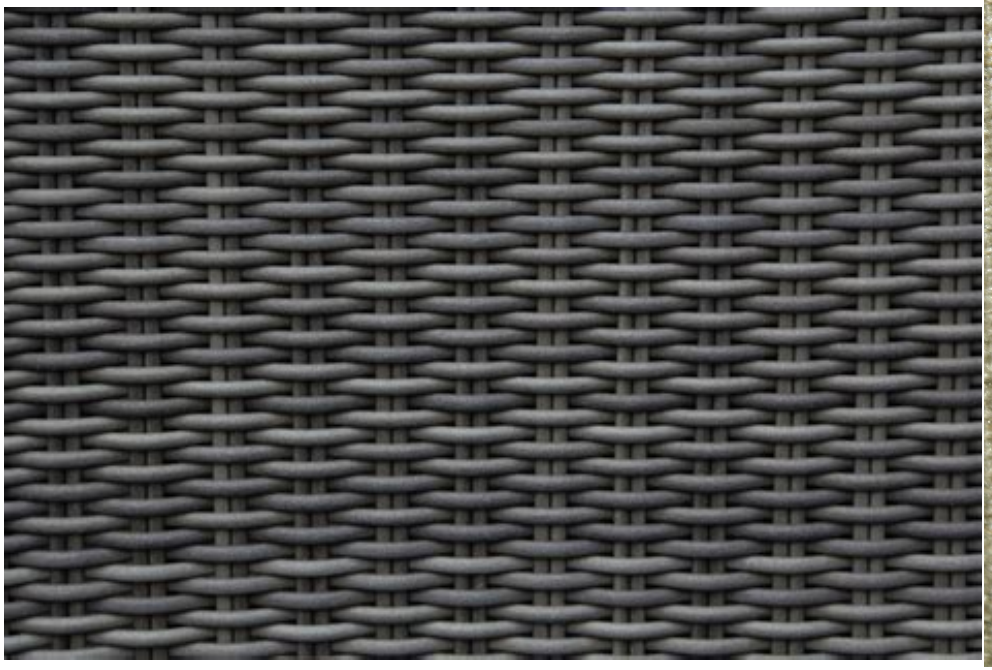
Fig. 13. *Our method suffers indecisiveness, as do others; this is a persistent problem in intrinsic image methods. Figures show a decomposition of an outdoor image, using our method. Note the pronounced shadow leaves effects in both albedo and shading fields; versions of this effect for other methods can be seen in Figure 6.* Best viewed in color.

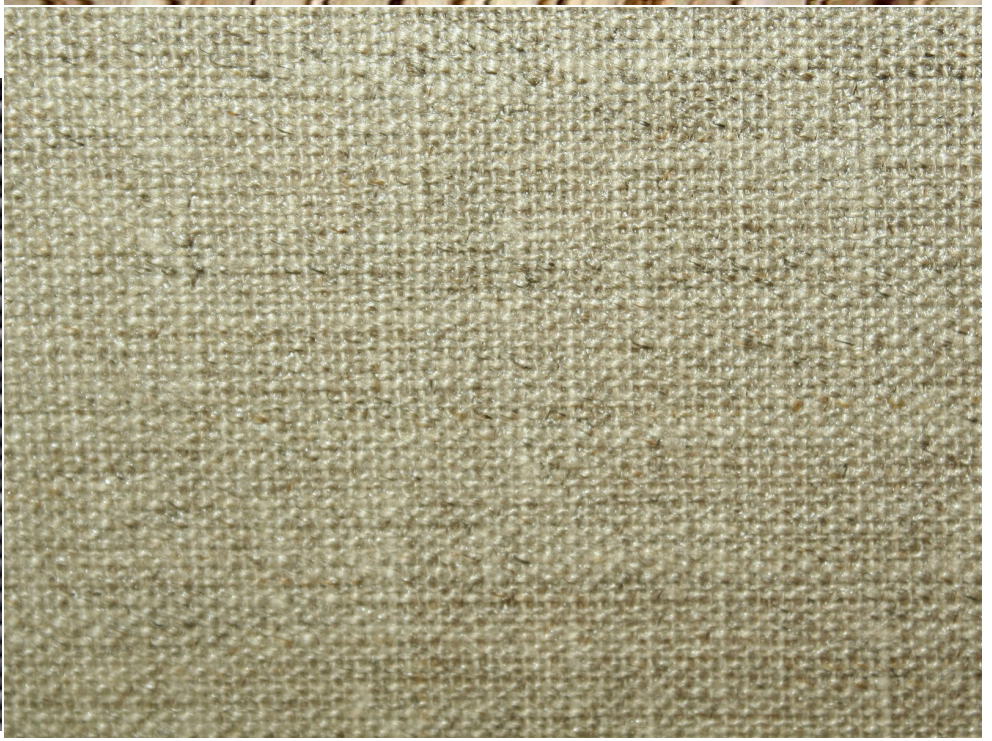
Other Possible Intrinsics

- Surface relief and material properties
  - and perhaps many of them
- Surface mechanical properties
- Surface glossiness
- Texture flow

Relief - intrinsic, because
small local shadows do not
move with illumination
(at least Koenderink+Van Doorn, 77)

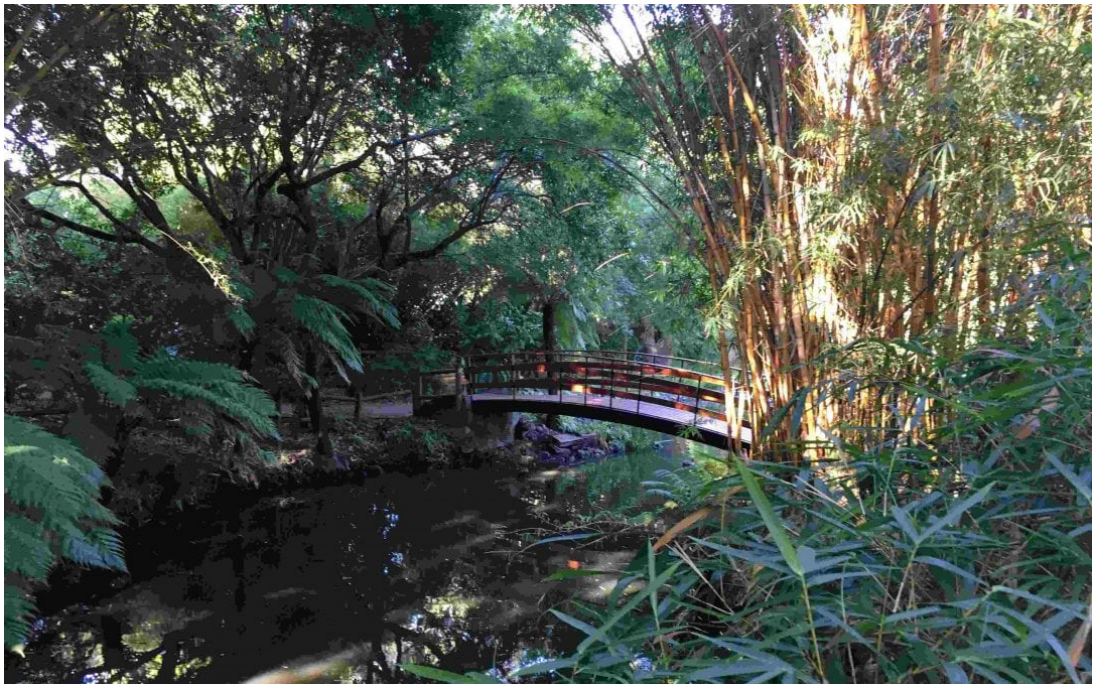
Relief - intrinsic, because
small local shadows do not
move with illumination
(at least Koenderink+Van Doorn, 77)



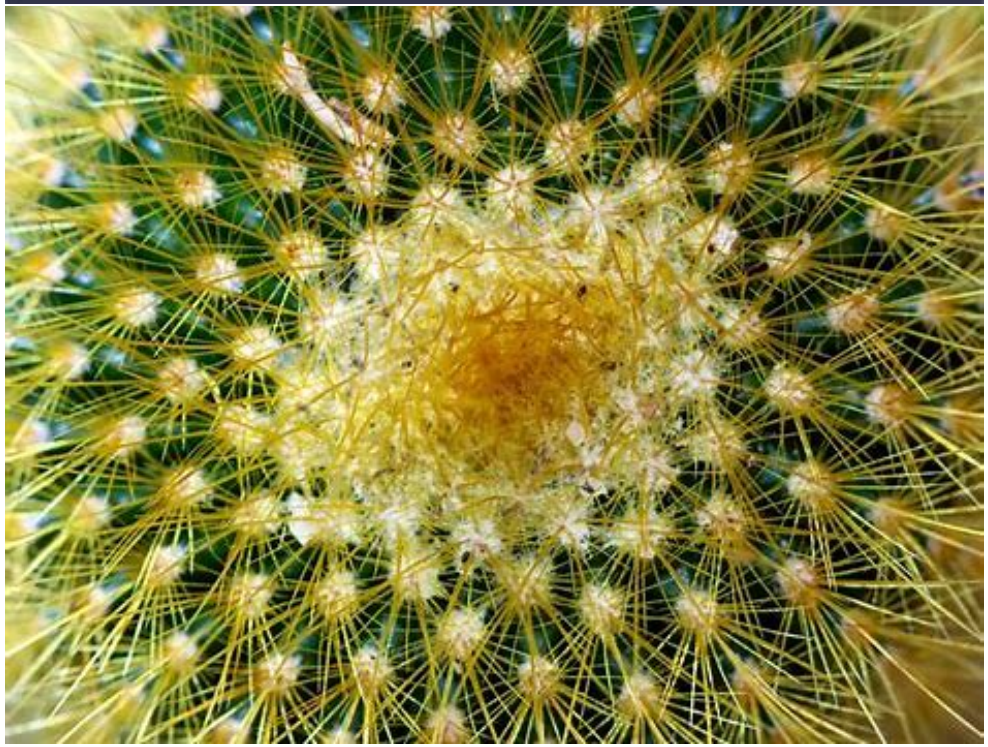


Fur - intrinsic, because
small local shadows do not
move with illumination
(at least Koenderink+Van Doorn, 77)

Relief - intrinsic (at least at this scale),
because small local shadows do not
move with illumination
(at least Koenderink+Van Doorn, 77)

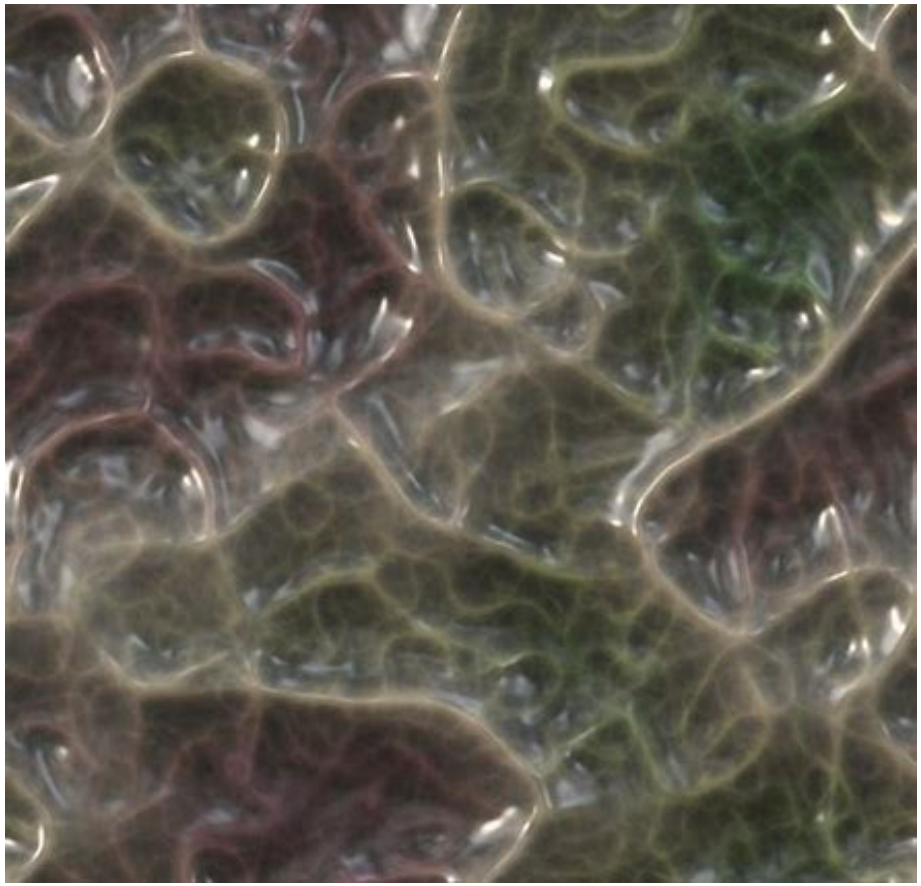


??? - intrinsic, because
mostly not a property of viewing
circumstances (?)

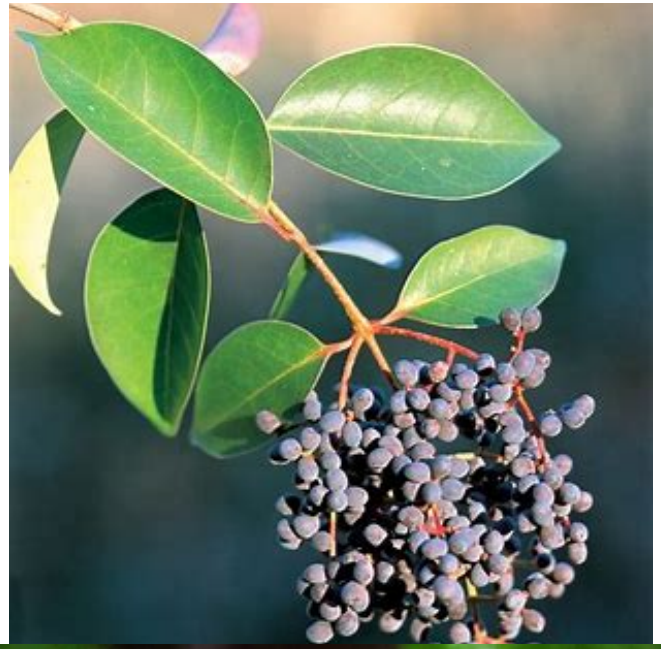


Iridescence
creating intrinsic gloss effects
intrinsic because the color effects will be
there for almost all illumination

??? - intrinsic, the specularities move but are always there



??? - intrinsic, the specularities move but are always there



Other Possible Extrinsics

- Glossy reflected component
- Luminaires
- Lens flare
- Rain effects
- etc.

Gloss/specular - clearly extrinsic, when the light moves, this moves

Lens flares - clearly intrinsic, product of viewing circumstances

Luminaires -
extrinsic or intrinsic?
worth knowing about, anyhow

Rain - multiple extrinsic phenomena,
including smoothing, raindrops, loss of saturation,
glossy/wet surfaces, etc. etc.

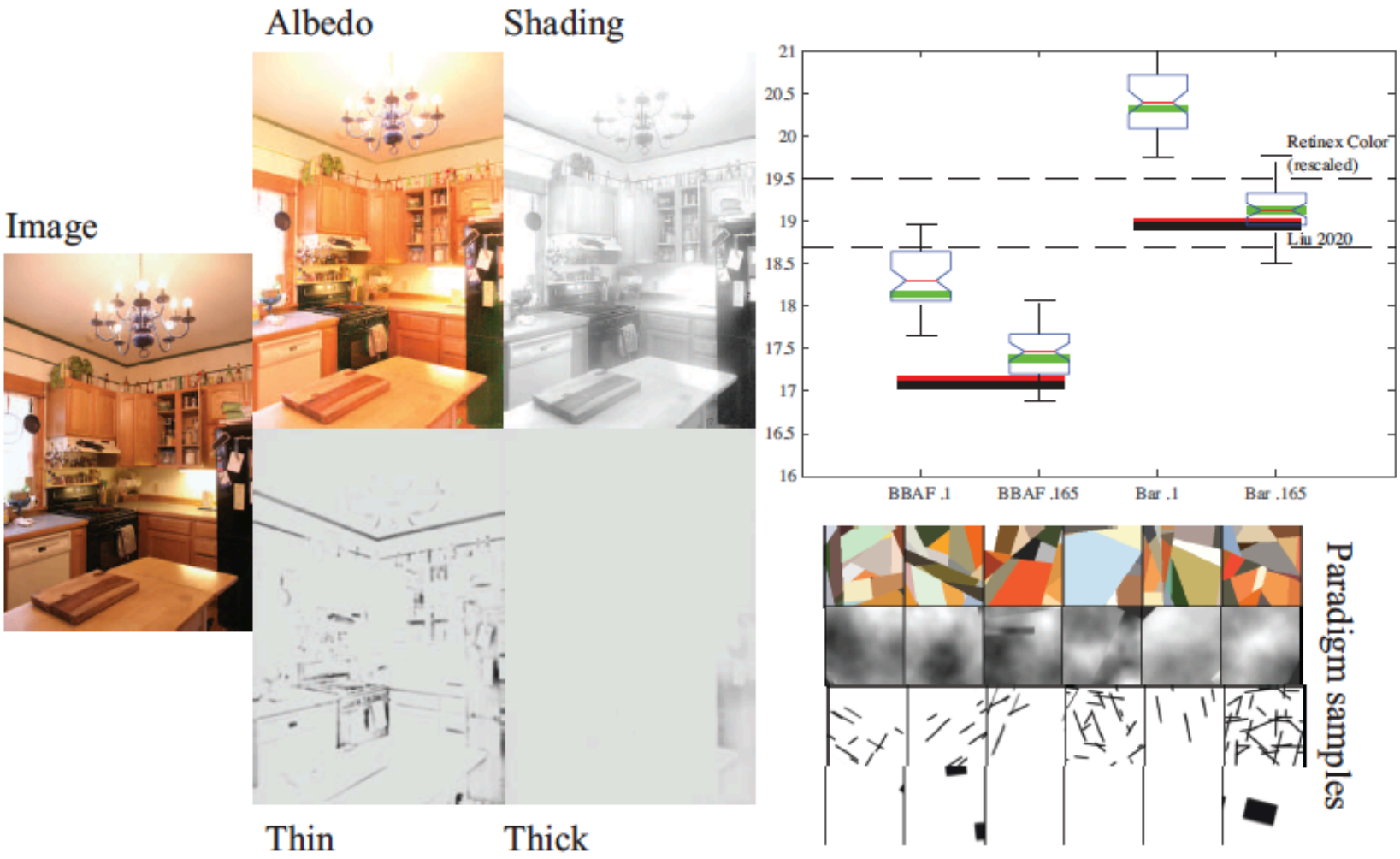


Fig. 12. *The method can be extended to capture thin and thick bars of darkness by extending the decomposer to have four heads (albedo, shading, thin bars, thick bars), and extending the paradigms (***bottom left** *shows examples). The advantage of doing so is that a decomposition will then capture the thin bars of darkness associated with grooves separately from albedo (example decomposition shown here). Qualitatively, these thin bars do appear to be associated with grooves (but note the thin dark paint bars on the ceiling, which also appear in this map). The cost in WHDR (***top right** *compares to* **BBAF***) is noticeable, but may be tolerable in some applications.* Best viewed in color.

No ground truth decompositions

- And there never will be
  - rendering is do-able (but hard)
  - modelling is hopeless
- Q: how do you train an image decomposition method when you don't know the right answer?
- Retinex provides clues - spatial statistics are the key