Today

- Interpretation tree
- Edges
- Bayes

Assignments

Take-home exam:

Given out Tuesday, March 15, due midnight, March 17.

Cannot collaborate on it.

Open book.

Problem set 2

- Can have until Monday 5pm to complete it.

6.869 projects

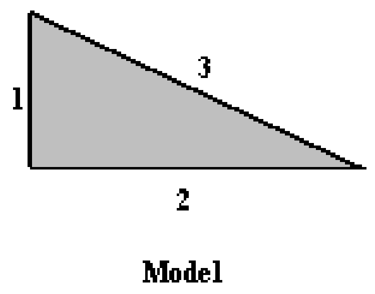
- Proposals to us by March 31 or earlier.
- We will ok them by April 5
- 3 possible project types:
  - Original implementation of an existing algorithm
  - Rigorous evaluation of existing implementation.
  - Synthesis or comparison of several research papers.

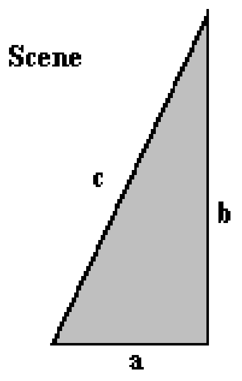
6.869 projects, continued

- Some possible projects
  - Evaluate the performance of local image feature descriptors.
  - Pose and solve a vision problem: make an algorithm that detects broken glass, or that finds trash. Implement and evaluate it.
  - Implement and evaluate the photographic/computer graphics discriminator.
  - Compare several motion estimation algorithms. Discuss how they're different, the benefits of each, etc. Put them in a common framework.

Interpretation Trees

- Tree of possible model-image feature assignments
- Depth-first search
- Prune when unary (binary, …) constraint violated
  - length
  - area
  - orientation





$$\begin{tikzcd}
 & (a,1) \arrow[dl, dash] \arrow[dr, dash] & \\
(b,2) \arrow[d, dash] & & \dots \\
\dots & &
\end{tikzcd}$$

Interpretation tree

The problem is to match the line primitives in the model, **{1, 2, 3}** to those in the scene, **{a, b, c}.** Select a scene feature at random, feature **a**, say. Choose a model feature at random. The choice **(a, 1)** represents a node in the tree. However, we could equally choose **(a, 2)** or **(a, 3)** as initial nodes. Thus there are three nodes at the first level of the tree.

Now expand each of these nodes. For example, if we choose to expand **(a, 1)** then the three children would be defined as **(b, 1), (b, 2)** and **(b, 3).** If we expand **(a, 2)** then the children are the same. Hence, for a **completely unconstrained** tree search matching a model of **n** primitives to a scene having **n** primitives there will **n** nodes at the first level, n^2 at the second level and so on until there are n^n nodes at the last level.

http://homepages.inf.ed.ac.uk/rbf/CVonline/LOCAL_COPIES/MARBLE/high/matching/tree.htm

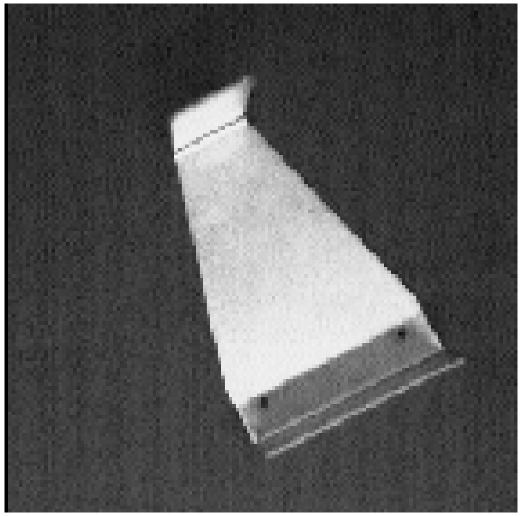
Interpretation tree

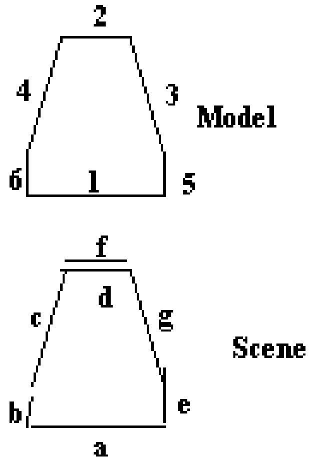
In general, we shall deal with **constrained** tree search. For example, is a scene labelling of **{(a, 3), (b, 3), (c,3)}** sensible ? Well it suggests that we can detect in the scene the hypoteneuses of three separate triangle, and that the other sides are occluded or otherwise undetected. Suppose we know a-priori that there is only one triangle in the scene ? Then, at the second level of the search tree we can only expand **(a, 1)** with **(b, 2)** and **(b, 3)**; this a *uniqueness constraint* by analogy with the stereo matching problem. Hence for each of **n** nodes at the first level, there are **n-1** children, then **n-2** children and so on.

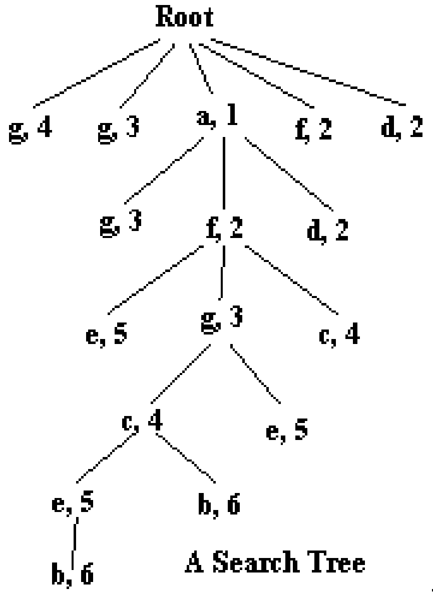
To reduce the combinatorics of the search still further, we should add additional constraints…**Unary** constraints apply to single pairings between model and scene features. For example we could introduce a constraint which says that lines can only be matched if they have the same length. **Binary** or **pairwise** constraints are based on pairs if features.

http://homepages.inf.ed.ac.uk/rbf/CVonline/LOCAL_COPIES/MARBLE/high/matching/tree.htm

Interpretation Trees







"Wild cards" handle spurious image features

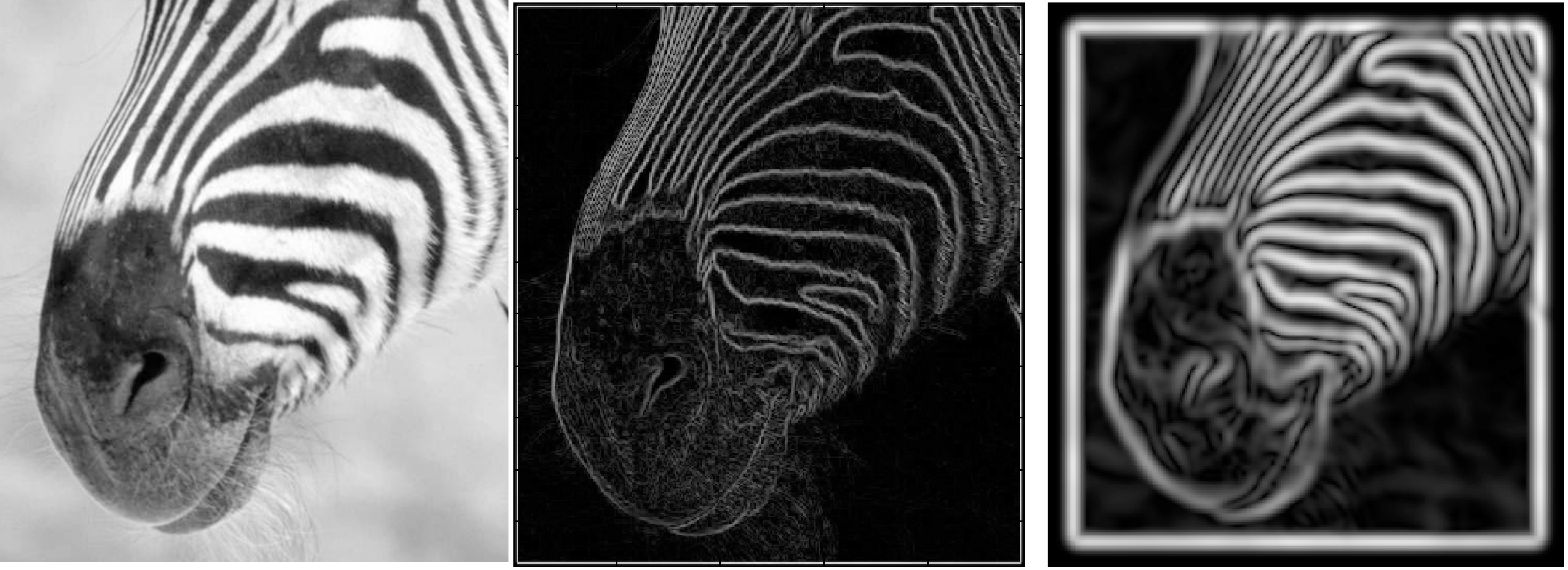
[A.M. Wallace. 1988.]

http://faculty.washington.edu/cfolson/papers/pdf/icpr04.pdf

Gradients and edges (Forsyth, ch. 8)

- Points of sharp change in an image are interesting:
  - change in reflectance
  - change in object
  - change in illumination
  - noise
- Sometimes called **edge points**

- General strategy
  - determine image gradient
  - now mark points where gradient magnitude is particularly large wrt neighbours (ideally, curves of such points).



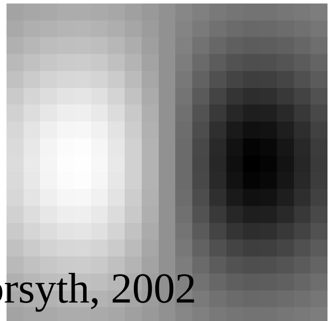
There are three major issues:

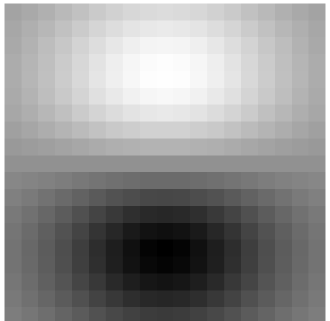
1) The gradient magnitude at different scales is different; which should we choose?
2) The gradient magnitude is large along thick trail; how do we identify the significant points?
3) How do we link the relevant points up into curves?

Forsyth, 2002

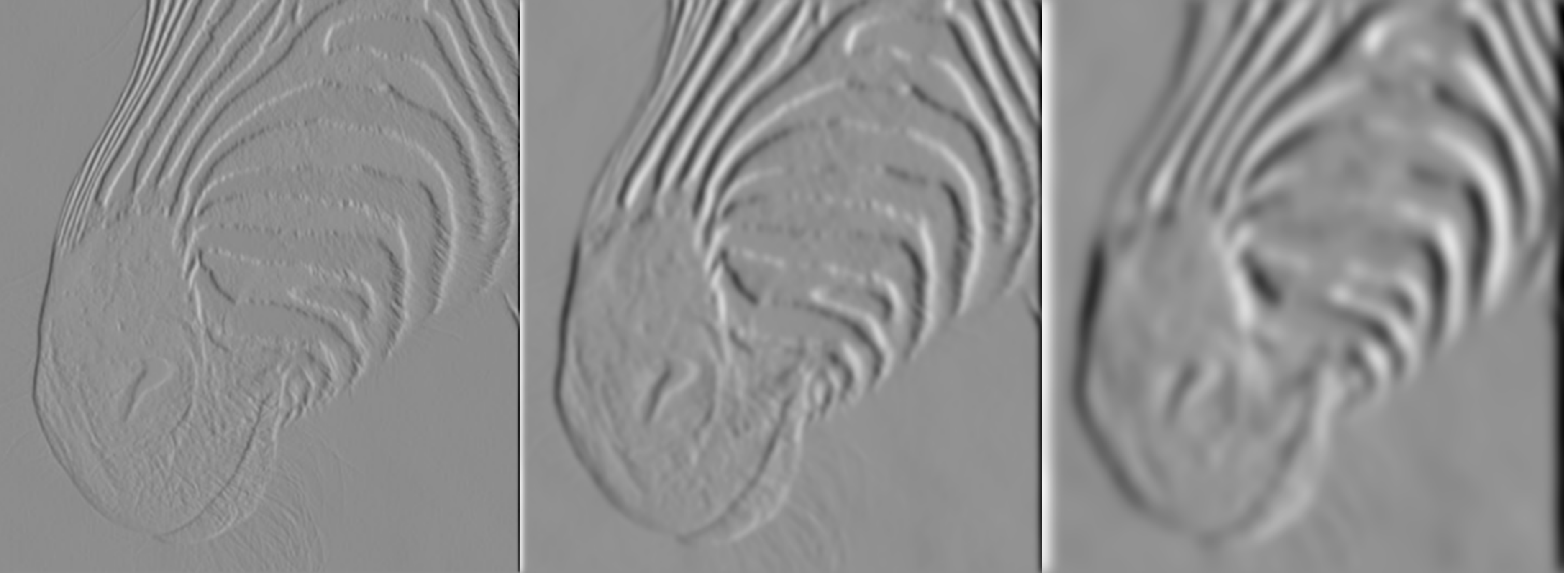
Smoothing and Differentiation

- Issue: noise
  - smooth before differentiation
  - two convolutions to smooth, then differentiate?
  - actually, no - we can use a derivative of Gaussian filter
    - because differentiation is convolution, and convolution is associative





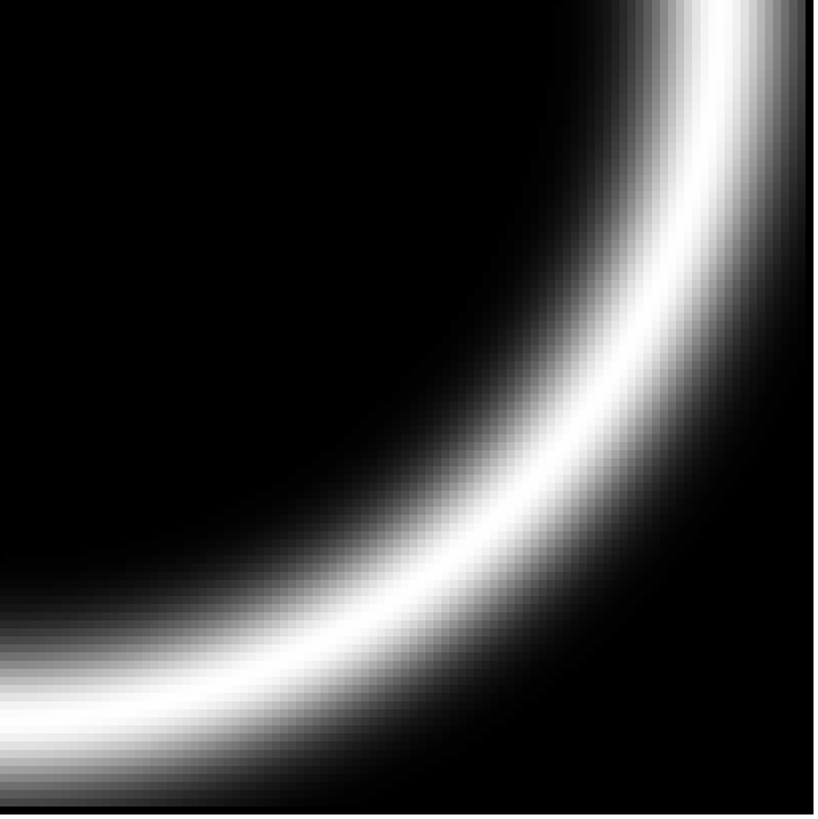
Forsyth, 2002

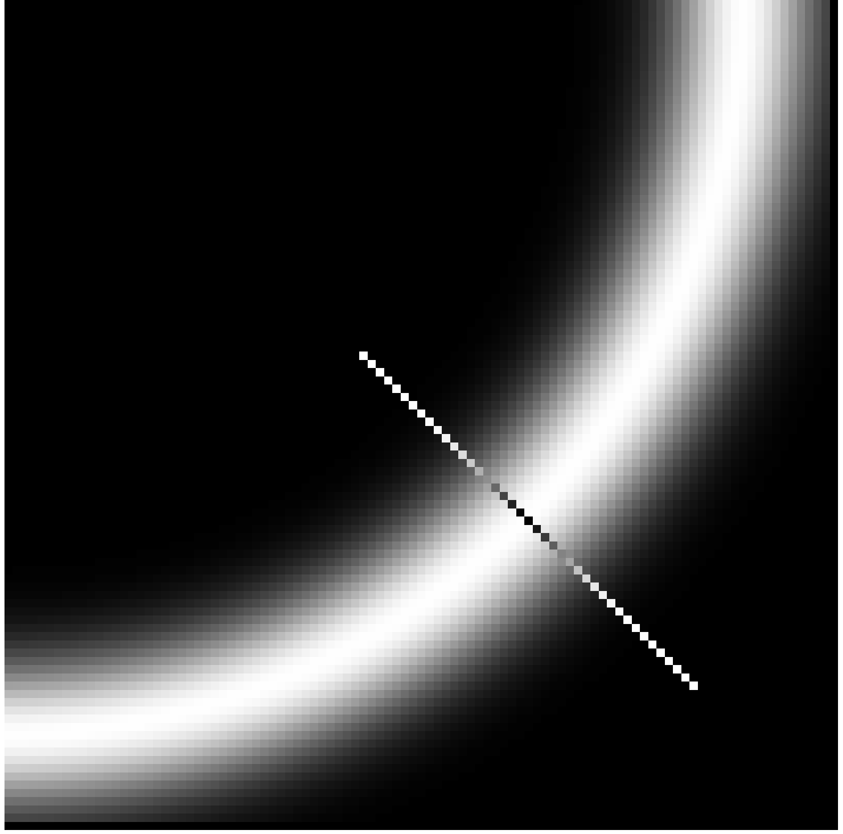


1 pixel 3 pixels 7 pixels

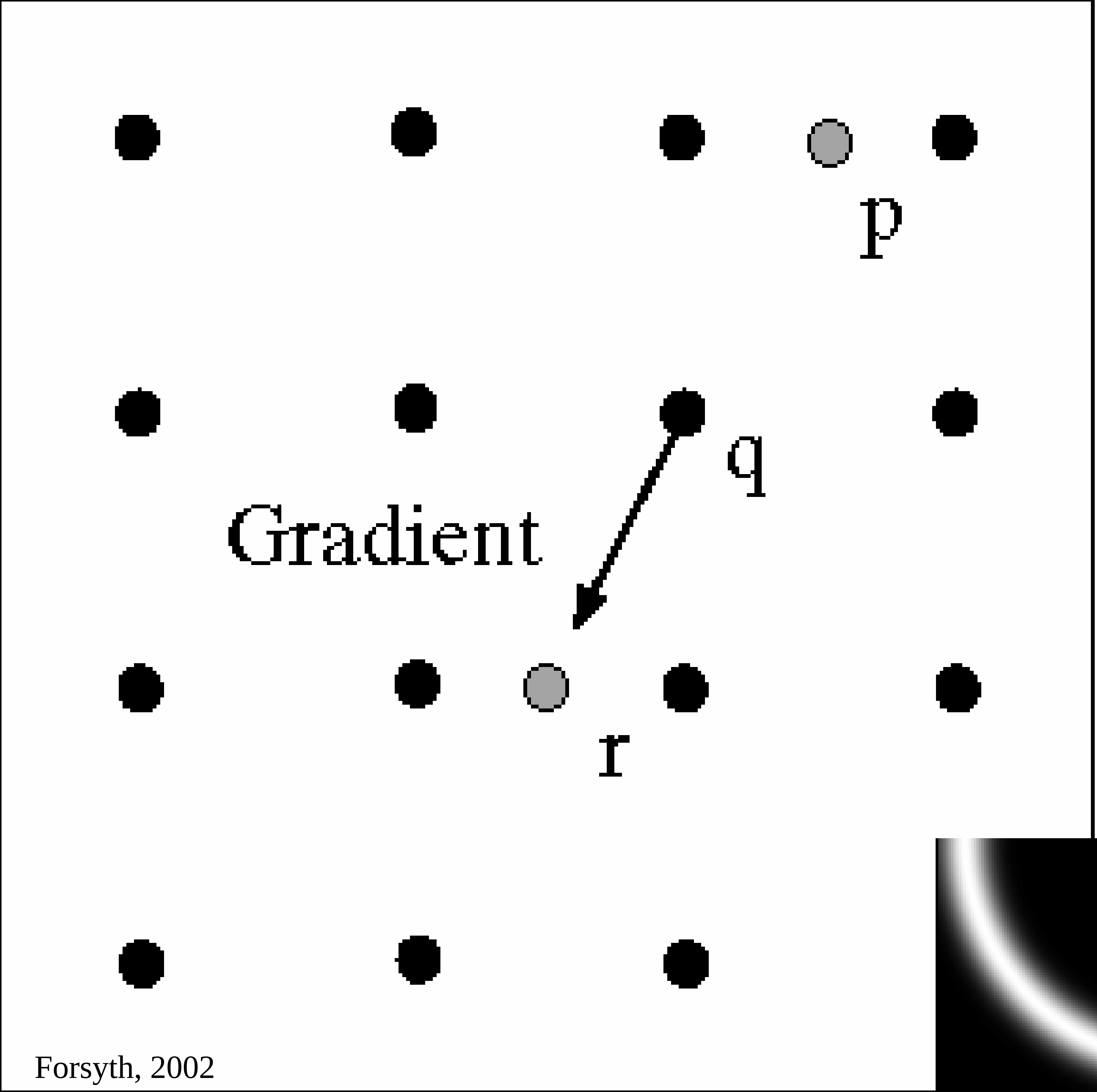
The scale of the smoothing filter affects derivative estimates, and also the semantics of the edges recovered.

Forsyth, 2002





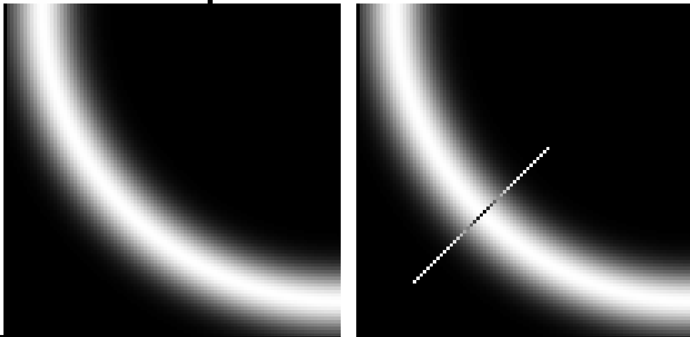
We wish to mark points along the curve where the magnitude is biggest. We can do this by looking for a maximum along a slice normal to the curve (non-maximum suppression). These points should form a curve. There are then two algorithmic issues: at which point is the maximum, and where is the next one?

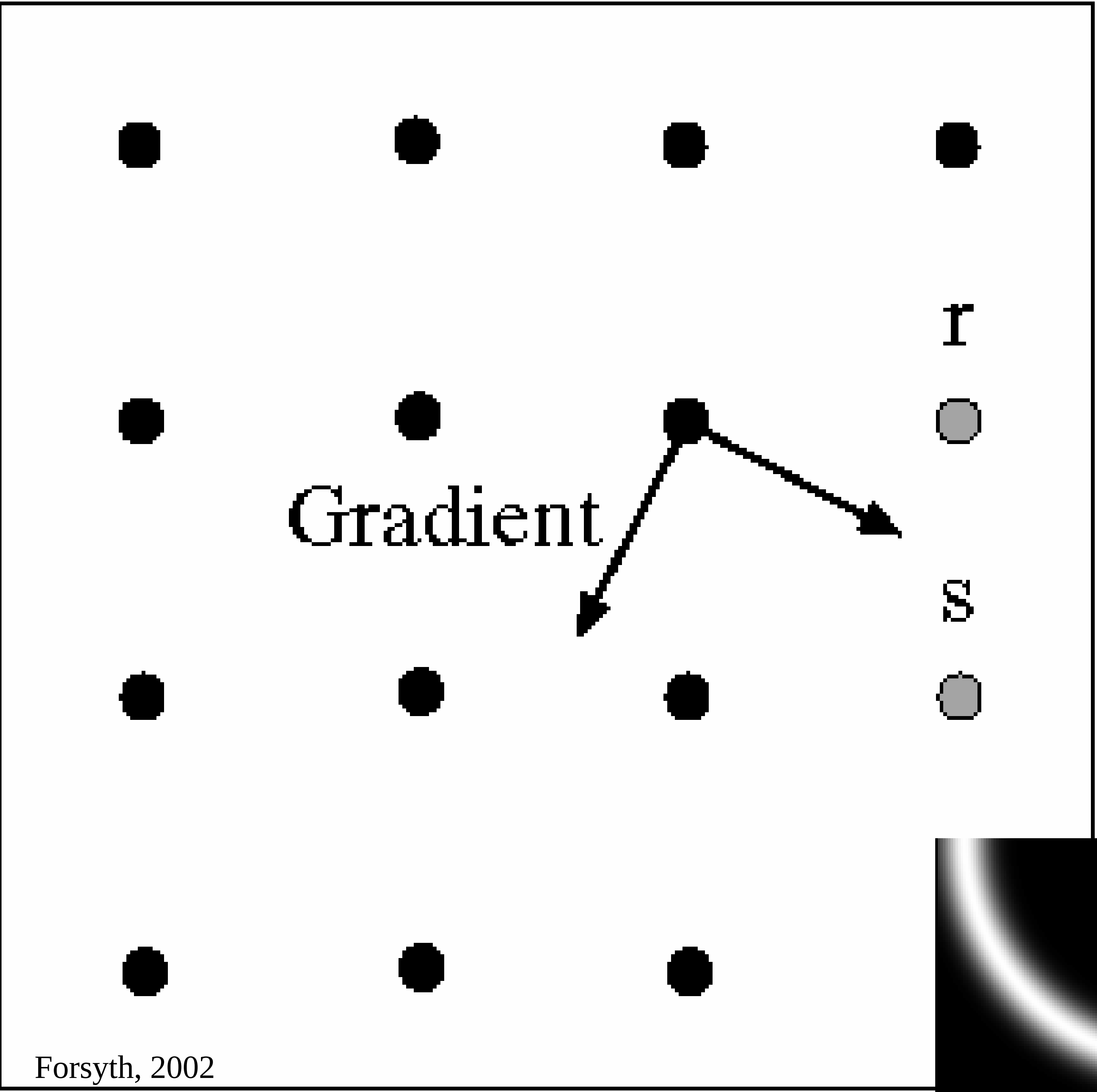


Forsyth, 2002

Non-maximum suppression

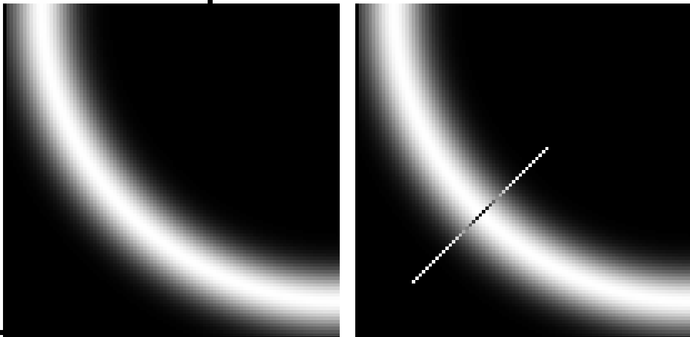
At q, we have a maximum if the value is larger than those at both p and at r. Interpolate to get these values.





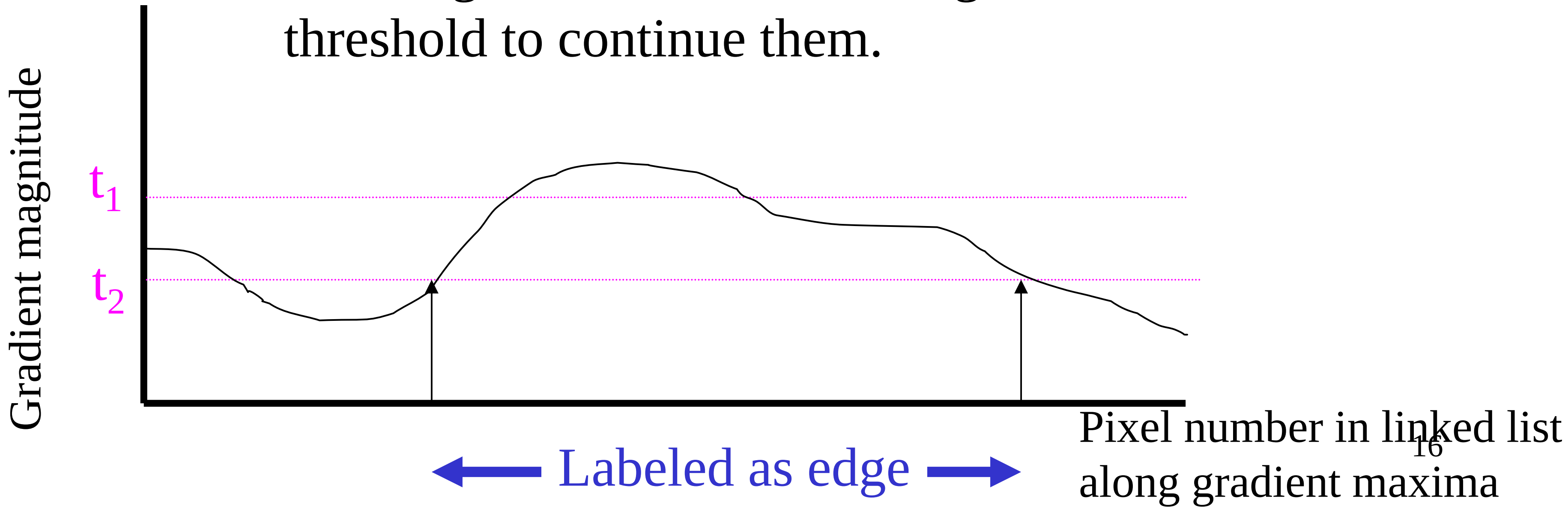
Predicting the next edge point

Assume the marked point is an edge point. Then we construct the tangent to the edge curve (which is normal to the gradient at that point) and use this to predict the next points (here either r or s).



Remaining issues

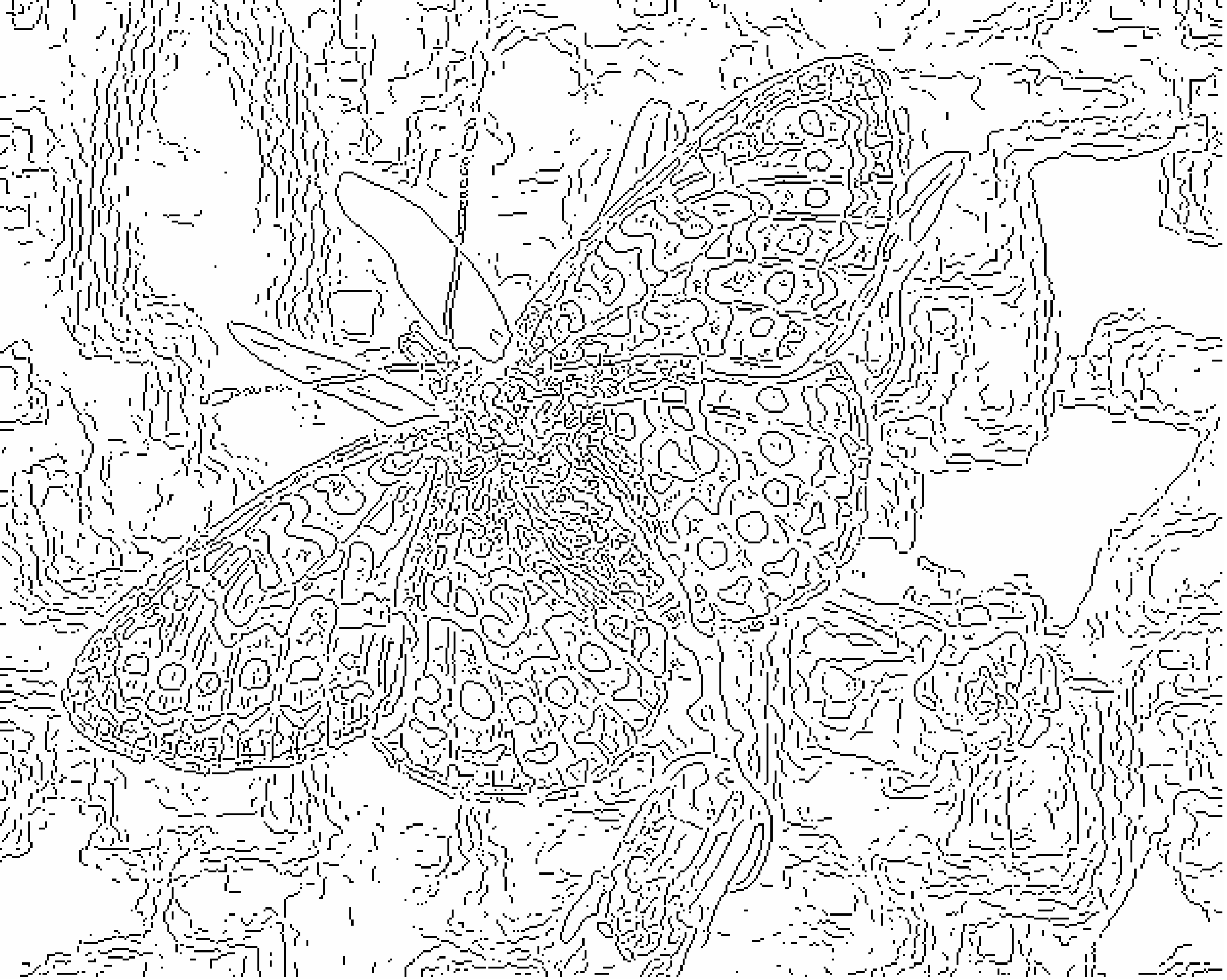
- Check that maximum value of gradient value is sufficiently large
  - drop-outs? use **hysteresis**
    - use a high threshold to start edge curves and a low threshold to continue them.



Notice

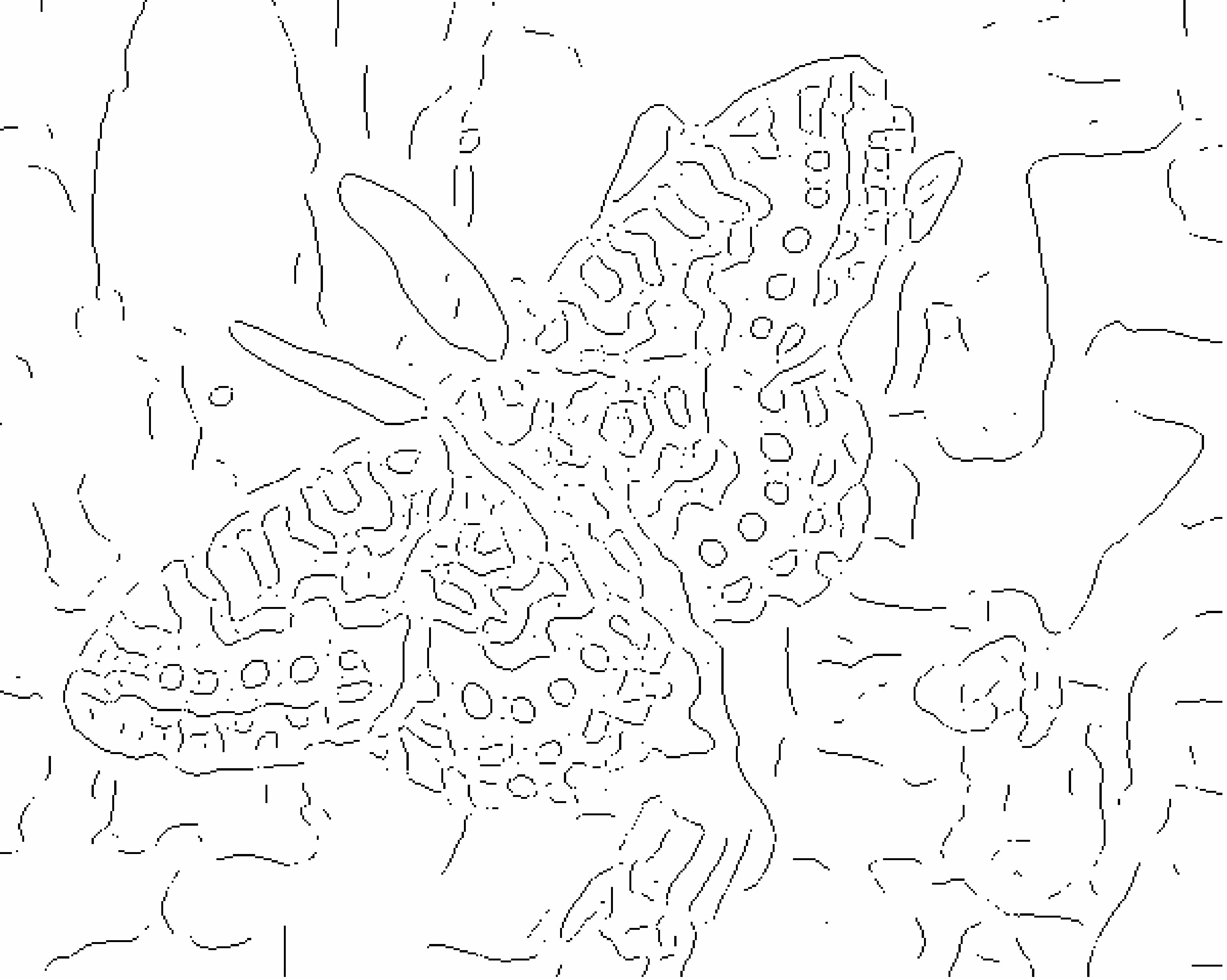
- Something nasty is happening at corners
- Scale affects contrast
- Edges aren’t bounding contours

Forsyth, 2002



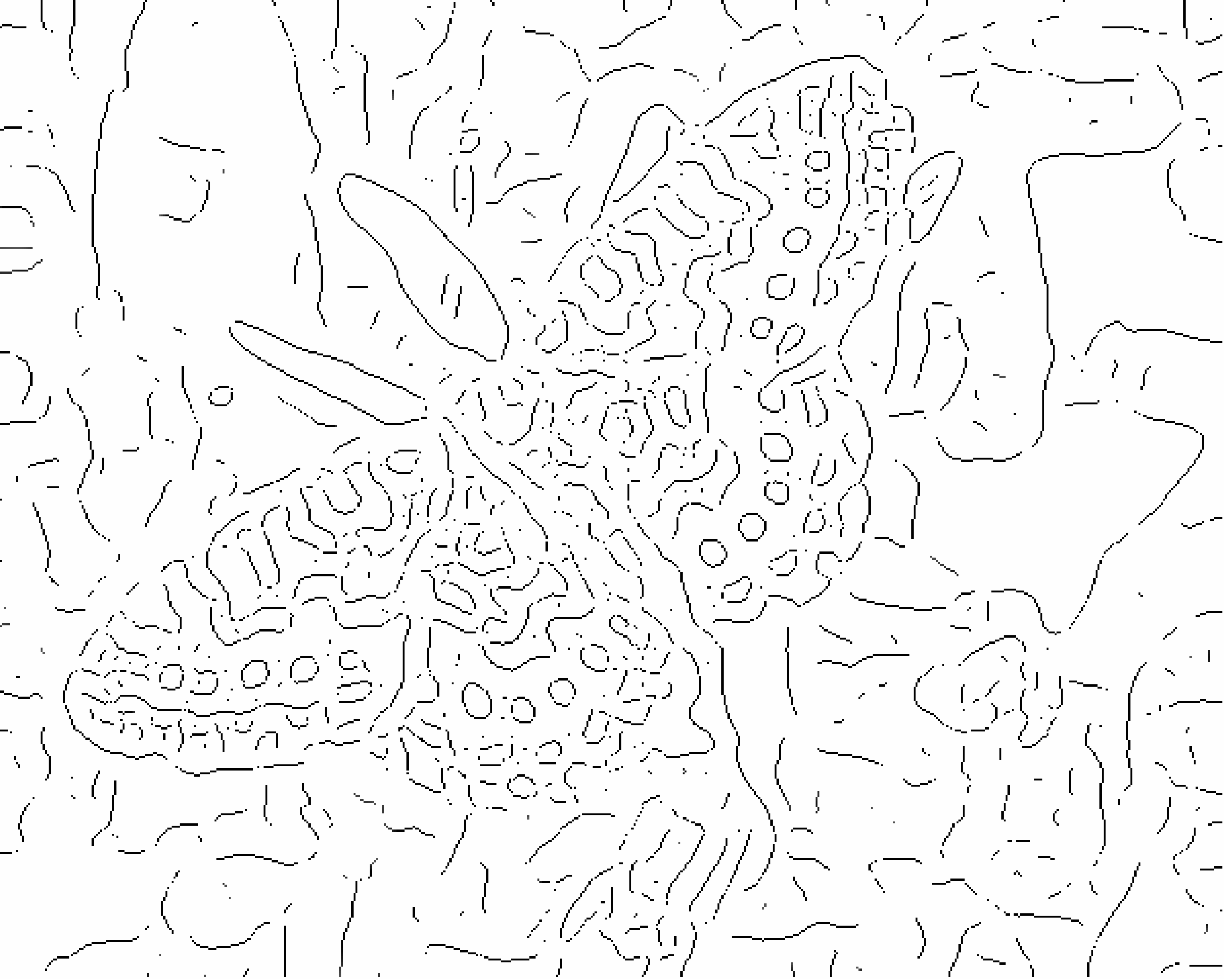
fine scale
high
threshold

Forsyth, 2002



coarse scale, high threshold

Forsyth, 2002



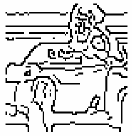
coarse
scale
low
threshold

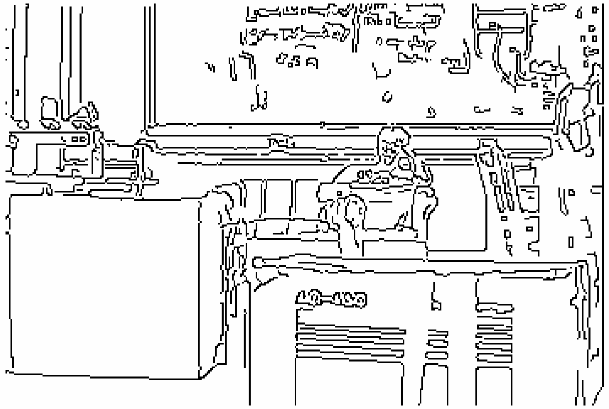
Forsyth, 2002

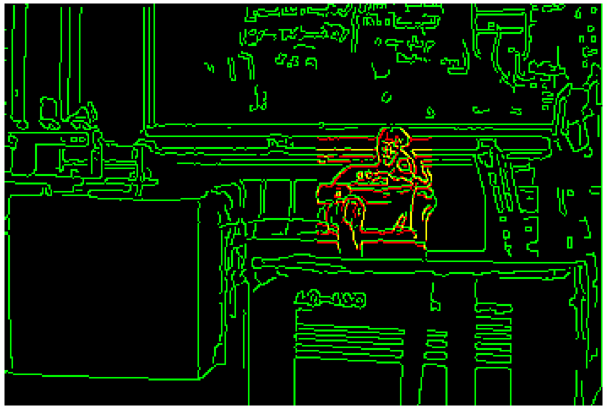
edges

- Issues:
  - On the one hand, what a useful thing: a marker for where something interesting is happening in the image.
  - On the other hand, isn’t it way to early to be thresholding, based on local, low-level pixel information alone?

Something useful with edges


is a two-dimensional bitmap that serves as a model view of an object (Kevin sitting on a couch)


is a two-dimensional bitmap (intensity edges) in which we want to locate this model, under the transformation of two-dimensional translation and scaling (that is the model is allowed to move in x and y, and also to scale separately in each of these dimensions, for a total of four transformation parameters).


shows the best transformation (translation and scaling) of the model with respect to the image, in the sense that it maximizes the fraction of model edge points that lie near image edge points (within 1 pixel diagonally). The green points are image egdes, the red points are transformed model edges, and the yellow points are locations where both an image edge and a transformed model edge are coincident. Note that there are many red locations adjacent to green ones (which would not be detected by a method such as binary correlation).

Dan Huttenlocher

http://www.cs.cornell.edu/~dph/hausdorff/hausdorff1.html

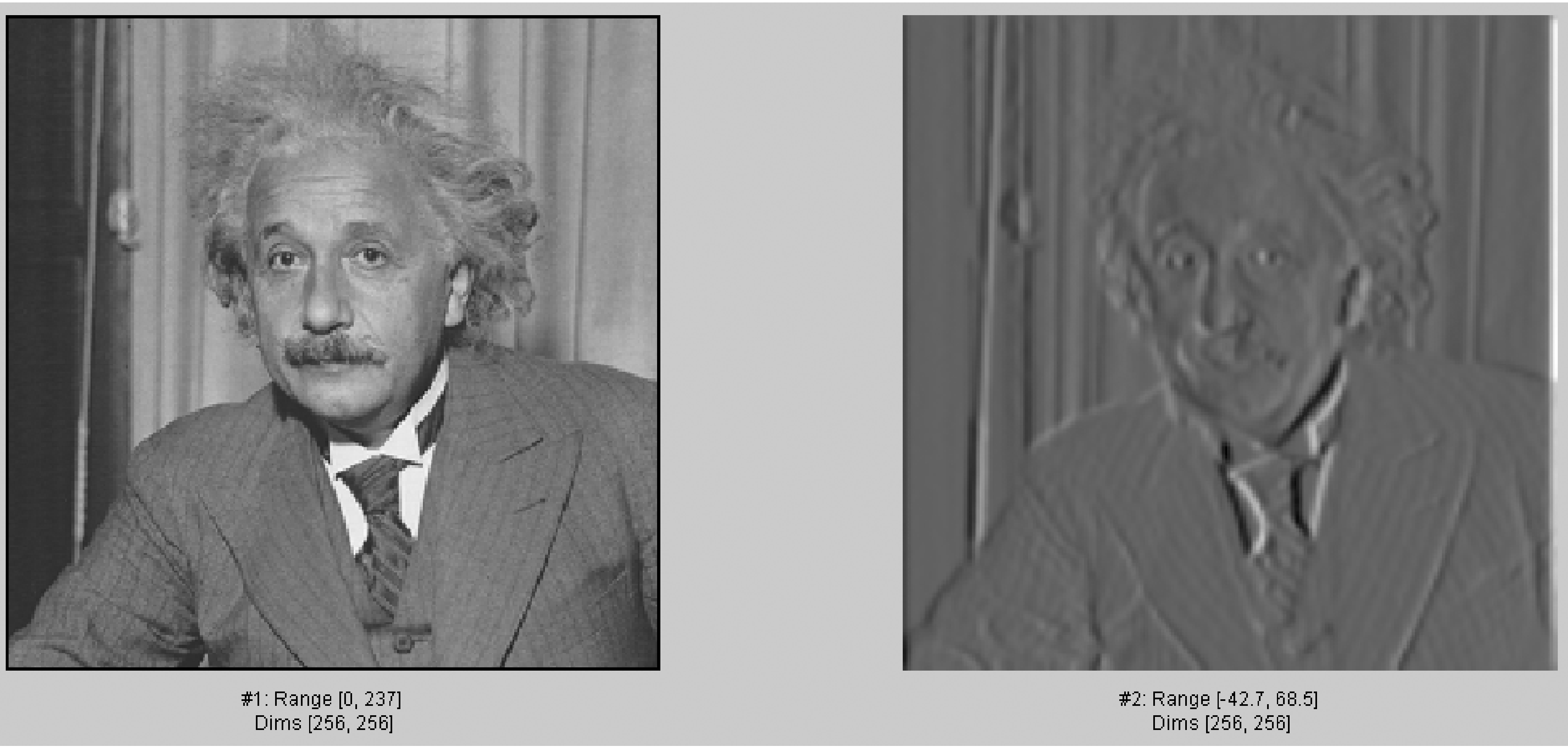
Another useful, bandpass-filter-based, non-linear operation: Contrast normalization

- Maintains more of the signal, but still does some gain control.
- Algorithm: bp = bandpassed image.

amplitude $\longrightarrow$ absval = abs(bp);

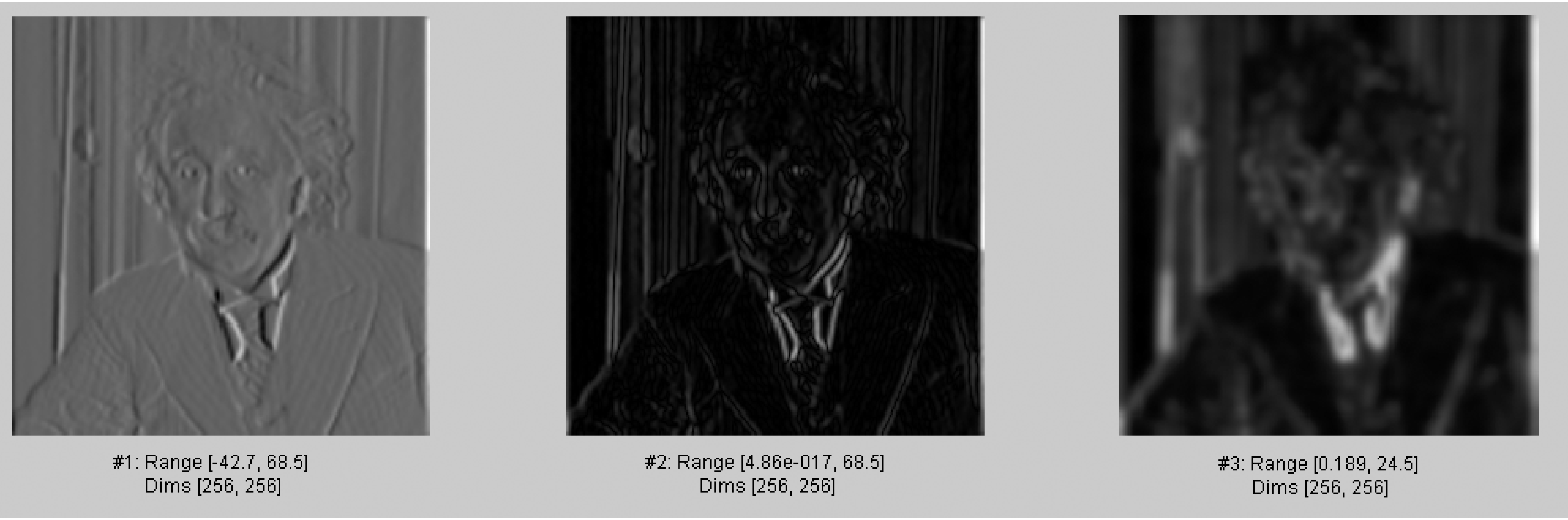
local contrast $\longrightarrow$ avgAmplitude = upBlur(blurDn(absval, 2), 2);

Contrast normalized output $\longrightarrow$ contrastNorm = bp ./ (avgAmplitude + const);



Original image

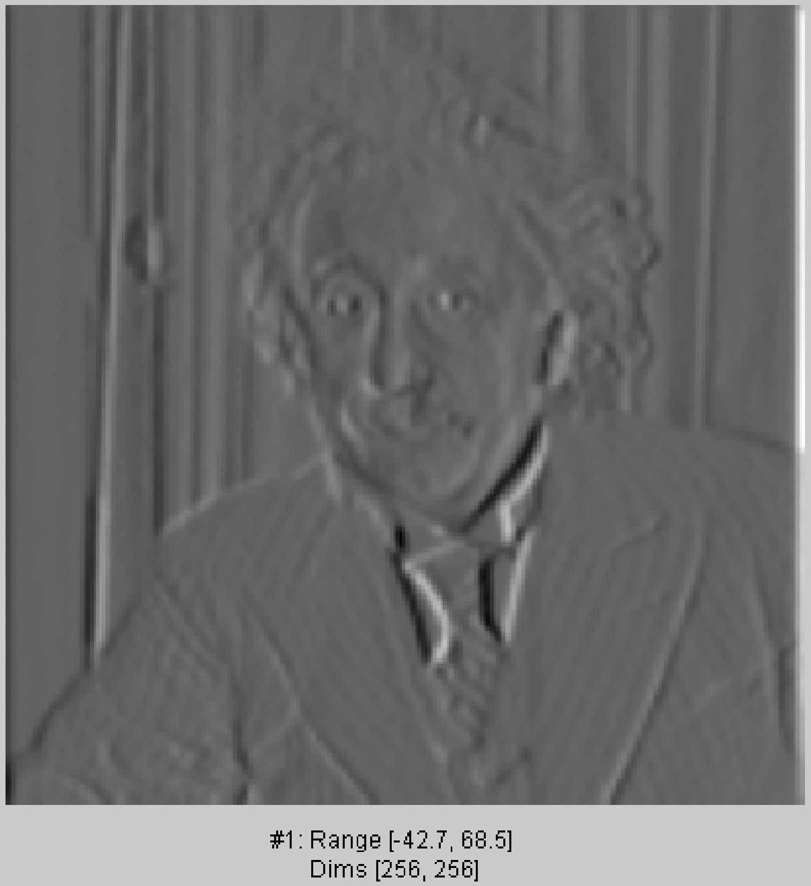
Bandpass filtered
(deriv of gaussian)



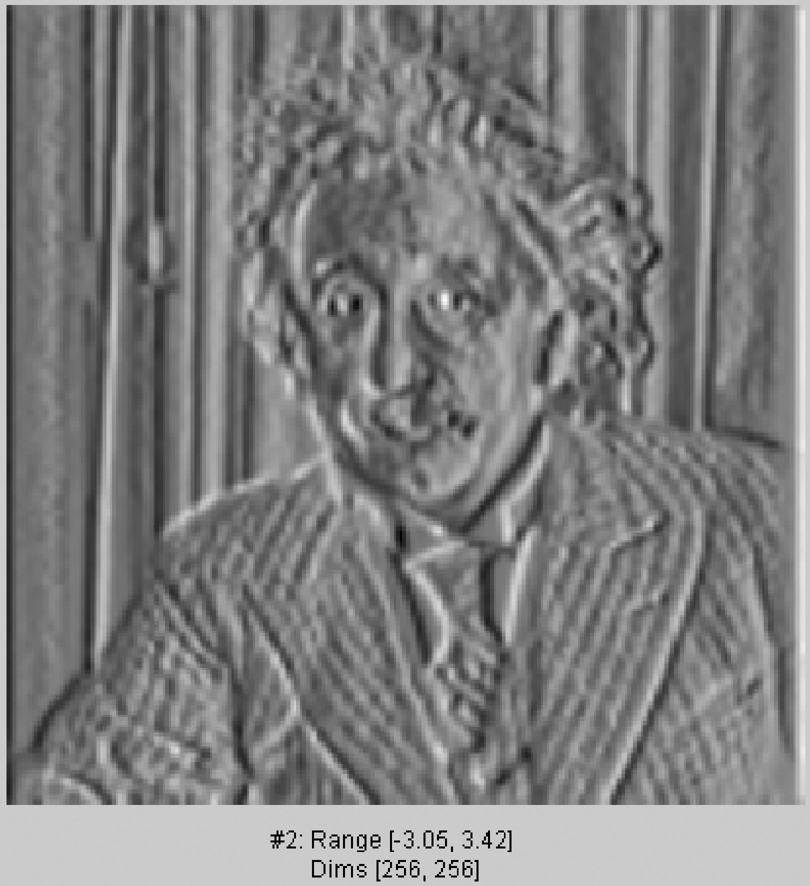
Bandpass filtered

Absolute value

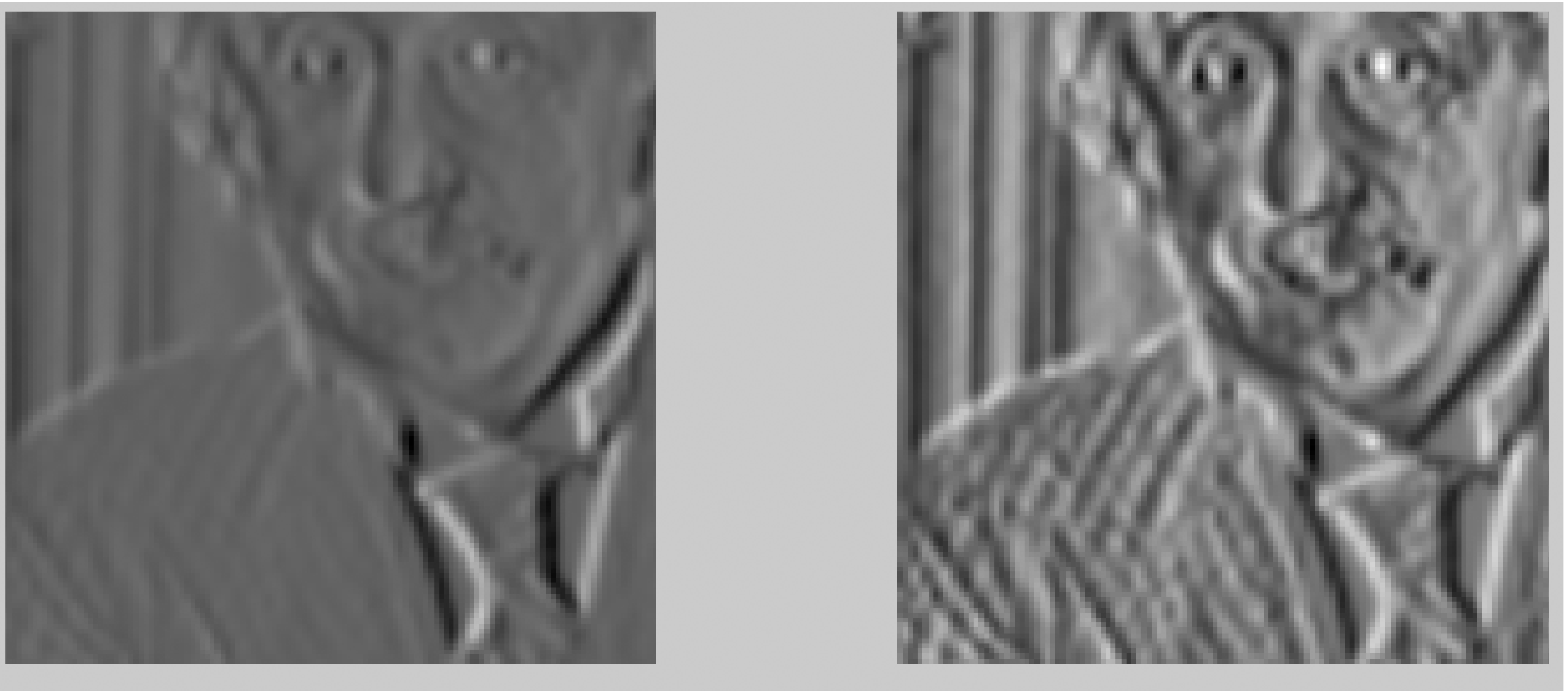
Blurred absolute value



Bandpass filtered



Bandpass filtered and contrast normalized



Bandpass filtered

Bandpass filtered and contrast normalized

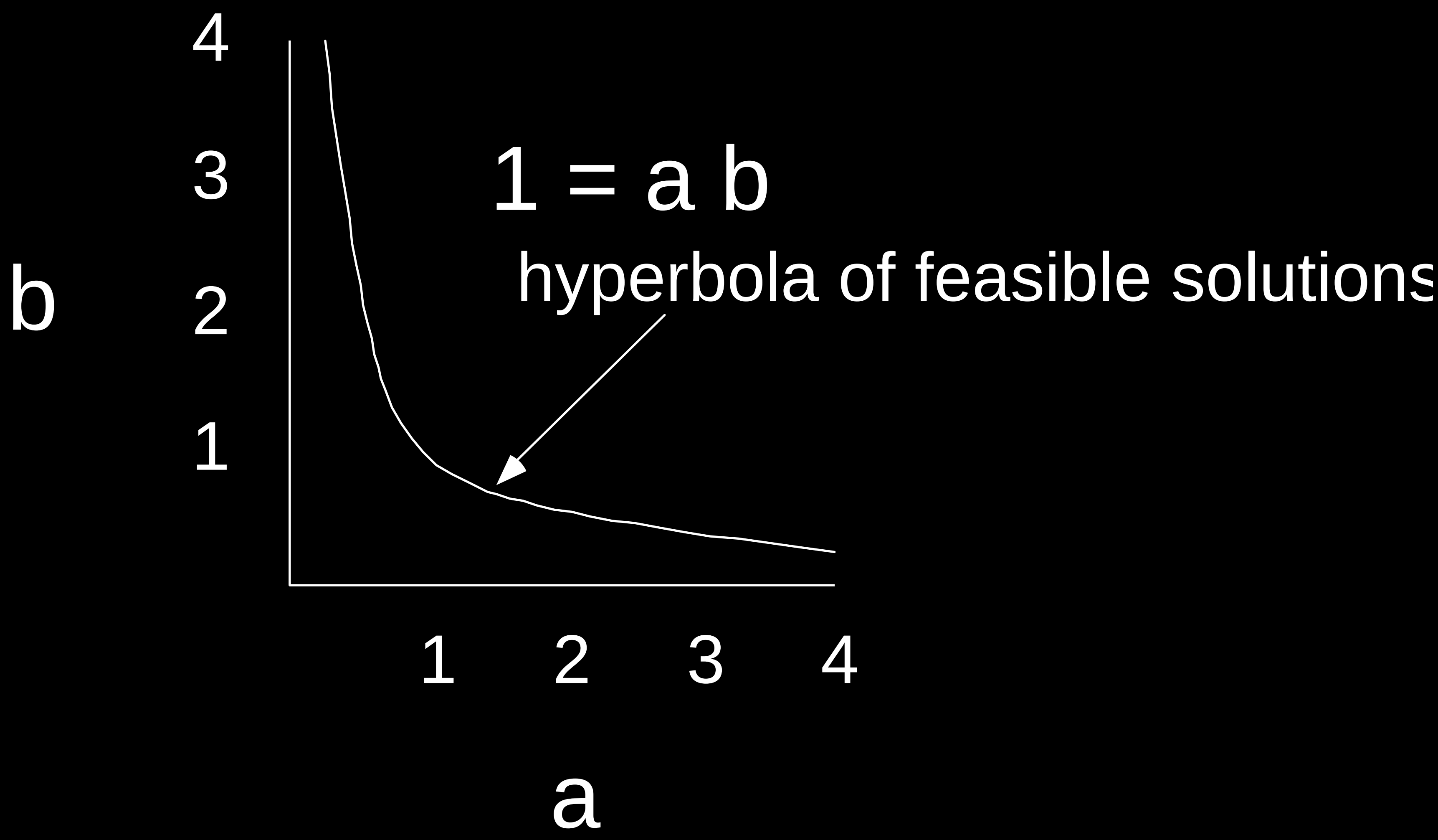
Bayesian methods

See Bishop handout, chapter 1 from “Neural Networks for Pattern Recognition”, Oxford University Press.

Simple, prototypical vision problem

- Observe some product of two numbers, say 1.0.
- What were those two numbers?
- Ie, 1 = ab. Find a and b.

- Cf, simple prototypical graphics problem: here are two numbers; what’s their product?


4
3
2
1
b
1 = a b
hyperbola of feasible solutions
1
2
3
4
a

Bayes rule

$$P(x|y) = P(y|x)\,P(x) / P(y)$$

Bayesian approach

- Want to calculate P(a, b | y = 1).
- Use P(a, b | y = 1) = k P(y=1|a, b) P(a, b).

Likelihood function

Prior probability

Posterior probability

Likelihood function, P(obs|parms)

- The forward model, or rendering model, taking into account observation noise.
- Example: assume Gaussian observation noise. Then for this problem:

$$P(y=1|a,b)=\frac{1}{\sqrt{2\pi\sigma}}e^{-\frac{(1-ab)^2}{2\sigma^2}}$$

A common criticism of Bayesian methods

- “You need to make all those assumptions about prior probabilities”.
- Response…?
- “Everyone makes assumptions. Bayesians put their assumptions out in the open, clearly stated, where they belong.”

Prior probability

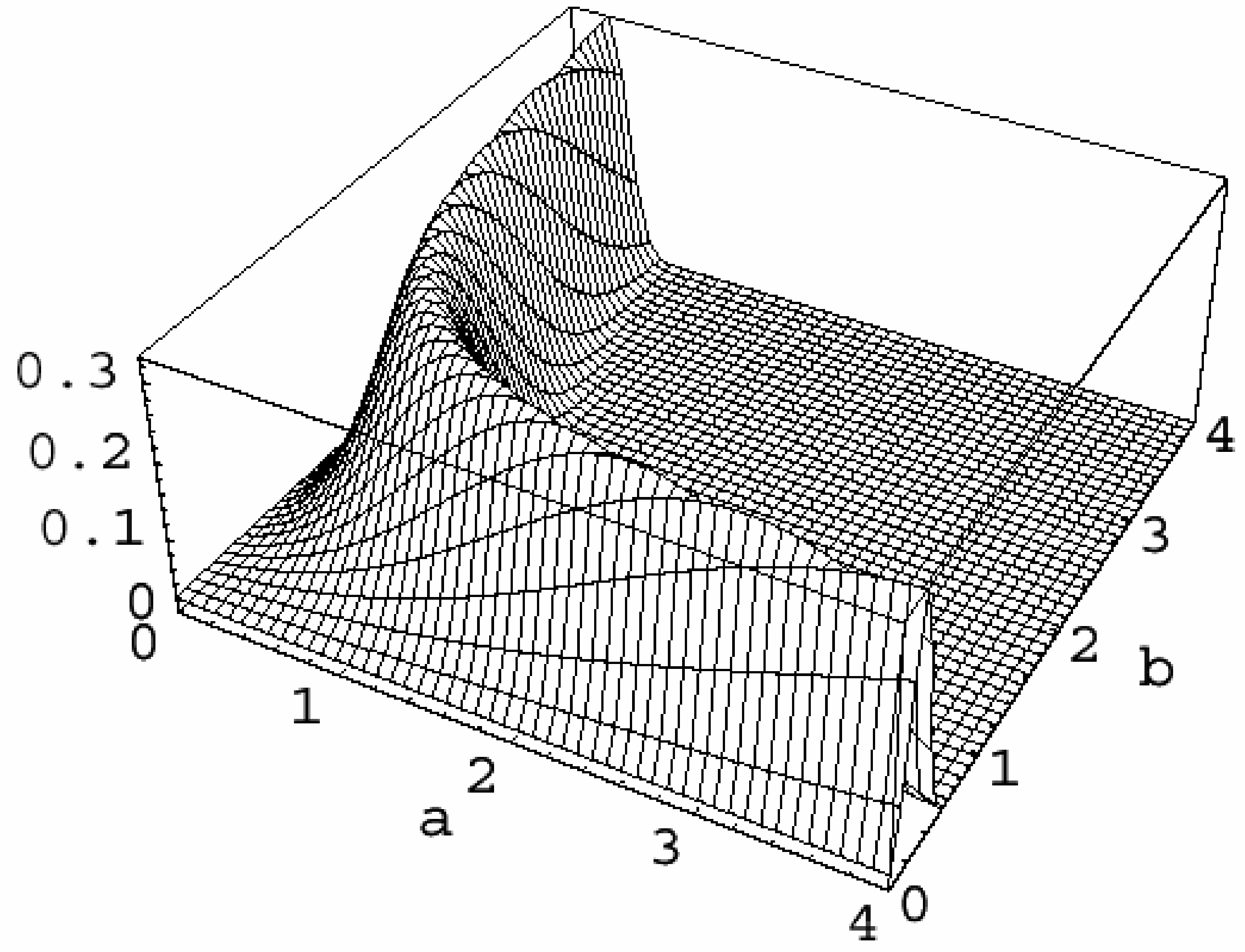
In this case, we'll assume $P(a,b)=P(a)P(b)$, and $P(a) = P(b) = \text{const.}$, $0<a<4$.

Posterior probability

Posterior = k likelihood prior

$$P(a,b \mid y=1) = ke^{-\frac{(1-ab)^2}{2\sigma^2}}$$

for 0 < a,b<4,
0 elsewhere

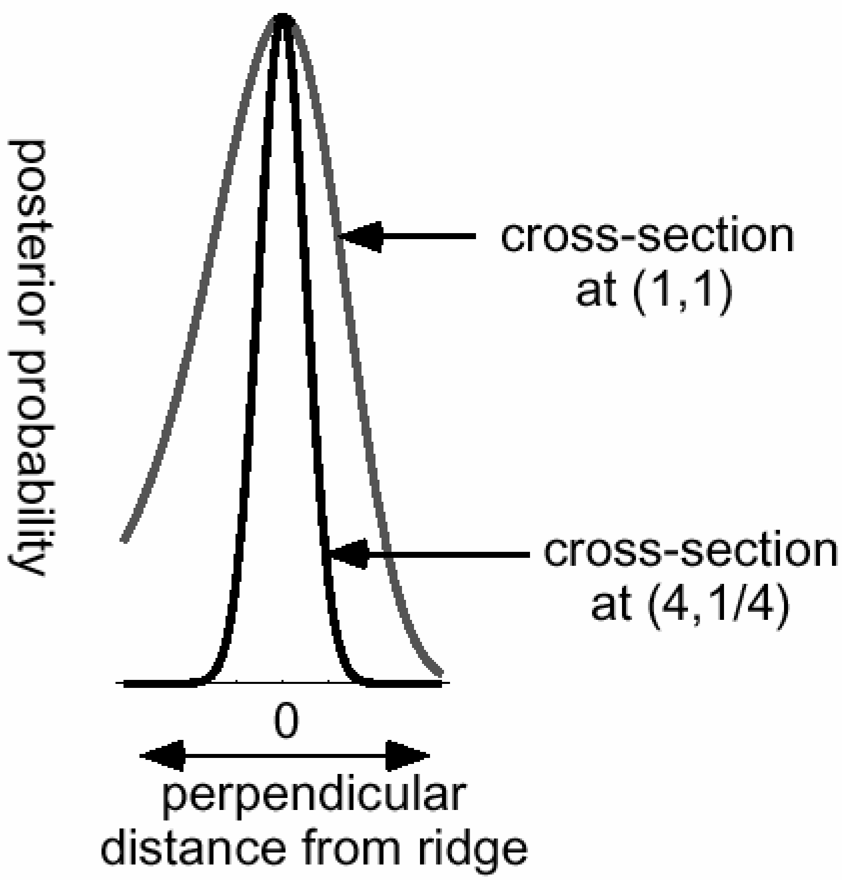


D. H. Brainard and W. T. Freeman, *Bayesian Color Constancy*, Journal of the Optical Society of America, A, 14(7), pp. 1393-1411, July, 1997

(a) Posterior Probability

For that posterior probability, what is the best pair of numbers, (a,b), to pick, given your observation ab = 1?

Loss functions



(b) Ridge Thickness Variations

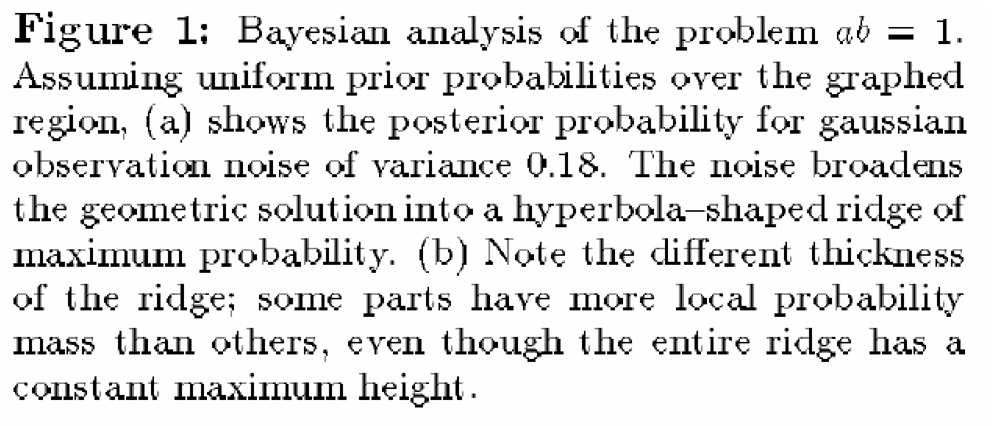
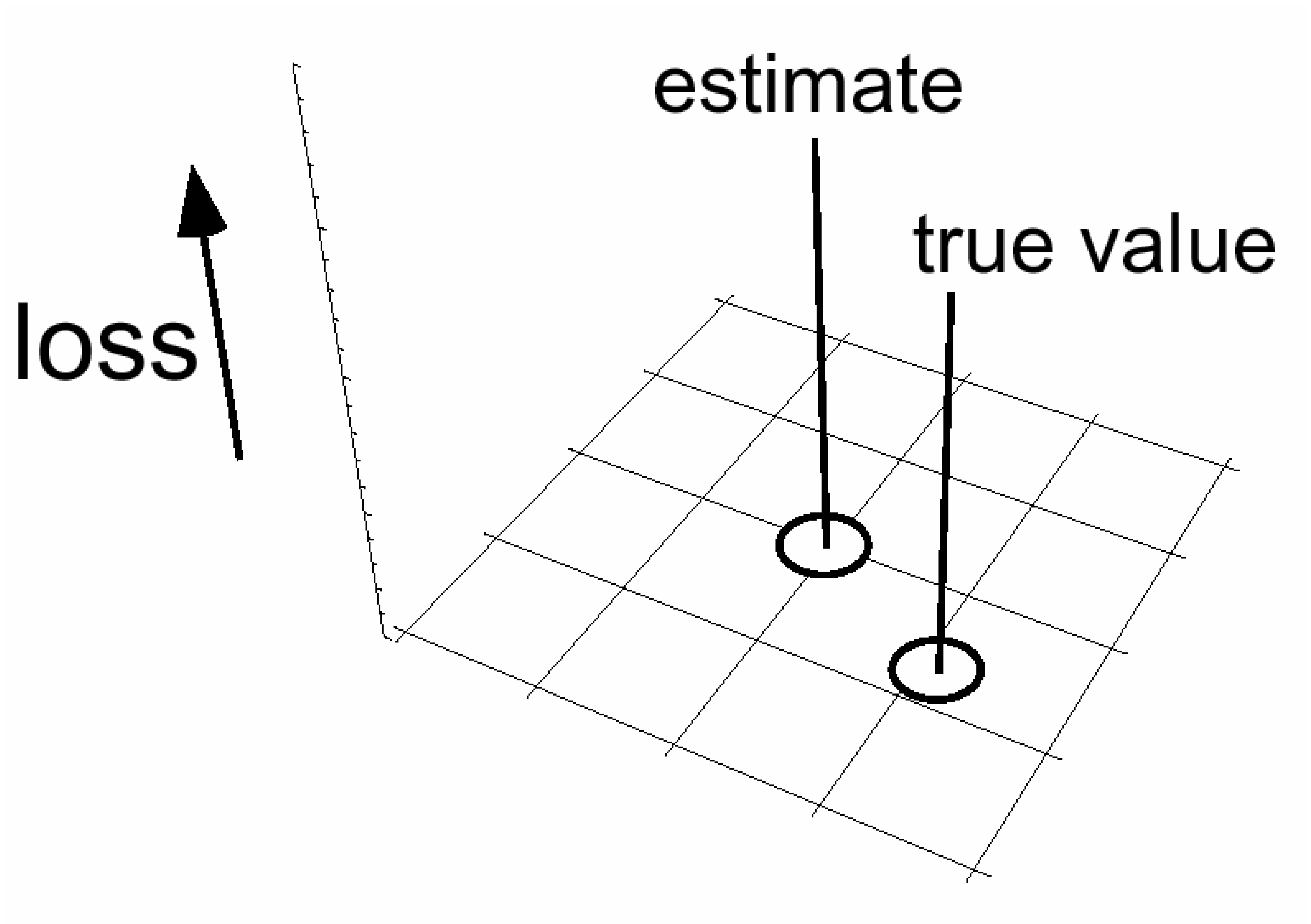
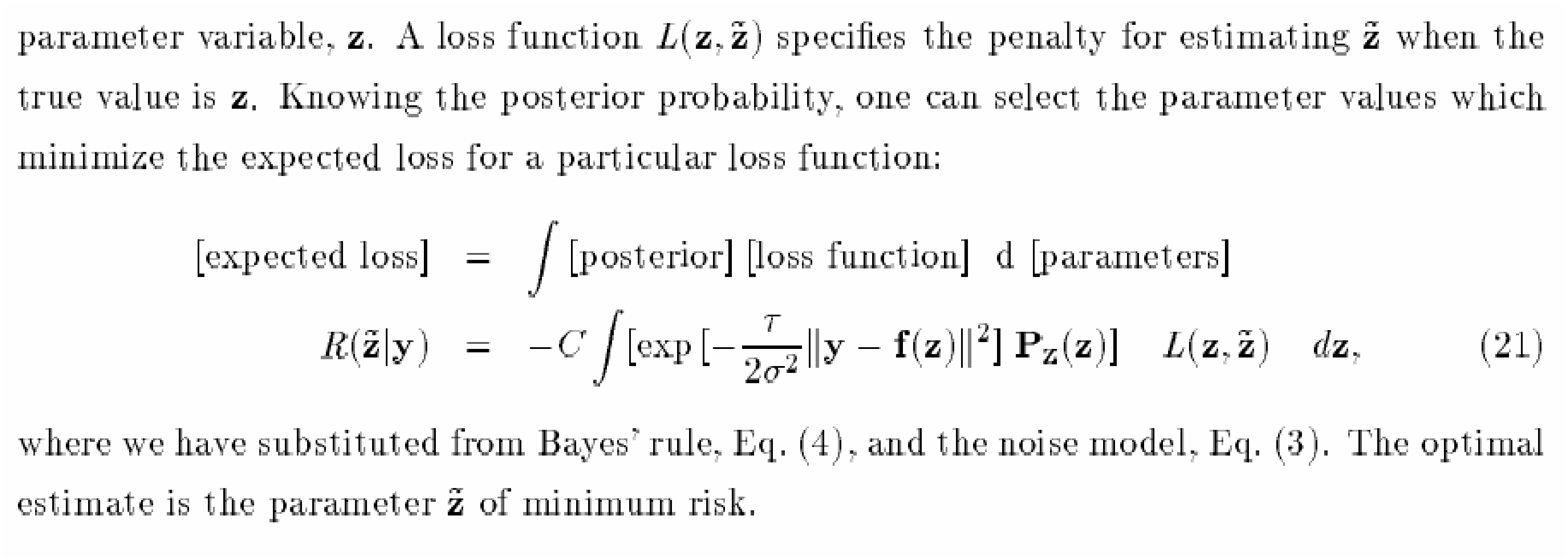


Figure 1: Bayesian analysis of the problem $ab = 1$. Assuming uniform prior probabilities over the graphed region, (a) shows the posterior probability for gaussian observation noise of variance 0.18. The noise broadens the geometric solution into a hyperbola–shaped ridge of maximum probability. (b) Note the different thickness of the ridge; some parts have more local probability mass than others, even though the entire ridge has a constant maximum height.

D. H. Brainard and W. T. Freeman, *Bayesian Color Constancy*, Journal of the Optical Society of America, A, 14(7), pp. 1393-1411, July, 1997


estimate
true value
loss

Bayesian decision theory



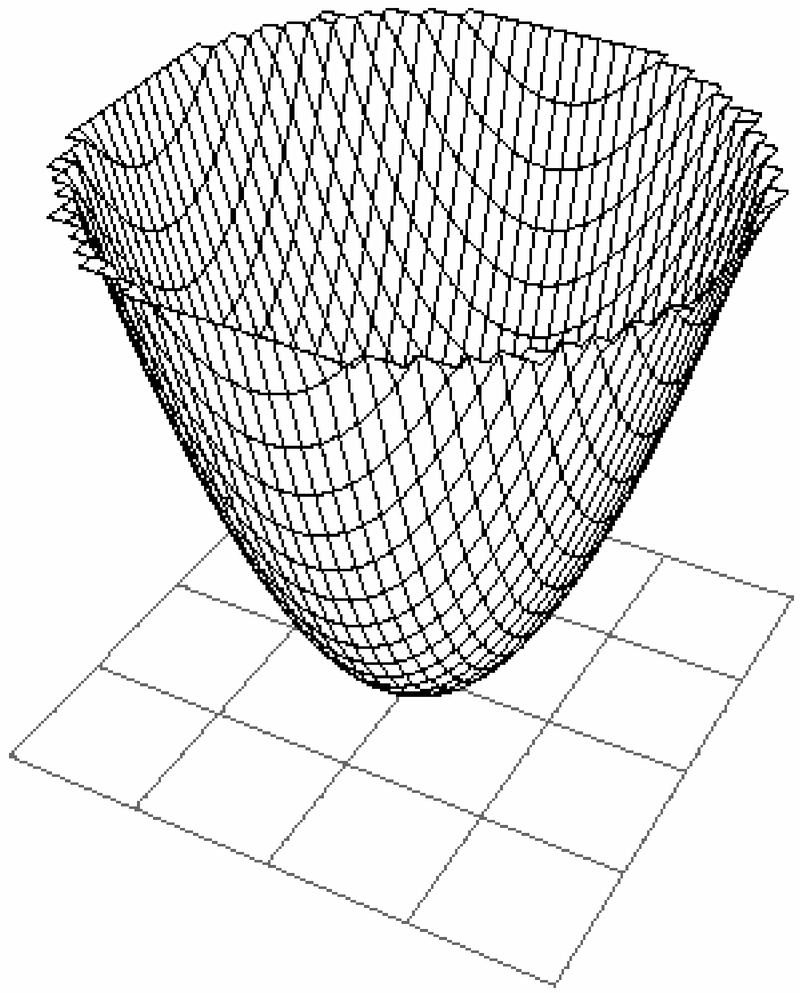
parameter variable, $\mathbf{z}$. A loss function $L(\mathbf{z}, \tilde{\mathbf{z}})$ specifies the penalty for estimating $\tilde{\mathbf{z}}$ when the true value is $\mathbf{z}$. Knowing the posterior probability, one can select the parameter values which minimize the expected loss for a particular loss function:

$$\begin{aligned}
[\text{expected loss}] &= \int [\text{posterior}]\,[\text{loss function}]\;\; \mathrm{d}\;[\text{parameters}] \\
R(\tilde{\mathbf{z}}|\mathbf{y}) &= -C \int [\exp[-\frac{T}{2\sigma^2} \|\mathbf{y} - \mathbf{f}(\mathbf{z})\|^2]\, \mathbf{P}_{\mathbf{z}}(\mathbf{z})] \quad L(\mathbf{z}, \tilde{\mathbf{z}}) \quad d\mathbf{z}, \qquad (21)
\end{aligned}$$

where we have substituted from Bayes' rule, Eq. (4), and the noise model, Eq. (3). The optimal estimate is the parameter $\tilde{\mathbf{z}}$ of minimum risk.

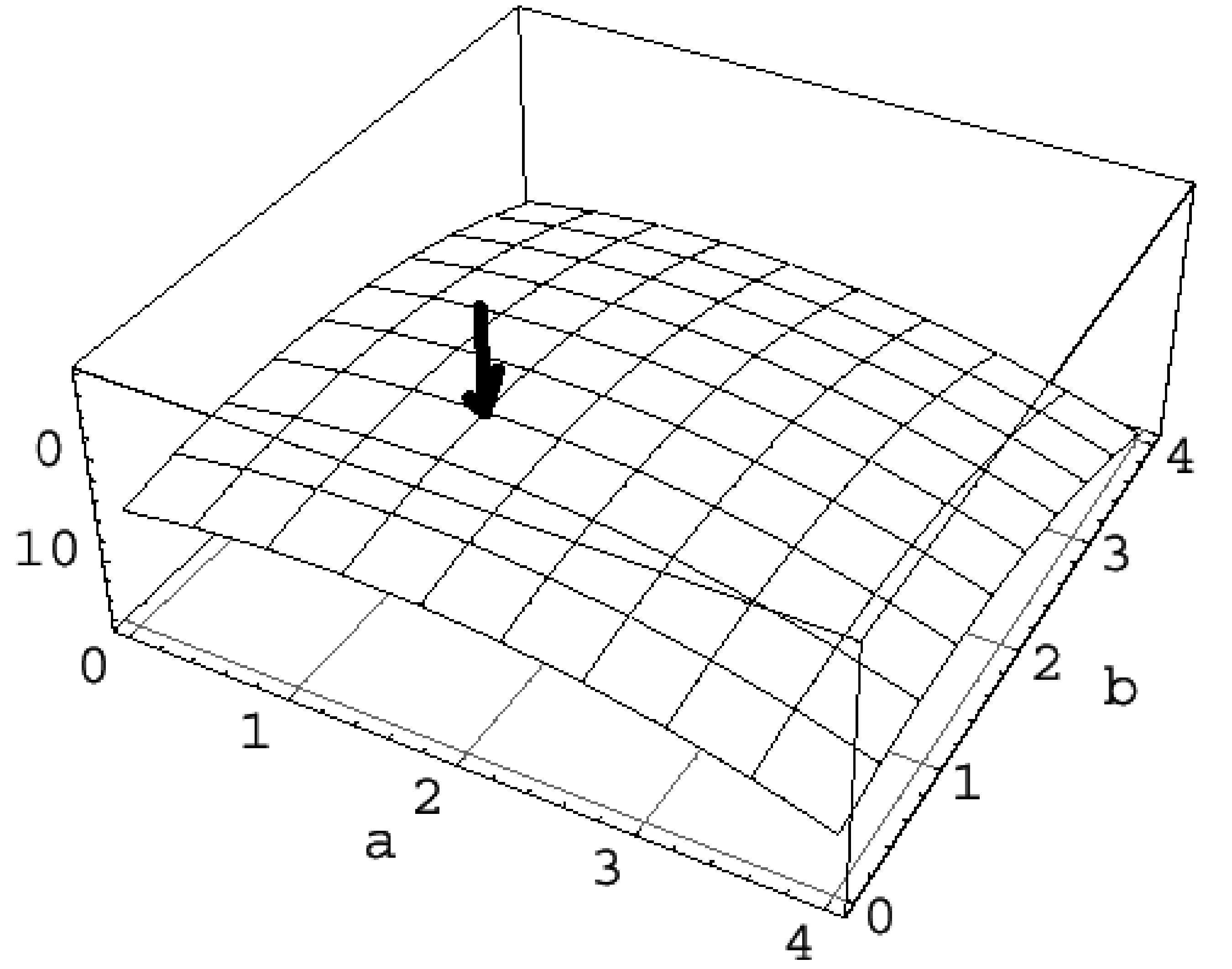
Convolve loss function with posterior

Typically, $L(z, \tilde{z}) = L(z-\tilde{z})$, and the integral for the expected loss becomes a convolution of the posterior probability with the loss function.



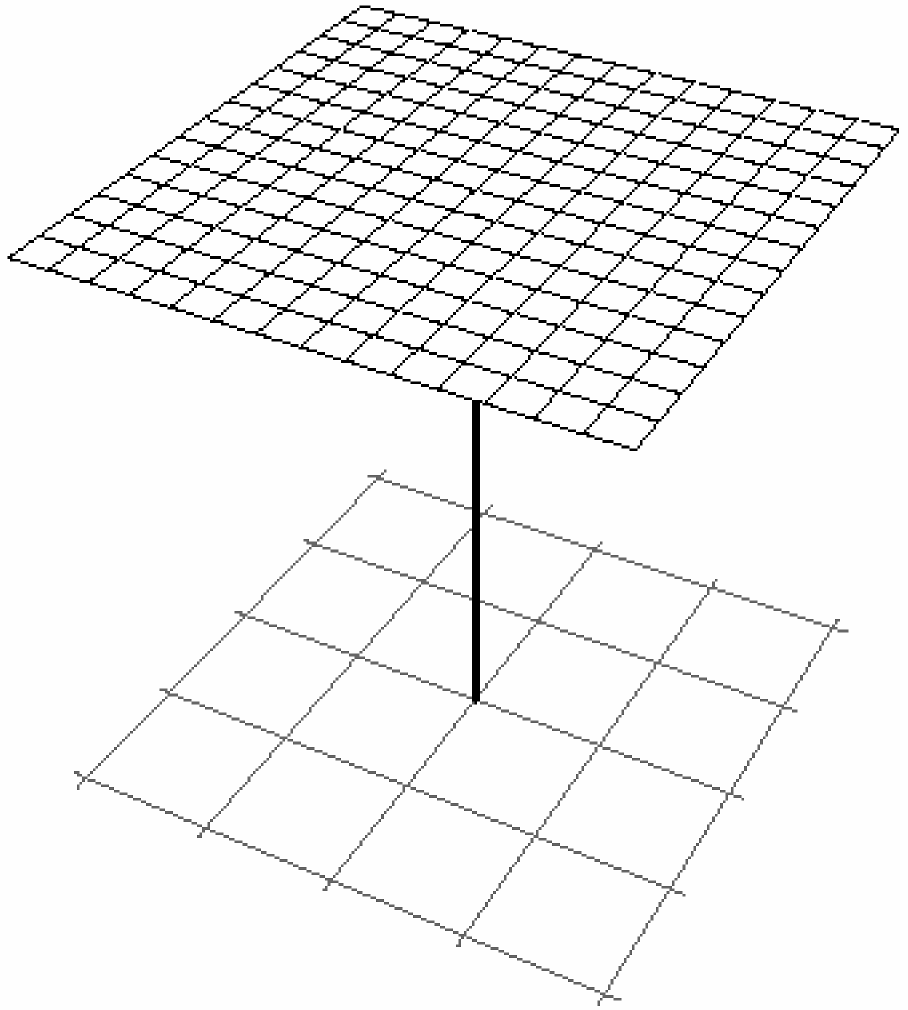
(b) MMSE loss fn.

D. H. Brainard and W. T. Freeman, *Bayesian Color Constancy*, Journal of the Optical Society of America, A, 14(7), pp. 1393-1411, July, 1997



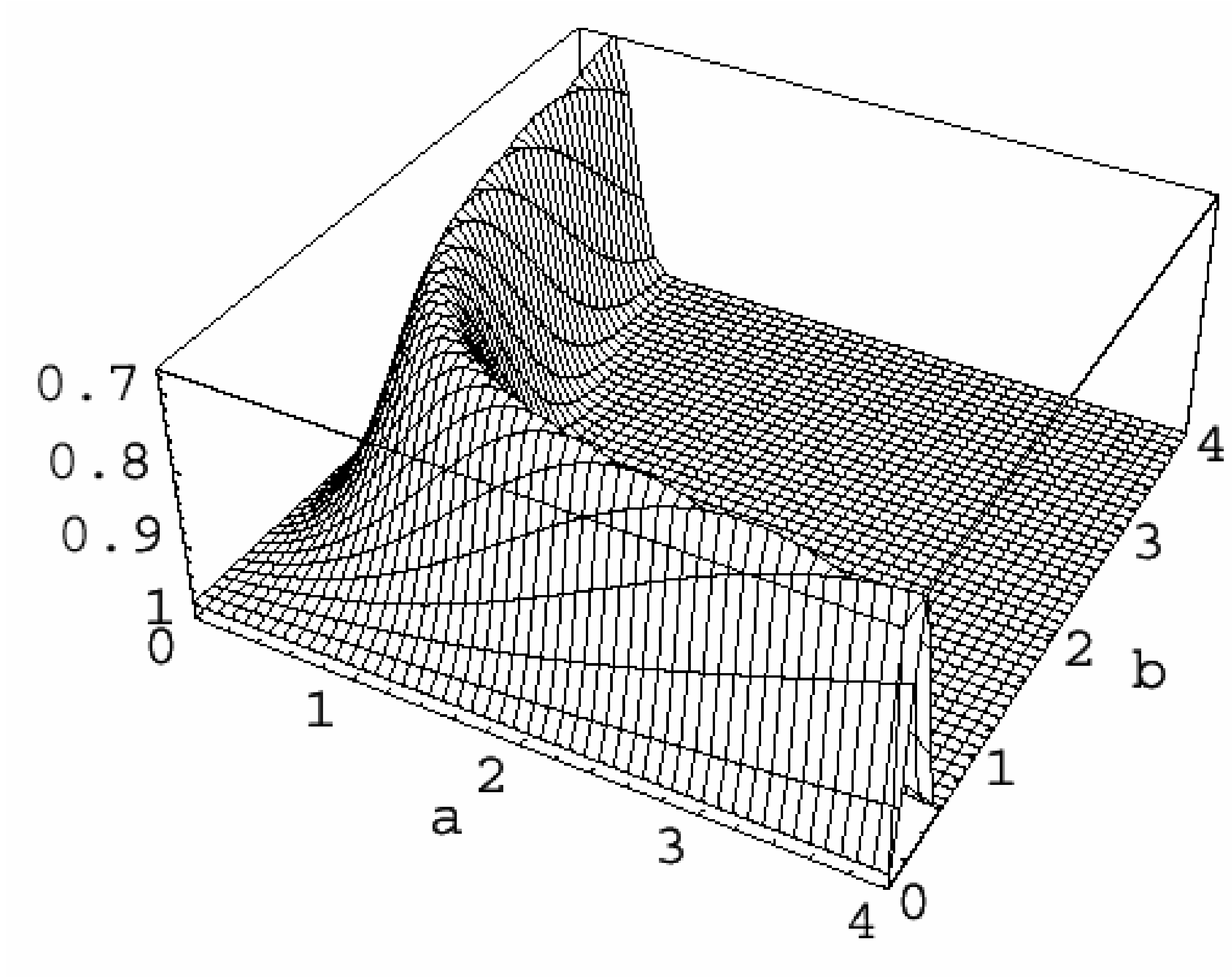
D. H. Brainard and W. T. Freeman, *Bayesian Color Constancy*, Journal of the Optical Society of America, A, 14(7), pp. 1393-1411, July, 1997

(e) (minus) MMSE risk



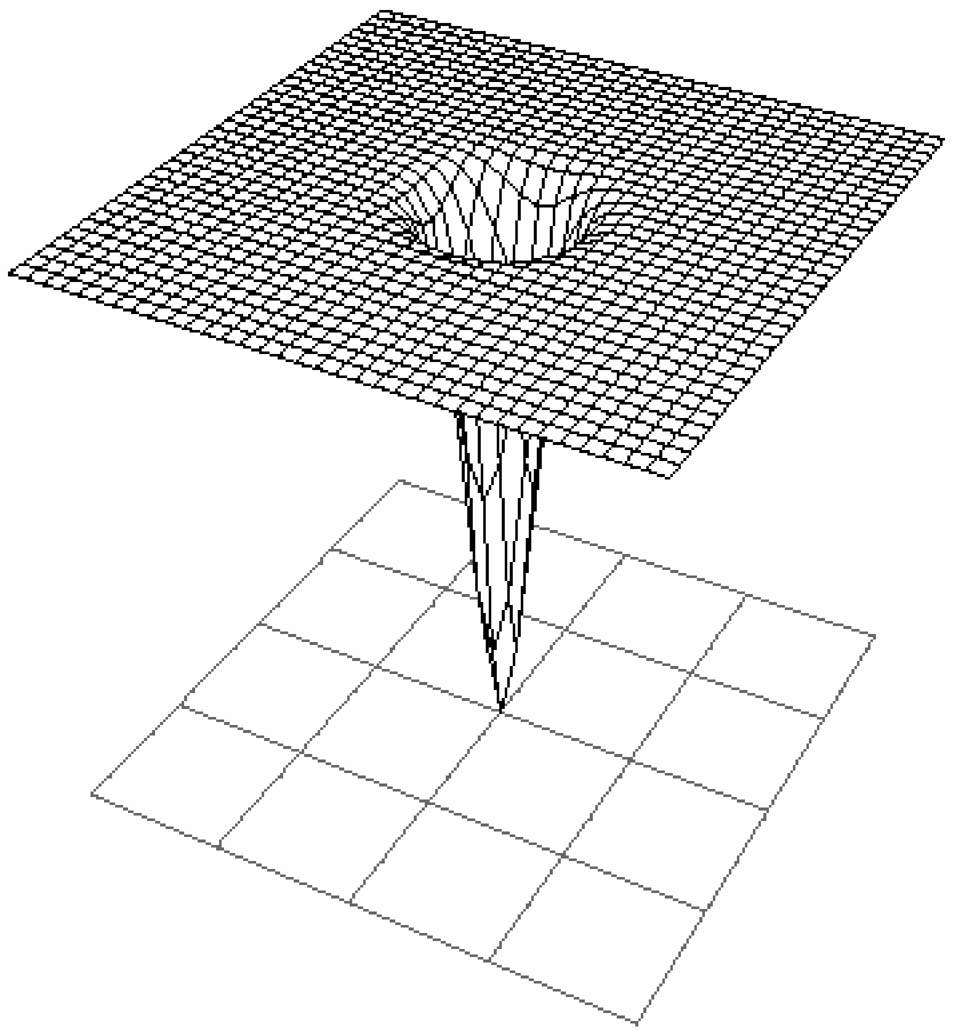
(a) MAP loss fn.

D. H. Brainard and W. T. Freeman, *Bayesian Color Constancy*, Journal of the Optical Society of America, A, 14(7), pp. 1393-1411, July, 1997



D. H. Brainard and W. T. Freeman, *Bayesian Color Constancy*, Journal of the Optical Society of America, A, 14(7), pp. 1393-1411, July, 1997

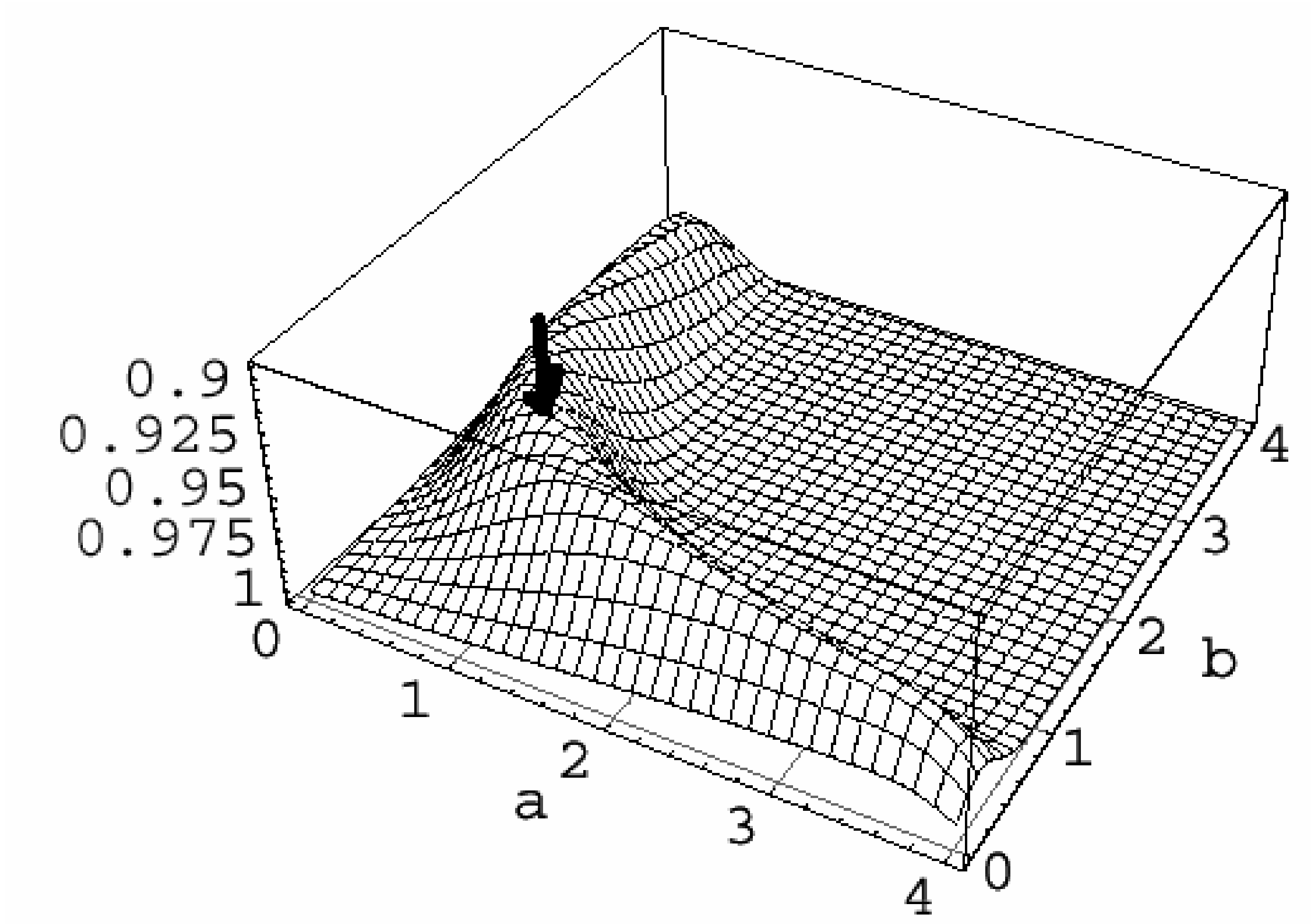
(d) (minus) MAP risk



(c) MLM loss fn.

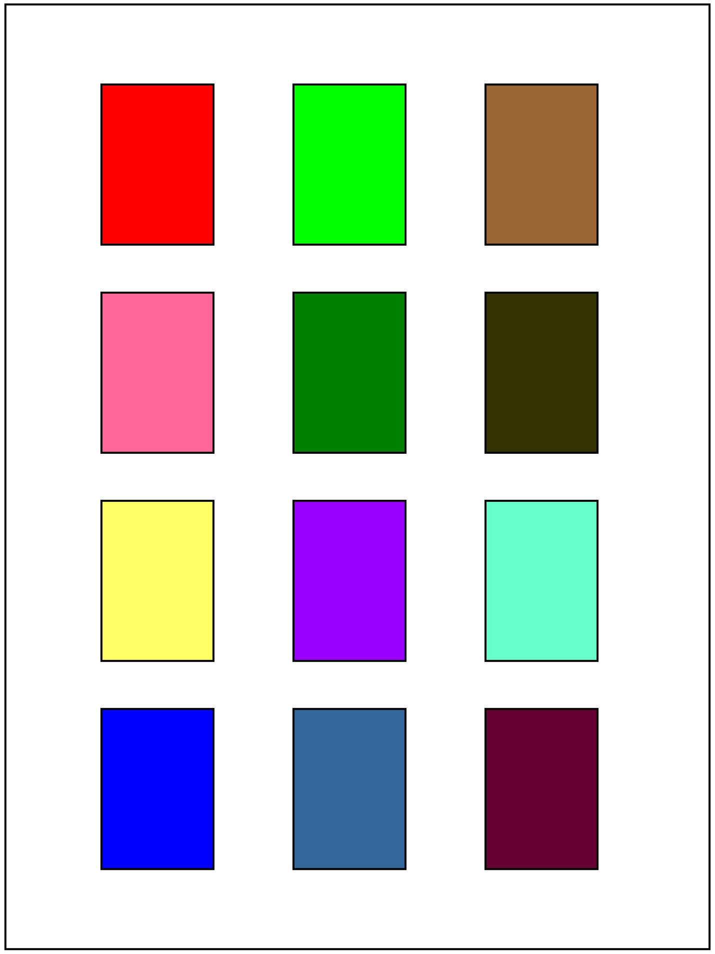
Local mass loss function may be useful model for perceptual tasks

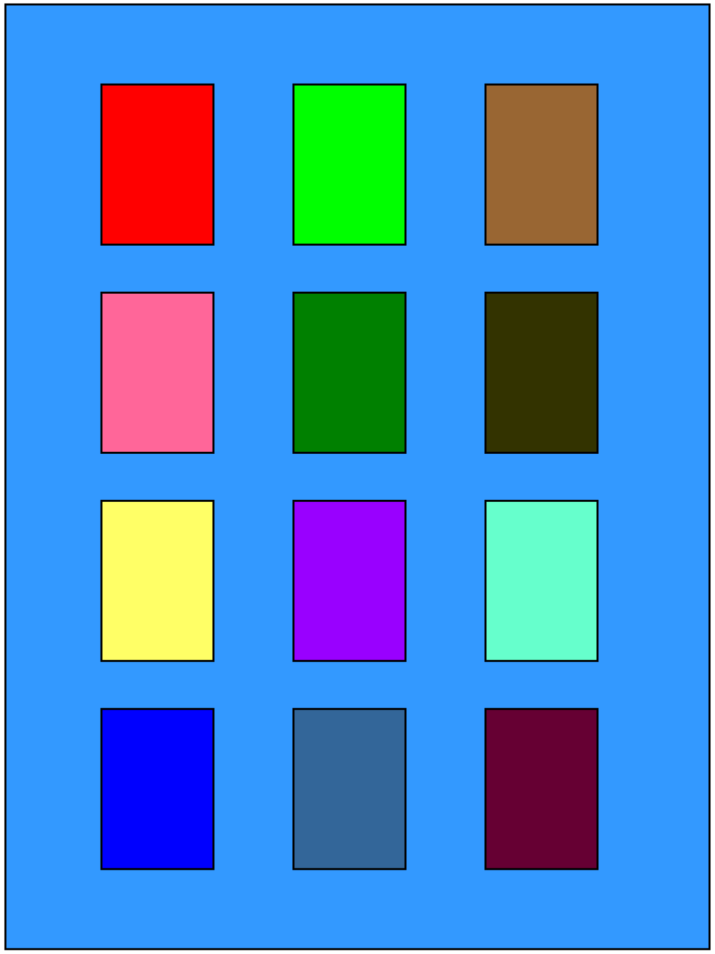
http://sportsillustrated.cnn.com/baseball/college/2000/college_world_series/news/2000/06/15/cws_notebook_ap/t1_borchard_ap_01.jpg

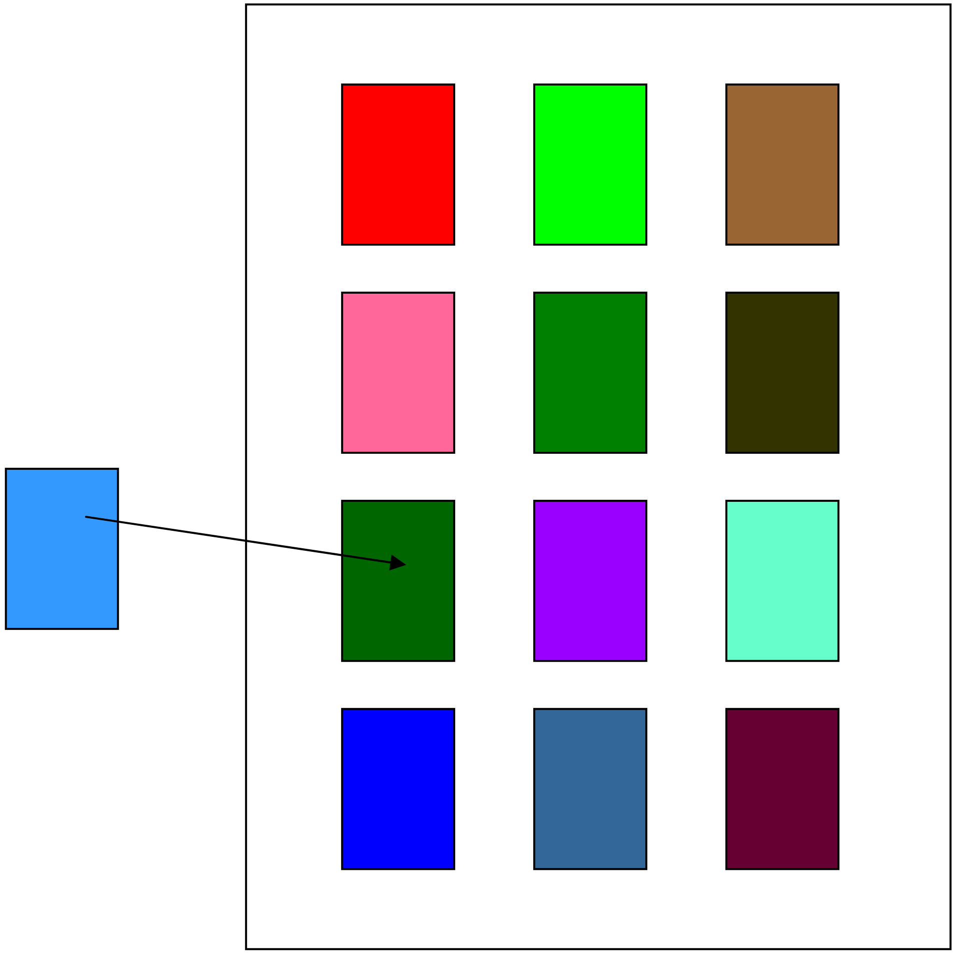


(f) (minus) MLM risk

Reminder of color constancy demo







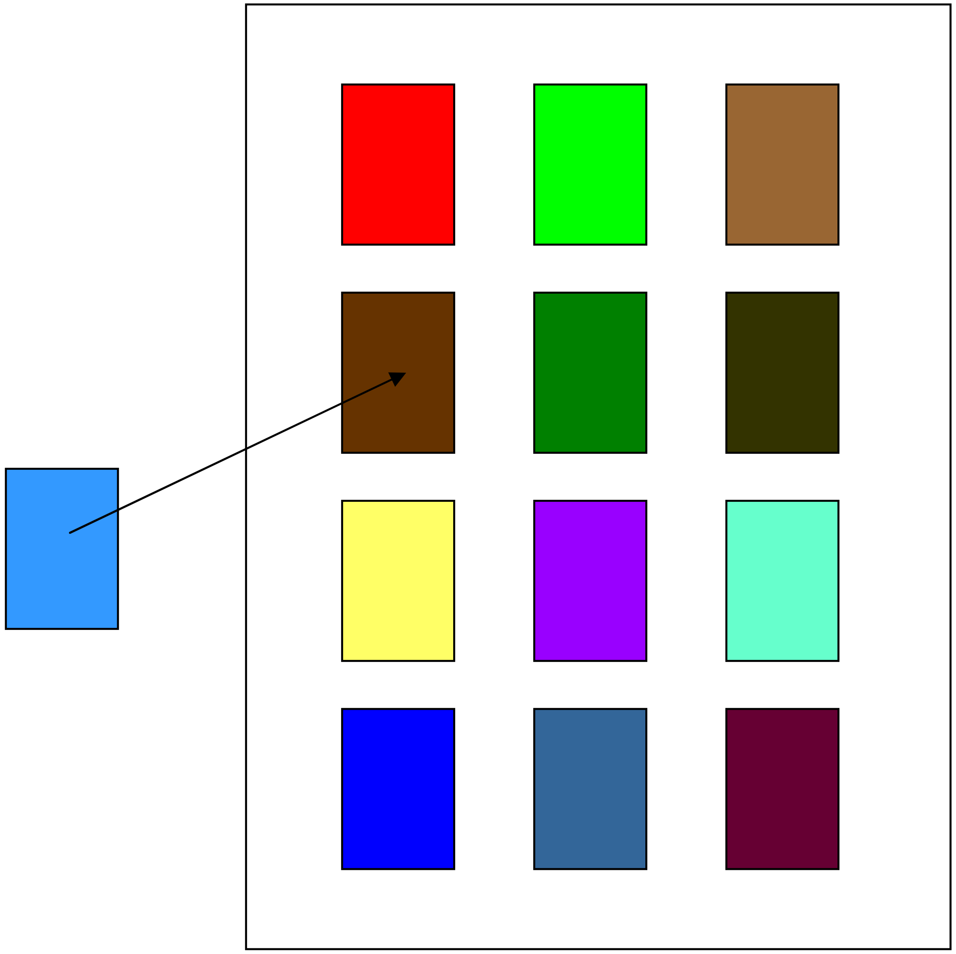
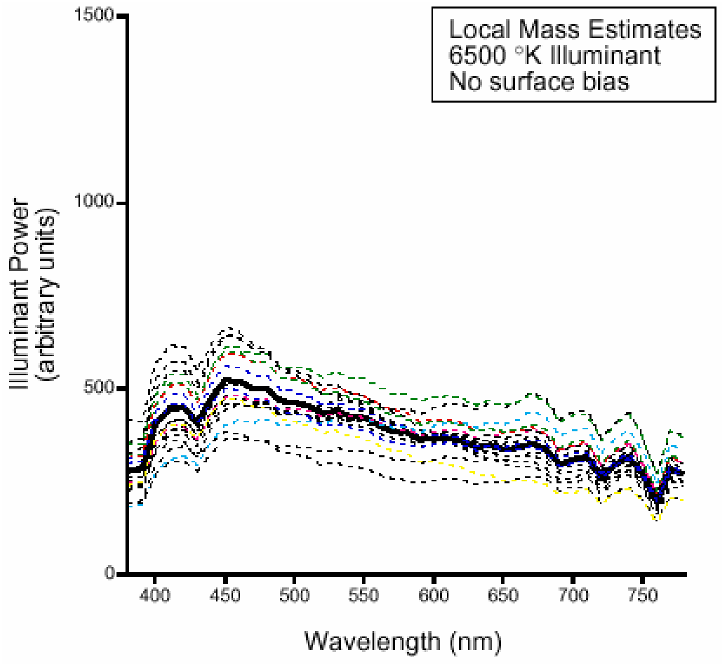
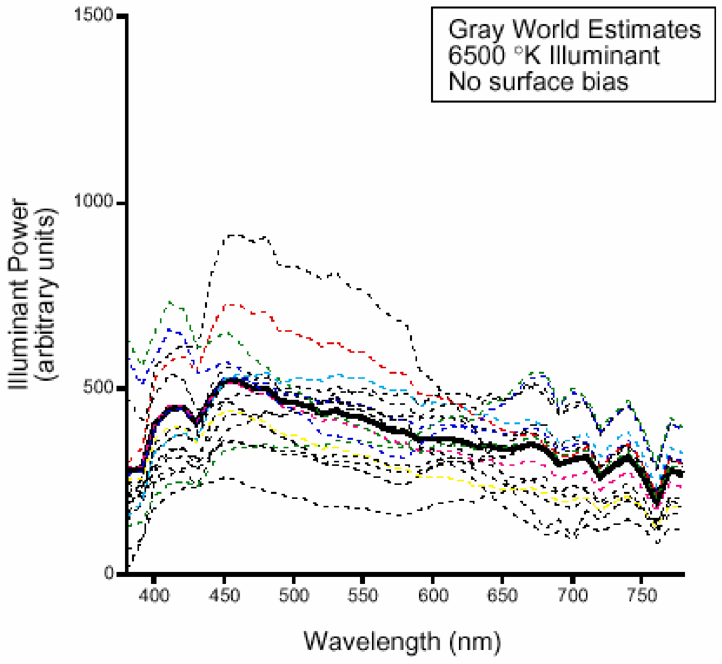


Figure 2: Left column: Three loss functions. Plots show penalty for guessing parameter values offset from the actual value, taken to be the plot center. (a) Minus delta function loss, assumed in MAP estimation. Only *precisely* the correct answer matters. (b) Squared error loss (a parabola), used in MMSE estimation. Very wrong guesses can carry inordinate influence. (c) Minus local mass loss function. Nearly correct answers are rewarded while all others carry nearly equal penalty. Right column: Corresponding expected loss, or Bayes risk, for the $y = ab$ problem. Note: loss *increases* vertically, to show extrema. (d) Expected loss for MAP estimator is minus the posterior probability. There is no unique point of minimum loss. (e) The minimum mean squared error estimate, (1.3, 1.3) (arrow) does not lie along the ridge of solutions to $ab = 1$. (f) The minus local mass loss favors the point (1.0, 1.0) (arrow), where the ridge of high probability is widest. There is the most probability mass in that local neighborhood.



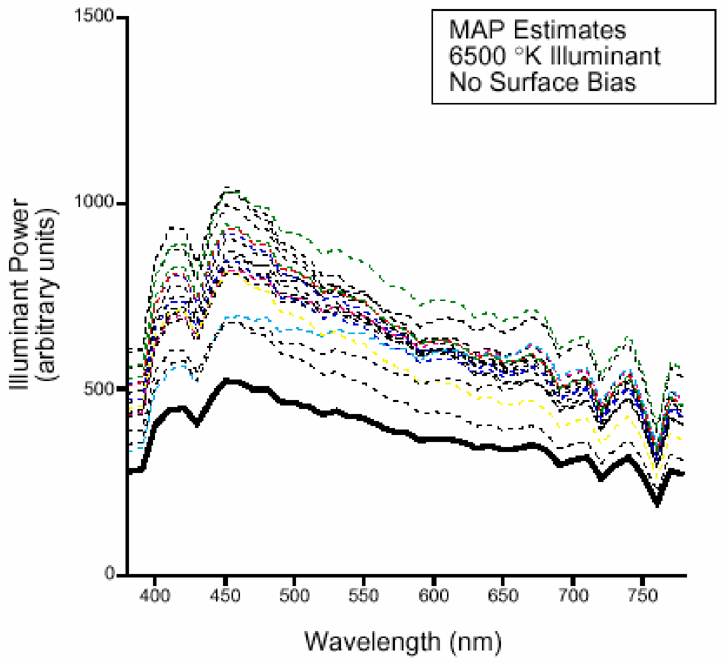
Local Mass Estimates
6500 °K Illuminant
No surface bias
Illuminant Power (arbitrary units)
Wavelength (nm)

(a)

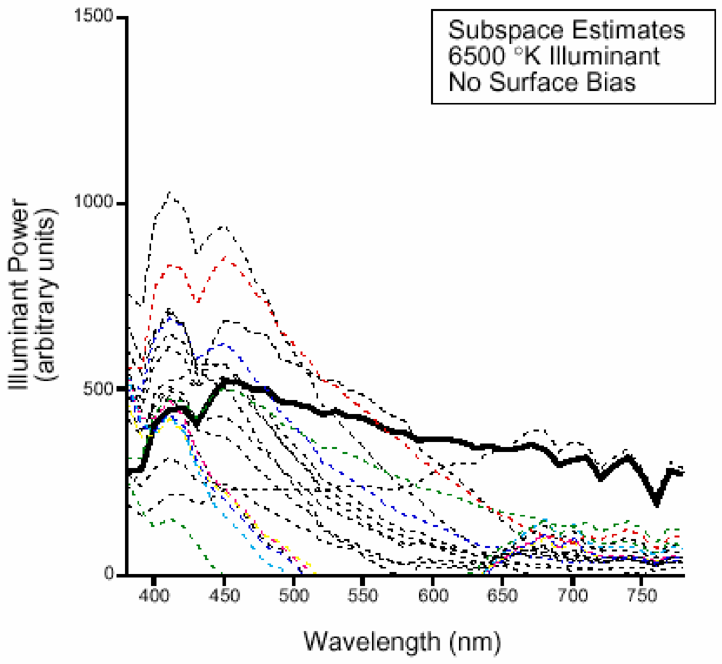


Gray World Estimates
6500 °K Illuminant
No surface bias
Illuminant Power (arbitrary units)
Wavelength (nm)

(b)



MAP Estimates
6500 °K Illuminant
No Surface Bias
Illuminant Power (arbitrary units)
Wavelength (nm)



Subspace Estimates
6500 °K Illuminant
No Surface Bias
Illuminant Power (arbitrary units)
Wavelength (nm)

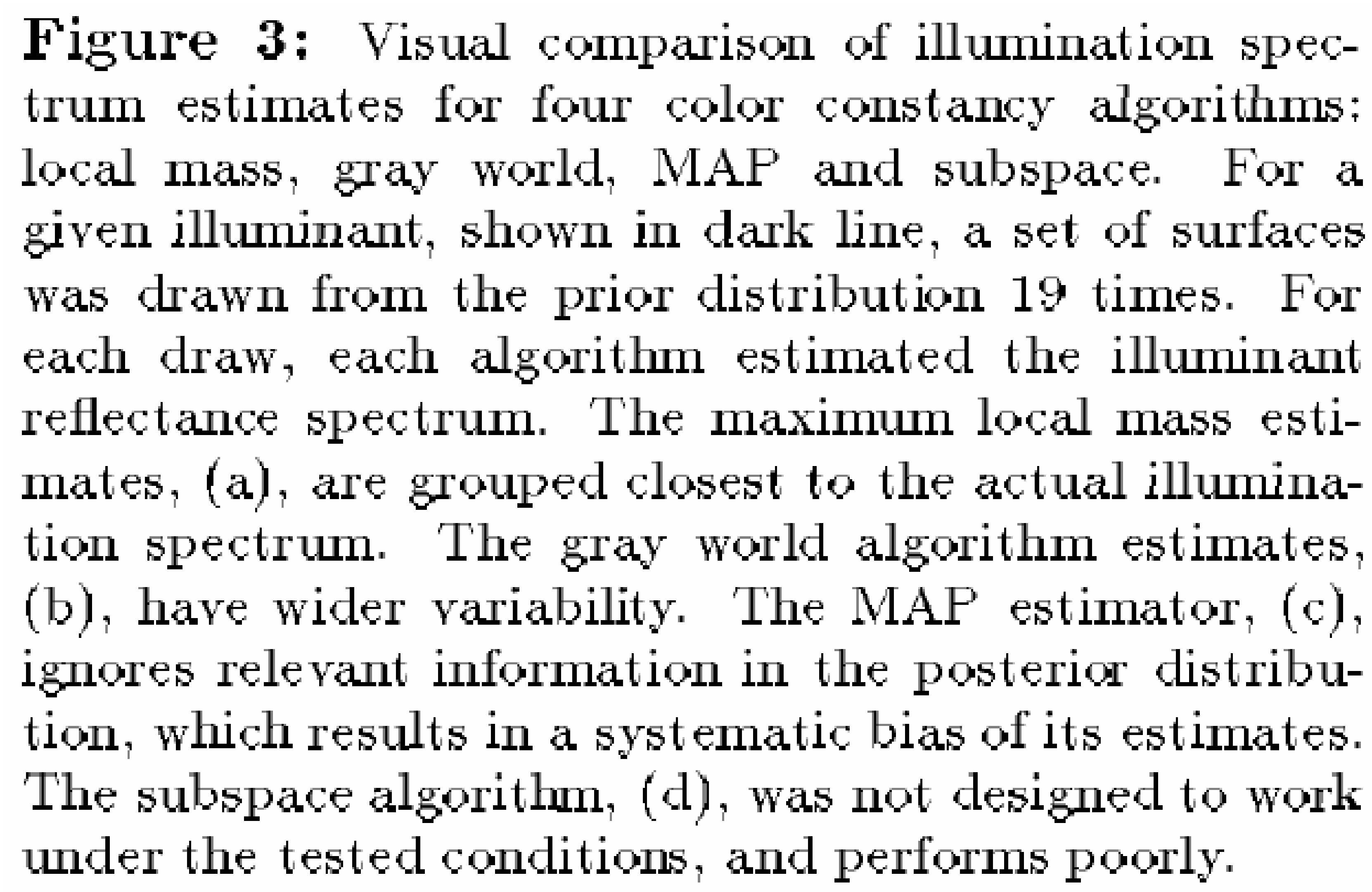
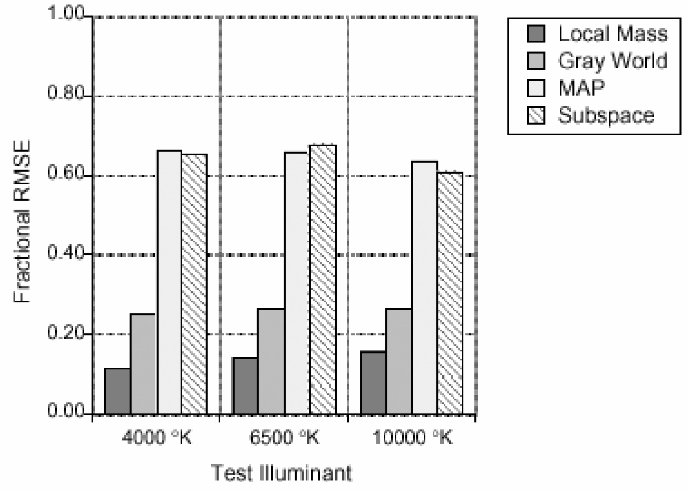
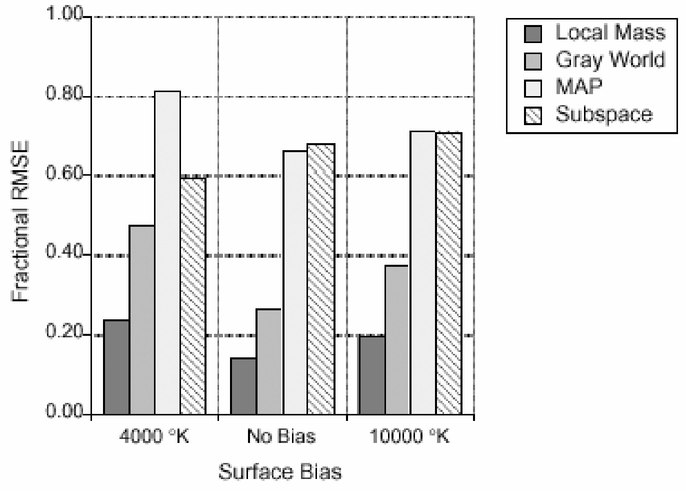


Figure 3: Visual comparison of illumination spectrum estimates for four color constancy algorithms: local mass, gray world, MAP and subspace. For a given illuminant, shown in dark line, a set of surfaces was drawn from the prior distribution 19 times. For each draw, each algorithm estimated the illuminant reflectance spectrum. The maximum local mass estimates, (a), are grouped closest to the actual illumination spectrum. The gray world algorithm estimates, (b), have wider variability. The MAP estimator, (c), ignores relevant information in the posterior distribution, which results in a systematic bias of its estimates. The subspace algorithm, (d), was not designed to work under the tested conditions, and performs poorly.



(a)



(b)

Figure 4: Summary results. (a) shows the performance of all four algorithms for three illuminants. (b) shows the performance of all four algorithms for three surface draw conditions. The performance measure is the average (over 19 individual runs) fractional root mean squared error (RMSE) between the estimate and true illuminant. For all conditions, the MLM estimate performs substantially better than the other algorithms. It is seen to be robust against these violations of its prior assumptions.

D. H. Brainard and W. T. Freeman, *Bayesian Color Constancy*, Journal of the Optical Society of America, A, 14(7), pp. 1393-1411, July, 1997

Regularization vs Bayesian interpretations

Regularization:
minimize

$$(1-ab)^2+\lambda(a^2+b^2)$$

Bayes:
maximize

$$\underbrace{e^{-\frac{(1-ab)^2}{2\sigma^2}}}_{\text{likelihood}}\;\underbrace{e^{-\lambda(a^2+b^2)}}_{\text{prior}}$$

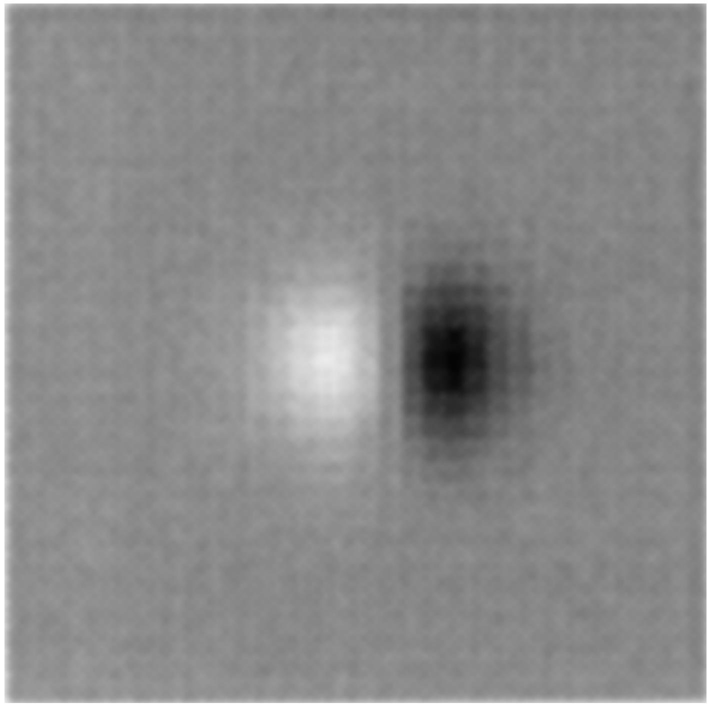
Bayesian interpretation of regularization approach

- For this example:
  - Assumes Gaussian random noise added before observation
  - Assumes a particular prior probability on a, b.
  - Uses MAP estimator (assumes delta fn loss).

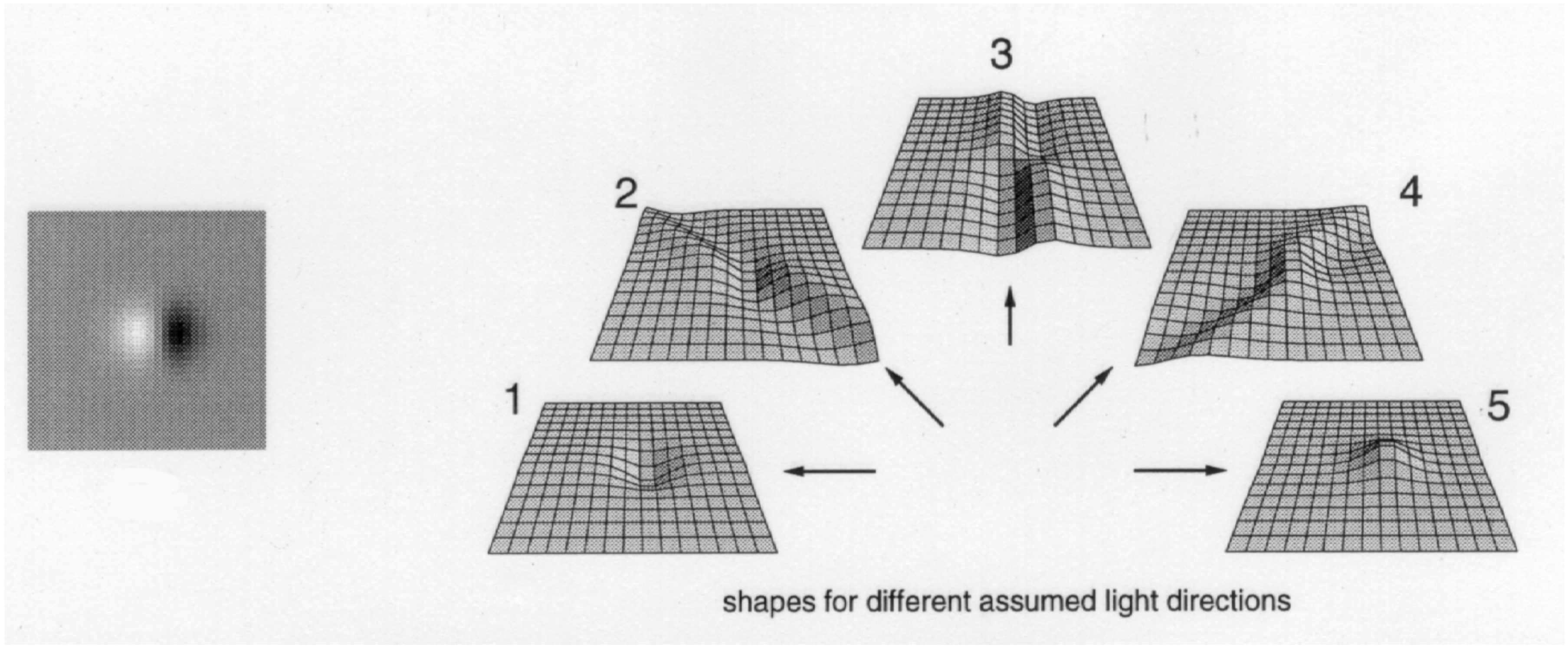
Why the difference matters

- Know what the things mean
- Speak with other modalities in language of probability
- Loss function
- Bayes also offers principled ways to choose between different models.

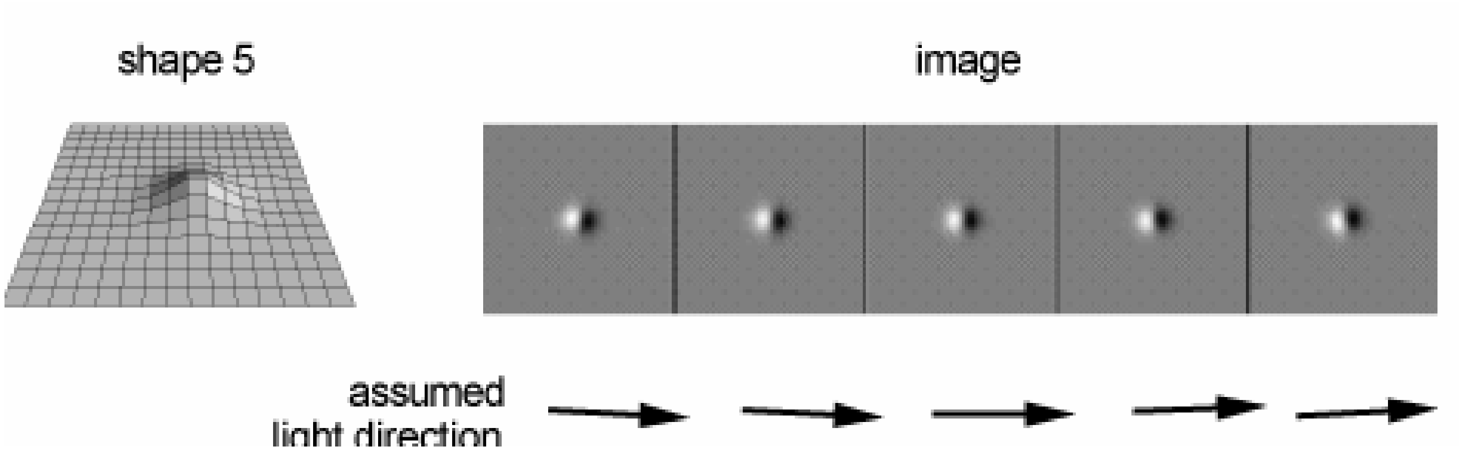
Example image

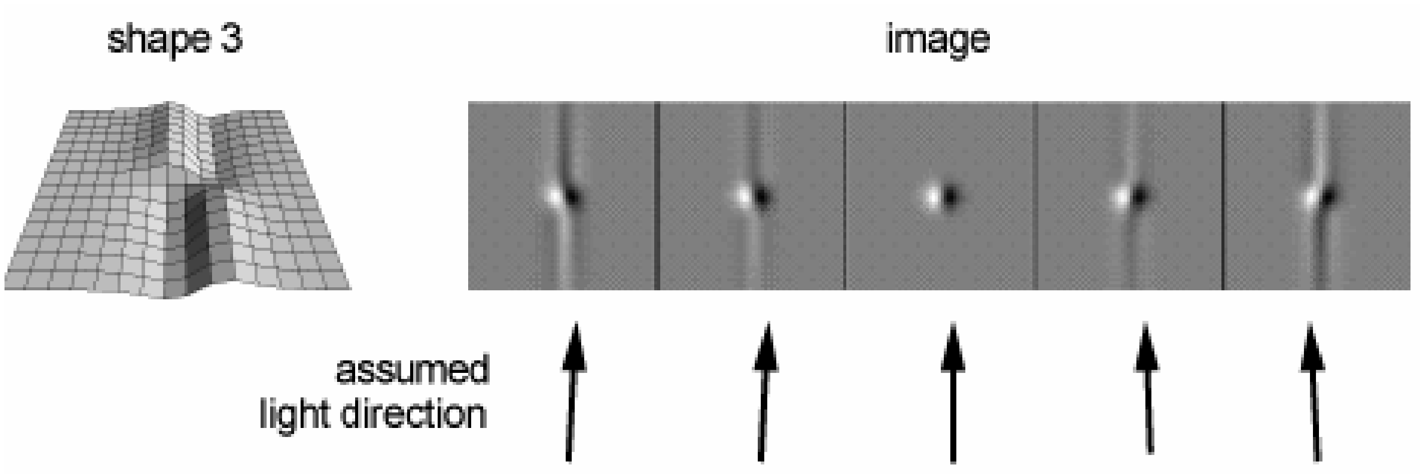


Multiple shape explanations



Generic shape interpretations render to the image over a range of light directions





Loss function

$$L(s,\theta \mid y) = \int P(s',\theta' \mid y) l(s,\theta,s',\theta') ds' s\theta'$$

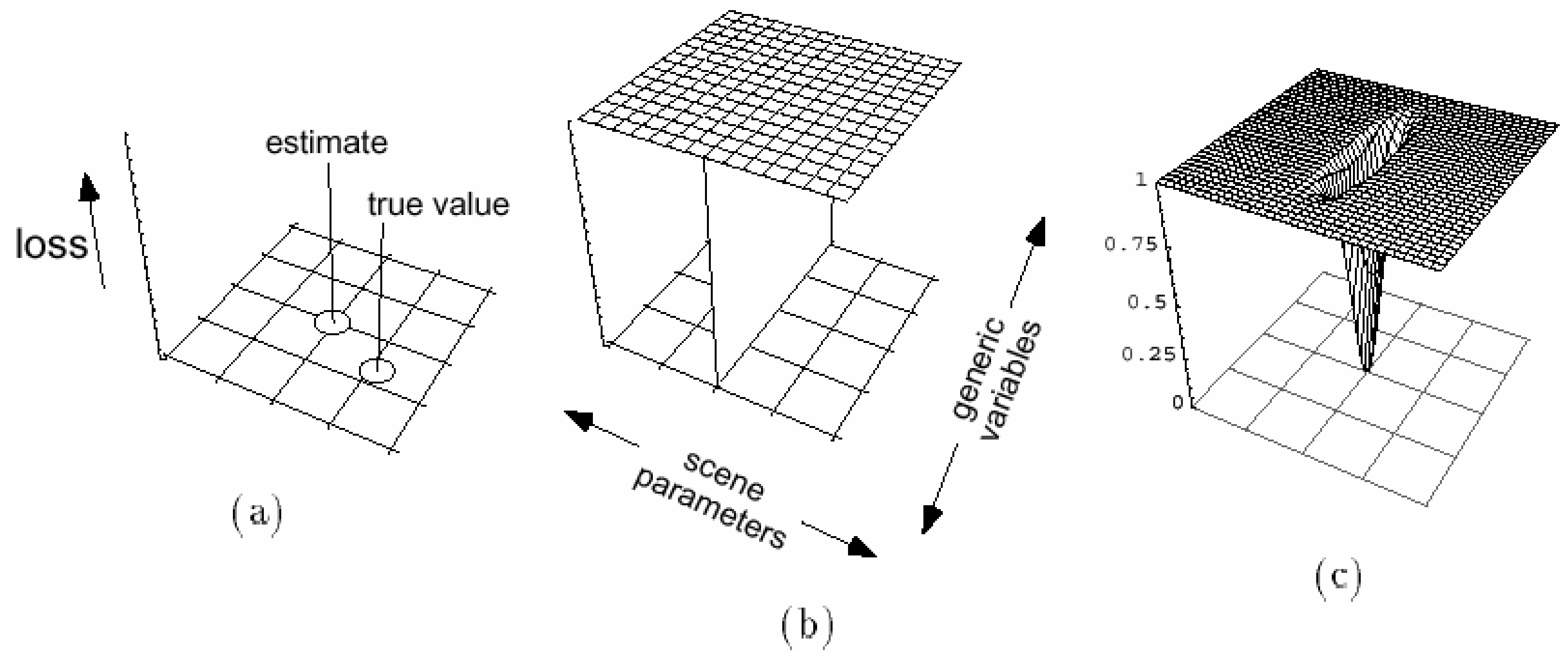
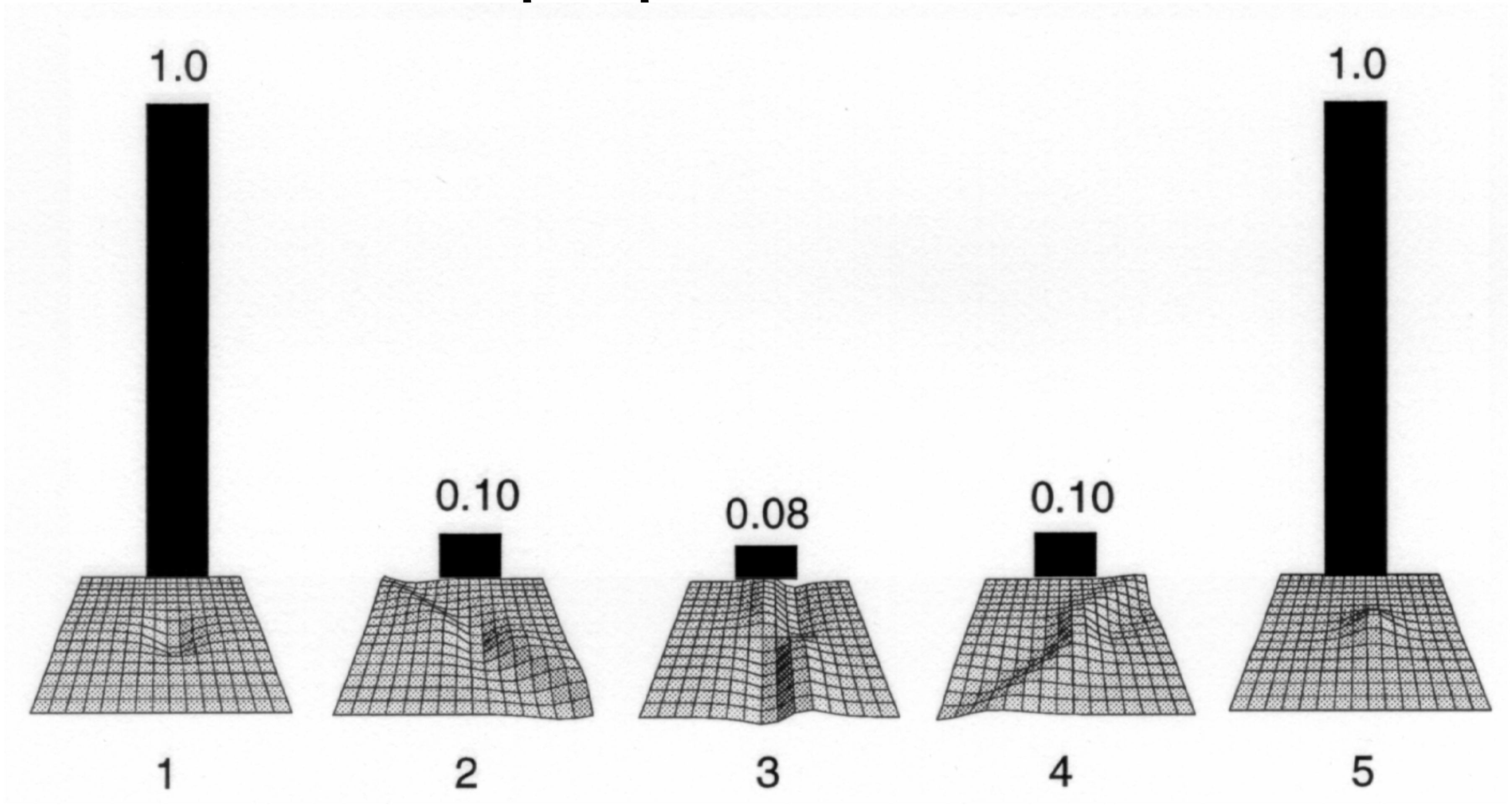
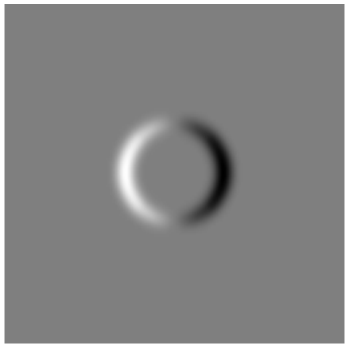


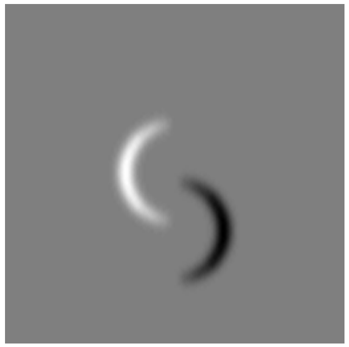
Figure 10: Loss function interpretation of generic viewpoint assumption. (a) shows the general form for a shift invariant loss function. The function $L(\mathbf{z}, \bar{\mathbf{z}})$ describes the penalty for guessing the parameter $\bar{\mathbf{z}}$ when the acutal value was $\mathbf{z}$. The marginalization over generic variables of Eq. (5) followed by MAP estimation is equivalent to using the loss function of (b). (c) Shows another possible form for the loss function, discussed in [11, 23, 24, 65].

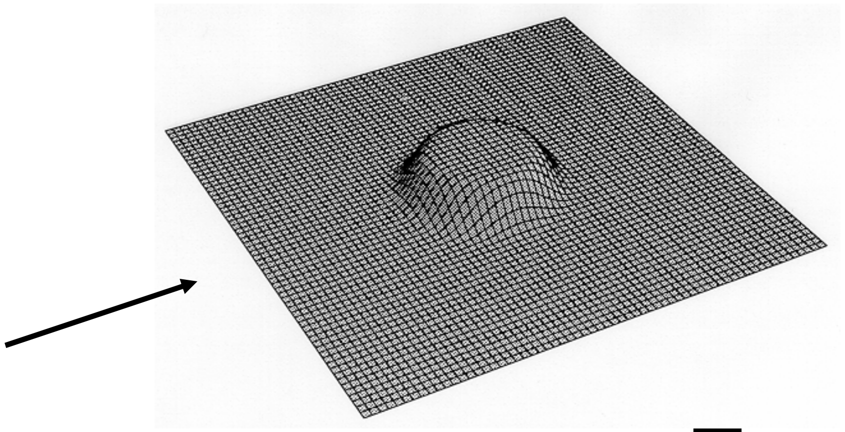
Shape probabilities

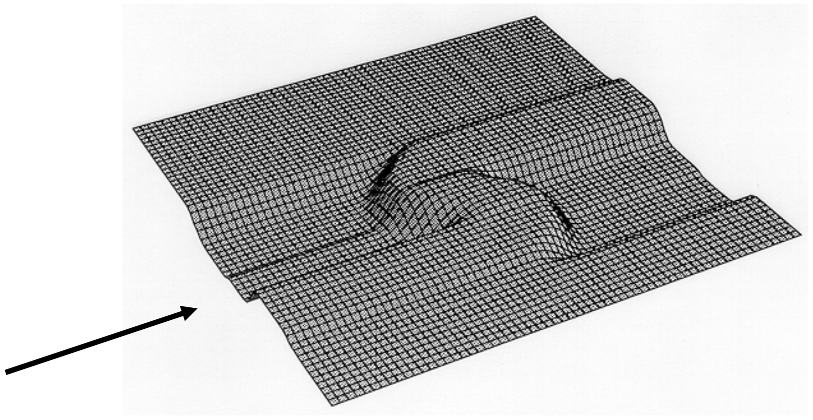


Comparison of shape explanations









- Lighting "genericity" of the shape explanation: