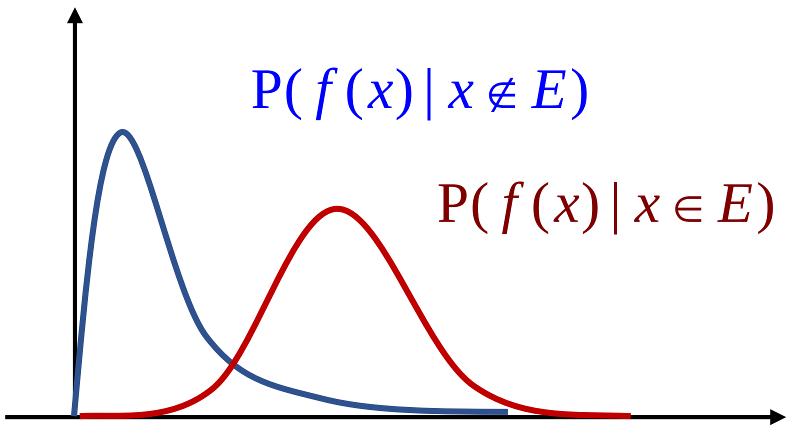
One strategy to learn decision rule (Konishi et al., 1999, 2003)

: **Learn distributions** $\begin{cases} \mathrm{P}(f(x) \mid x \in E) \\ \mathrm{P}(f(x) \mid x \notin E) \end{cases}$

$E = \{x : e(x) = 1\}$: set of edges



Example (1-dimension): $f(x) = \left|\dfrac{dI}{dx}\right|$

$\left|\dfrac{dI}{dx}\right|$ is typically larger on edges than not on edges

But, sometimes $\left|\dfrac{dI}{dx}\right|$ is small on edges and big not-on-edges

How to represent P? Parametric Distribution? (e.g. Gaussian)
or non-parametric? (e.g. histogram)

Learn $P\big(f(x)\,|\,x \in E\big), P\big(f(x)\,|\,x \notin E\big)$

Note: Memorization and Generalization

- Learn theses probability distribution using only part of the dataset
- Evaluate/test on other parts of the dataset – testing dataset

This is ensure generalization and present over learning (Machine learning)

Decision Rule: $\alpha(x) = 1\,(\text{edge}) \quad \text{if } \log \frac{P(f(x)|x \in E)}{P(f(x)|x \notin E)} \geq T$ $\quad$ T: threshold

$$0\,(\text{not edge}) \text{ if } \log \frac{P(f(x)|x \in E)}{P(f(x)|x \notin E)} < T$$

Changing the threshold affects:

i. The false positives – number of image pixels wrongly decided to the edges

ii. The false negatives – number of image pixels which really are edges, but are decided to be non-edges

What threshold to use? ➔ Depends on task

- Impossible to find a threshold which gives perfect results (i.e., has no false positive and no false negatives)
- Best not to make hard decisions too early
  - Use context and higher level information to decide

Comparisons **Canny** vs. $\log \frac{P\left(f(x) \mid x \in E\right)}{P\left(f(x) \mid x \notin E\right)}$

Performance depends on filters used

If $f(x) = \left|\nabla I(x)\right|$, then performance is similar

But for more sophisticated filter, then $\log \frac{P\left(f(x) \mid x \in E\right)}{P\left(f(x) \mid x \notin E\right)}$ is much better

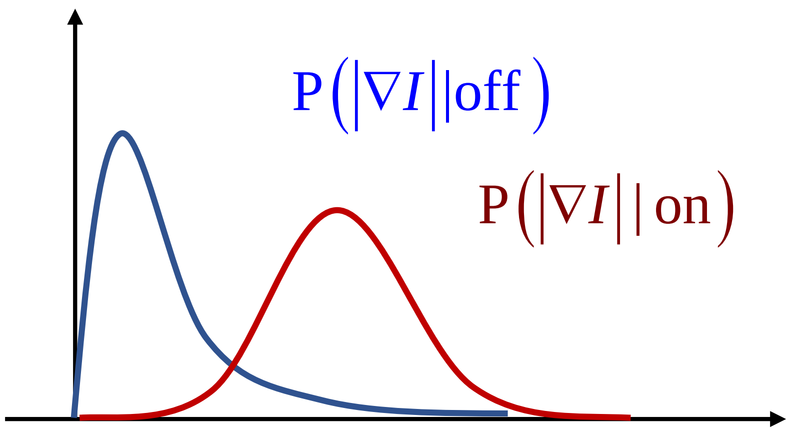
EG If $f(x) = \left(\left|\nabla I(x)\right|, \left|\nabla G * I(x)\right|\right)$, where $G * I$ is a smoothed image.

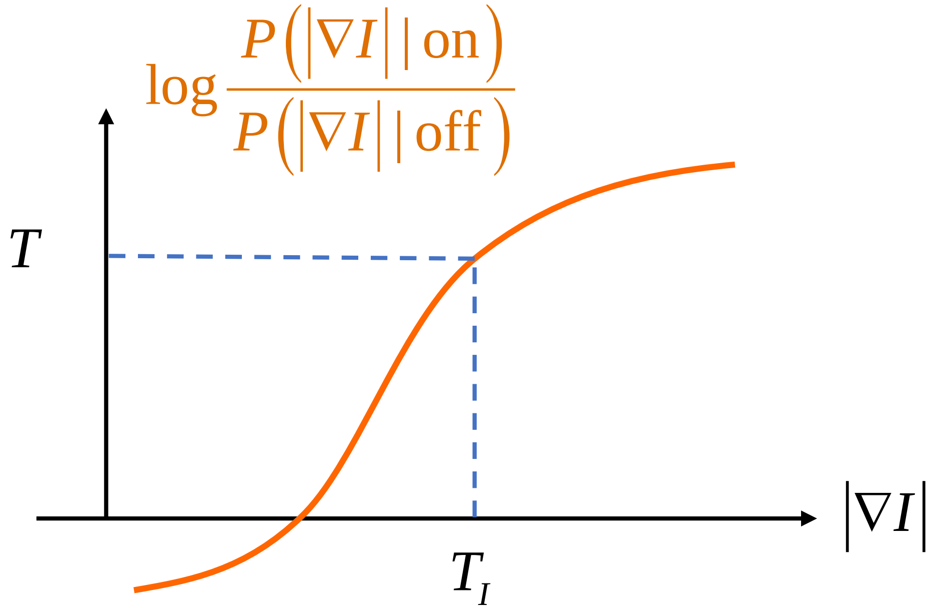
➔i.e., combine information at different scales

Problem: if we use more sophisticated filters, then we need more data to learn

Note: If you only use $|\nabla I|$ filter then results are disappointing

$\log \frac{P(|\nabla I| \mid \text{on})}{P(|\nabla I| \mid \text{off})}$ is monotonic in $|\nabla I|$





➔ Thresholding $\log \frac{P(|\nabla I| \mid \text{on})}{P(|\nabla I| \mid \text{off})}$ reduces to thresholding $|\nabla I|$

(so we rediscover the Sobel edge detector)

Datasets: Problems and Perils

Dataset Bias: Is the data representative of the set of natural images?

Experimental Design

EG South Florida dataset –contains images with little texture

→ edge detection is easy

Sowerby dataset – contains images with a lot of texture (e.g. vegetation)

→ edge detection is hard

An edge detector trained on South Florida will perform badly on Sowerby

An edge detector trained on Sowerby will perform well on South Florida

➔ South Florida dataset is biased

Labeling bias → where does ground-truth come from?

In Berkeley BDS datasets, 5~6 students label each image independently

But, not all labelers agree

- “Strong Edges” are labeled by all 5 students
- “Weak Edges” are labeled by 1 student only

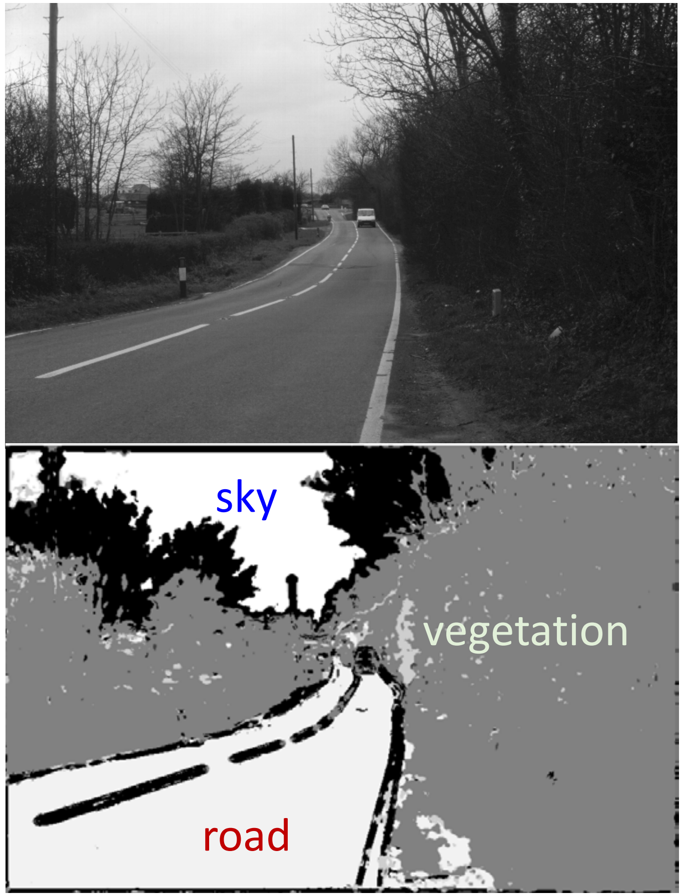
Psychophysics experiments (X. Hou et al., CVPR 2014) show that weak edges are poorly defined

→ Problematic to use weak edges for training and testing

Summary

- Use visual cues for edges to turn it into a classification task.
  - Apply S & ML techniques
- Use datasets with ground-truth to train/learn and evaluated different methods
  - Konishi *et al*., Malik *et al*., Dollar *et al*.
- Edge detection is impossible locally (context helps)
  - Balance false positives and false negatives
  - Typically better to have few false negatives on more false positives

Extend to other vision tasks / What else can we do locally?



(Konishi & Yuille, 1999)

EG Classify an image pixel as {sky, vegetation, road, other}

Dataset: Sowerby

Strategy: same as before

Learn $\mathrm{P}\big(f(x) \mid s\big)$

s is a label → e.g., sky, vegetation, etc.

$f(x)$: filters

Classify x by $\hat{s} = \arg\max P\big(f(x) \mid s\big)$