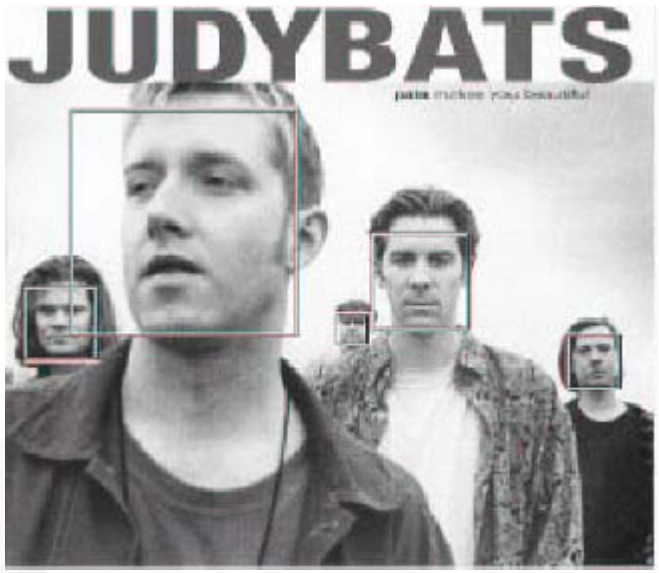
CORNELL

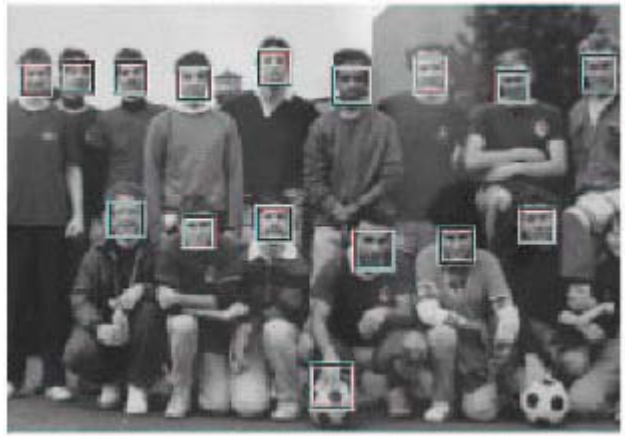
CS 664 Lecture 1
Fast Detection Algorithms

Prof. Dan Huttenlocher
Fall 2003

Fast Object Detection

- For example finding faces at video rates



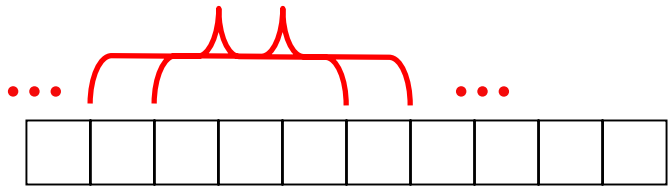


Dynamic Programming (DP)

- General algorithmic technique
  - Not specific algorithm
  - Analogous to "divide and conquer" – bottom up
- Methods that cache solutions to sub-problems rather than re-computing them
  - E.g., Fibonacci, substring matching
- Applies to problems that can be decomposed into sequence of stages
  - Each stage expressed in terms of results of fixed number of previous stages

Simple DP Example: Box Sum

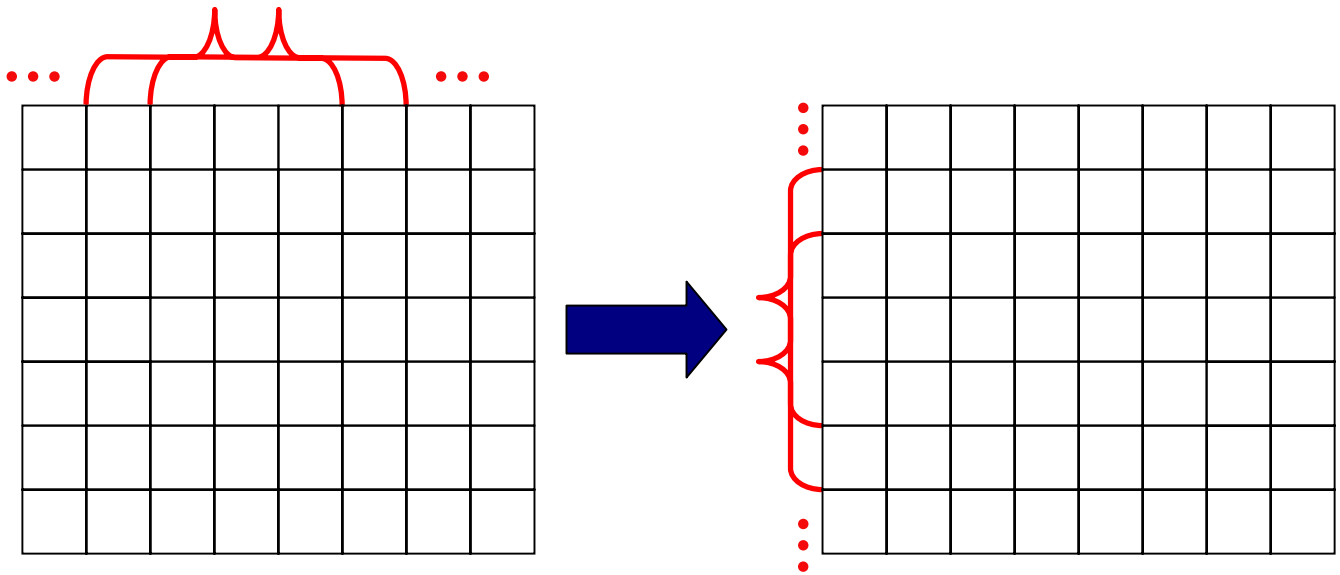
- Sum n-vector over sliding k-window
  - $W_k[x] = f[x] + \ldots + f[x+k]$
  - Note: often k odd, sum between $x \pm (k-1)/2$



- Explicit summation O(k*n) additions
- Recurrence yields O(n+k) time method
  - $W_k[x] = W_k[x-1] + f[x+k] - f[x-1]$
  - Each element of sum differs from previous by just two values

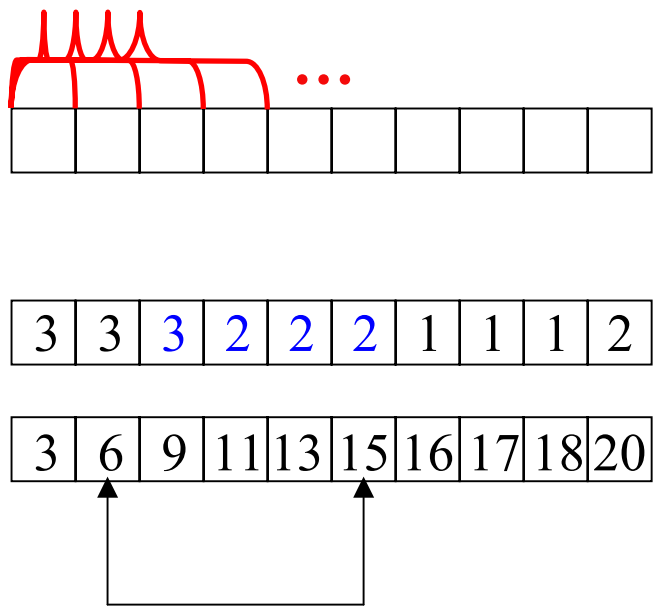
Box Sums in d Dimensions

- One pass along each dimension
  - Sum intermediate result from previous pass
  - 2D case: horizontal then vertical (or vice versa)
    - m by n image, O(mn+wh) time vs. O(mnwh)
    - E.g., 10 by 10 summation window, 100x faster



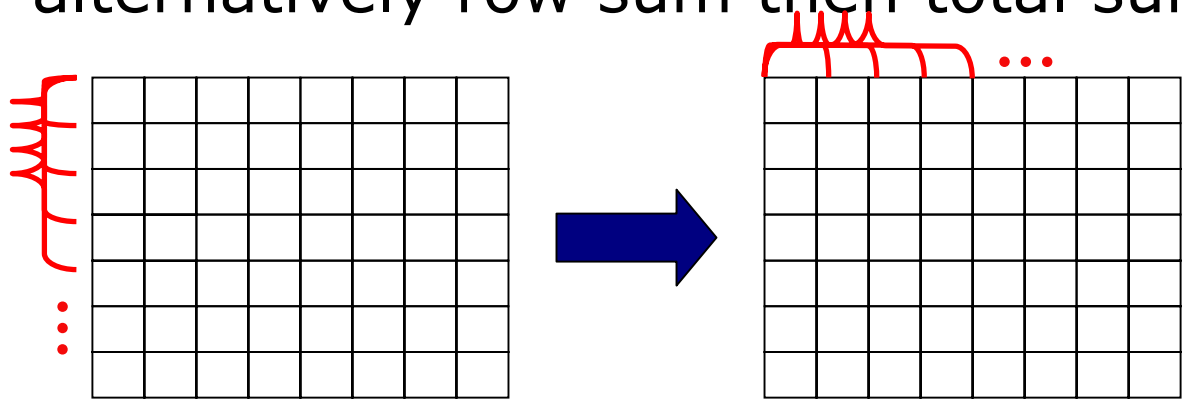
1d Integral Images

- Fast summations over different sized regions (non spatially uniform)
- Cumulative sum
  - S[x] = f[0] + … + f[x]
- DP recurrence O(n) time
  - S[x] = S[x-1] + f[x]
- Sum over window of f[x] independent of size k
  - W_k[x] = S[x+k-1]-S[k-1]



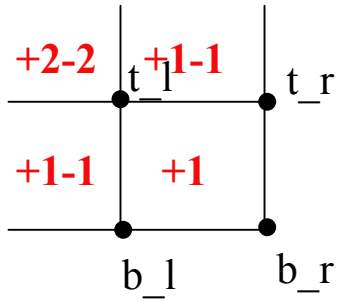
n-d Integral Images

- Analogous for higher dimensions, 2D:
  - S[x,y] = f[0,0] + … + f[0,y] + …
    f[x,0] + … + f[x,y]
- Separate recurrence per dimension
  - C[x,y] = C[x,y-1] + f[x,y] (column sum)
  - S[x,y] = S[x-1,y] + C[x,y] (total sum)
  - Or alternatively row sum then total sum



Fast Region Sums With II

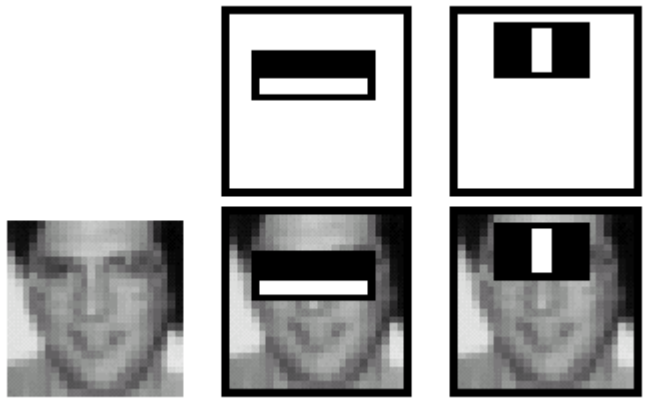
- Sum over a rectangle, constant time
  - S[b_r] + S[t_l-(1,1)] – S[b_l-(1,0)] – S[t_r-(0,1)]



- Sum over arbitrary region, linear time
  - Running time proportional to length of boundary not area

Fast Detection With II

- Features formed from combinations of sums over rectangles
  - For example positive and negative regions
  - Running time independent of rectangle size
- Viola and Jones use for face detection at approximately video rates



Fast Detection With II

- Also useful for arbitrary shaped regions
  - Decompose into rectangles
    - With no holes in worst case this is number of scan lines (not too bad with holes either)
    - Proportional to boundary length rather than area
  - Construct chain-code representation of boundary and sum values
    - Positive for downward links and negative for upward (reverse for holes)
  - Note relation to work of Jermyn and Ishikawa on boundary integrals