

Agenda

- Introduction
- Call for presentations
- Discussion of paper



University of Rochester
Computer Vision Seminar

February 9th, 2007

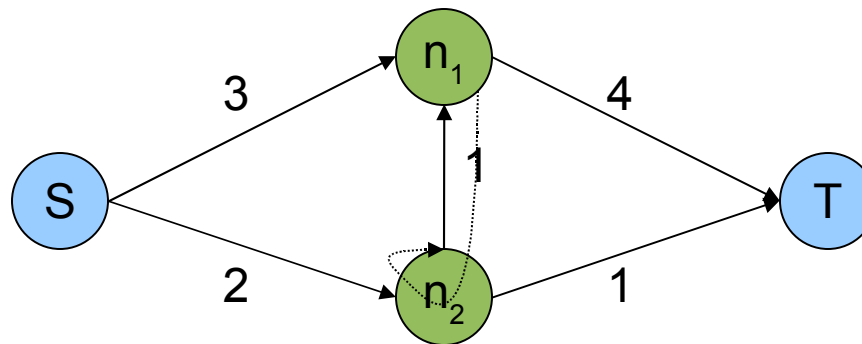
“Exact Voxel Occupancy with Graph Cuts”

Overview and Discussion

Slides available at <http://www.cs.rochester.edu/~ardis/seminar.html>

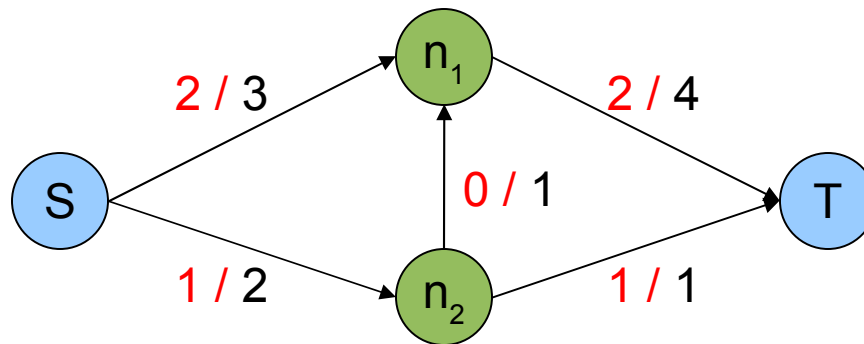
Background

- A *voxel* is a volumetric primitive
- An *S-T graph* is a directed, weighted, acyclic graph with source node S and sink node T, where the weights are capacities



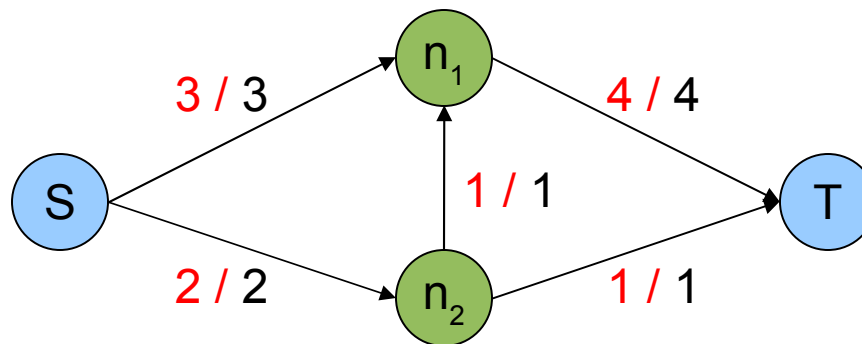
Background

- A *flow* is a setting of usages such that each edge doesn't have its capacity exceeded and the sum of flows in equals sum of flows out (S and T excluded)



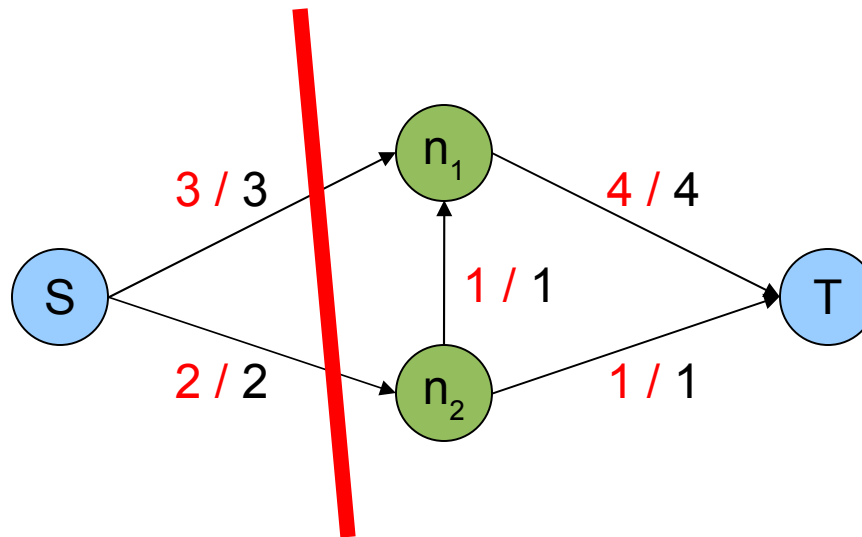
Background

- A *max flow* is a flow where the maximum amount flows out of S and flows into T (equivalent, since only S and T can have unbalanced in/out)



Background

- A *min cut* is a partition of vertices into two disjoint sets, with S and T not in the same set, such that the sum of the capacities between sets is minimized



Notation


- $N(v)$ Neighboring voxels to v
- f_v Label (1 = occupied) for v
- $D_v(f_v)$ Penalty for assigning f_v to v
- λ Penalty for differently-labeled neighboring voxel
- δ Impulse function (1 at origin, 0 otherwise)

Energy Minimization

$$E(f) = \sum_{v \in V} [D_v(f_v) + \lambda \sum_{v, v' \in N(v)} (1 - \delta(f_v - f_{v'}))]$$

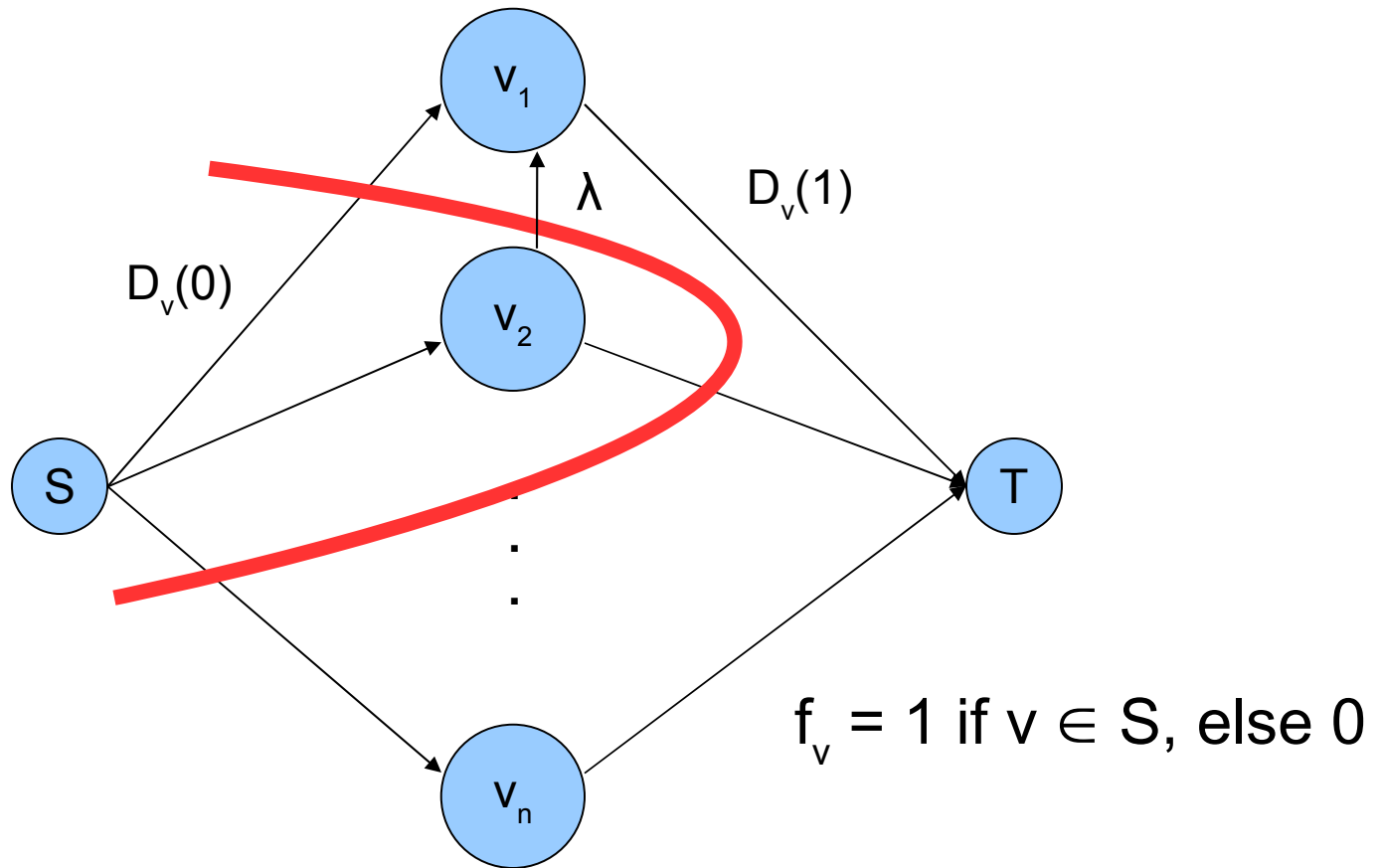
Sum of all voxel penalties

Sum of “neighbor
difference” penalty
over all neighbors



General labeling penalty
for applying f_v to v

Minimization via Cut



Experiments

- $D_v(1) = 300$

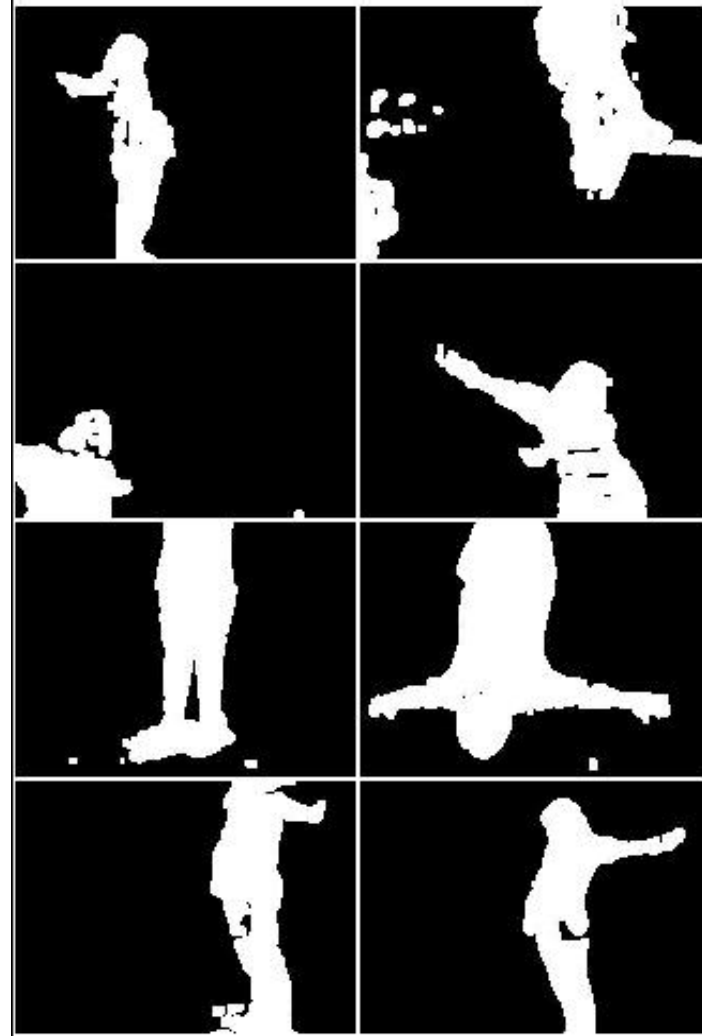
- $D_v(0) = [\sum_{\Delta \in O(v)} \min(\Delta^2, 400)] / 16$

- $\lambda = 30$



Difference in pixel intensity with all pixels that intersect within voxel v

Results



Thoughts for Discussion

- Larger neighborhood?
- Uneven lighting conditions?
- Colorspace information / varied neighbor penalties based on other knowledge?

