

PROCESSING DIGITAL CAMERA IMAGES

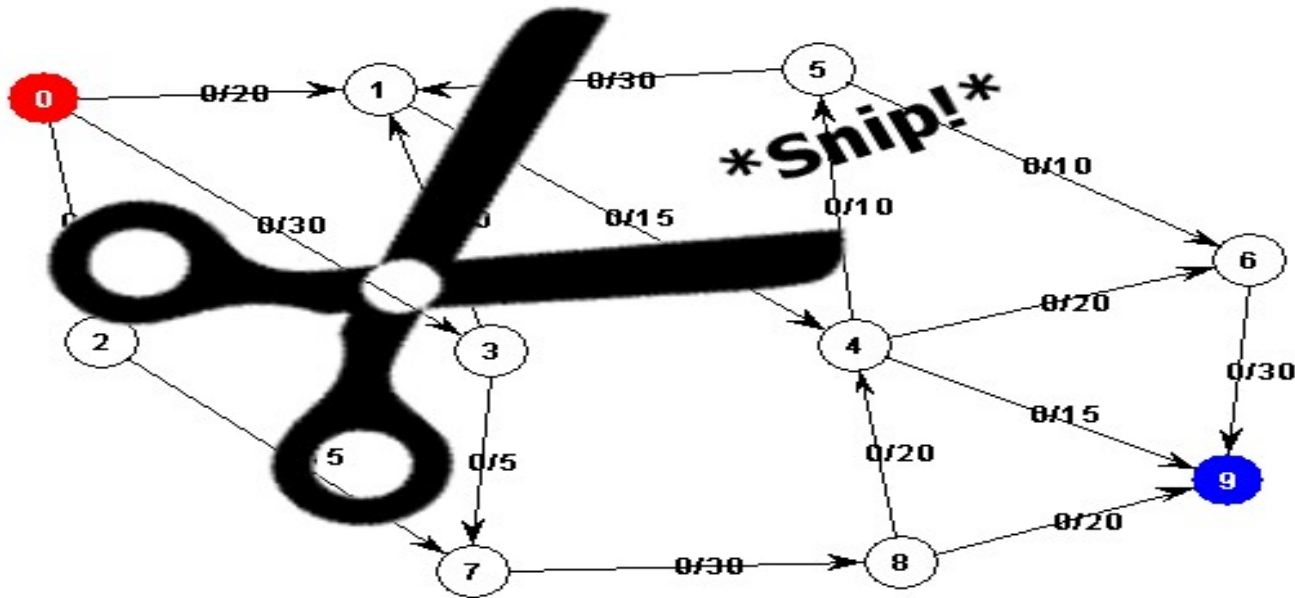
Graph Cut

Arwed Meyer

What is „Graph Cut“?

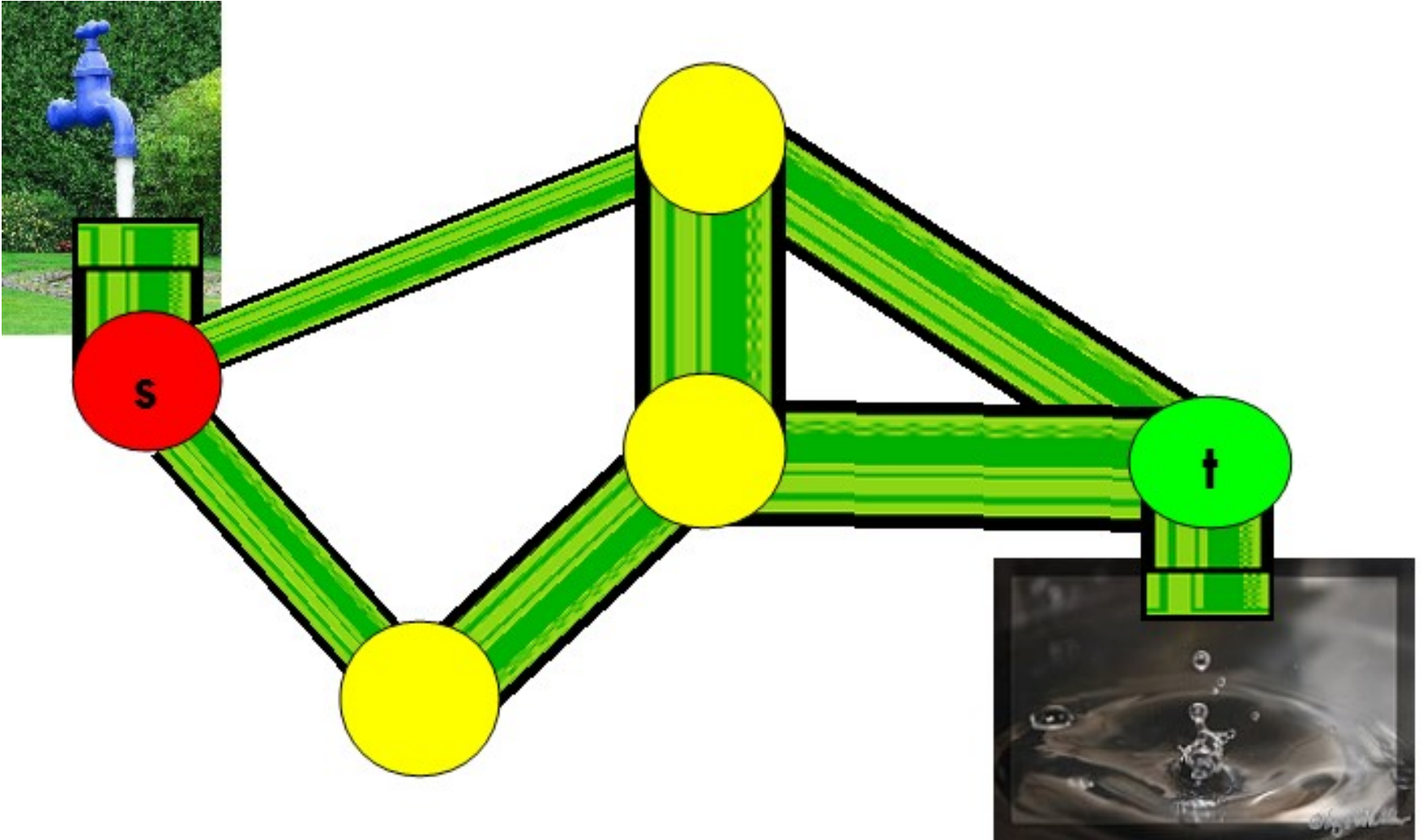
2

Graph-Cut is a class of algorithms that use the min. cut/max flow of graphs for optimization problems.



Max. Flow in Graphs

3

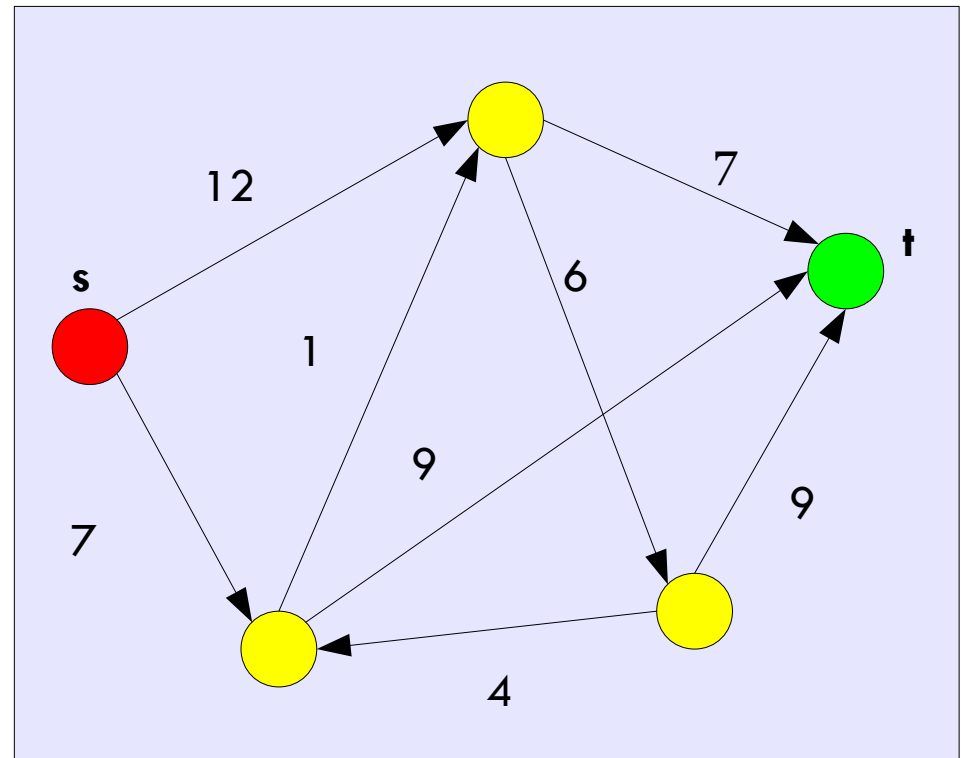


Some Graph Theory

4

Search for the maximum flow from a source vertex “s” to a sink vertex “t” in directed graphs with weighted edges.

- $f(u,v) \leq c(u,v)$
- $f(u,v) = -f(v,u)$
- All flows into a vertex equal all flow from a vertex (except s & t)



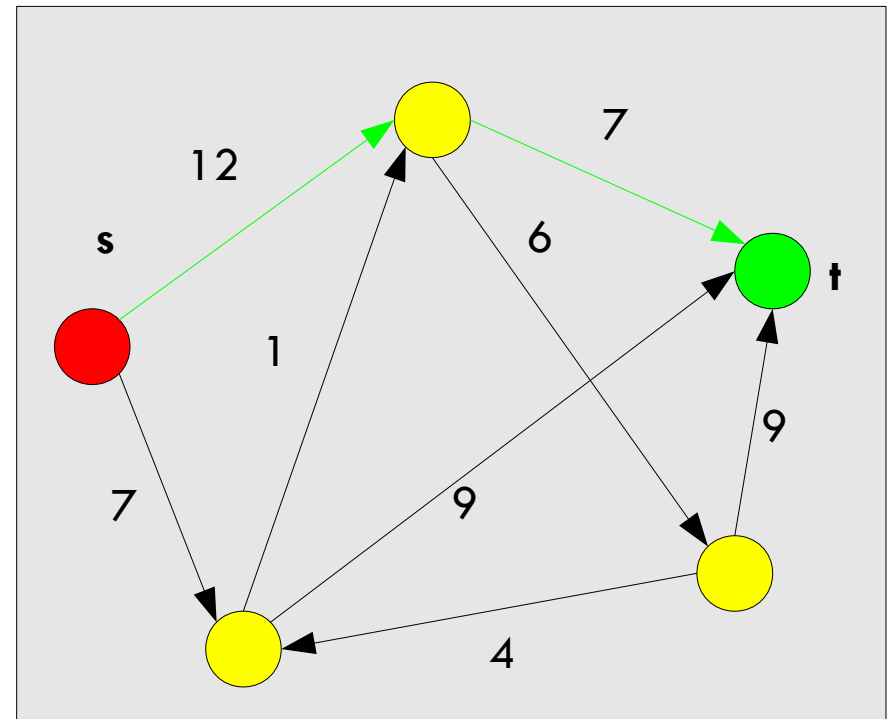
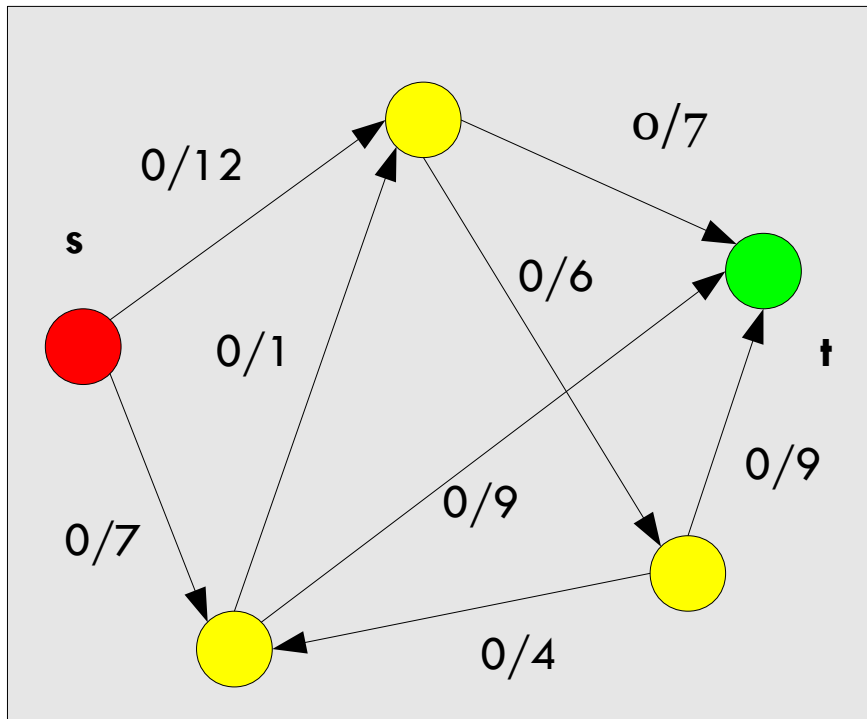
Ford-Fulkerson

5

Preparations/ 1^{st} step: Graph and residue graph

Find a path that increases the flow

max flow $f = 0$



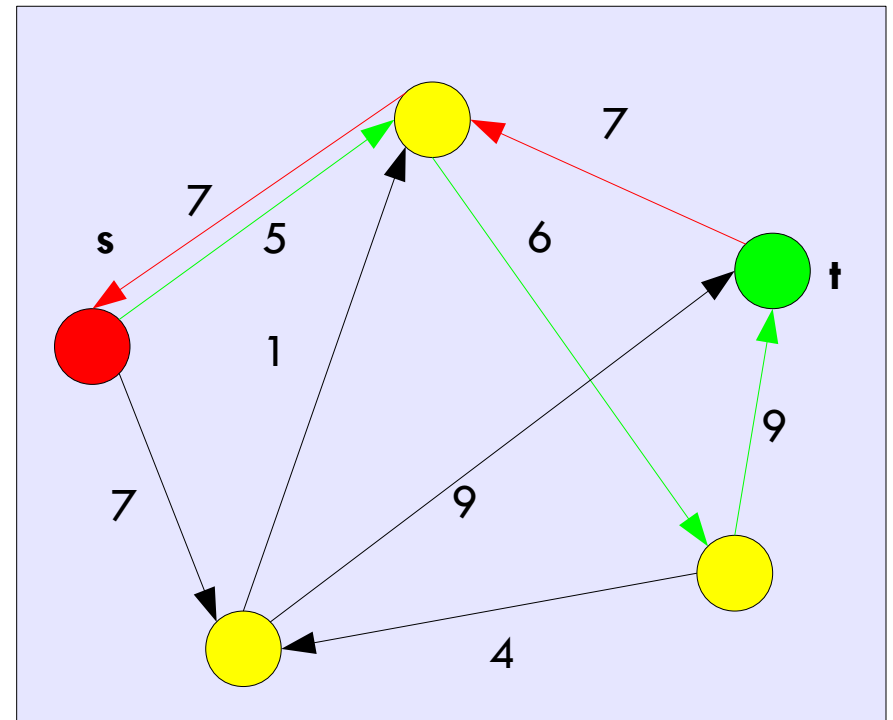
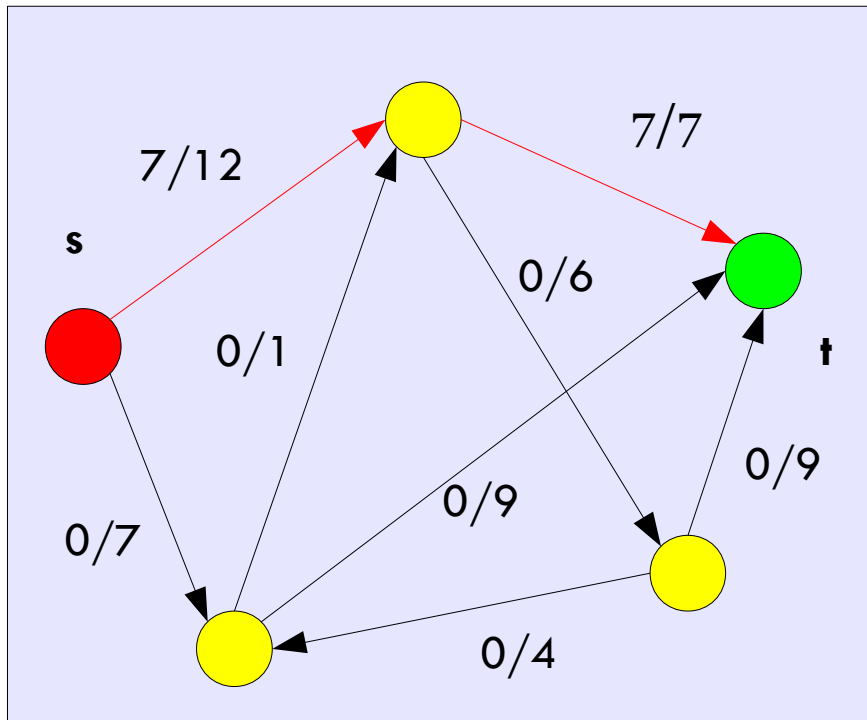
Ford-Fulkerson

6

2nd step:

add flow from augmentation path (red), update residue graph and find new path in residue graph (green)

$f=7$

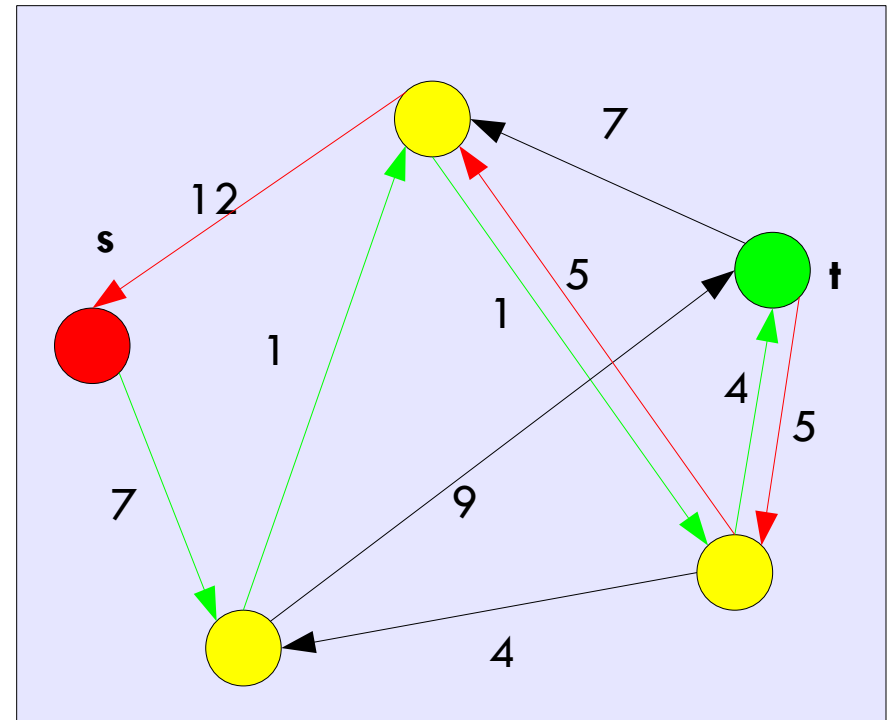
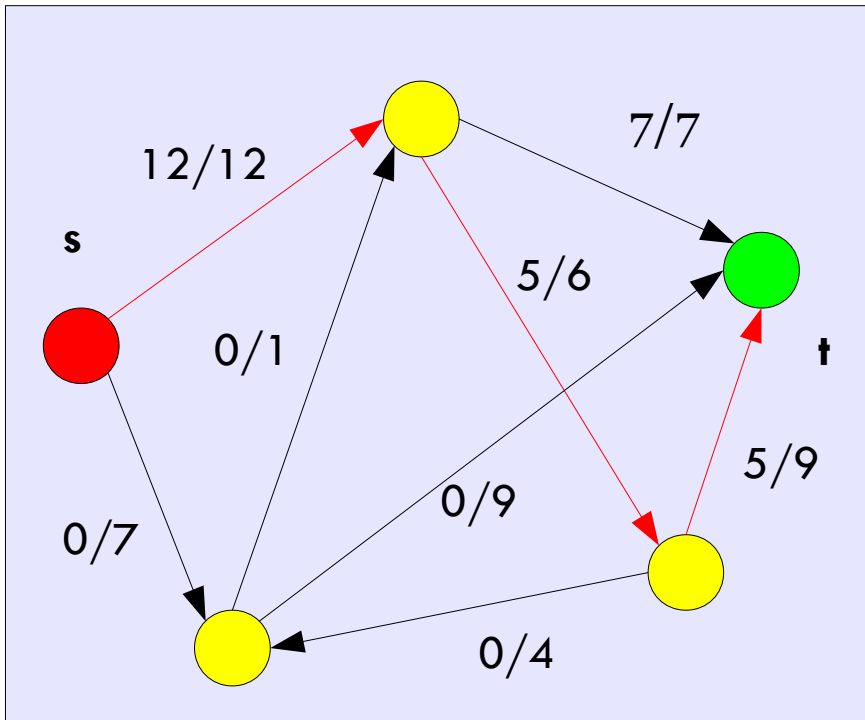


Ford-Fulkerson

7

3rd step:

$f=12$

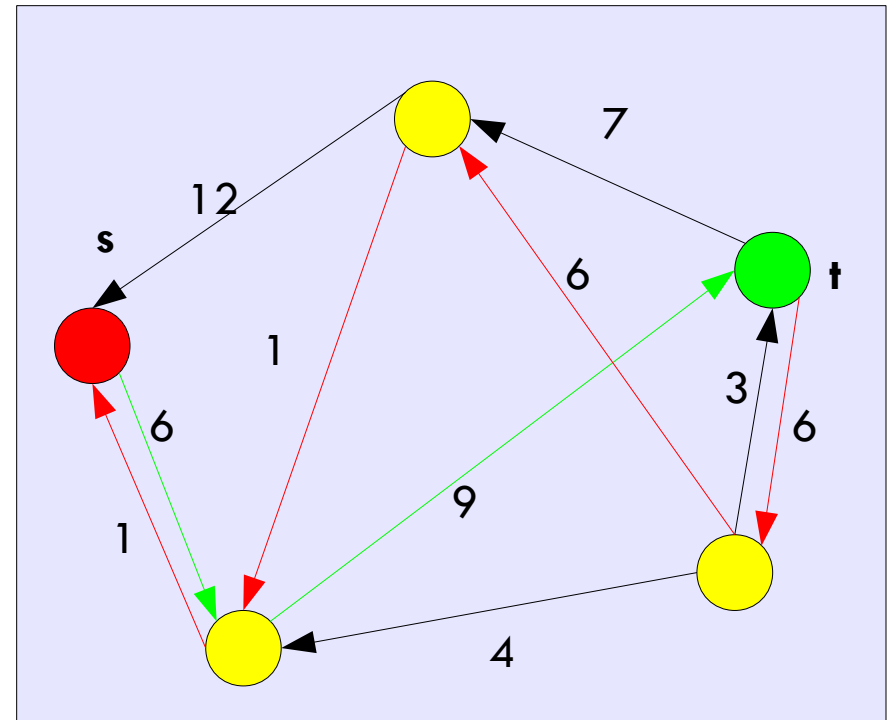
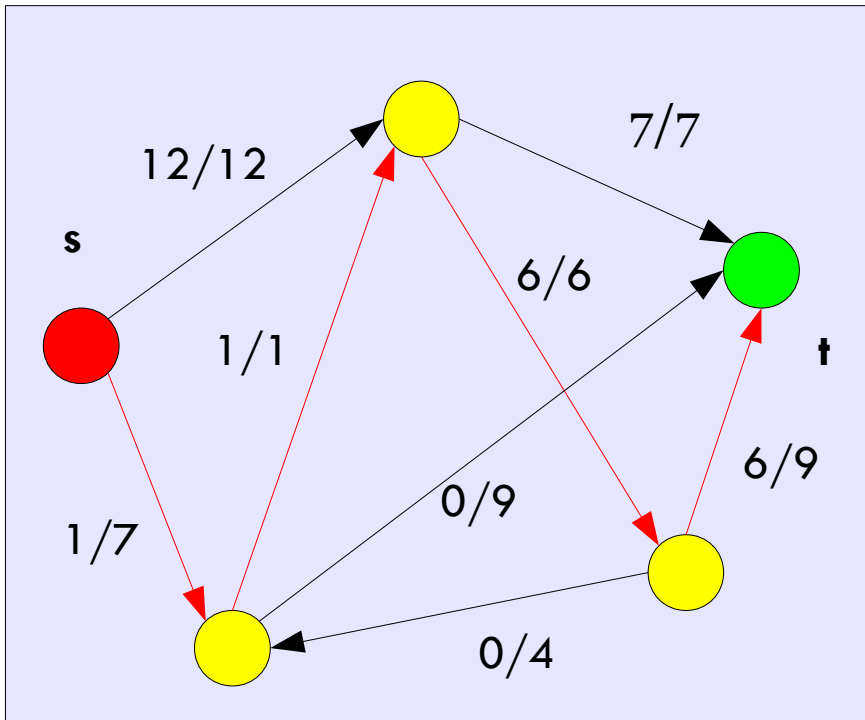


Ford-Fulkerson

8

4th step:

$f=13$



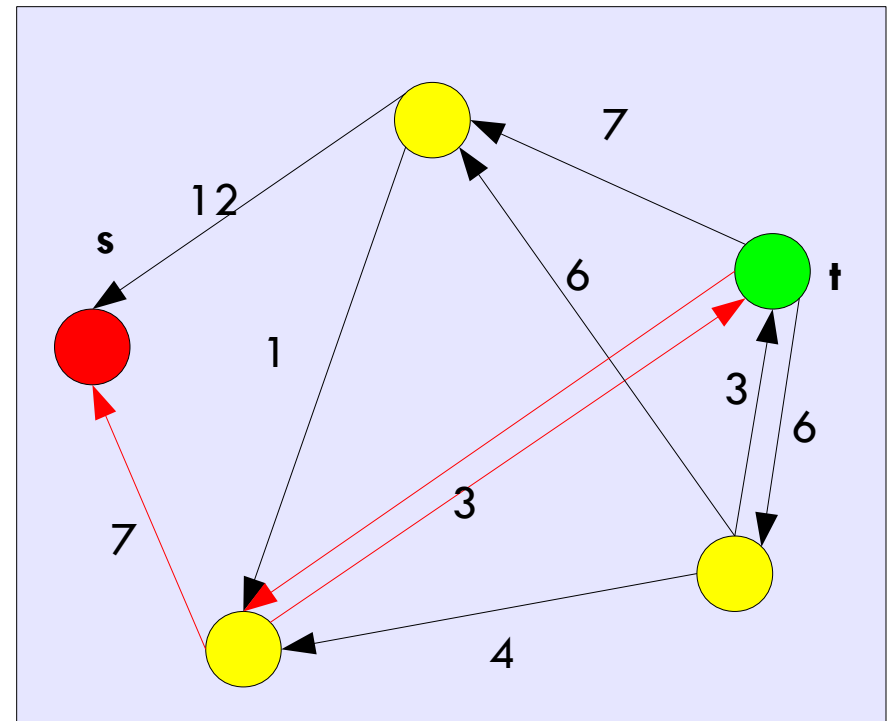
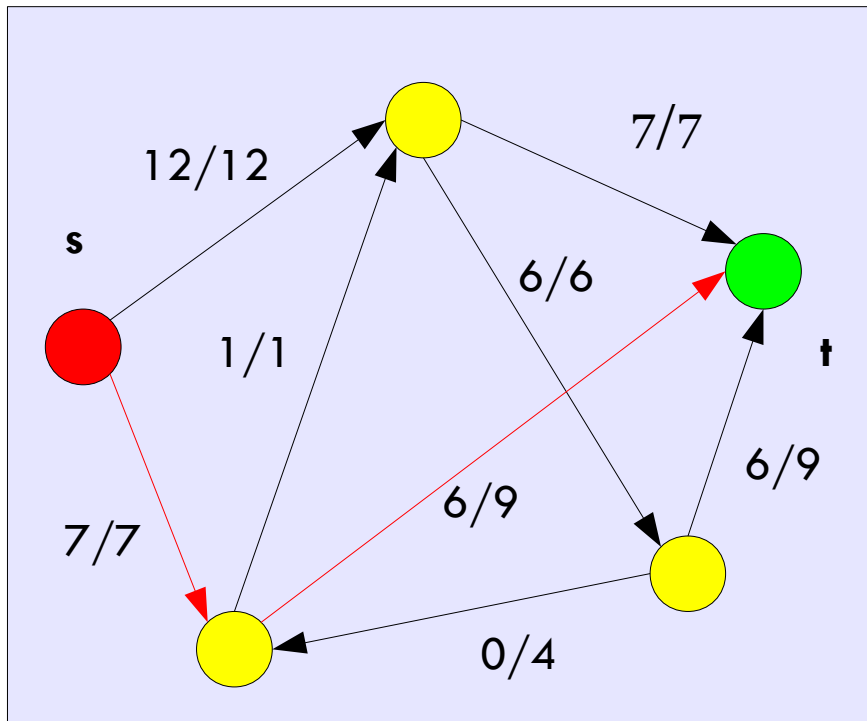
Ford-Fulkerson

9

5th step:

There is no other path from s to t that increases the flow, so we're done!

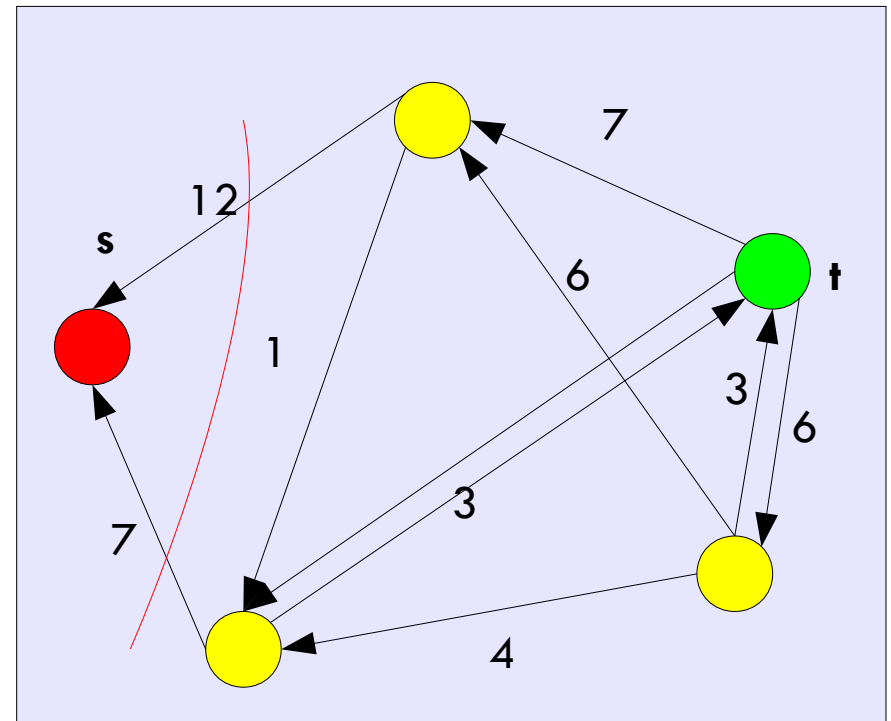
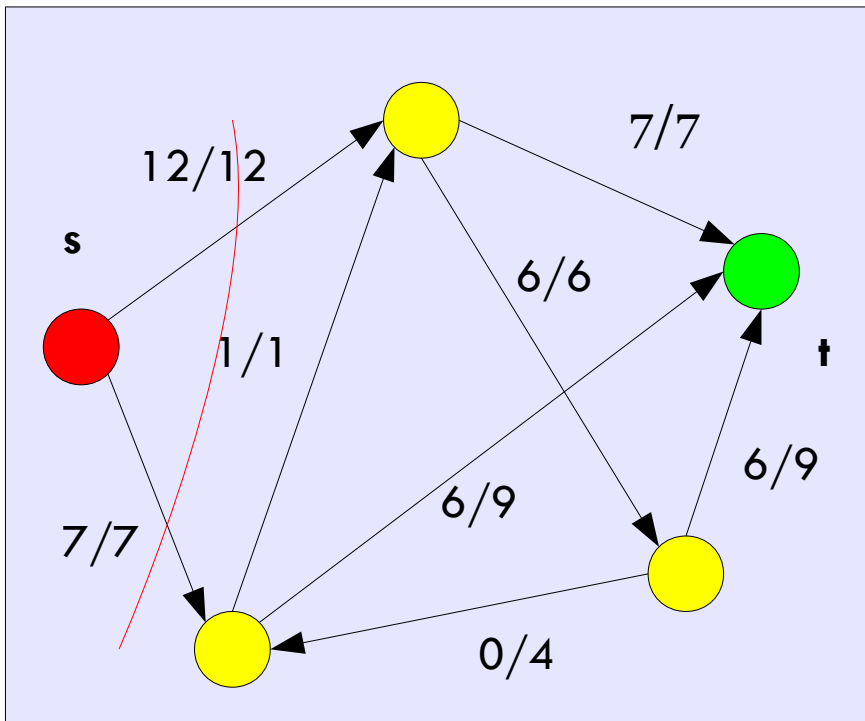
Max flow $f=19$



From max flow to min cut

10

The min cut is placed between nodes S reachable from s and all other nodes T .



Ford-Fulkerson

11

A basic implementation in pseudo code:

worst case: $O(|f_{\max}| * |E|)$

```
1 ford_fulkerson(G, s, t) {
2     foreach (edge (u,v) in E) {
3         f[u,v] = 0; f[v,u] = 0
4     }
5
6     while (p = depth_first_path(s t, Gf)) {
7         cap_p = min(res_cap(u,v) : (u,v) in p)
8
9         foreach (edge (u,v) in p) {
10            f[u,v] = f[u,v] + cap_p
11            f[v,u] = -f[u,v]
12        }
13    }
14
15    return f
16 }
```

Optimizations

12

Edmonds-Karp: Use breadth-first path search

worst case: $O(|V| * |E|)$

Boykov (new algorithm)

Push-Relabel algorithms

Applications

13

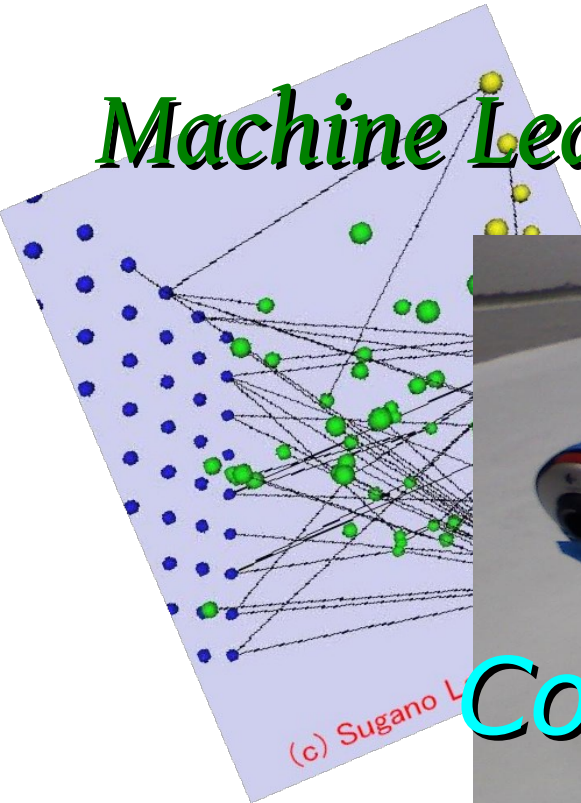
Now that we found the min cut, what are we gonna do with it?

Machine Learning

Computer Graphics

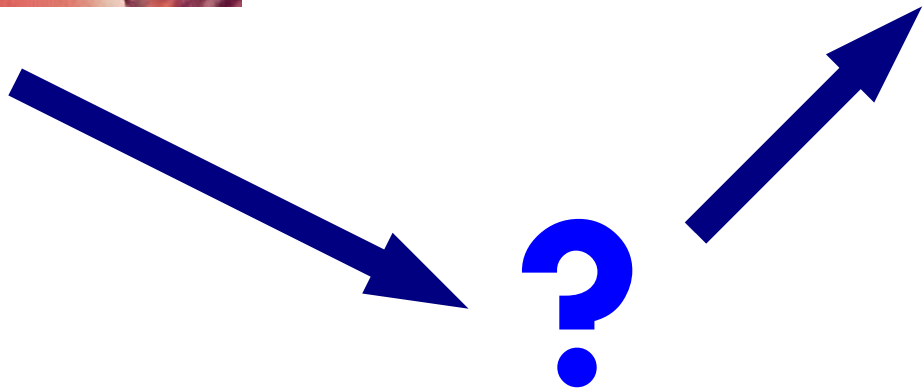
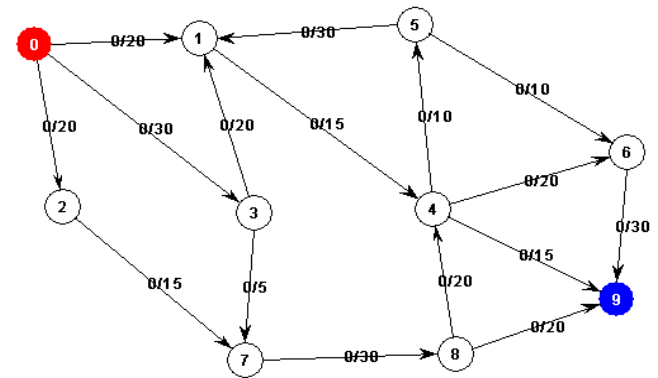
Computer Vision

(c) Sugano L



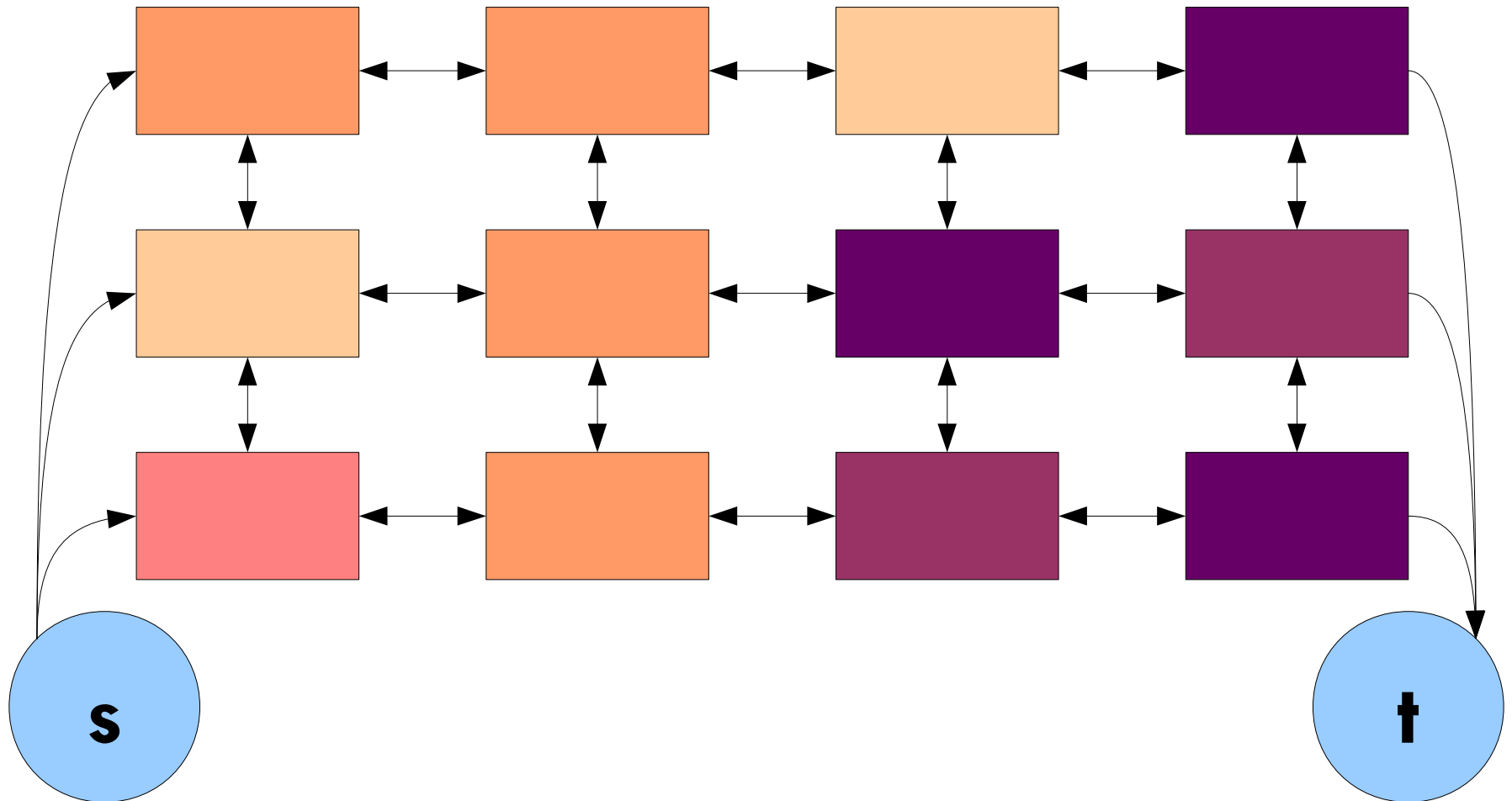
From Picture to Graph

14



From Picture to Graph

15

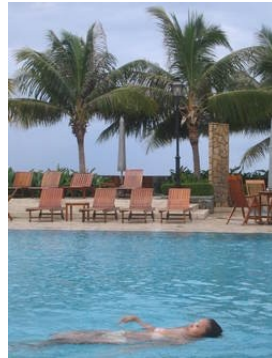


Seam Finding

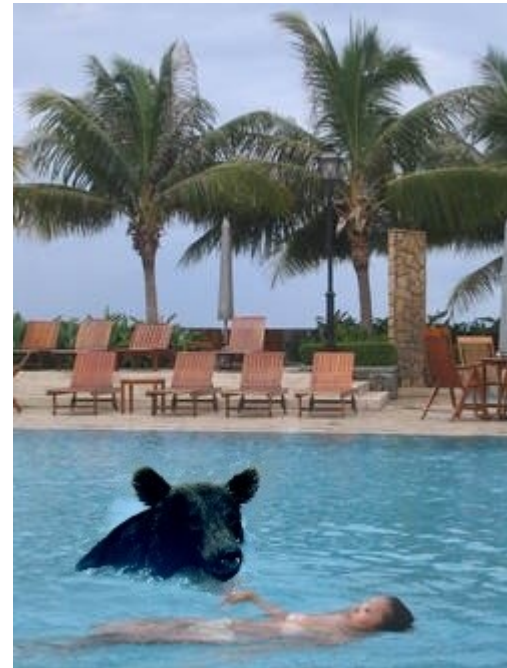
16



+



=



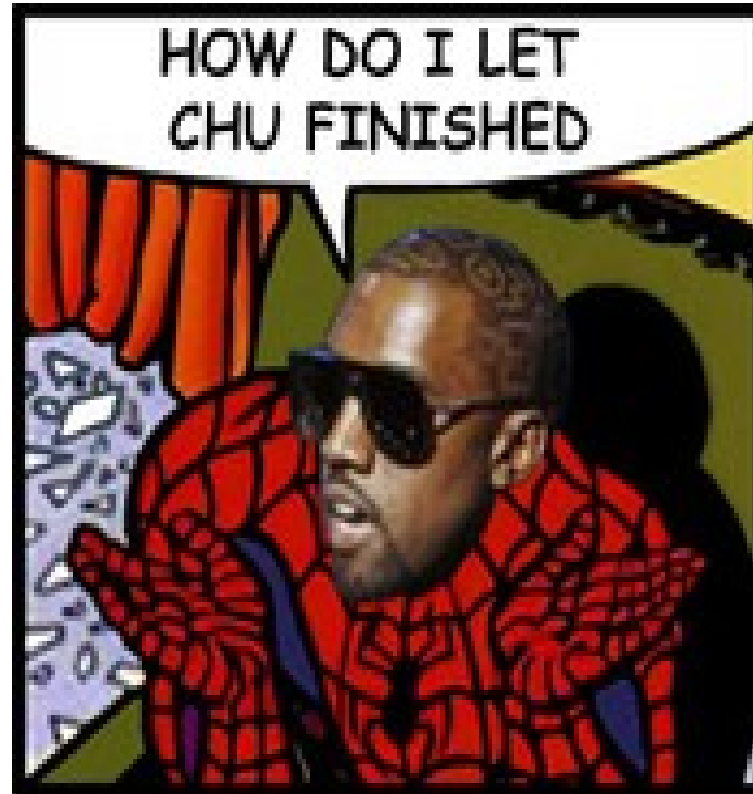
Seam Boundaries

Sample Texture



Questions?

17



References

18

http://de.wikipedia.org/wiki/Algorithmus_von_Ford_und_Fulkerson

<http://theorie.informatik.uni-ulm.de/Lehre/SS5/GA/FordFulkerson.pdf>

http://www.vuse.vanderbilt.edu/~mip-web/seminar_files/GraphCuts.ppt

Graph Cut Algorithms in Vision, Graphics and Machine Learning An Integrative Paper; Sinha et. al.

Graphcut Textures: Image and Video Synthesis Using Graph Cuts; Kwatra et. al.

Pictures:

<http://www.cs.brown.edu/courses/csci1950-g/results/proj2/edwallac/>

<http://www.chaoszone.de/blog/media/blogs/webnews/>

<http://www.in60.de/wp-content/gallery/2009-01-04-wassertropfen/>

<http://www.sugano.mech.waseda.ac.jp/main.html>