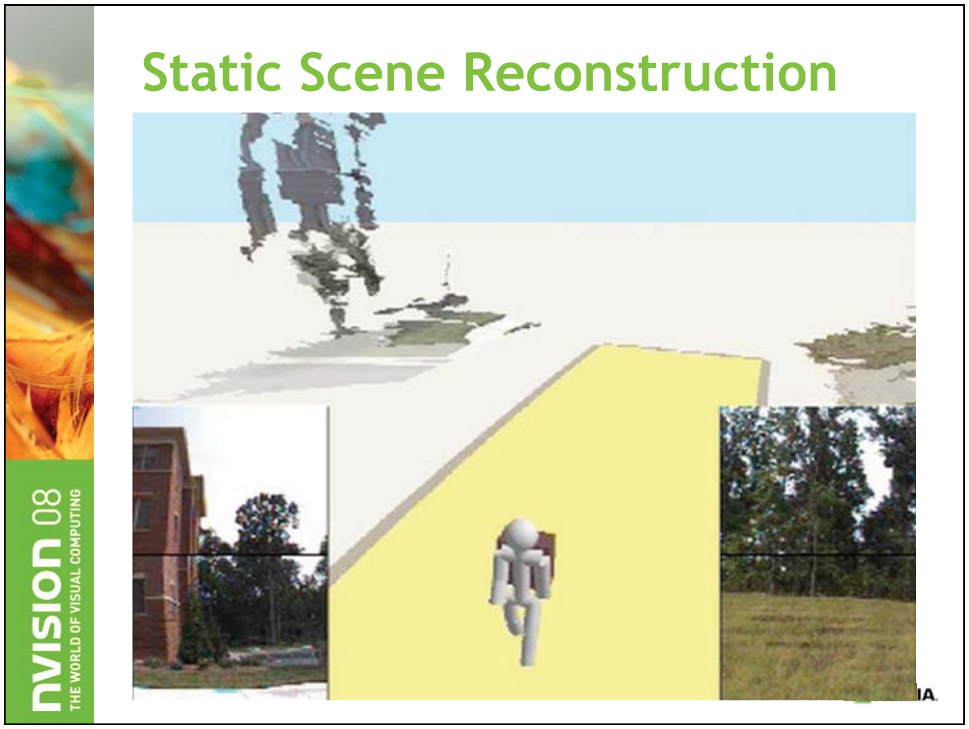

nVISION 08
THE WORLD OF VISUAL COMPUTING
GPU supported Real-Time Scene Reconstruction with a Single Camera
Jan-Michael Frahm, 3D Computer Vision group, University of North Carolina at Chapel Hill
NVIDIA


Static Scene Reconstruction
nVISION 08
THE WORLD OF VISUAL COMPUTING

Capture on campus & in downtown Chapel Hill

2x4 cameras,
1024x768@30Hz

GPS location &
orientation

~$(400 GPUs)

- Chapel Hill (15-30mph, 30fps)
- **2.6M** frames (8 cams, 3 hours)
- **2 TB** video data

NVISION 08 THE WORLD OF VISUAL COMPUTING

NVIDIA

Capture on campus & in downtown Chapel Hill

NVISION 08 THE WORLD OF VISUAL COMPUTING

NVIDIA

Capture on campus & in downtown Chapel Hill

2x4 cameras,
1024x768@30Hz

GPS location &
orientation

~$(400 GPUs)

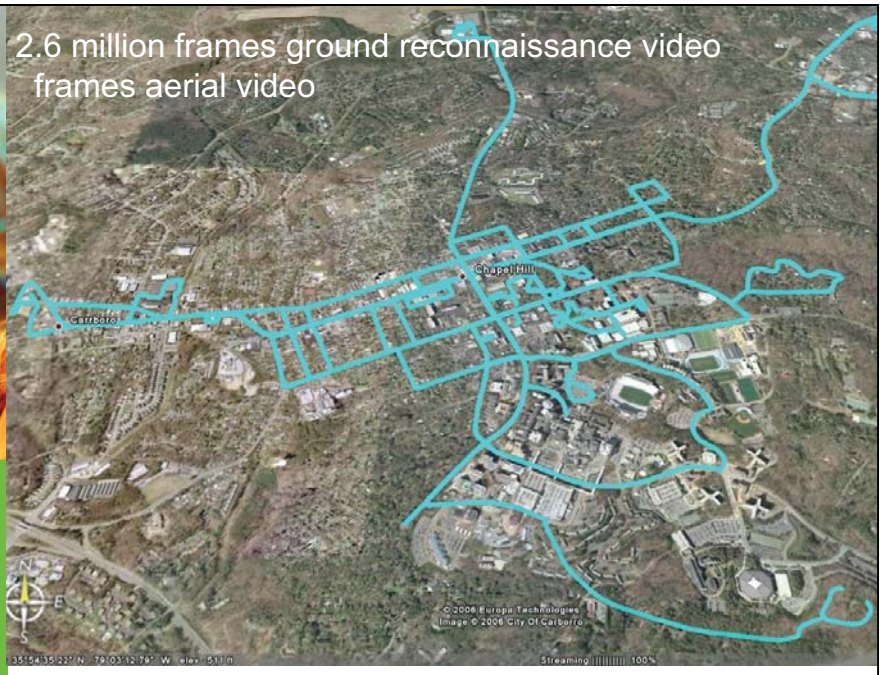
6 camera-head

GPS/Inertia sensors

NVIDIA

NVISION 08 THE WORLD OF VISUAL COMPUTING

2.6 million frames ground reconnaissance video frames aerial video

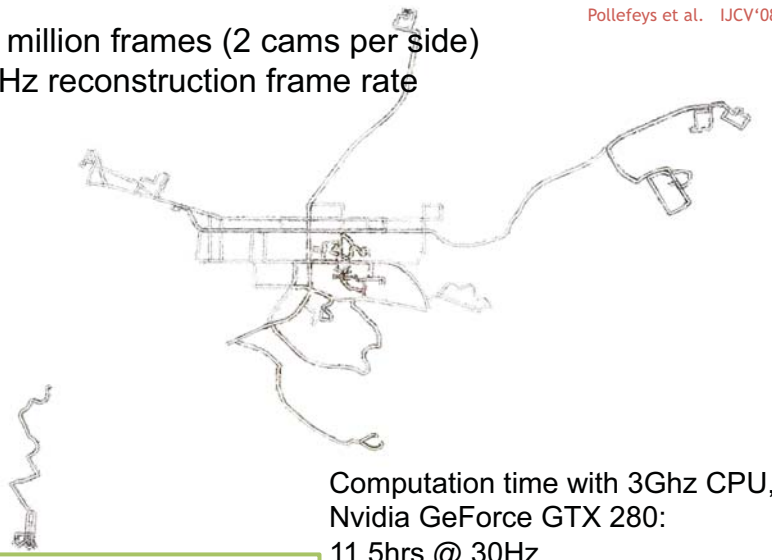


NVIDIA

NVISION 08 THE WORLD OF VISUAL COMPUTING

Pollefeys et al. IJCV'08

- 1.3 million frames (2 cams per side)
- 30 Hz reconstruction frame rate



Computation time with 3Ghz CPU, Nvidia GeForce GTX 280:
11.5hrs @ 30Hz
~2 weeks @ 1 Hz
~8 weeks @ 0.25 Hz

David Kirk:
Factor 100 is fundamentally different!

Speedup of ~120x

NVIDIA

3D Reconstruction Result

NVIDIA

3D model from video

Capture & 2D processing

3D model from video

Capture & 2D processing

images, 2D feature correspondences

3D model from video

Capture & 2D processing

images, 2D feature correspondences

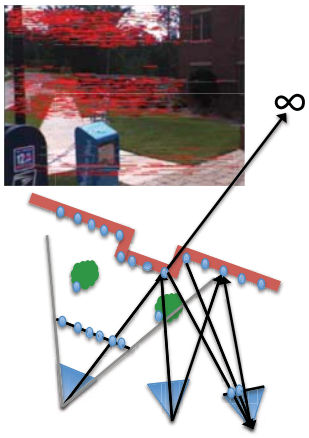
3D model from video

Capture & 2D processing

images, 2D feature correspondences

3D points & camera poses

images, camera poses



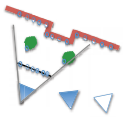
3D model from video

Capture & 2D processing

images, 2D feature correspondences

3D points & camera poses

images, camera poses



3D model from video

Capture & 2D processing

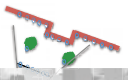
images, 2D feature correspondences

3D points & camera poses

images, camera poses

Dense 3D

images, camera poses, depth maps



The brighter the farther away

3D model from video

Capture & 2D processing

images, 2D feature correspondences

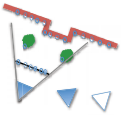
3D points & camera poses

images, camera poses

Dense 3D

images, camera poses, depth maps

Model generation



NVISION 08 THE WORLD OF VISUAL COMPUTING

NVIDIA

3D model from video

Capture & 2D processing

images, 2D feature correspondences

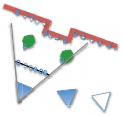
3D points & camera poses

images, camera poses

Dense 3D

images, camera poses, depth maps

Model generation



NVISION 08 THE WORLD OF VISUAL COMPUTING

3D model from video

Capture & 2D processing

images, 2D feature correspondences

3D points & camera poses

images, camera poses

Dense 3D

images, camera poses, depth maps

Model generation

GPU speedup ~190

GPU speedup ~140

NVISION 08 THE WORLD OF VISUAL COMPUTING

NVIDIA

3D model from video

Capture & 2D processing

images, 2D feature correspondences

3D points & camera poses

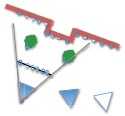
images, camera poses

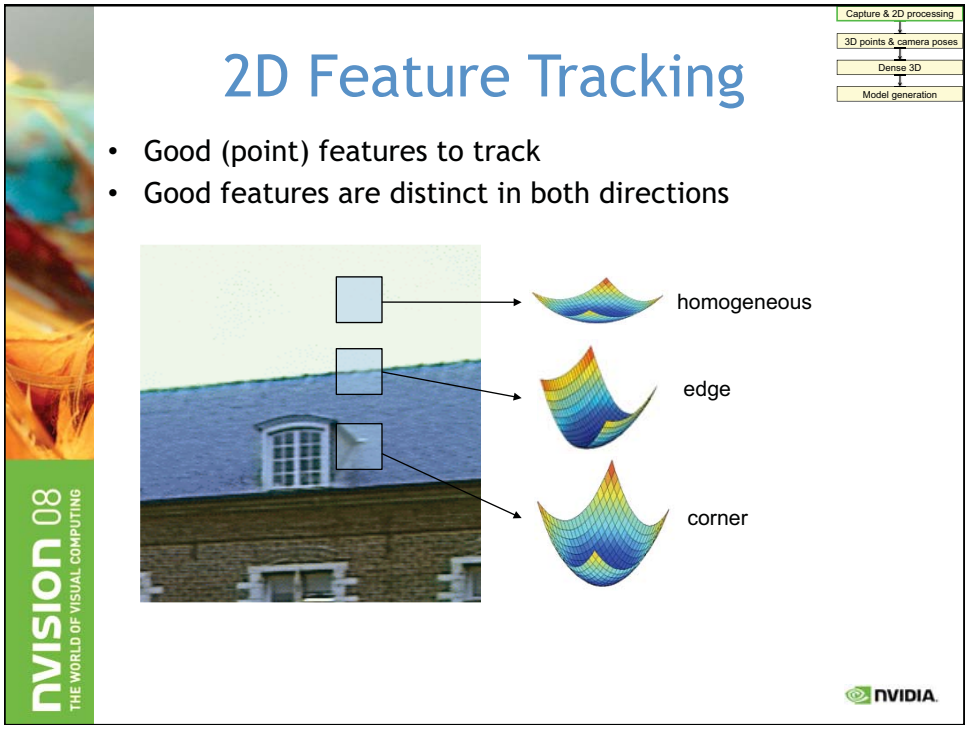
Dense 3D

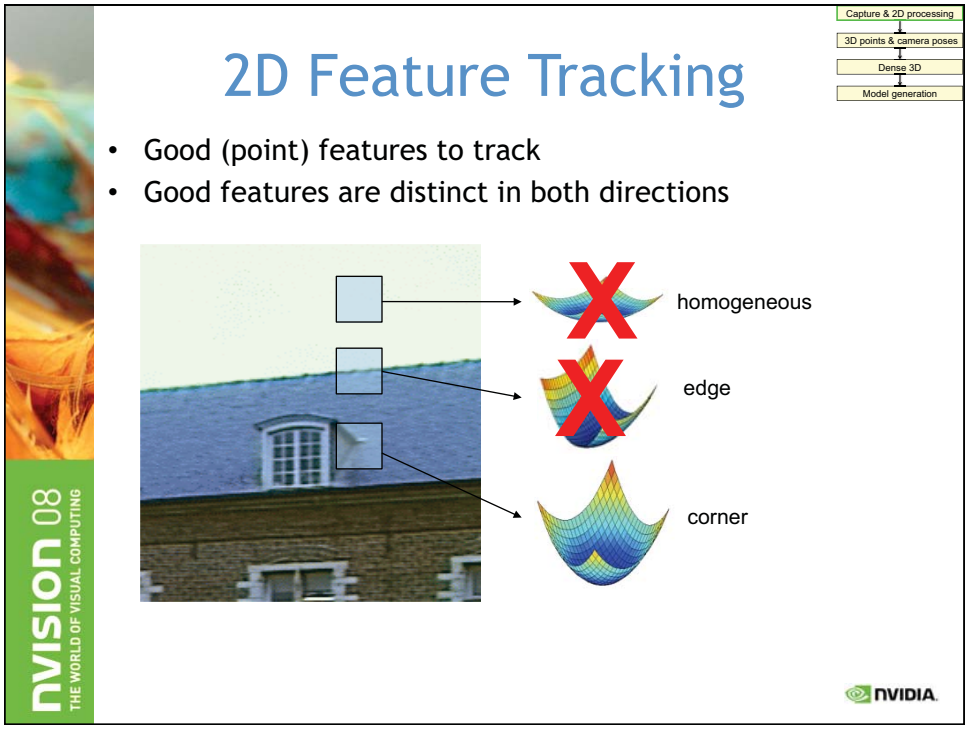
images, camera poses, depth maps

Model generation

NVISION 08 THE WORLD OF VISUAL COMPUTING



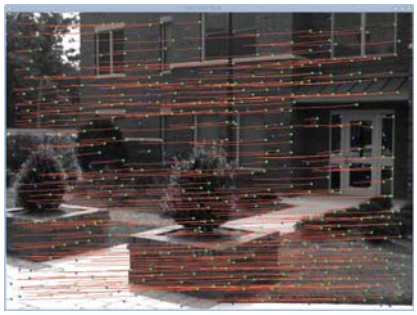

Capture & 2D processing
3D points & camera poses
Dense 3D
Model generation
2D Feature Tracking
• Good (point) features to track
• Good features are distinct in both directions
homogeneous
edge
corner
NVISION 08
THE WORLD OF VISUAL COMPUTING
NVIDIA


Capture & 2D processing
3D points & camera poses
Dense 3D
Model generation
2D Feature Tracking
• Good (point) features to track
• Good features are distinct in both directions
homogeneous
edge
corner
NVISION 08
THE WORLD OF VISUAL COMPUTING
NVIDIA

2D Feature Tracking

Capture & 2D processing
3D points & camera poses
Dense 3D
Model generation

- Establish correspondences between identical points in images
- Assumption: image content varies slowly/smoothly
  - Video sequences

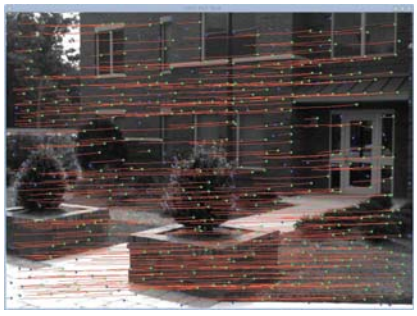
NVISION 08 THE WORLD OF VISUAL COMPUTING

NVIDIA

2D Feature Tracking

Capture & 2D processing
3D points & camera poses
Dense 3D
Model generation

- Establish correspondences between identical points in images
- Assumption: image content varies slowly/smoothly
  - Video sequences

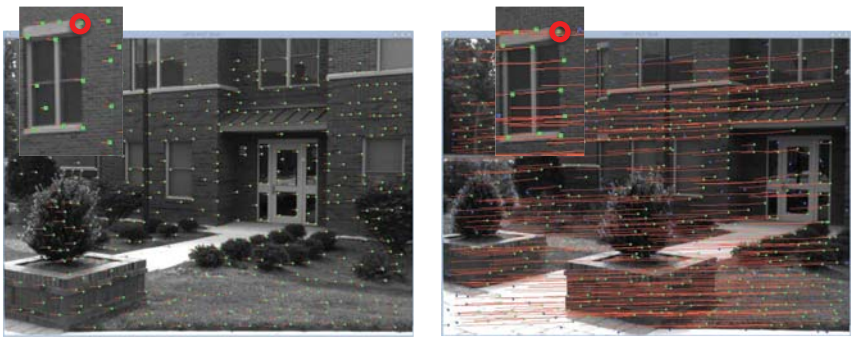
NVISION 08 THE WORLD OF VISUAL COMPUTING

NVIDIA

2D Feature Tracking

Capture & 2D processing
3D points & camera poses
Dense 3D
Model generation

- Establish correspondences between identical points in images
- Assumption: image content varies slowly/smoothly
  - Video sequences



NVISION 08 THE WORLD OF VISUAL COMPUTING

NVIDIA

Gain-Adaptive KLT-Tracking

Capture & 2D processing
3D points & camera poses
Dense 3D
Model generation

Kim et al. ICCV 2007

Video with fixed gain

- Simultaneous tracking and radiometric calibration
  - But: not data parallel – hard for GPU acceleration
- Continuous energy optimization [Zach et al. CVGPU'08]
  + Data parallel, very efficient on GPU

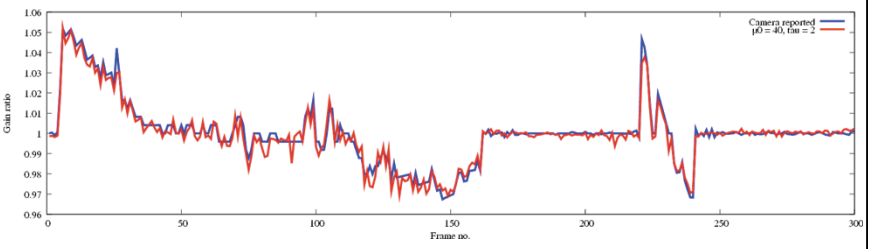
NVISION 08 THE WORLD OF VISUAL COMPUTING

NVIDIA

Gain Estimation

Capture & 2D processing
3D points & camera poses
Dense 3D
Model generation

- Camera reported (blue) and estimated gains (red)



NVISION 08 THE WORLD OF VISUAL COMPUTING

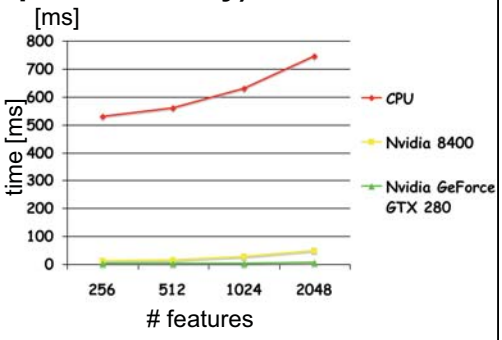
NVIDIA

GPU-KLT

Sinha, Frahm, Pollefeys and Genc, Machine, Vision and Applications 07
Zach, Gallup, Frahm CVGPU'08

Capture & 2D processing
3D points & camera poses
Dense 3D
Model generation

On CPU with 1000 features in 1024x768 video in 630 ms (Daimler only needs ~100ms on their bold machine without gain and no sub-pixel accuracy)



On GeForce GTX 280 3.3 ms! Speedup of ~190x (still ~ 30 compared to Daimlers CPU)

Code available at: www.cs.unc.edu/~cmzach/klt

NVISION 08 THE WORLD OF VISUAL COMPUTING

NVIDIA

3D model from video

Capture & 2D processing

images, 2D feature correspondences

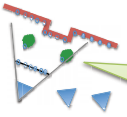
3D points & camera poses

images, camera poses

Dense 3D

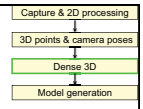
images, camera poses, depth maps

Model generation



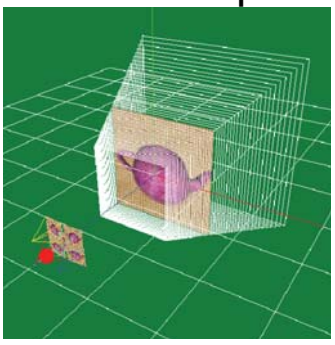
Research Roundtable: Computer Vision and Image Processing (1pm, SJCC J4)

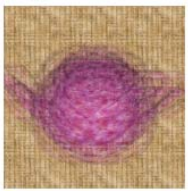
Fast GPU-based plane-sweeping stereo

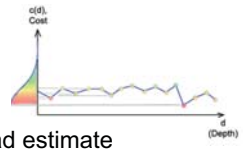


(Yang & Pollefeys, CVPR'03)

- Plane-sweep multi-view depth



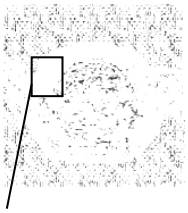




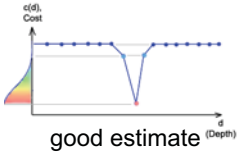
bad estimate

Confidence:

$$\left(\sum_{d \neq d_0} e^{-(\sigma(d) - \sigma(d_0))^2 / \sigma^2} \right)^{-1}$$



- For each plane similarity SSD = $(I_1-I_2)^2$
- Chose maximum similarity as best match



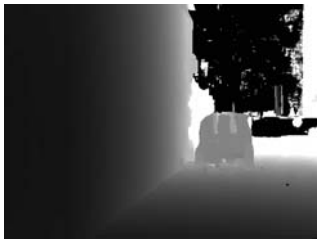
good estimate

NVIDIA

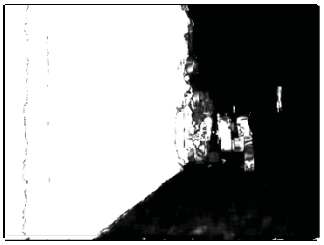
Confidence Results

Capture & 2D processing
3D points & camera poses
Dense 3D
Model generation

Captured Image



Stereo Depth Map



Confidence Map

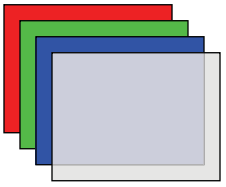
NVISION 08 THE WORLD OF VISUAL COMPUTING

NVIDIA

Compute SSAD per plane

Capture & 2D processing
3D points & camera poses
Dense 3D
Model generation

- 4 planes at a time, 1 plane per color channel



plane 0
plane 1
plane 2
plane 3

- Compute plane homographies
  - Warps matching view into reference view, as if it were on a given plane.
  - Uses projective texture mapping

NVISION 08 THE WORLD OF VISUAL COMPUTING

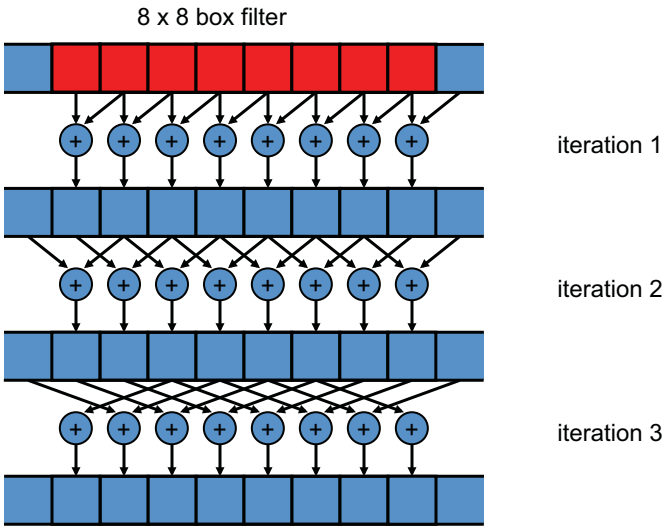
NVIDIA

Compute SSAD per plane

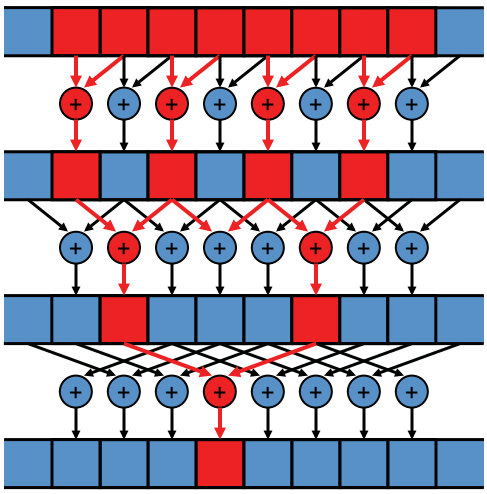
- SAD fragment shader:
  - N is compile-time constant

```
float4 sad = float4(0,0,0,0);
for(i=0; i<N; i++) {                              // N matching views
    float4 color;
    color[0] = tex2Dproj(tex[i],coord[i][0]);  // plane 0
    color[1] = tex2Dproj(tex[i],coord[i][1]);  // plane 1
    color[2] = tex2Dproj(tex[i],coord[i][2]);  // plane 2
    color[3] = tex2Dproj(tex[i],coord[i][3]);  // plane 3
    sad += abs(refColor-color);
}
```


Aggregate with box filter



Aggregate with box filter



iteration 1

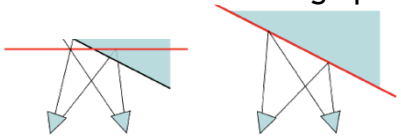
iteration 2

iteration 3

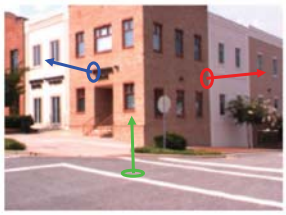
Multi-directional planes

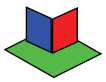
Gallup, Frahm, Mordohai, Pollefeys CVPR 07

- Improved stereo through plane alignment



- Urban scenes typically have three dominant surface normals

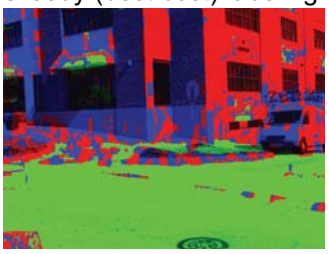




- Planes can be automatically estimated from planar camera motion assumption and vanishing points

Depth Results from Stereo

Greedy (best cost) labeling



NVIDIA

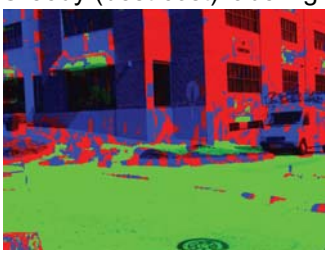
GPU-Based Global Labeling

Goal: assign a label to each pixel, such that

- The overall data (label) cost is small, and
- Neighboring pixels prefer the same label
  - Spatial regularization or smoothness
- Graph cut optimization 2s
- Continuous energy with Potts model 60 ms

Speedup ~33 on Nvidia GeForce 8800 GTX

Greedy (best cost) labeling

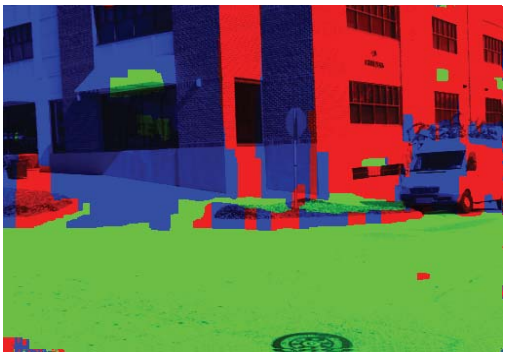


NVIDIA

Multi-way plane-sweep Stereo

Gallup, Frahm, Mordohai, Pollefeys CVPR 07

Capture & 2D processing
3D points & camera poses
Dense 3D
Model generation



Color labels correspond to selected sweeping direction

After label optimization

NVIDIA

NVISION 08 THE WORLD OF VISUAL COMPUTING

Stereo results

Capture & 2D processing
3D points & camera poses
Dense 3D
Model generation

Video side camera

Depth side camera

NVIDIA

NVISION 08 THE WORLD OF VISUAL COMPUTING

3D model from video

Capture & 2D processing

images, 2D feature correspondences

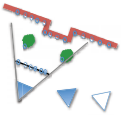
3D points & camera poses

images, camera poses

Dense 3D

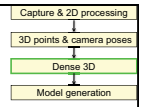
images, camera poses, depth maps

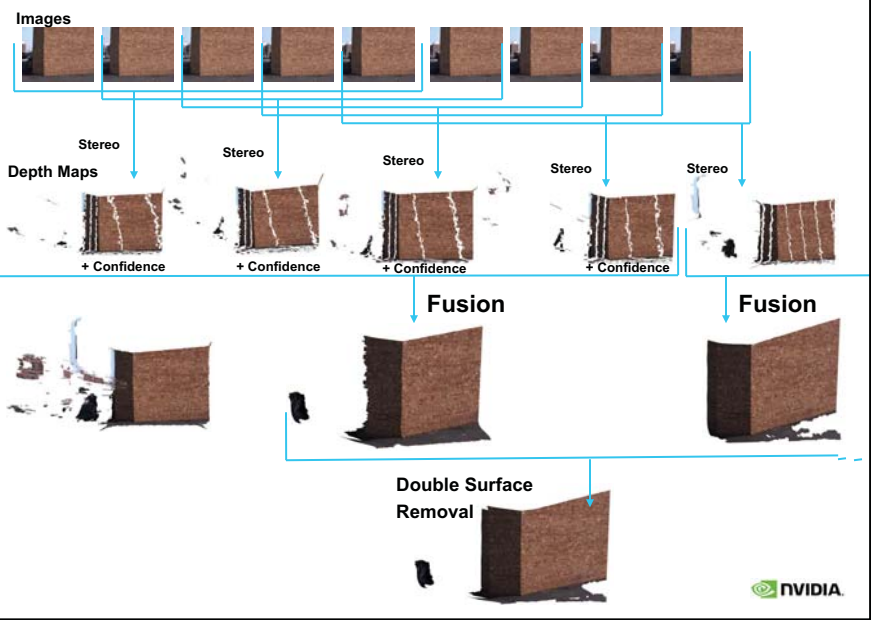
Model generation



Consensus Surface Fusion on GPU

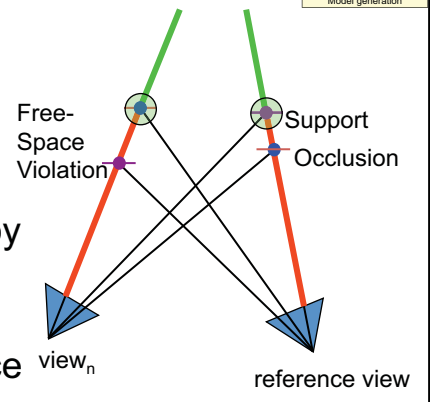
Merrel et al. ICCV'07



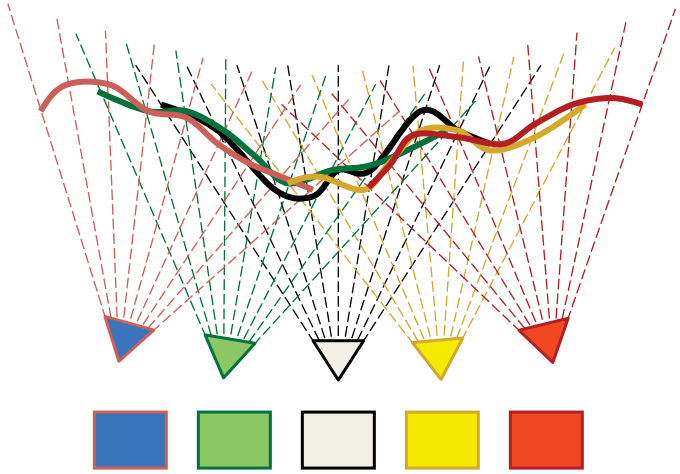


Visibility Relationships

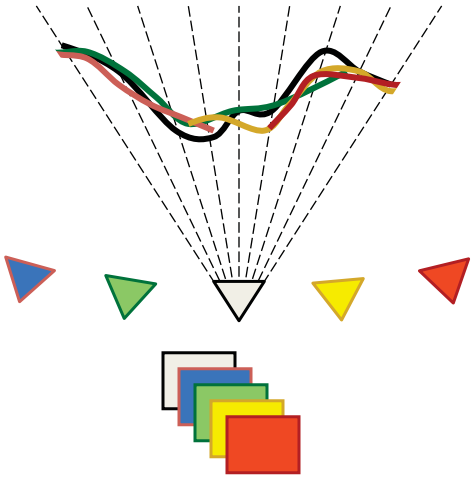
- Render all depth maps into reference view
- Choose most confident points along rays
- Add support from nearby points
- Remove support for occlusions and free-space violations
- Reject points with too little support



Render depthmaps to reference

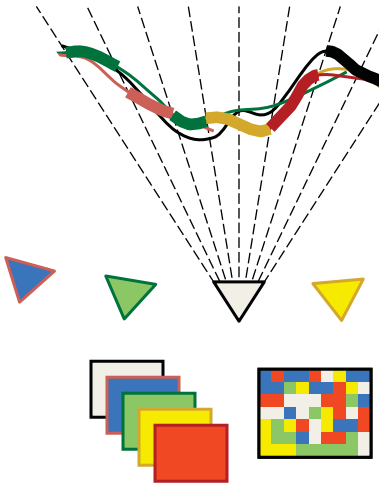


Render depthmaps to reference



- Point rendering
- Z buffer

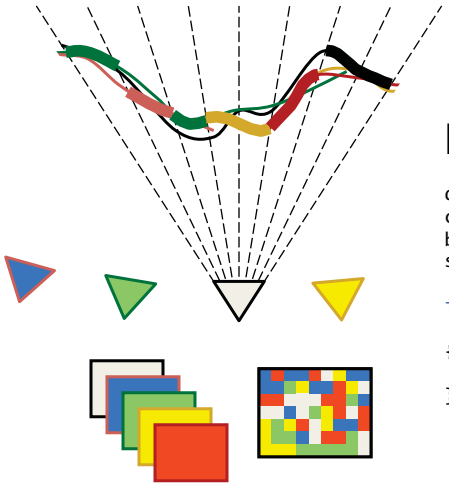
Choose most Confident



For each depthmap:

```
depth     = tex2D(tex,coord).x;
conf      = tex2D(tex,coord).y;
bestDepth = tex2D(texBest,coord).x;
bestConf  = tex2D(texBest,coord).y;

if(conf > bestConf)
{
    bestDepth = depth;
    bestConf  = conf;
}
OUT.color =
    float2(bestDepth,bestConf);
```

Add support from nearby points



Capture & 2D processing
3D points & camera poses
Dense 3D
Model generation

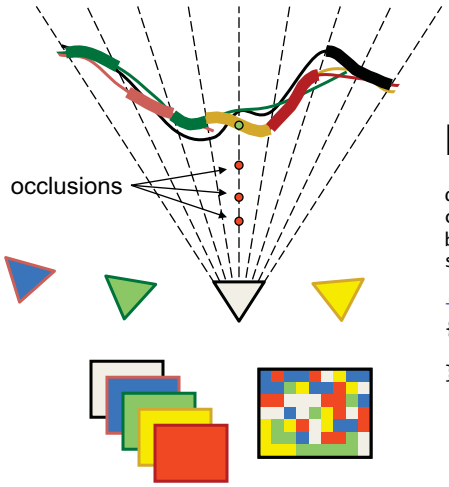
For each depthmap:

```
depth     = tex2D(tex,coord).x;
conf      = tex2D(tex,coord).y;
bestDepth = tex2D(texBest,coord).x;
support   = tex2D(texBest,coord).z;

if(depth < bestDepth*1.05
   && depth > bestDepth*0.95)
{
    support += conf;
}
```

NVISION 08 THE WORLD OF VISUAL COMPUTING

NVIDIA

Remove support for occlusions



Capture & 2D processing
3D points & camera poses
Dense 3D
Model generation

For each depthmap:

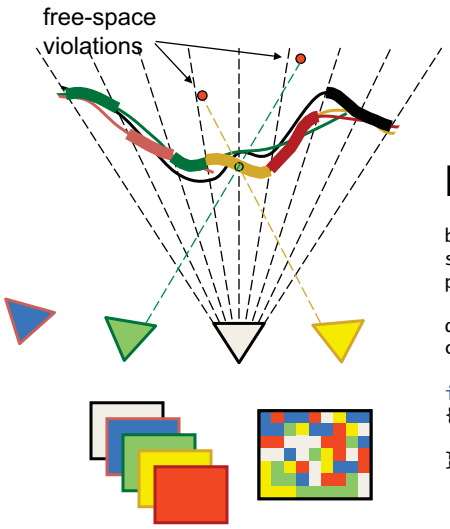
```
depth     = tex2D(tex,coord).x;
conf      = tex2D(tex,coord).y;
bestDepth = tex2D(texBest,coord).x;
support   = tex2D(texBest,coord).z;

if(depth < bestDepth*0.95)
{
    support -= conf;
}
```

NVISION 08 THE WORLD OF VISUAL COMPUTING

NVIDIA

Remove support for free-space violations

Capture & 2D processing
3D points & camera poses
Dense 3D
Model generation



For each depthmap:

```
bestDepth = tex2D(texBest,coord).x;
support   = tex2D(texBest,coord).z;
pt = mul(P,float4(coord.x,coord.y
                  bestDepth,1));
depth = tex2Dproj(tex,pt).x;
conf  = tex2Dproj(tex,pt).y;

if(depth > pt.z*1.05)
{
    support -= conf;
}
```

points with too little support are then rejected

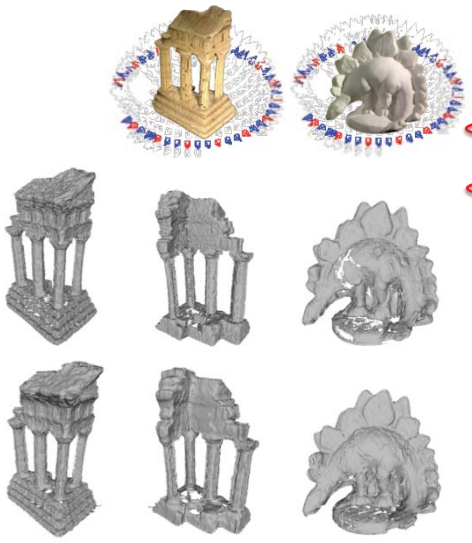
NVIDIA

NVISION 08 THE WORLD OF VISUAL COMPUTING

Middlebury 3D Evaluation

Merrell et al., ICCV 07 & VRML 07

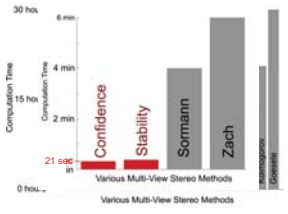
Capture & 2D processing
3D points & camera poses
Dense 3D
Model generation



Ring datasets: 47 images

	Accuracy	Completeness	Time
TEMPLE			
Stability	0.85 mm	87.0%	22 sec
Confidence	0.76 mm	85.2%	21 sec
DINO			
Stability	0.73 mm	73.1%	28 sec
Confidence	0.84 mm	83.1%	21 sec

Results competitive but much, much faster (30 hours → 30 seconds)



NVIDIA

NVISION 08 THE WORLD OF VISUAL COMPUTING

Example Curb Reconstruction

Capture & 2D processing
3D points & camera poses
Dense 3D
Model generation

Courtesy Daimler

Example Curb Reconstruction

Capture & 2D processing
3D points & camera poses
Dense 3D
Model generation

NVISION 08 THE WORLD OF VISUAL COMPUTING
NVIDIA

3D model from video

Capture & 2D processing
3D points & camera poses
Dense 3D
Model generation

Capture & 2D processing

images, 2D feature correspondences

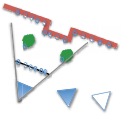
3D points & camera poses

images, camera poses

Dense 3D

images, camera poses, depth maps

Model generation



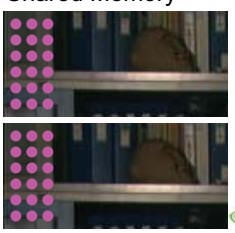
CUDA Stereo

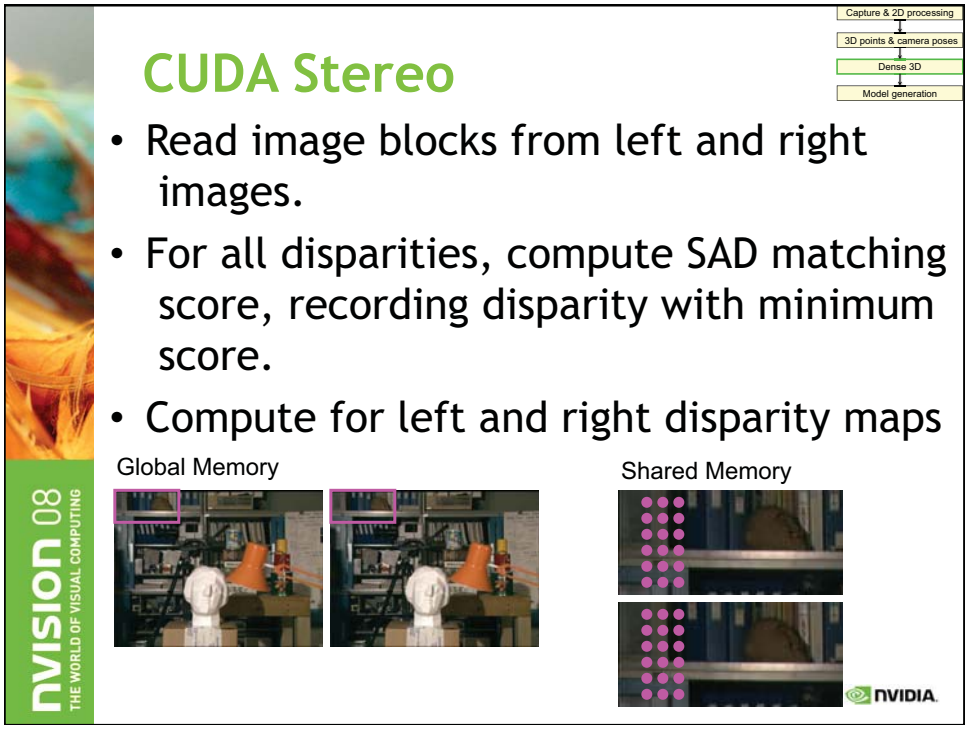
Capture & 2D processing
3D points & camera poses
Dense 3D
Model generation

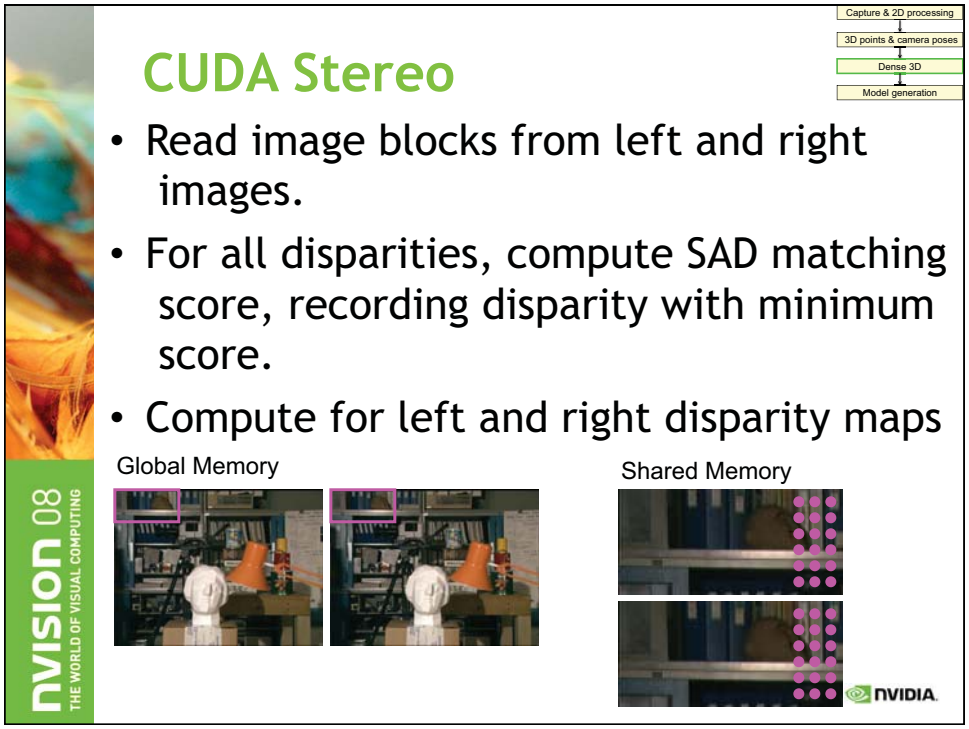
- Read image blocks from left and right images.
- For all disparities, compute SAD matching score, recording disparity with minimum score.
- Compute for left and right disparity maps

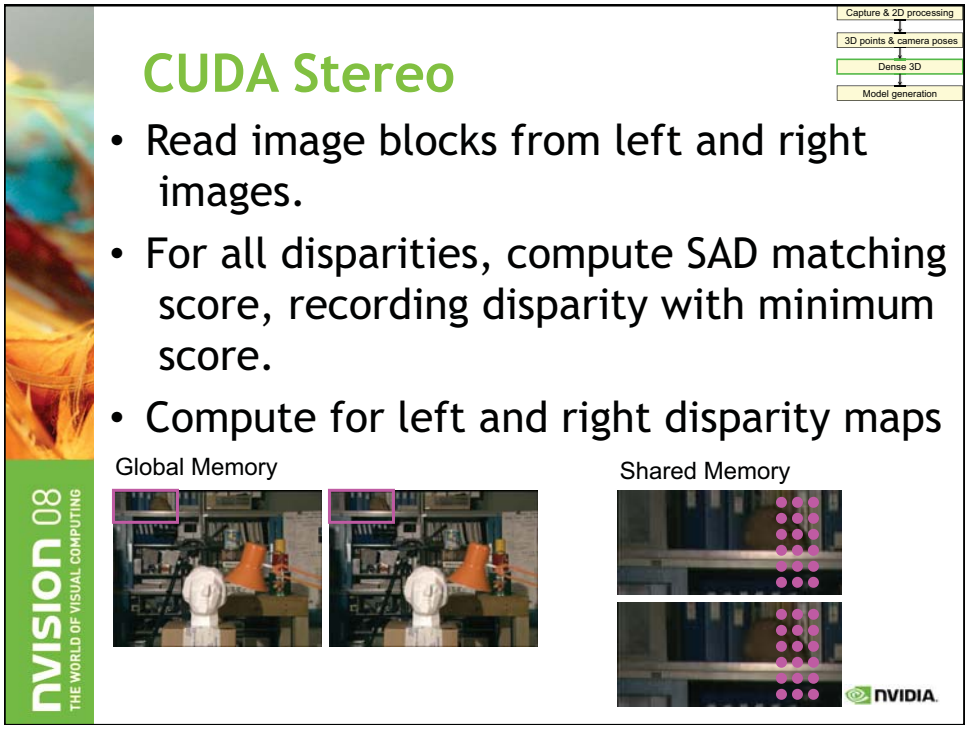
Global Memory

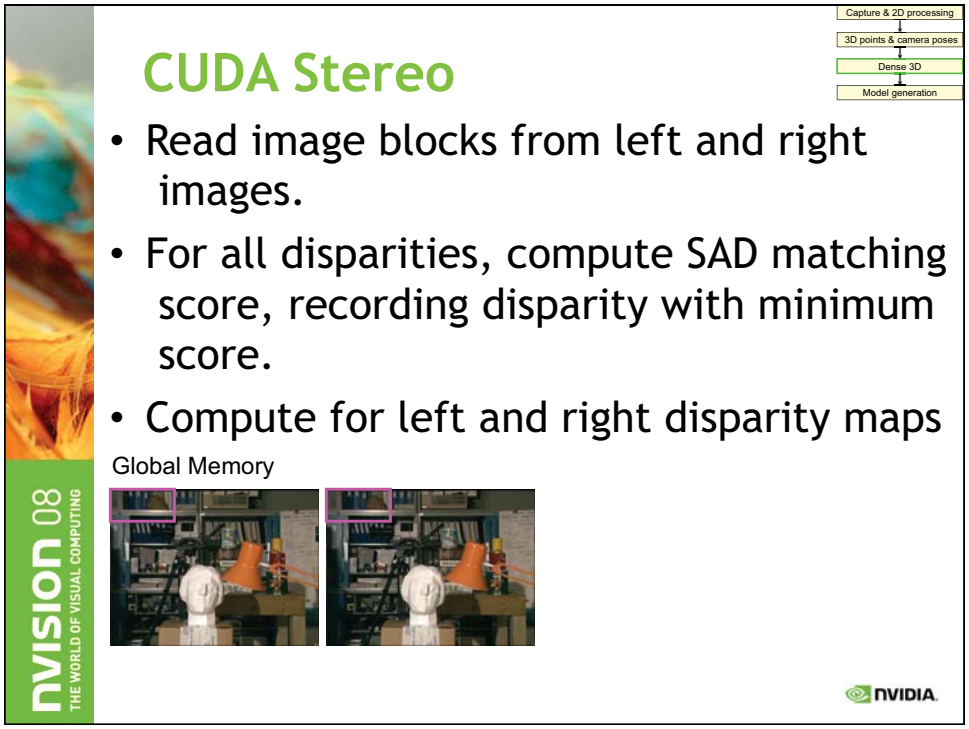
Shared Memory

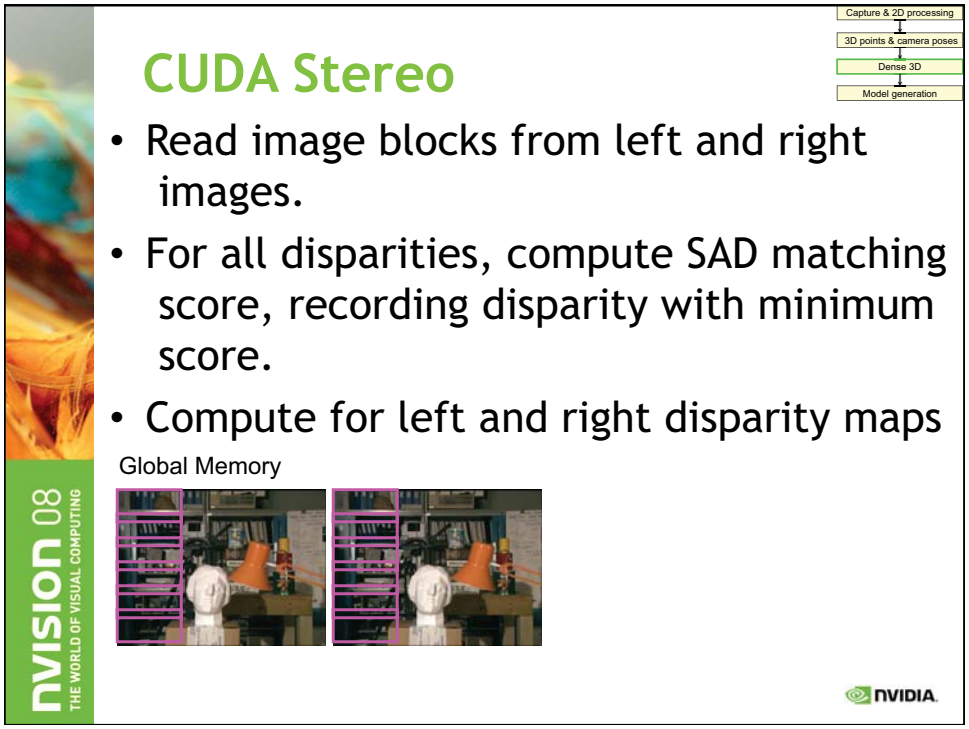


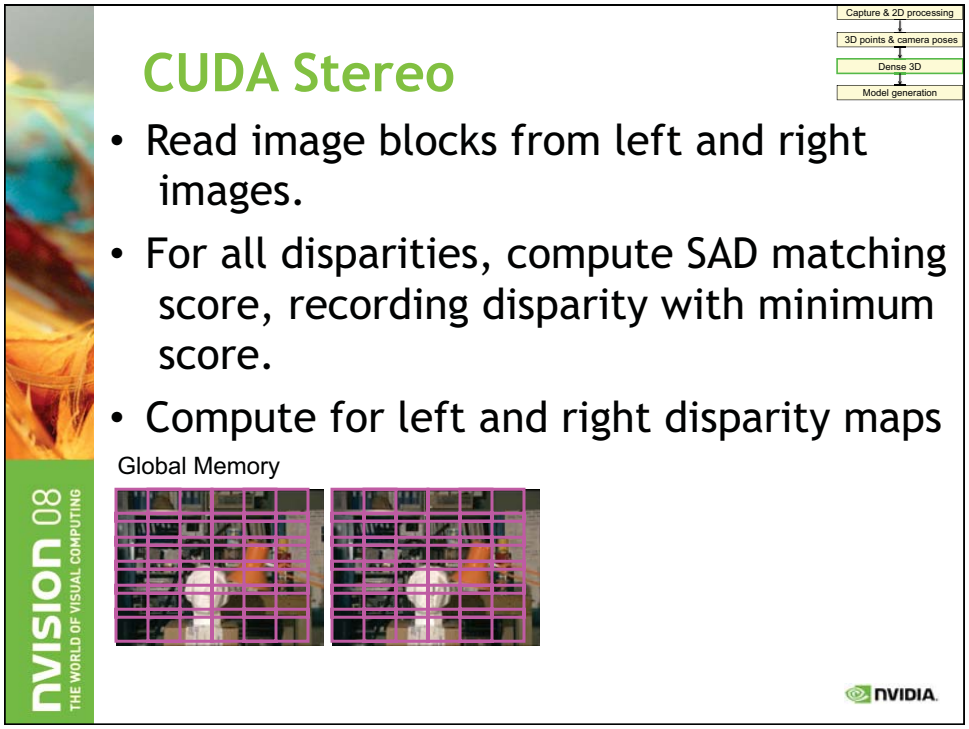

Capture & 2D processing
3D points & camera poses
Dense 3D
Model generation
CUDA Stereo
• Read image blocks from left and right images.
• For all disparities, compute SAD matching score, recording disparity with minimum score.
• Compute for left and right disparity maps
Global Memory
Shared Memory
NVISION 08
THE WORLD OF VISUAL COMPUTING
NVIDIA


Capture & 2D processing
3D points & camera poses
Dense 3D
Model generation
CUDA Stereo
• Read image blocks from left and right images.
• For all disparities, compute SAD matching score, recording disparity with minimum score.
• Compute for left and right disparity maps
Global Memory
Shared Memory
NVISION 08
THE WORLD OF VISUAL COMPUTING
NVIDIA


Capture & 2D processing
3D points & camera poses
Dense 3D
Model generation
CUDA Stereo
• Read image blocks from left and right images.
• For all disparities, compute SAD matching score, recording disparity with minimum score.
• Compute for left and right disparity maps
Global Memory
Shared Memory
NVISION 08
THE WORLD OF VISUAL COMPUTING
NVIDIA


Capture & 2D processing
3D points & camera poses
Dense 3D
Model generation
CUDA Stereo
• Read image blocks from left and right images.
• For all disparities, compute SAD matching score, recording disparity with minimum score.
• Compute for left and right disparity maps
Global Memory
NVISION 08
THE WORLD OF VISUAL COMPUTING
NVIDIA


Capture & 2D processing
3D points & camera poses
Dense 3D
Model generation
CUDA Stereo
• Read image blocks from left and right images.
• For all disparities, compute SAD matching score, recording disparity with minimum score.
• Compute for left and right disparity maps
Global Memory
NVISION 08
THE WORLD OF VISUAL COMPUTING
NVIDIA


Capture & 2D processing
3D points & camera poses
Dense 3D
Model generation
CUDA Stereo
• Read image blocks from left and right images.
• For all disparities, compute SAD matching score, recording disparity with minimum score.
• Compute for left and right disparity maps
Global Memory
NVISION 08
THE WORLD OF VISUAL COMPUTING
NVIDIA

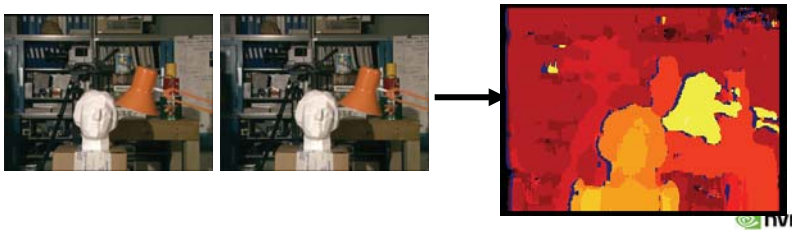
CUDA Stereo

Capture & 2D processing
3D points & camera poses
Dense 3D
Model generation

- Detect occlusions and outliers by left /right depthmap consistency check.
  - If pixel x in left image corresponds to x' in right image, then x' should correspond to x, or one of its neighbors.

$$\left|D_L(x) - D_R(x + D_L(x))\right| \le C$$

- **10% speedup compared to OpenGL**



NVISION 08 THE WORLD OF VISUAL COMPUTING

NVIDIA

Two-view Stereo Results

Capture & 2D processing
3D points & camera poses
Dense 3D
Model generation

NVISION THE WORLD OF VISUA

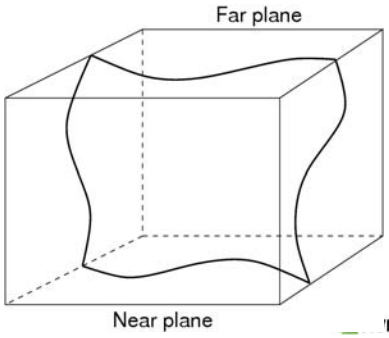
NVIDIA

Global Stereo

Capture & 2D processing
3D points & camera poses
Dense 3D
Model generation

- Shares some similarity with global labeling
  - More labels (all possible depth values)
  - Embedded in higher dimension
- Find surface separating the near from the far plane
  - With minimal costs
    - Data cost (NCC), and
    - Smoothness cost
  - Local updates of values according to PDEs
  - Data parallel!

GPU speedup 80x
but still 4s!



NVISION 08 THE WORLD OF VISUAL COMPUTING

NVIDIA

Global Stereo results

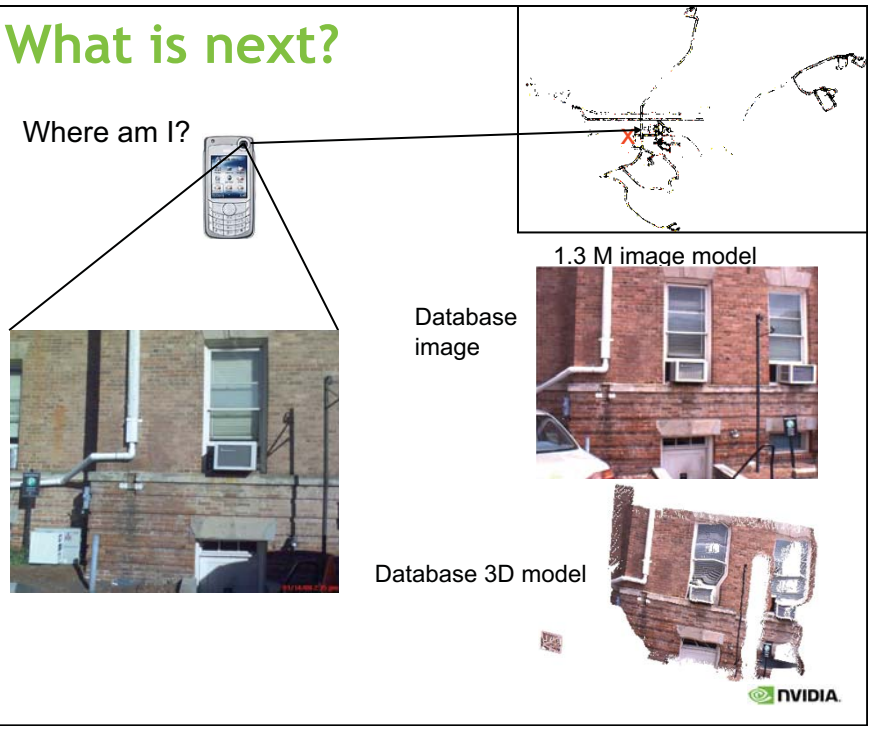
NVISION THE WORLD OF VISUAL COMPUTING

NVIDIA

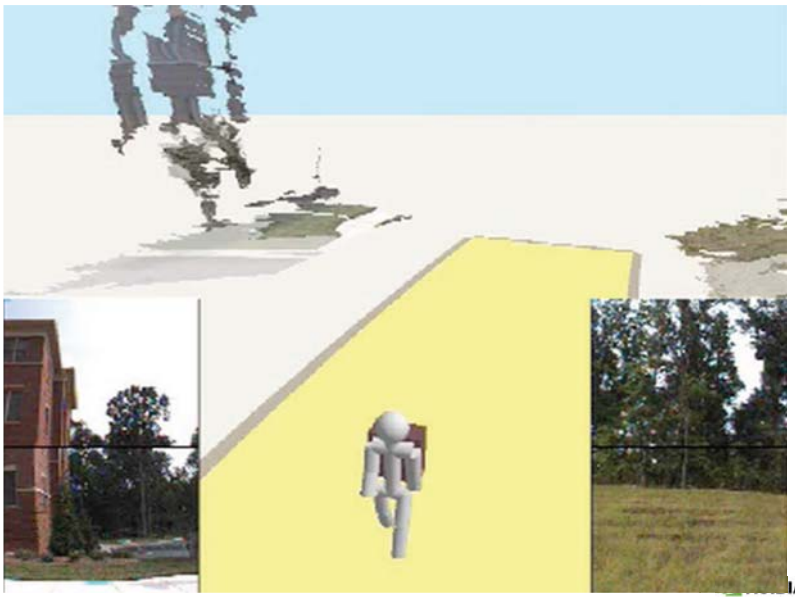
Conclusions

- GPU based real-time scene reconstruction system
  - 2D tracking on GPU
  - Stereo estimation on GPU
  - Dense map fusion on GPU
- Algorithmic redesign to better deploy the GPU

What is next?



Static Scene Reconstruction



NVISION 08 THE WORLD OF VISUAL COMPUTING

Thank you!

NVIDIA

NVISION 08 THE WORLD OF VISUAL COMPUTING