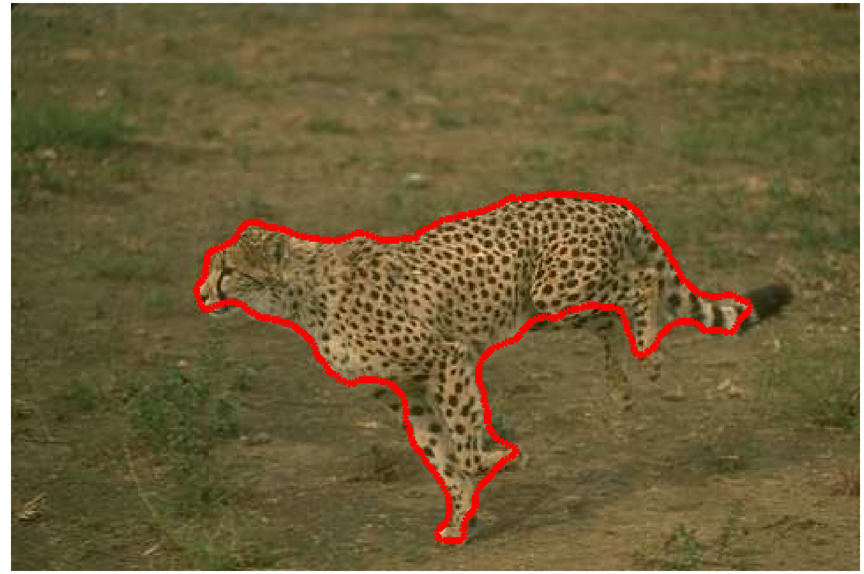
Project Presentations for the GPU Course

Thomas Möllenhoff,Mohamed Souiai,Rudolph Triebel, Emanuele Rodula , Michael Moeller, Jakob Engel, Caner Hazirbas, Jan Stühmer

Unsupervised Figure-Ground Segmentation



Solve a following variational formulation of the following problem:

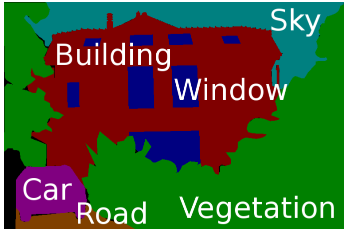
$$\min_{F \subset \Omega} \mathrm{Per}(F;\Omega) - \lambda \mathcal{D}(P_F, P_G)$$

Goal: Solve above problem in parallel using a GPU in order to accelerate the algorithm towards unsupervised real time segmentation.

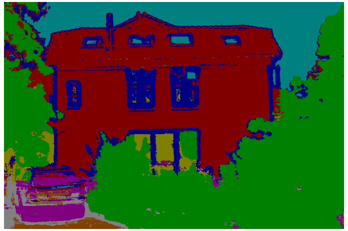
Random Forests on GPU
Caner Hazirbas

Research Interest : **Object Detection/Recognition,**
Semantic Scene Understanding
Deep Learning

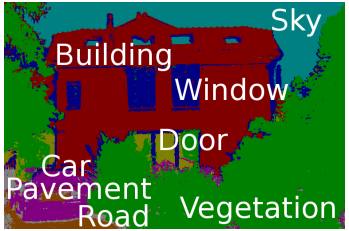
Input Image



Ground Truth



Pixel-wise Recognition

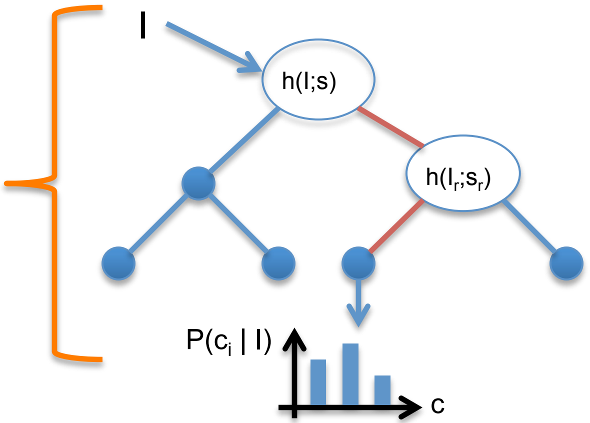


Segmented Image

Caner Hazirbas c.hazirbas@tum.de / room : 02.09.041

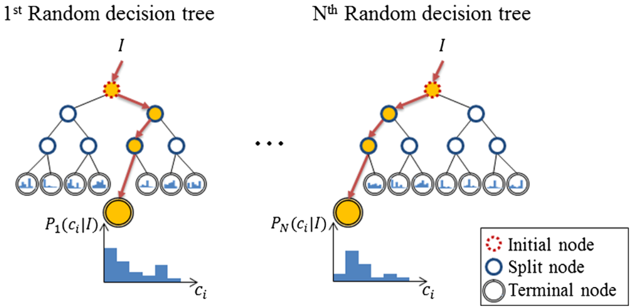
Random Forests

- ✧ Ensemble of Decision Trees
- ✧ Capable of Multi-Class Object Recognition
- ✧ Fast Classification
- ✧ Can be parallelizable on GPU
  - each tree
  - each node at same level



$I_1 = \{f_1,..,f_i,..,f_n \,;\, c^1\}$
$I_2 = \{f_1,..,f_i,..,f_n \,;\, c^2\}$
$I_m = \{f_1,..,f_i,..,f_n \,;\, c^m\}$

I ; training data
h ; split function (weak learner)
s ; split parameters
m; number of samples
c ; class label of sample
i ; class index



Prediction :

$$\max_c \frac{1}{N} \sum_{n=1}^{N} P_n(c|I)$$

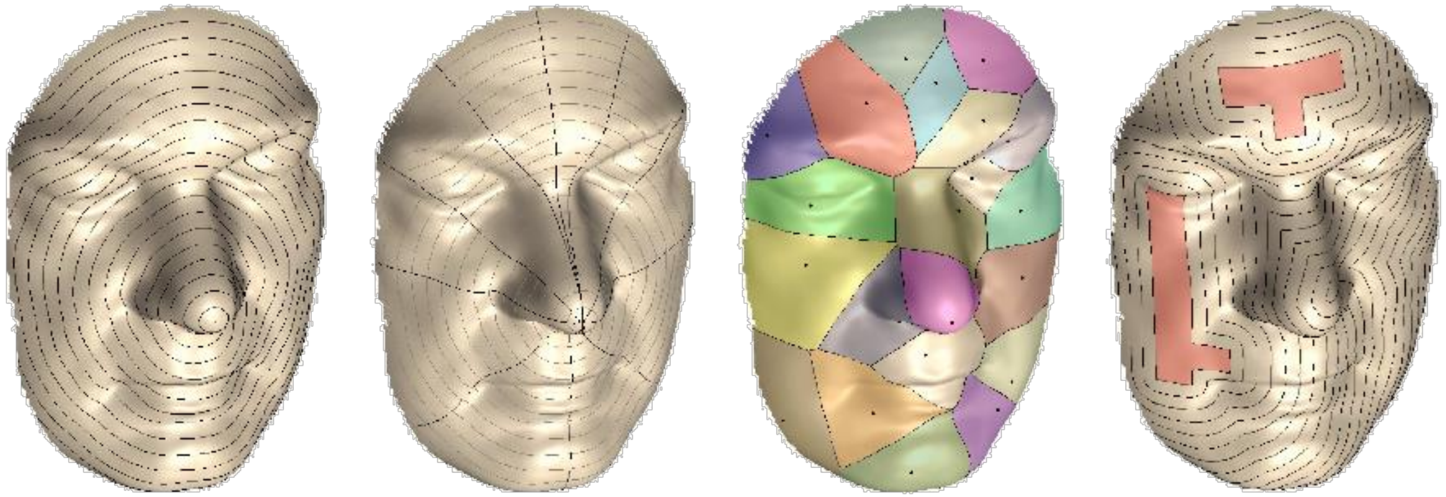
*http://opticalengineering.spiedigitallibrary.org/article.aspx?articleid=1653276

Caner Hazirbas
c.hazirbas@tum.de / room : 02.09.041

APPROXIMATED GEODESICS ON CURVED SURFACES

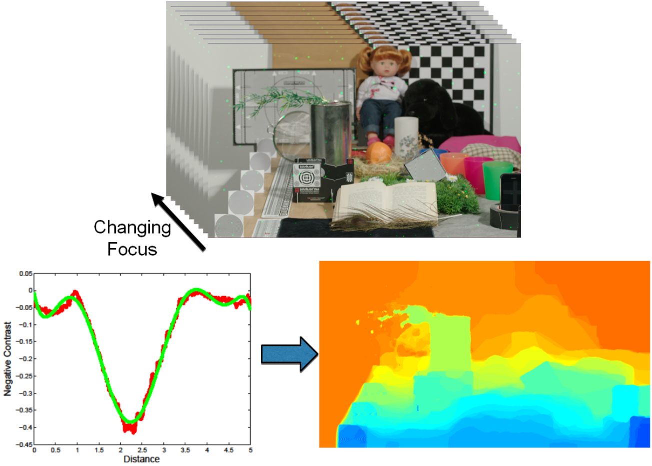
Problem: Compute geodesic distances and paths on arbitrary 3D surfaces.

Goal: Implement the state-of-the-art Parallel Marching method described in the paper.



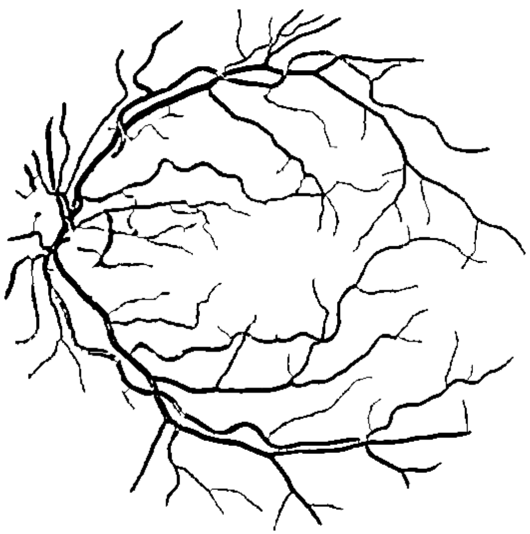
(trailer time!)

Reconstruct a depth map from differently focused images

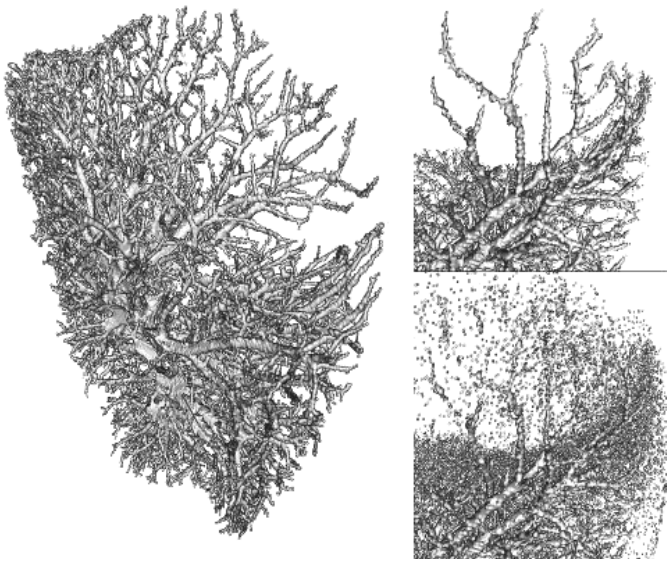


$$\min_{d} \int_{\Omega} -C(d(x,y)) + |\nabla d(x)| \; dx \, dy$$

Connectivity Constraints in Image Segmentation



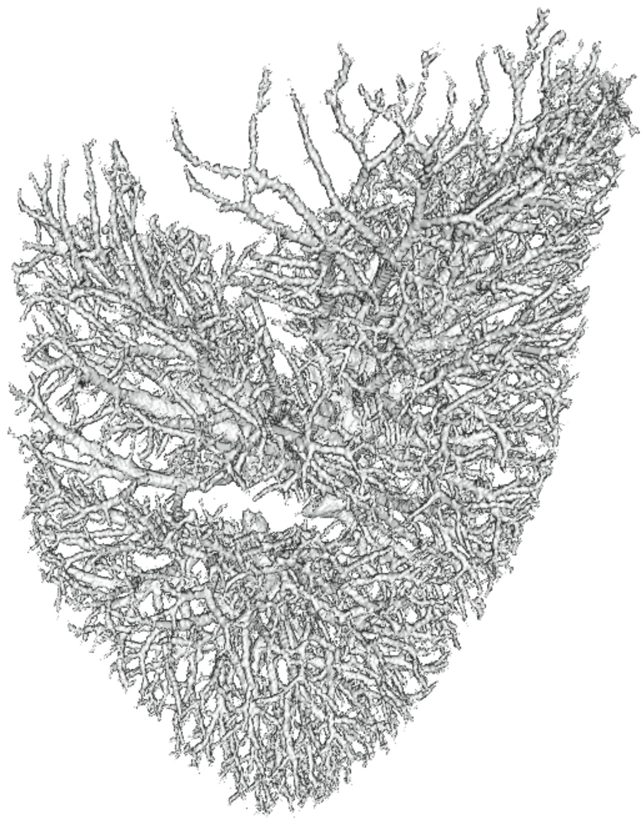
Retinal Blood Vessels



3D CT Angiography of the Lung

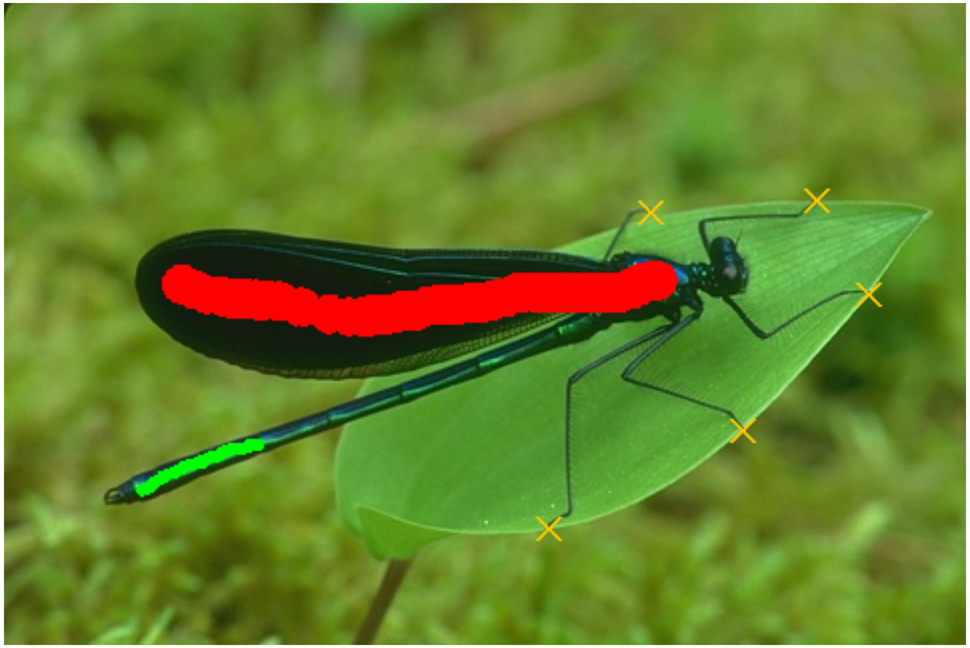
Connectivity Constraints in Image Segmentation

Connectivity Constraints in Image Segmentation



Interactive Segmentation

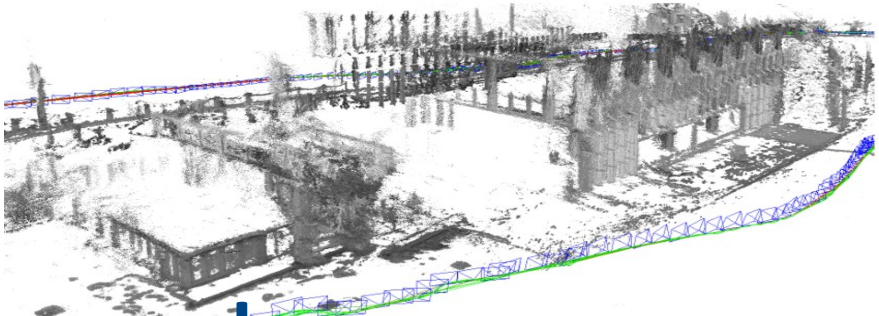
Interactive Segmentation



Interactive Segmentation

Interactive Segmentation

Porting LSD-SLAM to GPU



Large-Scale

Monocular

Direct

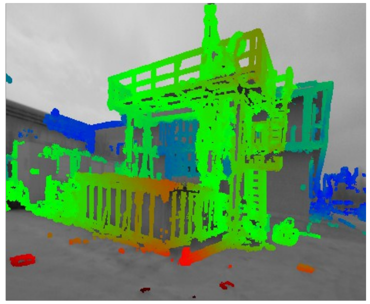
Project Proposal:

Port part of
LSD-SLAM to the GPU.

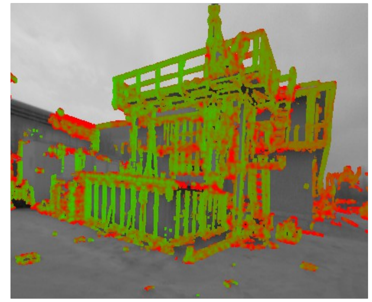
https://github.com/tum-vision/lsd_slam

Semi-Dense Depth map for Image Tracking

image



inverse depth



inverse depth variance

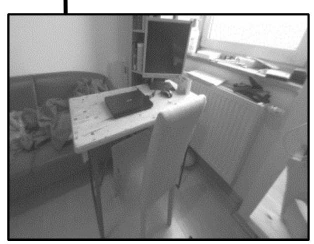
Depth estimate

- **Gaussian inverse depth coding**
- **Triangulation and Pixelwise filtering**
- **Information selection**

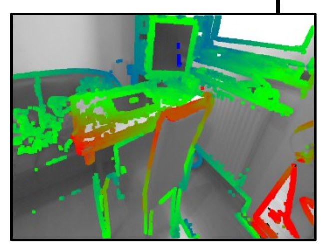
- **Tracking**: Minimize photometric error using **Gauss-Newton**

$$E(\boldsymbol{\xi}) = \sum_{\mathbf{x}\in\Omega_{\mathrm{KF}}} \left(I_{\mathrm{KF}}(\mathbf{x}) - I(\omega(\mathbf{x}, D_{\mathrm{KF}}(\mathbf{x}), \boldsymbol{\xi}))\right)^2 =: \|\mathbf{r}(\boldsymbol{\xi})\|_2^2$$

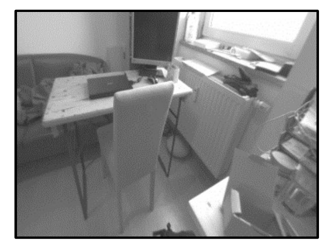
Camera Pose in $\mathfrak{se}(3)$



KF image



KF depth



back-warped new frame

Do it on GPU !

- Embarassingly Parrallel workload (Lot of independant projections for computing error)
- Real-time requirement limit us to 320x240 images on CPU.
- Heavy task in LSD-SLAM (used for finding loop closure constraints)
- Computationnal needs become more important with complicated projection functions.
  - (Omnidirectional Camera...)

So, why using a GPU here ?

- It still needs a powerful CPU
- Speed is robustness
- ➢ Would help to offload CPU
- ➢ Could go embedded ...

Why would YOU want to do that ?

- Some great improvement should be doable in three weeks.
- You will integrate your code in a state of the art system.
  - (ICCV 2013, ECCV 2014, ISMAR 2014,...)
- You could end with the fastest dense SLAM system existing around.
- It might be possible, and fun, to try to port it on Tango tablet if you are still interested afterward. Yes the lab has two of them (: