3D Photography

Marc Pollefeys and Torsten Sattler

Spring 2015
3D Photography

• Understanding geometric relations between images and the 3D world, as well as between images
• Obtaining 3D information from our 3D world
Motivation

- Applications in many different areas
- A few examples ...
Architecture

Survey
Stability analysis
Plan renovations
Architecture

Survey
Stability analysis
Plan renovations
Interactive 3D Modeling

(Sinha et al. Siggraph Asia 08)
collaboration with Microsoft Research (and licensed to MS)
Interactive 3D Architectural Modeling from Unordered Photo Collections

Paper # 0062
3D urban modeling

UNC/UKY UrbanScape project
Robot navigation

ESA project
our task: Calibration + Terrain modelling + Visualization
Self-Driving Cars
Self-Driving Cars

Infrastructure-Based Calibration of a Multi-Camera Rig

Lionel Heng\textsuperscript{1}, Mathias Bürki\textsuperscript{2}, Gim Hee Lee\textsuperscript{1},
Paul Furgale\textsuperscript{2}, Roland Siegwart\textsuperscript{2}, and Marc Pollefeys\textsuperscript{1}

\textsuperscript{1}Computer Vision and Geometry Lab
\textsuperscript{2}Autonomous Systems Lab
ETH Zürich

Virtual Tourism

Photo Tourism
Exploring photo collections in 3D

Noah Snavely  Steven M. Seitz  Richard Szeliski
University of Washington  Microsoft Research

SIGGRAPH 2006
Virtual Tourism

[Image of a virtual tour interface with a gallery exhibit and a figure navigating through the scene.]

[Microsoft Photosynth logo and interface elements.]
Improving Holiday Photos

Photo Uncrop

Qi Shan⁺  Brian Curless⁺  Yasutaka Furukawa°
Carlos Hernandez*  Steven M. Seitz**

⁺University of Washington  *Google
°Washington University in St. Louis

ECCV 2014
Geo-Tagging Holiday Photos

(Li et al. ECCV 2012)
Clothing

- Scan a person, custom-fit clothing
Industrial inspection

- Verify specifications
- Compare measured model with CAD
Scanning industrial sites

as-build 3D model of off-shore oil platform
Scanning cultural heritage
Cultural heritage

Virtual Monticello

Allow virtual visits
Cultural heritage

Stanford’s Digital Michelangelo

Digital archive
Art historic studies
Archaeology

accuracy ~1/500 from DV video
(i.e. 140kb jpegs 576x720)
Archaeology

Record different excavation layers

Generate 4D excavation record

Layer 1

Layer 2

Generate & verify construction hypothesis
Mobile Phone 3D Scanner

Turning Mobile Phones into 3D-Scanners

ETH Zürich
Forensics

- Crime scene recording and analysis
Leica Geosystems

- when it has to be right
Forensics
Surgery - simulation

- simulate results of surgery
- allows preoperative planning
Surgery - teaching

Capture models of surgery for interactive learning
Surveillance
Surveillance
Computer games

Content capture

HCI
Raw Kinect output: Color + Depth

http://grouplab.cpsc.ucalgary.ca/cookbook/index.php/Technologies/Kinect
Bodytracking

Randomized forest (Shotton et al.)

based on simple depth difference tests at each node
Human-machine interface

Control Humanoid Robot with Kinect
3D Video with Kinect
Autonomous micro-helicopter navigation

Use Kinect to map out obstacles and avoid collisions
Google: Project Tango
Google: Project Tango
Performance capture

High-Quality Passive Facial Performance Capture using Anchor Frames

Thabo Beeler, Fabian Hahn, Derek Bradley, Bernd Bickel, Paul Beardsley, Craig Gotsman, Robert W. Sumner, Markus Gross
Performance capture

(Oswald et al. ECCV 14)
Motion capture
Augmented Reality

(Middelberg et al. ECCV 2014)
Mixed Reality
Course objectives

• To understand the concepts that relate images to the 3D world and images to other images

• Explore the state of the art in 3D photography

• Implement a 3D photography system/algorithm
Material

Slides and more

http://www.cvg.ethz.ch/teaching/3dphoto/

Also check out on-line “shape-from-video” tutorial:

http://www.cs.unc.edu/~marc/tutorial/

Other interesting stuff:

• Book by Hartley & Zisserman, Multiple View Geometry
• Book by Szeliski, Computer Vision: Algorithms and Applications
Learning approach

• Introductory lectures:
  • Cover basic 3D photography concepts and approaches.

• Further lectures:
  • Short introduction to topic
  • Paper presentations (you)
    (seminal papers and state-of-the-art, related to projects)

• 3D photography project:
  • Choose topic, define scope (by week 4)
  • Implement algorithm/system
  • Presentation/demo and paper report

Grade distribution
• Paper presentation & discussions: 25%
• 3D photography project: 75%
3D photography course team

Marc Pollefeys
CNB G105
marc.pollefeys [at] inf.ethz.ch

Torsten Sattler
CNB 102.2
torsten.sattler [at] inf.ethz.ch

Yagiz Aksoy
CAB G 84.2
yaksoy [at] inf.ethz.ch
# Schedule (tentative)

<table>
<thead>
<tr>
<th>Date</th>
<th>Topic</th>
</tr>
</thead>
<tbody>
<tr>
<td>Feb 16</td>
<td><strong>Introduction</strong></td>
</tr>
<tr>
<td>Feb 23</td>
<td>Geometry, Camera Model, Calibration</td>
</tr>
<tr>
<td>Mar 2</td>
<td>Features, Tracking / Matching</td>
</tr>
<tr>
<td>Mar 9</td>
<td>Project Proposals by Students</td>
</tr>
<tr>
<td>Mar 16</td>
<td>Structure from Motion (SfM) + 2 papers</td>
</tr>
<tr>
<td>Mar 23</td>
<td>Dense Correspondence (stereo / optical flow) + 2 papers</td>
</tr>
<tr>
<td>Mar 30</td>
<td>Bundle Adjustment &amp; SLAM + 2 papers</td>
</tr>
<tr>
<td>Apr 6</td>
<td>Easter</td>
</tr>
<tr>
<td>Apr 13</td>
<td>Multi-View Stereo &amp; Volumetric Modeling + 2 papers</td>
</tr>
<tr>
<td>Apr 20</td>
<td>Project Updates</td>
</tr>
<tr>
<td>Apr 27</td>
<td>3D Modeling with Depth Sensors + 2 papers</td>
</tr>
<tr>
<td>May 4</td>
<td>3D Scene Understanding + 2 papers</td>
</tr>
<tr>
<td>May 11</td>
<td>4D Video &amp; Dynamic Scenes + 2 papers</td>
</tr>
<tr>
<td>May 18</td>
<td>Final Demos</td>
</tr>
<tr>
<td>Apr 6</td>
<td>Pentecost</td>
</tr>
</tbody>
</table>

---

*Note: Papers mentioned may be subject to change.*
Fast Forward!

- Quick overview of what is coming...
Camera models and geometry

Pinhole camera

\[
\lambda \begin{bmatrix} x \\ y \\ 1 \end{bmatrix} = \begin{bmatrix} f_x & p_x \\ f_y & p_y \\ 1 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} R & t \\ 0 & 1 \end{bmatrix} \begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix}
\]

or \( \lambda x = PX \)

Geometric transformations in 2D and 3D
Camera Calibration

- Know 2D/3D correspondences, compute projection matrix

  also radial distortion (non-linear)
Feature tracking and matching

Harris corners, KLT features, SIFT features

key concepts: invariance of extraction, descriptors to viewpoint, exposure and illumination changes
3D from images

Triangulation
- calibration
- correspondences
Epipolar Geometry

Fundamental matrix \[ x^T F x = 0 \]
\[ F \leftrightarrow P, P' \]

Essential matrix \[ x^T [t]_x R x = 0 \]
\[ E \leftrightarrow P, P' \]

Also how to robustly compute from images
Structure from motion

- **Initialize Motion**
  
  \( (P_1, P_2 \text{ compatible with } F) \)

- **Initialize Structure**
  
  (minimize reprojection error)

- **Extend motion**
  
  (compute pose through matches seen in 2 or more previous views)

- **Extend structure**
  
  (Initialize new structure, refine existing structure)
Visual SLAM

- Visual Simultaneous Navigation and Mapping

ICCV Paper Number 450
A New Minimal Solution to the Relative Pose
of a Calibrated Stereo Camera with
Small Field of View Overlap

(Clipp et al. ICCV’ 09)
Stereo and rectification

Warp images to simplify epipolar geometry

Compute correspondences for all pixels
Multi-View Stereo
Joint 3D reconstruction and class segmentation

(Haene et al CVPR13)

reconstruction only
(isotropic smoothness prior)

joint reconstruction and segmentation
(ground, building, vegetation, stuff)

- Building
- Ground
- Vegetation
- Clutter
Structured-light

- Projector = camera
- Use specific patterns to obtain correspondences
Papers and discussion

- Will cover recent state of the art
  - Each student team will present a paper, followed by discussion
  - Papers will be related to projects/topics
- Will distribute papers later (depending on chosen projects)
Course project example: Build your own 3D scanner!

Example: Bouguet ICCV’ 98

Students can work in alone or in pairs.
Interactive 2D-3D Image Conversion on Mobile Devices

**Goal:** An Android app will be developed that can generate a stereo pair from a single image

**Description:**
An Android app will be developed that is designed to generate stereo pairs from single images. The system will consist of an interactive segmentation step followed by stereo view rendering step using several simple hole handling techniques. The stereo pair will then be shown to the user either on a regular Android phone in anaglyph 3D or on a phone with an autostereoscopic display such as HTC Evo 3D.


**Requirements / Tools:**
- Required: Java or C++, some experience with image processing
- Recommended: Experience with Android app development and with OpenCV

**Supervisor:**
Yagiz Aksoy
www.inf.ethz.ch/personal/aksoyy
Video Deblurring for Hand-Held Cameras

**Goal:** Removing blur due to camera shake from videos recorded with a mobile phone

**Description:**
In videos recorded with a hand-held camera, some frames appear blurred. In this project, students will record their own videos and detect which frames are sharp. Then, using samples from the sharp frames, motion-blurred frames will be recovered.


**Requirements / Tools:**
Required: C++, some experience with image processing
Recommended: Experience with OpenCV

**Supervisor:**
Yagiz Aksoy
www.inf.ethz.ch/personal/aksoyy
**Goal:** Recover scene structure and camera motion using the intersection of lines in the image as features.

**Description:**
Classical Structure-from-Motion is approached using point-features, however, indoor environments often lack enough texture for point features to be tracked effectively. One solution is to use straight lines to track the motion of the camera which frequently appear in man-made environments, however relocalizing against lines can be hard. This project proposes to first detect point-features based on the intersection of lines [1], and then use this features as an input to existing Structure-from-Motion software to recover the location of these points in 3D as well as the camera motion.


**Requirements / Tools:**
Matlab or C++

**Supervisor:**
Federico Camposeco
Fede.camposeco@inf.ethz.ch
CNB D102
Image Deblurring using 3D Camera Rotation

**Goal:** Remove motion blur effects in an image or a video using deconvolution with a non-uniform blur kernel depending on the 3D camera rotation.

**Description:**
Ever taken a blurry image? Let’s remove it!

Motion blur is mostly due to the 3D rotation. The goal here is to estimate the blur kernel which depends on this rotation in order to „Deblur“ (or deconvolve) the original image. The project contains two main steps:

- Blur analysis and blur kernel estimation based on the camera rotation
- Image deconvolution to remove the blur effect.
  (using Wiener Filtering or similar techniques)

Additionally, this can eventually be implemented on a smartphone, where the rotation of the camera can be easily recovered from the motion sensors (gyroscopes).


**Requirements / Tools:**
C++/Matlab/...

Optional: Android / Smartphone development experience

**Supervisor:**
Amaël Delaunoy  
Amael.delaunoy@inf.ethz.ch  
CNB G104

3D Photography, Spring Semester 2013
Obstacle Fusion for Autonomous Car

**Goal:** Fusion of obstacle data over multiple frames from a 2D obstacle sensor with high uncertainty

**Description:**
In the V-Charge project [1] we have an obstacle sensor that extracts 2D obstacle information from fisheye images based on the motion of the car. The data has a high uncertainty and hence needs to be fused properly over multiple frames to get good results. In computer vision this task has been done in different ways for 3D data for example [2,3]. The idea for this project is to implement a fusion for the 2D obstacle data.


**Requirements / Tools:**
C++

**Supervisor:**
Christian Häne
chaene@inf.ethz.ch
http://people.inf.ethz.ch/chaene/
**Real-time Obstacle Detection**

**Goal:** Obstacle Detection based on low latency rolling shutter depth maps

**Description:**
A prototype board with a combination of FPGA and mobile CPU generates depth maps using rolling shutter cameras in real-time and low latency.

Based on a histogram of the depth map, obstacles can be detected to support any mobile robot or micro aerial vehicle during maneuvering [1].

**Tasks:**
- Generate histogram and update line by line to exploit the rolling shutter effect.
- Detect obstacles in generated histogram and track over time.

The algorithm will be implemented in Matlab first and ported on a mobile CPU with the Android operating system later.


**Requirements / Tools:**
- Matlab / C++
- embedded programming

**Supervisor:**
Dominik Honegger
donnik_honegger@inf.ethz.ch
http://people.inf.ethz.ch/dominiho/
EpicFlow: Edge-Preserving Interpolation of Correspondences for Optical Flow

Goal: Implement state-of-the-art optical flow algorithm

Description:
A novel approach for optical flow estimation, targeted at large displacements with significant occlusions. It consists of two steps:

i) Dense matching by edge-preserving interpolation from a sparse matches
ii) Variational energy minimization initialized with the dense matches

EpicFlow: Edge-Preserving Interpolation of Correspondences for Optical Flow
https://hal.inria.fr/hal-01097477/document

Requirements / Tools:
C++, some experience with image processing

Supervisor:
Lubor Ladicky
www.inf.ethz.ch/personal/ladickyl
3D from motion vectors

**Goal:** Recover 3D shape of objects from encoded video motion vectors.

**Description:**
Encoded videos contain estimated motion vectors describing the displacement of image-blocks from one frame to the next. This information could potentially be used to estimate the 3D shape of the observed object (as well as potentially the camera motion). The advantage would be a potentially very fast and efficient 3D reconstruction approach on mobile devices (which have hardware accelerators to encode video).

The goal of this project is to explore this possibility and determine what can be achieved. In a first instance we could assume the camera motion has been determined using a separate approach.

**Requirements / Tools:**
Matlab or C++

**Supervisor:**
Fabio Maninchedda
fabiom@inf.ethz.ch
CNB G100.9
Infrared Super-Resolution

**Goal:** Recover super-resolution images from low-res infrared images

**Description:**
Infrared cameras start to become available in small, low-cost form factors. They have some interesting properties as they are continuously integrating the image, and only “resetting” pixels by cooling them (actively or passively). Each pixel is therefore a “leaking” convolution of the last few frames, a property that should allow to recover a higher resolution image, in particular when camera motion is present.

The goal of this project is to devise a super-resolution approach using a low-resolution infrared image sequence. The image sensor / interface will be provided.

**Requirements / Tools:**
C++ / Matlab

**Supervisor:**
Lorenz Meier
lm@inf.ethz.ch
http://www.inf.ethz.ch/personal/lomeier/
Wide-spectrum Imaging

**Goal:** Recover thin objects from fused RGB + IR images

**Description:**
Infrared cameras start to become available in small, low-cost form factors. Their spectral properties allow interesting scene understanding / reconstruction applications; however, their resolution is not suitable for fine-grained detail.

The scope of this project is to calibrate a megapixel high-res camera with a thermal camera and use the registered image data to create a wide-spectrum image which preserves image detail but covers the image data from long-wavelength infrared to visible light.

**Requirements / Tools:**
C++ / Matlab

**Supervisor:**
Lorenz Meier
lm@inf.ethz.ch
http://www.inf.ethz.ch/personal/lomeier/
**Goal:** Estimate the motion of objects in an image sequence.

**Description:**

The goal of this project is the estimation of a dense optical flow field, that is, for every pixel in the image a 2D motion vector is estimated for every frame. In the images above the motion field is visualized color-coded: the color encodes the motion direction and the intensity encodes the vector length.

The task is to reimplement method [1], which proposes a novel vectorial data term which improves the motion estimation of small-scaled structures.


**Requirements / Tools:**

Required: C++, some experience with image processing

Recommended: Good programming and math skills.

**Supervisor:**

Martin Oswald

[mailto: martin.oswald@inf.ethz.ch](mailto: martin.oswald@inf.ethz.ch)


---

3D Photography, Spring Semester 2015
**Goal:** Upsample the resolution of image sequences by using the information from neighboring frames

**Description:**
Given an image sequence and the corresponding motion field (optical flow), the resolution of each image can be super-resolved by using the information from all images.

The task is to reimplement method [1]. The optical flow estimation will be provided.


**Requirements / Tools:**
Required: C++, some experience with image processing
Recommended: Good programming and math skills.

**Supervisor:**
Martin Oswald
martin.oswald@inf.ethz.ch
http://people.inf.ethz.ch/moswald/
Robust Large-Scale Localization

Goal: Robustly register a novel image against a 3D SfM point cloud.

Description:
Given a Structure-from-Motion point cloud, image-based localization methods estimate the position and orientation from which a new photo was taken. This is done by establishing 2D-3D matches between features in the image and 3D points in the model [1,2]. For large scenes, finding good matches becomes harder. Robust localization approaches thus relax the matching criterion and exploit co-visibility information to handle the larger amount of wrong matches [1].

In this project, you will implement and evaluate such a robust localization method [1].


Requirements / Tools:
C++, Computer Vision

Supervisor:
Torsten Sattler
torsten.sattler@inf.ethz.ch
CNB G102.2
Automatically Annotation Structure-from-Motion Models

**Goal:** Detect and label buildings in a SfM model.

**Description:**
Structure-from-Motion techniques produce a sparse point cloud of a scene. For urban environments, it is desirable to automatically detect and label buildings in the model. Fortunately, OpenStreetMaps provides building boundaries and also building names, which can be used to accomplish this task.

In this project, you will (semi-automatically) register a given SfM point cloud onto an OpenStreetMap. By projecting the points into a 2D plane, you can then automatically detect buildings and label the point cloud. Finally, you will label the images used for the reconstruction by projecting the 3D model into them.

**Requirements / Tools:**
C++, Computer Vision

**Supervisor:**
Torsten Sattler  
torsten.sattler@inf.ethz.ch  
CNB G102.2
Visual Tracker for Cardboard VR

Goal: Implement a real-time visual tracker to enhance VR user experience

Description:
Recently Google released a low cost virtual reality (VR) kit, called Cardboard [1], allowing users to turn their mobile device into a VR stereo screen. The head tracking (camera tracking) is solely based on the Inertial Measurement Unit (IMU) and therefore only allows for rotational motion (rotation of the head). The task of this project is to include translational motion, which can be computed from visual features. More precisely KLT [2] tracks should be used to find point correspondences between frames and a least-squares translational motion should be computed from the correspondences. Key challenges are low latency and high pose estimation rate, for optimal user experience.

[1] https://developers.google.com/cardboard/overview

Requirements / Tools:
Required: C++, Java
Recommended: Anrroid SDK and NDK, Opencv

Supervisor:
Olivier Saurer
saurero@inf.ethz.ch
http://people.inf.ethz.ch/saurero/
Large-scale semantic 3D reconstruction

**Goal:** City-scale semantic 3D reconstruction.

**Description:**
Semantic 3d reconstruction involves not only inferring the occupancy map in 3D, but also the semantic label of every point in space (for a city those labels are Building, Vegetation, Ground etc.). In this project you start with existing code for this task to make it scale to city reconstruction.

**Requirements / Tools:**
Required: C++

**Supervisor:**
Nikolay Savinov
nikolay.savinov@inf.ethz.ch
CAB G 85.1
Real-time point-wise depth estimation from video

**Goal:** Implement the 3D reconstruction algorithm proposed in [1]

**Description:**
Videos are well-suited as input for 3D reconstruction systems due to the large amount of data they offer. At the same time, processing this amount of data in a real-time system is challenging. One strategy is to extract individual points with a context window from arbitrary frames of a video and iteratively estimate their depth, narrowing down the search range in every iteration. This project is about implementing a 3D reconstruction algorithm employing this strategy which has been proposed in [1]. The system's input will be a video sequence with given camera poses for each frame, and it is expected to run in real-time on recent desktop PCs by utilizing GPGPU computing. The output will be a reconstruction of the environment as a colored point cloud.


**Requirements / Tools:**
C++, and OpenCL or CUDA

**Supervisor:**
Thomas Schöps
schoepst@inf.ethz.ch
CAB G84.2
Simple Motion Tracking System

Goal: Build a simple motion tracking system using 2 or more cameras that are able to track three colored balls in 3D.

Description:
We provide a fixed two camera setup. Calibrate the intrinsic and extrinsic parameters of both cameras. Use OpenCV to detect colored balls in the camera view and triangulate the 2D image positions of the cameras into 3D positions.

Detect 3 different balls fixed rigidly together, add a function to define a fixed setup of 3 balls and track them as one object.

Optional: Add additional cameras to the system. Use Kalman filters to robustify the tracking and model pose estimation.

Requirements / Tools:
C++, OpenCV
2+ cameras, e.g. webcams (provided)

Supervisor:
Petri Tanskanen, tpetri@inf.ethz.ch
Lorenz Meier, lm@inf.ethz.ch
CNB G 102.4, CAB G 86.3
Gradient reconstruction from DVS sensor

**Goal:** Recovering image and image gradient from event based camera

**Description:**
An event camera is a silicon retina which outputs not a sequence of video frames like a standard camera, but a stream of asynchronous spikes, each with pixel location, sign and precise timing, indicating when individual pixels record a threshold log intensity change. By encoding only image change, it offers the potential to transmit the information in a standard video but at vastly reduced bitrate, and with huge added advantages of very high dynamic range and temporal resolution. According to the given reverence the idea is to recover the original grayscale image using such an event based camera.


**Requirements / Tools:**
- Matlab, Filtering (Particle and Kalman) some experience with image processing

**Supervisor:**
Petri Tanskanen
tpetri@inf.ethz.ch
http://people.inf.ethz.ch/tpetri/