3D surface reconstruction and automatic camera calibration
a Bayesian approach

March 2004

André Jalobeanu

Esfandiar Bandar, Peter Cheeseman,
Frank Kuehnel, John Stutz, Donald Tae

Bayesian Vision Group
NASA Ames Research Center - Moffett Field, CA - USA

Summary

- Goal?
- Proposed vs. classical methods
- Bayesian approach and graphical model
- The surface model*
- Observation parameters
- Forward problem: rendering*
- Inverse problem: optimization
- Preliminary results
- Applications
- Conclusion
- Extensions and future work

* my contributions
Case 1: infer the geometry

Example - Asteroid 433 Eros (NEAR mission)

- N optical images; very large number N>10000
- Light direction ~ known
- Camera parameters ~ known
- Albedo ~ constant

2D data fusion is not sufficient

Single model:
- 3D geometry
  arbitrary resolution
- Multispectral albedo
  arbitrary number of bands
  super-resolution

N calibrated images --> Geometry
Case 2: infer topography and scalar albedo

N aerial or satellite images

Cameras
Light sources

Elevation (height field)
Albedo (panchromatic)

3D reconstruction: existing methods I

Shape from Stereo [Zhang et al. 94]

- The 2D projection of a surface point into the left and right cameras can be re-projected to give a point in the 3D space

Shape from Shading [Horn & Brooks 89]

- Image gradients are related directly to surface gradients.
- Integrating from known boundary condition gives heights.

Drawbacks:
- Relies on finding point matches in both images
- The density of the recovered surface points is data dependent

Assumptions: Lambertian reflection and constant albedo
3D reconstruction: existing methods II

**Generalized stereo** [Fua & Lecerf 94]

- Use textured triangles on a dense 3D mesh as stereo features
- Optimize a cost function including a data term and a smoothness penalty term
- Combines stereo and shading information

**Drawbacks:**
- Not a Bayesian method: difficult to infer prior model parameters, and the data term is not natural

**Other methods**

- [Morris & Kanade 03]
  Keep vertices fixed, search for good triangulations
- [Isidoro & Sclaroff 02, Vogiatzis et al. 03]
  Stochastic algorithm
  Iteratively shrink and deform a mesh

**Drawbacks:**
- Stochastic: slow, complex
- Others: sensitive to local optima

---

Advantages of the proposed approach

Existing methods are not compatible with our long-term goal:

- **dense** reconstruction of both geometry and reflectance,
  taking advantage of all the information in the N images

**The proposed approach:**

- Enables us to arbitrarily choose the surface **density**
  (e.g. according to the amount of available data)
- Can use any **reflectance model**, from the simple Lambertian BRDF to complex realistic BRDFs
- Can use data from **any sensor**, whatever the pixel size;
  in principle we should be able to use non-optical data (e.g. altimetry)
- Should be able to easily integrate **new images**
  once a initial estimate has been computed
The Bayesian approach

posterior density

\[
P(\text{surface, cameras | images}) \propto \frac{\mathcal{L}(Y)}{\mathcal{L}(\theta)} P(\text{surface}) \prod P(\text{camera}_i) \prod P(\text{image}_i | \text{surface, camera}_i)
\]

prior density

\begin{align*}
\text{Surface model} & \quad \text{Prior camera pose} \\
\text{Gaussian fractal model} & \quad \text{Gaussian model}
\end{align*}

likelihood

\begin{align*}
\text{Image formation model} & \\
\text{Rendering, blur, noise}
\end{align*}

Graphical model (generative model)

- **Surface**
  - Mean: 3D object
  - Uncertainty on the object

- **Observations**
  - Mean: Rendering * Blur
  - Uncertainty from noise

- **Observation parameters**
  - Mean: calibration
  - Uncertainty on camera pose

**Graphical model**:

Relationship between random variables (prior and conditional densities)
The different densities

Surface model $P(S)$
- **Geometry:** dense triangular mesh: vertices + neighborhood
- **Reflectance:** one albedo per vertex
- **Topology:** arbitrary – DEM (flat), planet/asteroid (spherical)
  - Subdivided mesh, has the topology of the initial mesh

Observation parameter density $P(\theta)$
- **Camera pose:** position, orientation
- **Camera physics:** PSF, noise variance
- **Light source:** orientation, ambient and direct intensity

Image formation model $P(Y / I(S, \theta))$
- **Rendering:** realistic synthetic image, $I(S, \theta)$ non-linear w.r.t. $S, \theta$
- **Degradation:** blur by PSF, Gaussian noise

Fractal appearance of natural surfaces
Surface analysis and modeling

What tool to use to analyze and model fractal surfaces?

Check the **statistical self-similarity**:
- first compute the **power spectrum** of the object to analyze.
- Height fields on planar surfaces: 2D Fourier Transform
- Height fields on spherical surfaces: Spherical Harmonics
- **Arbitrary surfaces**: irregular sampling, arbitrary topology
  
  Wavelets on subdivided meshes
  Spectrum: feature size = fct(scale)

Topology, geometry and regularity

**Topological support**
- Set of sites (vertices) + neighborhood system
- Support regularity = neighborhood regularity
- **Semi-regular mesh**: 5 or 6 neighbors

**Geometry**
- 3D point for each topological site
- Objects can have an irregular geometry, but the wavelets are defined on a semi-regular topological support.
Subdivided meshes

Creation of a new topological vertex at the midpoint of each edge
Each triangle is replaced by 4 smaller triangles

→ ideal framework for a multiresolution analysis

Vertex prediction

Geometric **Subdivision**: new vertex creation using a **prediction rule**

Prediction of a new vertex at level j+1 using 8 neighbors at level j:
**interpolation scheme**

\[ m = \frac{1}{2} (v'_1 + v'_2) + \frac{1}{8} (v'^2_1 + v'^2_2) - \frac{1}{16} (v'^3_1 + v'^3_2 + v'^3_3 + v'^3_4) \]

[Dyn et al. 90, Sweldens & Schroeder 95]
Wavelets on a triangular mesh

Wavelet coefficients at level $j + 1 = \text{vertices at level } j + 1 - \text{prediction from level } j$

encode the details at level $j + 1$

Multiresolution analysis

Approximations: $a_1 \ldots a_n$
from coarse to fine, different versions of the same surface

Details: $d_1 \ldots d_n$
differences between two successive approximations
Defining a local scale

- dilation / x
- dilation / y
- skew

details = absolute geometric variations
(independent of the local mesh resolution)

→ we need to define a local scale estimate:

\[ s = L \left( \frac{3}{4} \frac{L^2}{l^2} (\cos \alpha + \sin \alpha)^2 + 4 \right)^{-1/2} \]

Defining a local direction

Wavelet details are 3D vectors

\[ W = W_\perp + W_\parallel \]  

normal + parallel decomposition

g geometric detail

sampling irregularity

vertex at level j+1
Amplitude spectrum of 433 Eros, data from the NEAR mission lidar

\[ \log \sigma = \log \sigma_0 - q \log r \]

fractal exponent \( q=1.3 \)
fractal dimension \( D=2.4 \)

A new multiscale model

- Scale-invariant adaptive Gaussian model on \( \mathbb{R}^3 \):
  prior roughness of the surface

\[ P(\{w_{ij}\}) \propto \prod_i \exp \left( -\lambda_i \frac{w_{ij}}{s_i} - 2q \right) \]

Gaussian random variables
spatially adaptive parameters
local scales

\sim 3D analog of the fractional Brownian motion in 2D
(wavelet coefficients instead of Fourier coefficients)
Efficient description of the power spectrum of natural images

- Statistical model of \( \omega \): mesh regularity prior density
  (sampling regularity)
Samples from the surface model

Observation parameters

prior camera pose: mean $\mu$

uncertainty: covariance $\Sigma$

Prior on the observation parameters (camera, light, PSF, noise):

- Knowledge of the camera motion and orientation (rover: odometers; space probe: gyroscopes, etc.)
- Use the navigation parameters if available
- Sensors: orientation w.r.t. sun or stars
- Prior knowledge of the PSF and noise level
**Forward problem: rendering**

**Rendering:**
- known surface
- known reflectance
- known light source
- known camera param. (internal & external)

Compute the intensity for each pixel...and its derivatives!

---

**Why do we need an accurate rendering?**

- **Sub-pixel** accuracy required for good reconstruction accuracy and super-resolution
- Compute the derivatives of the pixel intensity w.r.t. all the parameters (surface, camera, light) to perform optimization
- **Photometric accuracy:** ambient light, reflectance functions, etc.
- **Occlusions** (hidden surface removal)
- **Shadows** (remove surfaces hidden from the light source)

*We need to work in the object space*

Most of existing algorithms work in image space
Image space or object space?

- Very fast, real time (OpenGL, etc.)
- Image = non-continuous function of the surface S
- Aliasing
- Limited to big triangles

Object

- Sum of contributions / pixel
  \[ I_p = \sum \Phi_\alpha \]
- Image = continuous fct. of S
- No aliasing (better sampling)
- Works with any triangle size
- Independence image/model

The rendering method

Core of the algorithm: occlusion removal

Hidden surfaces = occlusions / camera projection
Surfaces in shadow = occlusions / light source projection

Compute the contribution to a pixel by a partially hidden triangle

- Polygonal approach
- Recursive polygon-triangle subtractions
- Trick:
  only on ridge lines
- At the end: polygon moments + derivatives
Rendering method - details

Intersection triangle/pixel

Occluding triangles subtraction

Shadowing triangles subtraction

Camera projection
Camera projection
Camera + light projection

The rendering tree...

Under development

Vertices
Light
Camera
Camera projection
BRDF
Albedo
Irradiances
Continuous shading

Visibility polygons
Camera + light projection
Moments
Blur

Render

30/45
Derivative computation

The chain rule:

Examples of derivatives:

- $\frac{\partial P}{\partial v}$ projected vertex
- $\frac{\partial P}{\partial v}$ 3D vertex
- $\frac{\partial n_i}{\partial P}$ 2D polygon vertex
- $\frac{\partial n_i}{\partial P}$ normal
- $\frac{\partial A}{\partial n_i}$ polygon area
- $\frac{\partial L_i}{\partial n_i}$ irradiance
- $\frac{\partial W}{\partial A}$ pixel intensity

Rendered images: asteroid
Rendering examples 1

Light source rotation

Rendering examples 2

Object rotation
Solving the inverse problem: optimization

Maximum A Posteriori (MAP) - iterative optimization

- **Linearize** the intensity (rendering):

  \[ I(S, \Theta) = I(S_0, \Theta_0) + \left[ \frac{\partial I}{\partial S} \right] (S - S_0) + \left[ \frac{\partial I}{\partial \Theta} \right] (\Theta - \Theta_0) \]

  \[ \Rightarrow - \log P(S, \{\Theta\} \mid \{Y\}) \text{ approx. by a quadratic form} \]
  \[ \rightarrow \text{optimization of } S, \{\Theta\} \text{ using a conjugate gradient} \]

- Result: used to initialize the next iteration
- Convergence: small variation of \( S, \{\Theta\} \)

Optimization over \( S \):

- **3D geometry and reflectance recovery**

Optimization over \( \{\Theta\} \):

- **automatic camera and light calibration**

Estimating the uncertainties

- **Estimated pose**: mean \( \mu \)
- **Uncertainty**: covariance \( \Sigma \)

- Prior pose
- Prior uncertainty

- At the end: keep the **inverse covariance matrix** (related to the uncertainty), not only the estimated parameters \( S, \{\Theta\} \)

- Very useful to initialize a new estimation procedure, so that new data can be added

- **Recursive refinement** of \( S, \{\Theta\} \):
  - Update the 3D (process large amounts of data recursively instead of batch)
  - Refine the camera pose - Simultaneous Localization And Mapping (SLAM)
Preliminary surface reconstruction results
Duckwater, Nevada

- Physical model built using USGS elevation model
- Hand-painted albedo

Images: CMOS camera
(10 bits, monochromatic)
Light source: sun

**Goal: reconstruct DEM+albedo**
Assumptions:
- no shadows, no occlusions [old]

**Preliminary estimation:**
- sun direction (sun dial)
- camera param. (checkerboard)

One of the 8 observed images

---

Geometry (height field)

Ground truth (USGS DEM)
Inferred DEM
(max error<10mm, RMS error<2mm)
Special case: uniform albedo

Ground truth (USGS DEM)

Inferred DEM (max error ~ 15mm)

Estimating the camera parameters

Camera pose estimation
Method used: joint MAP = estimation of \( \{ \Theta, S \} \)
Alternate optimizations w.r.t. surface and camera (sub-optimal)

Estimation error

Camera calibrated from points
Camera calibrated using MAP
Potential applications

- **3D object reconstruction from multiple images:**
  - Asteroids (albedo ~uniform, spherical topology)
  - Planetary surfaces (variable albedo, planar/spherical topology)
  - Rover situation: MER mission (variable albedo, overhangs)

- **Simultaneous Localization And Mapping (SLAM):**
  - Simplified terrain model: use 3D features instead of meshes

- **Space probe localization and 3D object recovery:**
  - approach and flyby: recursive trajectory estimation and object model refinement

- **Multi-sensor data fusion:**
  - Optical (multi- and hyperspectral), radar (SAR), lidar, ...

- **Fractal geometry and synthetic images:**
  - Generate realistic fractal surfaces
  - Compute photorealistic BRDFs for natural surfaces

---

Inverse rendering and computer vision

- **Computer vision**
  (3D model reconstruction from multiple observations):
  inverse problem of rendering

- **Bayesian inference**
  applied to this inverse problem:
  everything is described by random variables

- **2D data fusion into a single 3D model**
  becomes a **parameter estimation** problem

- **It can be solved by existing efficient optimization techniques**
### Main contributions

- **Rendering:**
  - Take into account shadows and occlusions
  - Visibility polygons (recursive subtraction)
  - Derivative computation
  - Extensions

- **Surface modeling:**
  - New wavelet transform on surfaces of arbitrary topology:
  - Normal/parallel wavelet detail separation
  - Local scale computation
  - Fractal model for natural surfaces

---

### Extensions and future work I - Rendering

- **Extensions (in progress):**
  - Accurate shadows
  - Continuous shading (visibility polygon moments)
  - Continuous PSF, take into account the blur
  - Complex BRDFs (≠ Lambert)

- **Future extensions:**
  - Multispectral
  - Adaptive subdivision
  - Adaptive BRDFs
  - Secondary reflections
  - Other types of camera (push-broom, etc.)
  - Approximate methods
Extensions and future work II
models and inference

• Model extensions:
  Study real surfaces (Earth, Mars):
  – Reflectance/geometry interactions
  – Multi- and hyperspectral albedos

• Inference method extensions:

  Marginalization:
  – Simultaneous reconstruction and calibration
  – Separated geometry and albedo inference

  Bayesian model selection:
  – Infer the topology (overhangs, etc.)
  – Dynamic and adaptive subdivision