Mathematical Optimization in Graphics and Vision

Luiz Velho
Paulo Cezar Pinto Carvalho

IMPA - Instituto de Matemática Pura e Aplicada

SIGGRAPH 2003
SAN DIEGO

Course Schedule

Module 1 – *Computer Graphics and Optimization*
Luiz Velho (1:45 minutes)

Module 2 – *Continuous and Variational Optimization*
Paulo Cezar Pinto Carvalho (1:45 minutes)

Module 3 - *Combinatorial Optimization*
Luiz Velho (1:45 minutes)

Module 4 - *Global Optimization*
Paulo Cezar Pinto Carvalho (1:45 minutes)

Questions we will Try to Answer

- *Why optimization is important for graphics?*
  - Problems and Solutions
- *How optimization can be used in graphics?*
  - Basic Principles
- *What are the main optimization techniques?*
  - Mathematical Concepts
- *Where optimization has been applied?*
  - Examples of Applications

Additional Material

- Website
  - Course Notes
  - Presentation Slides
  - Links

⇒ URL
http://www.visgraf.impa.br/otim-03/

- Course Notes
  - Contents of Presentations

* OBS: *Probability and Optimization*

Computer Graphics and Optimization

Luiz Velho

IMPA - Instituto de Matematica Pura e Aplicada

SIGGRAPH 2003
SAN DIEGO

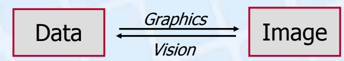
Outline

- Basic Concepts
  - Graphical Objects
  - Operators
- Graphics Problems
  - Direct / Inverse
  - Well-Posed / Ill-Posed
- Optimization Methods
  - Formulating the Problem
  - Classification of Techniques
- Applications
  - Main Criteria
  - Examples in Different Areas

Motivation

- Optimization is a Basic Tool for Graphics
  - Widely Used
  - Flexible
- What Makes Optimization Important ?
  - Need to Understand Graphics Problems
  - Conceptual View

Graphics and Vision



- Relation with Physical Universe
  - Photography: (2D representation)
  - Human Visual System: (3D reconstruction)
- Concepts
  - What are the Models ?
  - What are the Problems ?

Graphical Object

$$O = (U, f), \quad f : U \subset \mathbf{R}^n \to \mathbf{R}^m$$

- Geometric Support: U

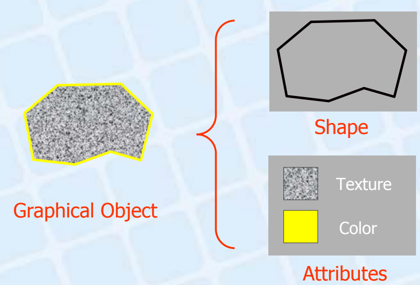
  (shape)
- Attribute Function: f

  (properties)

* *Dimension*
- * Object: $dim(U)$
- * Space: n

Simple Example

- 2D Drawing



More Examples

- Images:
  - Grayscale, Color

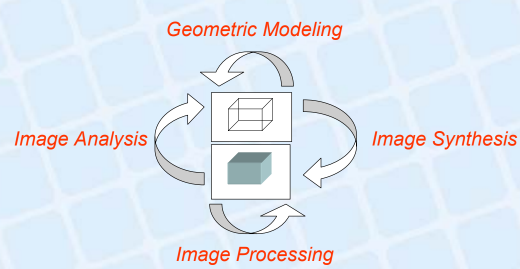
  (simple shape, complex attributes)

- Models (Data)
  - Surfaces, Solids

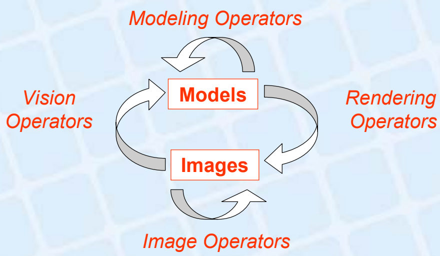
  (complex shape, simple attributes)

* *Graphical Object :- Comprehensive Concept*

Processing Graphical Objects



Operators on Graphical Objects



Graphical Problems

- Spaces of Graphical Objects
  (function spaces)

$$x \in O$$

- Operators on Spaces of Graphical Objects

$$T: O_1 \rightarrow O_2$$
$$T(x) = y$$

Types of Problems

$$T(x) = y$$

- Direct Problems
  - Given T and x , find y
- Inverse Problems
  - Given T and y , find x such that $T(x) = y$
  - Given x and y , find T

Example: Visualization

- x is the scene
  - Geometry
  - Illumination
  - Camera
- T is the *rendering* operator
  $T\,x = y$
- y is the rendered image

* *Direct Problem*

⇒ *OBS: Vision - Inverse Problem*

Problem Characterization

* Hadamard (1902)

- Well-Posed Problem
  - Existence of Solution
  - Uniqueness of Solution
  - Continuous Dependence on Initial Conditions
- Ill-Posed Problem
  (doesn't satisfy at least one of the above conditions)

Ill-Posed Problems

- Inherent in some Applications
  - Inverse Problems are usually Ill-Posed
- Sources of Ill-Posedness
  - Multiple Solutions
  - Numerical Errors

⇒ Need to *get around* Ill-Posedness

- Optimization Methods
  - Best Solution (*unique*)

Example of Ill-Posed Problem

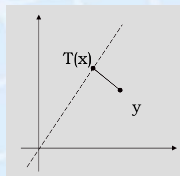
- Linear System: $T\,x = y$,

$$T = \begin{pmatrix} 1 & 3 \\ 2 & 6 \end{pmatrix}\ ,\ x = (u, v)\ ,\ y = (a, 2a)\ , \text{ and } a \in \mathbf{R}$$

- Solution y must lie on

  $2u - v = 0$

  – Ill-Conditioning / Perturbations

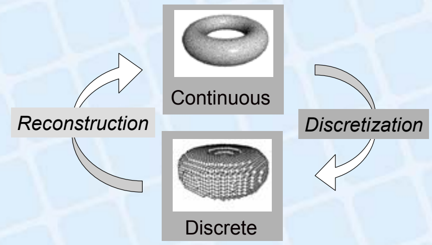


- Well-Posed Solution

  $y = \operatorname{argmin} |y, Tx|$

Describing Graphical Objects

- Computer Representation from Mathematical Model



Ill-Posed Problems in Graphics

- Representation Operator

  $R: \mathbf{O} \rightarrow \mathbf{O}_d$

- Discretization of Graphical Objects
  – Direct Problem
- Reconstruction of Graphical Objects
  – Inverse Problem

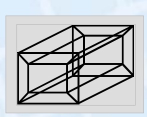
Reconstruction Issues

- Why do we need to reconstruct ?
  – Discretization gives incomplete information
  – Working in continuous domain avoid numerical errors
  – Semantics
- Reconstruction

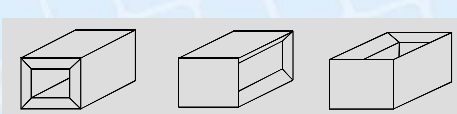
  (Invertibility of the Representation Operator)
  – Exact
  – Non-Exact
- Ill-Posed Reconstruction
  – Ambiguous Representation

Ambiguity in Models

- *Wireframe* Model



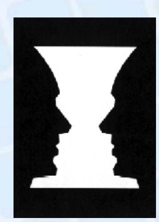
- Many Interpretations



Ambiguity in Images

- Two possible interpretations

2D / 3D



Foreground / Background

Optimization in Graphics and Vision

- Techniques for selecting the "*best*" solution
  - Solve Ill-Posed Problems
- Applications in Graphics
  - Good Representations (Unique, Compact and Efficient)
  - Robust Operators (Numerical Computation)
  - Automatic Selection of Desired Parameters (User Interface)
  - Minimal Energy Solutions (Physical Simulations)

Overview of Optimization Techniques

- Optimization:

Choosing the *best* among a set of *alternatives*

$$\min_{x \in S} \; f(x)$$

$f : S \to R$ is the objective function

S is the set of feasible solutions

Classification of Techniques

Three different Criteria:

- Nature of the Solution Set S
- Description of the Solution Set S
- Properties of the Objective Function f

Classification of Techniques I

- Nature of the Set S
  - Variational
  - Continuous
  - Discrete
  - Combinatorial

Classification of Techniques II

- Constraints on the Set S

$x \in S$ subject to

  - Equality Constraints

$$h_i(x) = 0, \; i = 1,..,m$$

  - Inequality Constraints

$$g_j(x) \le 0, \quad j = 1,\dots,n$$

Classification of Techniques III

- Properties of the function f
  - Linear
  - Quadratic
  - Convex
  - Sparse

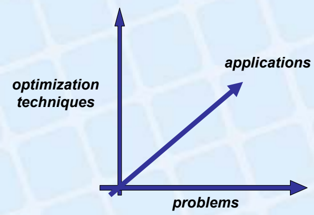
Classification of Techniques IV

Other Issues

- Local versus Global Solutions
- NP-complete Problems
- Probability and Optimization

Studying Optimization

- Approach:
  - Analyze from 3 different points of view



Optimization Techniques x Graphics Problems

	variational	continuous	combinatorial	global
optimal curves	✓	✓	✓	✓
parameter estimation				
...				

Ubiquitous Problem

Optimization in Graphics Applications

* *Typical applications from different areas*

- Selected Applications
  - Modeling: Variational Curves
  - Visualization: Camera Control
  - Vision: Edge Detection
  - Image Processing: Contour Following

Variational Modeling of Curves

- Energy Minimization Model

$$\gamma(t) = \arg\min E_{total}(\gamma)$$

- Energy Functional
  - Shape Quality
  - Model Control

$$E_{total}(\gamma) = E_{internal}(\gamma) + E_{external}(\gamma)$$

Internal Energy

- *Fair* Shape (Physical Model)

$$E_{internal}(\gamma) = \lambda E_{bend} + (1-\lambda) E_{stretch}$$

- Thin Plate

$$E_{bend}(\gamma) = \int_{\gamma} k^2(t)\,dt$$

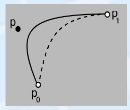
- Membrane

$$E_{stretch}(\gamma) = \int_{\gamma} \| \gamma'(t) \| \, dt$$

External Energy

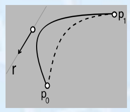
- Modeling Controls
  - Attraction / Repulsion Forces
- Punctual

$$E_{punctual}(\gamma) = \min_t \| \gamma(t) - p \|^2$$



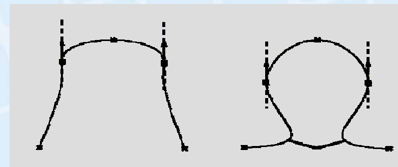
- Directional

$$E_{directional}(\gamma) = \min_t \| \gamma'(t) \times p \|$$



Applications

- Reconstruction from Points
- Interactive Modeling



Camera Control

- Projective Transformation

$$T(x) = p$$

- Inverse Problems
  - Compute Camera Transformation T
  - Compute Scene Points x

Pinhole Camera
(7 parameters)

* *Best Parameter Estimation*
- *Camera Paths*

Differential Camera Control

- Camera Motion with Constrains

minimize

$$E = \frac{1}{2} \left\| \frac{dc}{dt} - \frac{dc_0}{dt} \right\|^2$$

subject to

$$\frac{dp}{dt} - \frac{dp_0}{dt} = 0$$

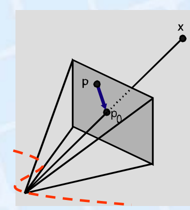
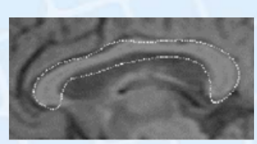


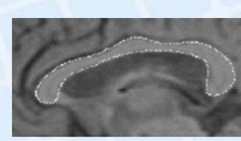
Image Analysis

- *Primal Sketch*
  - Edges
- David Marr's Conjecture
  - Image is Characterized by its Edges
- Boundary Operator
  (Computation of Edges)
  - Geometric Methods
    - * *snakes*

Snakes

- Geometric Computation of the Boundary
- Energy Minimization Approach
  - Boundary function (e.g. grad $I(u,v)$)





Classification of Techniques (again)...

- Variational optimization

  S has *infinite* dimension
  (elements of S are *functions*)

- Continuous optimization

  S has *finite* dimension, i.e., $S \subset R^n$
  (elements of S are *vectors*)

- Discrete optimization
  - usually, $S \subset Z^n$

- Combinatorial optimization
  - concise description by exploiting combinatorial structure

What type of optimization technique ?

- Often, the type of optimization technique to adopt is a matter of the user's choice.

- Basic question:
  At which level to discretize?

Concrete Example (Image Processing)

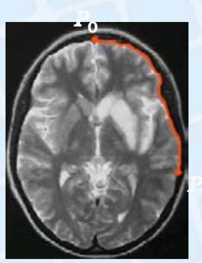
- Contour extraction

  Find x: [0, 1] $\to R^2$,
  with $x(0) = P_0$ and $x(1) = P_1$,
  as to minimize

$$\int_0^1 \|x'(t)\| \; w(x(t)) \; dt,$$

where

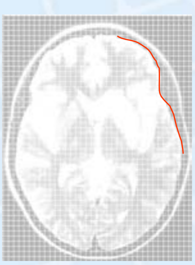
$$w(x) = \frac{1}{k + \|\nabla I(x)\|}.$$



Possible approaches ... (1)

- Tackling directly the variational problem
  - infinitely-dimensional space
  - Euler-Lagrange formulation
    (partial differential equations)
  - solution process requires discretization

Example (*Variational formulation*)



Euler-Lagrange Equations

$$f(t, \mathbf{x}, \mathbf{x}') = \|\mathbf{x}'(t)\| w(\mathbf{x}(t))$$

$$\min \int f(t, x', x) dt$$

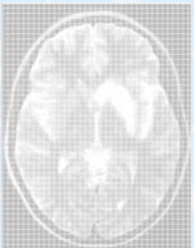
$$\frac{\partial f}{\partial x_i} = \frac{\partial}{\partial t}\left(\frac{\partial f}{\partial x_i'}\right), \; i = 1,..,n$$

Numerical Discretization

Possible approaches... (2)

- Approximating by an optimization problem in finite dimension
  - for instance, restrict attention to polynomials
    (characterized by a finite set of parameters)
  - use methods of continuous optimization in finite dimensions

Example (*Continuous formulation*)



Parametric Cubic Curves

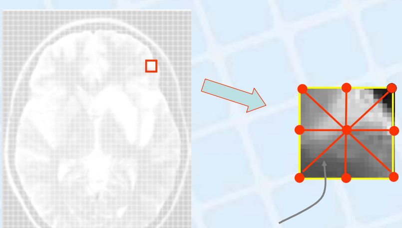
$$\mathbf{x}(t) = (x_1(t), x_2(t))$$
$$x_1(t) = c_{10} + c_{11}t + c_{12}t^2 + c_{13}t^3$$
$$x_2(t) = c_{20} + c_{21}t + c_{22}t^2 + c_{23}t^3$$

8-dimensional space

Possible approaches... (3)

- Approximating by a discrete optimization problem
  - discretization occurs at the *beginning* of the process
  - for instance, consider a regular grid and solve a shortest path problem

Example (*Discrete formulation*)



$$\text{cost} = \text{length} \cdot \frac{1}{k + \|\nabla I\|}$$

The need for discretization

- Discretization is always needed…
- … but can occur at different points of the solution process…
- … leading to different computational schemes

Conclusions

* *Optimization has many uses in Graphics*

- Representation
  - Solves Ambiguity
  - Incorporates User-defined Criteria
- Algorithms
  - Efficient
  - Robust
- User Interface
  - Intuitive Controls
- Optimization Techniques
  - Main Issue: *discretization*