An Algebraic Multigrid Approach for Shape from Photometric Stereo and Binarization

Ron Kimmel and Irad Yavneh
Department of Computer Science
Technion

Image Binarization

Original Image
Nonuniform Illumination

Tilted

Spherical
Naïve (threshold) binarization

Tilted
Naïve (threshold) binarization

Spherical
Yanowitz-Bruckstein Binarization

• Isolate the locations of edge centers, for example, the set of points,

\[ s = \{(x, y) : |\nabla I| > T\} \]

for some threshold \( T \).

• Use the values \( I(x,y) \), for \((x,y)\) in \( s \), as constraints for a threshold surface, \( u \), which elsewhere satisfies the equation

\[ \Delta u = 0. \]
Edges

Tilted

Spherical
Results

Tilted

Spherical
Algebraic Multigrid for Shape from Photometric Stereo

1. What is “shape from photometric stereo”?
2. Why is the “standard approach” insufficient?
3. What can we do about it?
4. How does (algebraic) multigrid help?
5. Some results
The Problem

“Shape from photometric stereo” deals with the problem of shape reconstruction from 2D projections of the real world onto a camera in the case where the camera is fixed and several images are obtained with different lighting conditions.
Input images with the same camera position and head object but three different lighting directions.
**Lambertian reflectance model:** Given three images, $I_1, I_2, I_3$, of the same object with three different lighting directions, $l_1, l_2, l_3$, we assume

$$I_i = \rho \langle l_i, N \rangle, \quad i = 1, 2, 3,$$

where $\rho(x, y)$ is the albedo (which depends on the properties of the object), and $N$ is the normal to the surface, $z(x, y)$, given by

$$N = \frac{(-z_x, -z_y, 1)}{\sqrt{1 + z_x^2 + z_y^2}}.$$
The approximate surface gradient,
\[
(p, q)^T \approx \nabla z
\]
can be extracted easily from the images.

We wish to reconstruct \(z(x,y)\), keeping in mind that there are errors in the model and in the measurements.
Variational formulation: find the surface which minimizes the functional

\[
\iint_\Omega w(x, y) \left\| (p, q)^T - \nabla z \right\|^2 \, dx \, dy,
\]

with \( w(x,y) > 0 \). The Euler-Lagrange equation is

\[
\nabla \cdot \left( w(x, y) \left( p - z_x, q - z_y \right)^T \right) = 0, \quad (x, y) \in \Omega,
\]

\[
\nabla z \cdot n = (p, q)^T \cdot n, \quad (x, y) \in \partial \Omega,
\]

where \( n \) is the outwards normal to \( \partial \Omega \).
The choice $w(x, y) \equiv 1$ yields the Poisson equation. The resulting reconstruction is often unsatisfactory due to errors in the model and in the measurements, shaded regions, etc.

Additional data: constraints at points where the height $z$ is known accurately by some independent measurement.

We then minimize the functional subject to the $c$ constraints,

$$z(x_k, y_k) = z_k, \quad k = 1, \ldots, c.$$
Now, however, $w(x,y)=1$ will result in spurious “spikes” at the constrained points. Indeed, one can show that in order to maintain $p$ continuous derivatives of the reconstructed shape at $(x, y) = (x_k, y_k)$ we must have

$$w(x, y) \approx r_k^{-\alpha},$$

with $\alpha > p$ in a neighborhood of $(x_k, y_k)$, where

$$r_k(x, y) = \sqrt{(x - x_k)^2 + (y - y_k)^2}.$$

Accordingly, we set

$$w(x, y) = \left( \min_k r_k \right)^{-2}.$$

There exists an efficient algorithm to compute $w(x,y)$. 
Computationally, the problem is now more difficult due to the constraints and the singular $w$.

Approaches based on simple multigrid or FFT are inefficient.

Hence, we choose a novel robust algebraic multigrid approach:

- Galerkin coarsening
- Specialized Prolongations
Some Results: Side and perspective views of the reconstructed surface, with the frontal textured mapped onto it. **Left:** unconstrained; **Right:** eight constrained points.
A multigrid approach for multidimensional scaling (MDS)

Michael M. Bronstein, Alexander Bronstein
Ron Kimmel, Irad Yavneh

Department of Computer Science
Technion – Israel Institute of Technology

Based on: [B^2,K,Y: NLAA 13(2-3), 149-171, 2006]
Plan

- Isometric embedding problem
- Motivation from 3D face recognition
- Basic MDS algorithm
- Multigrid MDS
- Results
Isometric embedding

A mapping between finite metric spaces

$$\varphi: \left( \{\xi_1, ..., \xi_N \} \subseteq S, \Delta \right) \rightarrow \left( \{x_1, ..., x_N \} \subseteq \mathbb{R}^m, D \right)$$

such that

$$\delta_{ij} \approx d_{ij} = \| x_i - x_j \| \quad \forall i, j = 1, ..., N.$$

- $\mathbb{R}^m$ - $m$-dimensional embedding space
- $\Delta = (\delta_{ij})$ - $N \times N$ matrix of original geodesic distances
- $D = (d_{ij})$ - $N \times N$ matrix of distances in the embedding space

A. Elad, R. Kimmel, CVPR 2001
Isometric embedding in cartography

GLOBE (HEMISPHERE)

Exact isometric embedding of the sphere into any $\mathbb{R}^n$ does not exist

PLANAR MAP

A. Bronstein, M. Bronstein and R. Kimmel, “Three-dimensional face recognition”
Multidimensional scaling (MDS)

- **Exact** isometric embedding does not exist in most cases
- MDS = minimization of embedding error criterion (stress), e.g. LS:

\[
s(X) \equiv \sum_{i<j} \left( d_{ij}(X) - \delta_{ij} \right)^2
\]

- \( X = (x_1;...;x_N) \) - \( N \times m \) matrix of coordinates in \( \mathbb{R}^m \)
- \( Nm \) optimization variables
- Optimum defined up to an isometry group in \( \mathbb{R}^m \)

### Caveats

- **Non-convex** and **nonlinear** optimization problem
- Hessian is structured but **full (dense)**
- Computational complexity of $s(X)$ and $\nabla_s(X)$ is approximately the same
- Newton algorithm is prohibitive for large $N$
- Line search is disadvantageous
Isometric embedding in 3D face recognition

FACIAL SURFACES

EXPRESSION-INVARIANT REPRESENTATIONS (EMBEDDING INTO $\mathbb{R}^3$)

M. Bronstein, A. Bronstein, R. Kimmel, “Expression-invariant representation for human faces”
3DFACE face recognition system
3DFACE face recognition system

SCANNER OUTPUT

CROPPING → SMOOTHING → SUBSAMPLING

BOTTLENECK
~ 35% TOTAL TIME

MDS → DISTANCE MEASUREMENT → MASK

EXPRESSION-IN Variant REPRESENTATION
SMACOF – a basic MDS algorithm

\[ \min_X \sum_{i<j} \left( d_{ij}(X) - \delta_{ij} \right)^2 \]

- First-order gradient-descent type optimization method
- Gradient of the stress function:
  \[ \nabla_X s(X) = 2UX - 2B(X)X \]

where

\[ u_{ij} = \begin{cases} 
-1 & \text{if } i \neq j \\
N-1 & \text{if } i = j 
\end{cases} \quad b_{ij} = \begin{cases} 
-\delta_{ij}d_{ij}^{-1}(X) & \text{if } i \neq j \text{ and } d_{ij} \neq 0 \\
0 & \text{if } i \neq j \text{ and } d_{ij} = 0 \\
-\sum_{j \neq i} b_{ij} & \text{if } i = j 
\end{cases} \]

- A gradient descent step can be performed with a multiplicative update

\[ X^{(k+1)} = U^\dagger B(X^{(k)})X^{(k)} \]

\[ \Leftrightarrow \quad X^{(k+1)} = X^{(k)} - \frac{1}{2} U^\dagger \nabla_X s(X^{(k)}) \]

SMACOF STEP

Multigrid MDS components

- **Hierarchy of grids:** \( \Omega_1 \supset \Omega_2 \supset \ldots \Omega_R \)

- **Restriction / interpolation operators:**

  \[
  \text{Points: } X_{r+1} = P_r^{r+1} X_r \quad X_r = P_r^r X_{r+1} = \left( P_r^{r+1} \right)^T X_{r+1}
  \]

  \[
  \text{Distances: } \Delta_{r+1} = \tilde{P}_r^{r+1} \Delta_r \left( \tilde{P}_r^{r+1} \right)^T
  \]
Towards multigrid MDS

- Convex nonlinear optimization is equivalent to a nonlinear equation

\[ \nabla s(X) = 0 \iff \min_X s(X) \]

- **Multigrid spirit**: solve problems of the form

\[ \nabla s(X) = T \iff \min_X s(X) - \text{trace}(X^T T) \]

at different resolution levels.

- **T** - residual transferred from finer resolution levels.
Modified stress

- **Problem:** the function \( s(X) - \text{trace}(X^T T) \) is unbounded
- **Modified stress:** force the center of gravity of \( X \) to be zero

\[
\hat{s}(X; \Delta) = \sum_{i<j} (d_{ij}(X) - \delta_{ij})^2 + \lambda \sum_{j=1}^{m} \left( \sum_{i=1}^{N} x_{ij} \right)^2
\]

The modified stress is bounded
Multigrid MDS (FAS V-cycle)

\[ \text{vcycle} (X_r, T_r, \Delta_r, K_r, K'_r) \]

- **IF** \( r = R \) (coarsest level), \( \min_{X_R} s_R (X_R, T_R) \) and return
- **ELSE**
  - Apply \( K_r \) SMACOF iterations to \( s_r (X_r, T_r) \), return \( X'_r \)
  - Compute \( G'_r = \nabla s_r (X'_r) \);
    \[ X'_{r+1} = P^{r+1}_r X'_r; \]
    \[ G'_{r+1} = \nabla s_{r+1} (X'_{r+1}) ; \]
    \[ T_{r+1} = G_{r+1} - P^{r+1}_r G_r \]
  - Apply MG on a coarser resolution:
    \[ X''_{r+1} \leftarrow \text{vcycle} (X'_{r+1}, T_{r+1}, \Delta_{r+1}, K_{r+1}, K'_{r+1}) \]
  - Correction: \( X''_r \leftarrow X'_r + P^r_{r+1} (X''_{r+1} - X'_{r+1}) \)
  - Apply \( K'_r \) SMACOF iterations to \( s_r (X''_r, T_r) \), return \( X''_r \)
Error smoothing

BEFORE RELAXATION

AFTER RELAXATION

Error smoothing using SMACOF relaxation

M. Bronstein, A. Bronstein, R. Kimmel, I. Yavneh, “A multigrid approach for multidimensional scaling”
Simulations

- Isometric embedding of a facial surface

- Problems of different sizes: $N = 225, 625, 1425, 3249$ points

- Different number of resolution levels: $R = 3, 4$

- Different MG cycles: V-cycle and F-cycle

- Different initialization: original points and random points
Results: random initialization

Performance of SMACOF and MG (V-cycle, 3 resolution levels) using random initialization

M. Bronstein, A. Bronstein, R. Kimmel, I. Yavneh, “A multigrid approach for multidimensional scaling”
Results: Problem size

Boosting obtained by multigrid MDS (V-cycle) compared to SMACOF, 3 resolution levels, initialization by the original points

M. Bronstein, A. Bronstein, R. Kimmel, I. Yavneh, “A multigrid approach for multidimensional scaling”
Results: different MG cycles

Convergence of different MG cycles. Initialization by the original points.
Conclusions

- Multigrid MDS demonstrates **significantly better performance** compared to SMACOF (~ order of magnitude)

- The improvement is more pronounced for large $N$

- Multigrid MDS appears to be **less sensitive to initialization**
References

- A. Bronstein, M. Bronstein, R. Kimmel, Three-dimensional face recognition, IJCV, to appear
- M. Bronstein, A. Bronstein, R. Kimmel, Expression-invariant representation of faces, TR CIS-2005-01
- M. Bronstein, A. Bronstein, R. Kimmel, I. Yavneh, A multigrid approach for multidimensional scaling, MG Copper Mountain 2005
Scale Consistent Image Completion

Michal Holtzman Gazit and Irad Yavneh

Computer Science Department
Technion - Israel Institute of Technology

The Problem

- Complete missing information in images
  - Image altered by object removal
  - Text or scratch on an image
Vanishing  Crane
Objectives

- The objective
  - To complete the image so that it will “look natural”.
- Mathematically hard to define.
  - No good objective measures of success/failure yet.
- Naturalness is multi-scaled, and ultimately requires high-level knowledge about the world.

Nevertheless, there are several good low-level approaches and many algorithms which often work well.
Previous work

- **Inpainting Methods**
  - PDE based
  - Diffusion by convolution
  - Learning image Statistics

- **Texture Synthesis**
  - Synthesizing one pixel at a time
  - Copying full patches onto the missing region

- **Complex methods involving**
  - Segmentation
  - Rotation and scaling of patch
  - Image decomposition
  - Order of filling
  - User guidance
Our Contribution

- Systematic employment of another dimension: *scale*.
- The main idea: a good completion must be *scale consistent*. That is, regardless of what our criterion of success is, the completed image must satisfy it at all scales.
Abstract Description

Image: $I = I(\Omega): \Omega \to [0,1]^{d \times |\Omega|}$

Domain (set of pixels):
$$\Omega = \Omega_k \cup \Omega_m$$
where $I$ is known in $\Omega_k$ but missing in $\Omega_m$

An image completion algorithm is a function, $C: [0,1]^{d \times |\Omega|} \to [0,1]^{d \times |\Omega|}$ such that
$$\bar{I} = C(I(\Omega))$$
satisfies: $\bar{I}(\Omega_k) = I(\Omega_k)$
Abstract Description

A smoothing algorithm is a function, $S : [0,1]^{d \times |\Omega|} \rightarrow [0,1]^{d \times |\Omega|}$, such that $I_S = S(I)$ is a less detailed version of $I$.

(The size of the image remains fixed).
We say that a completion is *scale consistent* if \( C(S(I)) \approx S(C(I)) \).
Patch-Based Completion, $C$

Initialize: $\bar{I} = I$; Repeat until:

$\Omega_m = \emptyset$

- Choose target patch, $p$, such that
  
  $p_m = p \cap \Omega_m \neq \emptyset$,  

  $p_k = p \setminus p_m \neq \emptyset$

- Choose source patch, $T(p) \subset \Omega_k$
  
  where $T$ belongs to a set of simple transformations, e.g., translations.

- Set $\bar{I}(p_m) \leftarrow \bar{I}(T(p_m))$

- Redefine $\Omega_m \leftarrow \Omega_m \setminus p_m$
Patch-Based Completion, \( C \)

How should the target patch, \( p \) (i.e., ordering of filling), and the source patch, \( T(p) \), be chosen?

We adopt (but modify) the approach of Criminisi et al.:

Elements of $C$

- **Choosing $p$:**
  - fix size and shape (square), and center on a boundary point of $\Omega_m$
  - Maximize the product of
    - $\left| \frac{p_k}{p} \right|$ Confidence in that patch
    - $\nabla I^\perp \cdot \vec{n}$ The inner product between the normal to the boundary of $\Omega_m$ and the edge entering $\Omega_m$

- **Choosing $T(p)$:** minimize

$$\left\| I(p_k) - I(T(p_k)) \right\|$$
Three Criteria

1. Smoothed-image completion:
   \[ \overline{I}_S(T_S(p_k)) \approx \overline{I}_S(p_k) \]

2. Detailed-image completion:
   \[ \overline{I}(T(p_k)) \approx \overline{I}(p_k) \]

3. Scale consistency:
   \[ \overline{I}_S(T(p)) \approx \overline{I}_S(p) \]
Specific Algorithm

- Generate $n$ detail levels of $I$
- Complete a single patch in $I_S$
- Complete the same patch in $I$ while trying to satisfy $\bar{I}(T(p_k)) \approx \bar{I}(p_k)$ and $\bar{I}_S(T(p)) \approx \bar{I}_S(p)$ simultaneously, equally weighted.
- Multi-scale: recursive, coarse-to-fine.
- Fine to Coarse:
  - The best match in the finest image is eventually used to fill the location in all the levels.
Computational Complexity

- The total complexity for $n$ levels is only
  $$(1+0.07(n -1))(\text{Criminisi})$$
- Exhaustive search performed in coarse level
- Only $K$ (~3%) best matches from coarse level are used for the finer levels for each target patch.
- Filling order is set by the coarsest level
- Each level costs 7% of the computational complexity of the coarsest level.
Single-Scale Developments
Region consistent completion
- In choosing the best matching patch, take into account the **region** surrounding \( p \).
- Among the \( N \) best matching patches choose one which has a **similar surrounding** to the **surrounding** of \( p \).
- Give decreasing weight to the pixels far from the center point (due to lower relevance).
Experiments

- Systematic comparison on a synthetic image of 500x500 pixels containing 2 textures.
- To add randomness, tested 50 locations of the missing region
- Subjective grading
  - Q=1 visible defect
  - Q=2 good (slight defects)
  - Q=3 excellent
- Compared SCIC to Criminisi.
Examples: Quality
Examples: Quality

1

2
Examples: Comparison

<table>
<thead>
<tr>
<th>Q</th>
<th>Criminisi</th>
<th>SCIC</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>56%</td>
<td>18%</td>
</tr>
<tr>
<td>2</td>
<td>36%</td>
<td>18%</td>
</tr>
<tr>
<td>3</td>
<td>8%</td>
<td>64%</td>
</tr>
<tr>
<td>Mean Score</td>
<td>1.52</td>
<td>2.46</td>
</tr>
</tbody>
</table>
Examples: Input Image
Conclusions

- **Scale consistency** boosts the performance of an existing patch-based completion algorithm substantially.
- Fine to coarse and coarse to fine information flow.
- Region Consistency.
- Computational complexity – a fraction more than single scale.
- Future research: Other image applications that use scale consistency.