Planetary Crater Detection and Registration Using Marked Point Processes, Multiple Birth and Death Algorithms, and Region-based Analysis

David Solarna², Gabriele Moser², Jacqueline Le Moigne¹, and Sebastiano B. Serpico²

¹ NASA Goddard Space Flight Center
² University of Genoa

IGARSS 2017, Fort Worth, Texas, July 2017
Introduction

Crater Detection
- Marked Point Process Model
- Energy Function
- Multiple Birth and Death Algorithm
- Region-of-Interest Approach
- Experimental Results

Image Registration
- 2-step Approach
- Experimental Results

Conclusion
Introduction

Crater Detection
- Marked Point Process Model
- Energy Function
- Multiple Birth and Death Algorithm
- Region-of-Interest Approach
- Experimental Results

Image Registration
- 2-step Approach
- Experimental Results

Conclusion
Introduction

Need for automated methods for image registration

- Launch of several planetary missions
- Design of new and powerful sensors

Objective

- **Crater detection** in planetary images
- Development of an **image registration** method based on the extracted features
Content

Introduction

Crater Detection
- Marked Point Process Model
- Energy Function
- Multiple Birth and Death Algorithm
- Region-of-Interest Approach
- Experimental Results

Image Registration
- 2-step Approach
- Experimental Results

Conclusion
Marked Point Processes

Crater detection based on a marked point process (MPP) model

MPP: Stochastic Process \[\xrightarrow{\text{Realizations}}\] Configurations of objects, each described by a marked point

Mathematical Formulation

A point process \(X\), defined over a bounded subset \(P\) of \(\mathbb{R}^2\) maps from a probability space to a configuration of points in \(P\).

Realizations of the process \(X\) are random configurations \(x\) of points, \(x = \{x_1, \ldots, x_n\}\), where \(x_i\) is the location of the \(i^{th}\) point in the image plane \((x_i \in P)\).

A configuration of an MPP consists of a point process whose points are enriched with additional parameters, called marks and aimed at parameterizing objects linked to the points.

Bayesian approach: Maximum a posteriori (MAP) rule to fit the model to the image is equivalent to minimizing an energy function (computationally challenging)
Marked Point Process for Crater Detection

Modelling

Distribution of craters on the surface

Distribution of ellipses on the image plane

Crater

Ellipse

Realization Example

Point Process

\[ x = \{x_1, \ldots, x_7\} \]

Parameters

\[ x_i = (x_{0i}, y_{0i}, a_i, b_i, \theta_i) \]
Energy function of the configuration $X = \{x_1, x_2, ..., x_n\}$ wrt the extracted set $C$ of contour pixels (Canny):

$$U(X|C) = U_P(X) + U_L(C|X)$$

**Prior**

Repulsion coefficient based on the overlapping of the ellipses (overlapping craters are quite unlikely)

$$U_P(X) = \frac{1}{n} \sum_{x_i \land x_j > 0} \frac{x_i \land x_j}{x_i \lor x_j}$$

$x_i \lor x_j =$ area of union of ellipses $x_i$ and $x_j$

$x_i \land x_j =$ area of intersection of $x_i$ and $x_j$

**Likelihood**

Two terms, one based on a correlation measure, the other based on a distance measure (fit between contours and realization of $X$)

$$U_L(C|X) = \sum_{i=1}^{n} \left[ \frac{d_H(x_i^0, C)}{na_i} - \frac{|x_i^0 \cap C|}{|C|} \right]$$

$x_i^0 =$ set of pixels corresponding to ellipse $x_i$ in the image plane

$d_H(x_i^0, C) =$ Hausdorff distance between ellipse $x_i$ and the contours:

$$d_H(A, B) = \max \{ \sup_{\alpha \in A} \inf_{\beta \in B} d(\alpha, \beta); \sup_{\beta \in B} \inf_{\alpha \in A} d(\alpha, \beta) \}$$

Classical distance between sets $d(A, B) = 0$
**Crater Detection – Energy Minimization**

- **Markov chain Monte Carlo-type method**
  - Simulated Annealing scheme

- **Initialization**
  - **Contour Extraction** (Canny) and Parameter Initialization
  - Generation of the **Birth Map** used to speed up the convergence

- **Birth Step**
  - Computation of the **Birth Probabilities** for each pixel
  - **Birth Phase**

- **Death Step**
  - **Energy Computation** for all the ellipses
  - Computation of **Death Probability** and **Death Phase** according to **Likelihood**
  - **Configuration** refinement according to **Prior**

- **Update**
  - **Convergence Test**

Markov chain sampled by a multiple birth and death (MBD) algorithm
MBD – Birth and Death Steps

Birth Step

For each pixel \( s \) in the image, compute the birth probability as \( \min\{\delta \cdot B(s), 1\} \), where:

\[
B(s) = \frac{b(s)}{\sum_s b(s)}
\]

\( b(s) \) is the birth map computed from the contour map using generalized Hough transform and Gaussian filtering.

Death Step

For each ellipse \( x_i \) in the configuration, compute the death probability as \( d(x_i) \):

\[
d(x_i) = \frac{\delta \cdot a(x_i)}{1 + \delta \cdot a(x_i)}
\]

\[
a(x_i) = \exp[-\beta(U_L(X \setminus \{x_i\} | C) - U_L(X | C))]
\]
Crater Detection – Region Based Approach

Region-Based Approach

Why?
- MBD is *computationally heavy*
- Computational burden increases with *image size*

Region Based Flowchart and Example

Initialization

Detection of **regions on interest** based on the birth map

MBD in each **region**

Aggregation
Crater Detection – Data Sets

- 6 THEMIS (Thermal Emission Imaging System) images, TIR, 100m resolution, Mars Odissey mission
- 7 HRSC (High Resolution Stereo Color) images, VIS, ~20m resolution, Mars Express mission
- Image sizes from 1581 × 1827 to 2950 × 5742 pixels

Quantitative Performance Assessment of the crater detection algorithm: Detection Percentage ($D$), Branching Factor ($B$), and Quality Percentage ($Q$)

<table>
<thead>
<tr>
<th>Data</th>
<th>$D = \frac{TP}{TP + FN}$</th>
<th>$B = \frac{FP}{TP}$</th>
<th>$Q = \frac{TP}{TP + FP + FN}$</th>
</tr>
</thead>
<tbody>
<tr>
<td>Avg on all THEMIS</td>
<td>0.91</td>
<td>0.10</td>
<td>0.83</td>
</tr>
<tr>
<td>Avg on all HRSC</td>
<td>0.89</td>
<td>0.06</td>
<td>0.85</td>
</tr>
<tr>
<td>Avg on all images</td>
<td>0.90</td>
<td>0.09</td>
<td>0.84</td>
</tr>
</tbody>
</table>
## Crater Detection – Results

<table>
<thead>
<tr>
<th>Crater</th>
<th>Center $(x_0, y_0)$</th>
<th>Semi-axes $(a, b)$</th>
<th>Orientation $\theta$</th>
</tr>
</thead>
<tbody>
<tr>
<td>Crater 1</td>
<td>(139, 393)</td>
<td>(35, 33)</td>
<td>64°</td>
</tr>
<tr>
<td>Crater 2</td>
<td>(258, 756)</td>
<td>(51, 50)</td>
<td>115°</td>
</tr>
<tr>
<td>Crater 3</td>
<td>(343, 23)</td>
<td>(13, 12)</td>
<td>180°</td>
</tr>
<tr>
<td>Crater 4</td>
<td>(591, 215)</td>
<td>(19, 18)</td>
<td>31°</td>
</tr>
<tr>
<td>Crater 5</td>
<td>(919, 157)</td>
<td>(15, 14)</td>
<td>106°</td>
</tr>
</tbody>
</table>
Content

Introduction

Crater Detection
- Marked Point Process Model
- Energy Function
- Multiple Birth and Death Algorithm
- Region-of-Interest Approach
- Experimental Results

Image Registration
- 2-step Approach
- Experimental Results

Conclusion
Image Registration – 2-Step Optimization

Why a 2-step Optimization?

**Feature-based registration**
- Min Hausdorff distance ($d_H$) between extracted craters through genetic algorithm
- Fast but sensitive to accuracy of crater maps

**Area-based registration**
- Max Mutual Information ($MI$) through genetic algorithm
- Highly accurate but computationally heavy

RST (Rotation-Scale-Translation) transforms $p = (t_x, t_y, \theta, k)$

Initialization

Fast registration based on extracted craters $\to \hat{p}$

Refinement: registration based on mutual information in a neighborhood of $\hat{p} \to p^*$
**Image Registration – Region of Interest**

Transformation found for an interactively selected region of interest → $p_B^*$

$$p_B = (t_x, t_y, \theta, k)$$

Transformation derived for the entire image → $p_A^*$

$$p_A = (T_x, T_y, \beta, \alpha)$$

$$p_A^* = \begin{pmatrix} -k \cos(\theta) x_0 - k \sin(\theta) y_0 + t_x + x_0 \\ k \sin(\theta) x_0 - k \cos(\theta) y_0 + t_y + y_0 \\ \theta \\ k \end{pmatrix}$$

**Superposition of Reference and Input**
Semi-simulated image pairs

20 pairs composed of one real THEMIS or HRSC image and of an image obtained by applying a synthetic transform and AWGN

Quantitative validation with respect to the true transform (RMSE)

Real multi-temporal image pairs

Real multi-temporal pair of LROC (Lunar Reconnaissance Orbiter Camera) images

100m resolution

Only qualitative visual analysis is available, as no ground truth is available
### Registration Results with Semi-synthetic Data

#### RGB of original non-registered data

![RGB of original non-registered data](image)

#### RGB of registered data

![RGB of registered data](image)

<table>
<thead>
<tr>
<th>Data set</th>
<th>RMSE [pixel]</th>
</tr>
</thead>
<tbody>
<tr>
<td>THEMIS (10 data sets)</td>
<td>0.31</td>
</tr>
<tr>
<td>HRSC (10 data sets)</td>
<td>0.22</td>
</tr>
<tr>
<td>Average (20 data sets)</td>
<td>0.26</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th></th>
<th>Left Image</th>
<th>Right Image</th>
</tr>
</thead>
<tbody>
<tr>
<td>( p_{GT} )</td>
<td>(7.05, 35.91, 0.18°, 1.071)</td>
<td>(76.59, 19.96, 2.17°, 1.031)</td>
</tr>
<tr>
<td>( p^* )</td>
<td>(7.04, 35.92, 0.19°, 1.071)</td>
<td>(76.41, 20.06, 2.18°, 1.031)</td>
</tr>
<tr>
<td>RMSE 1(^{st}) Step</td>
<td>0.79</td>
<td>0.51</td>
</tr>
<tr>
<td>RMSE 2(^{nd}) Step</td>
<td>0.16</td>
<td>0.33</td>
</tr>
</tbody>
</table>
Registration Results with Real Data

Visually accurate matching between reference and registered images in the real multitemporal data set.

Checkerboard representation of the registered images (zoom on details)
Registration Results with Real Data

Visually accurate matching between reference and registered images in the real multitemporal data set.
Content

Introduction

Crater Detection

– Marked Point Process Model
– Energy Function
– Multiple Birth and Death Algorithm
– Region-of-Interest Approach
– Experimental Results

Image Registration

– 2-step Approach
– Experimental Results

Conclusion
Conclusions

- **Accurate** crater maps, useful for both image registration and planetary science, were obtained from data from different sensors.
- **Higher accuracy** as compared to previous work on crater detection (not shown for brevity)
- Reduced time for convergence thanks to a region-based approach
- **Sub-pixel accuracy and visual precision** in registration: effectiveness of the proposed 2-step registration method

Future Developments

- Test in conjunction with a **parallel** implementation (e.g. computer cluster)
- Validation with **multi-sensor** real images
- Extension to **other applications** requiring the extraction of ellipsoidal or circular features, e.g. optical Earth observation images or medical images


For each pixel in the image compute the Birth Probability as \( \min\{\delta \cdot B(s), 1\} \), where:

\[
B(s) = \frac{b(s)}{\sum_s b(s)}
\]

Being \( b(s) \) the Birth Map computed from the Canny Contour Map.
For each ellipse $x_i$ in the configuration compute the **Death Probability** as $d(x_i)$, where

$$d(x_i) = \frac{\delta \cdot a(x_i)}{1 + \delta \cdot a(x_i)}$$

and

$$a(x_i) = e^{-\beta (U_L(x_i|I_g) - U_L(x|I_g))} = e^{\beta U^I_L(x_i|I_g)}$$

The complete **Flowchart** of the **Death Step** is as follows:

1. **Computation of the Energy Terms**
2. **Computation of Death Prob. based on Likelihood**
3. **Elimination of ellipses with prob. $d(x_i)$**
4. **Configuration refinement thanks to Prior term**
**Similarity Measures**

**Hausdorff Distance**

\[
\text{Similarity} = \frac{1}{\text{c}} \left\{ \sum_{i=1}^{\text{N}_c} \sum_{t=1}^{\text{P}} d_H(x_i^c, x_t) \right\}
\]

- \(c\) = craters in Input Image
- \(\text{N}_c\) = sum(pixels in crater \(c\) in Input Image)
- \(\text{P}\) = sum(craters’ border pixels in Ref Image)
- \(x_i^c\) = coord of pixel \(i\) in crater \(c\) in Input Image
- \(x_t\) = coord of pixel \(t\) in Ref Image’s craters

**Mutual Information**

\[
\text{MI}(X, Y) = \sum_{x \in X} \sum_{y \in Y} p_{X,Y}(x, y) \log \left( \frac{p_{X,Y}(x, y)}{p_X(x) p_Y(y)} \right)
\]

- \(X\): pixel intensity in Reference Image
- \(Y\): pixel intensity in Input Image
- \(p_X(x)\): probability density function (pdf) of \(X\)
- \(p_Y(y)\): probability density function (pdf) of \(Y\)
- \(p_{X,Y}(x, y)\): joint pdf of \(X\) and \(Y\)

\(p_X(x)\) and \(p_Y(y)\) estimated through the corresponding image histograms.
Rotation – Scale – Translation Transformation

Transformation vector

\[ p = (t_x, t_y, \theta, k) \]

\( \{t_x, t_y\} \): Translations in \( x \) and \( y \)

\( \theta \): Rotation angle

\( k \): Scaling Factor

Matrix Formulation

\[
T_p(x, y) = \begin{pmatrix}
  k \cos(\theta) & k \sin(\theta) & t_x \\
-k \sin(\theta) & k \cos(\theta) & t_y \\
0 & 0 & 1
\end{pmatrix}
\begin{pmatrix}
  x \\
y \\
1
\end{pmatrix}
\]
Region of Interest Approach

$I_A(X, Y), I_B(x, y)$: Two Images
$I_B$: sub-image of $I_A$ such that $I_B(0,0) = I_A(x_0, y_0)$

$p_A = (T_x, T_y, \beta, \alpha)$: RST transformation vector transforming $I_A$ into $I_A^{tr}$
$p_B = (t_x, t_y, \theta, k)$: RST transformation vector transforming $I_B$ into $I_B^{tr}$

$I_B^{tr}(0,0) = I_A^{tr}(x_0, y_0)$

Given:

<table>
<thead>
<tr>
<th>Transformation: $p_B$</th>
</tr>
</thead>
<tbody>
<tr>
<td>Reference of Region: $(x_0, y_0)$</td>
</tr>
</tbody>
</table>

Find: Transformation: $p_A$

Expressing the transformation in Matrix Form

$$T_{p_A} = \begin{pmatrix} \alpha \cos(\beta) & \alpha \sin(\beta) \\ -\alpha \sin(\beta) & \alpha \cos(\beta) \end{pmatrix} \begin{pmatrix} T_x \\ T_y \end{pmatrix} = T_{p_A}(X, Y) = (X', Y')$$

$$T_{p_B} = \begin{pmatrix} k \cos(\theta) & k \sin(\theta) \\ -k \sin(\theta) & k \cos(\theta) \end{pmatrix} \begin{pmatrix} t_x \\ t_y \end{pmatrix} = T_{p_B}(x, y) = (x', y')$$

This should also hold

$$T_{p_A}(x + x_0, y + y_0) = (x' + x_0, y' + y_0)$$

From the image

$$\begin{align*}
X &= x + x_0 \\
Y &= y + y_0
\end{align*}$$

Plugging $T_{p_A}$ into this equation and replacing $x'$ and $y'$ according to $T_{p_B}$

$$\begin{align*}
k \cos(\theta) x + k \sin(\theta) y + t_x + x_0 &= \\
\alpha \cos(\beta)(x + x_0) + \alpha \sin(\beta)(y + y_0) + T_x &= \\
-k \sin(\theta) x + k \cos(\theta) y + t_y + y_0 &= \\
\alpha \sin(\beta)(x + x_0) + \alpha \cos(\beta)(y + y_0) + T_y &=
\end{align*}$$

Knowing $\alpha = k$ and solving in $P_1 = (0,0)$ and $P_2 = (-x_0, -y_0)$

$$p_A = \begin{pmatrix} -k \cos(\theta) x_0 - k \sin(\theta) y_0 + t_x + x_0 \\ k \sin(\theta) x_0 - k \cos(\theta) y_0 + t_y + y_0 \\ \theta \\ k \end{pmatrix}$$
RMS Error Computation

Ground Truth Transformation
\[ p_{GT} = (t_{x1}, t_{y1}, \theta_1, k_1) \rightarrow T_{p_{GT}}(x, y) = Q_{p_{GT}} \cdot [x, y, 1]^T \]

Computed Transformation
\[ p = (t_x, t_y, \theta, k) \rightarrow T_p(x, y) = Q_p \cdot [x, y, 1]^T \]

Error Transformation
\[ p_e = (t_{xe}, t_{ye}, \theta_e, k_e) \rightarrow Q_{p_e} = Q_p \cdot Q_{p_{GT}}^{-1} \]
\[ k_e = \frac{k_2}{k_1}, \quad \theta_e = \theta_2 - \theta_1 \]
\[ t_{xe} = t_{x2} - k_e(t_{x1}\cos(\theta_1) + t_{y1}\sin(\theta_1)) \]
\[ t_{ye} = t_{y2} - k_e(t_{y1}\cos(\theta_1) - t_{x1}\sin(\theta_1)) \]

\( (x, y) \in \text{Image}, [x', y', 1]^T = Q_{p_e} \cdot [x, y, 1]^T \)

\[ [x'] = k_e \begin{pmatrix} \cos(\theta_e) & \sin(\theta_e) \\ -\sin(\theta_e) & \cos(\theta_e) \end{pmatrix} [x] + [t_{xe} t_{ye}] \]

RMS Error:
\[ E(p_e) = \sqrt{\frac{1}{AB} \int_0^A \int_0^B (x' - x)^2 + (y' - y)^2 \, dx \, dy}, \quad \alpha = A^2 + B^2 \]

\[ E^2(p_e) = \frac{1}{AB} \int_0^A \int_0^B (k_e \cos(\theta_e) x + k_e \sin(\theta_e) y + t_{xe} - x)^2 + (-k_e \sin(\theta_e) x + k_e \cos(\theta_e) y + t_{ye} - y)^2 \, dx \, dy \]

\[ E^2(p_e) = \frac{\alpha}{3} (k_e^2 - 2k_e \cos(\theta_e) + 1) + (t_{xe}^2 + t_{ye}^2) - (At_{xe}^2 + Bt_{ye}^2)(1 - k_e \cos(\theta_e)) - k_e(At_{ye} - Bt_{xe}) \sin(\theta_e) \]