Automatic detection of lunar sub-km craters via deep learning

1. Introduction

**Purpose**
To detect sub-km craters from high-resolution lunar imagery using deep learning.

**Crater detection**
High accuracy impact crater databases are needed for estimating generated age of the body’s surface with crater chronology. 
- Crater chronology is a method for estimating generated age of surface from crater size-frequencies.
- Accuracy of crater chronology depends on accuracy of crater information (geoinformation and diameters).

Resolution of exploration data were improved. Sub-km craters can be found from these high-resolution imagers.
- Resolution of SELENE Terrain Camera Ortho (TCO) imagery is about 7.4 m/pixel.
To develop algorithms for detecting craters automatically is important. Manually detecting sub-km crater take an immense amount of time because number of crater is increase exponentially with decreasing diameter.

**Deep Learning**
- By designing the neural network architecture that has more multi layers, deep learning can represent the data more abstract.
- From supervised data, deep learning can learn optimal features that are needed for feature extraction process and data identification process.
- Especially, Convolutional Neural Network (CNN) keeps high performance in the field of computer vision.
- [1] suggest a possible beneficial effect of CNN at martian craters classification task.

2. Method

**Supervised data**
SELENE’s Terrain Camera (TC) Ortho imagery:
Divided a image into 3 regions. (2 regions: For learning, 1 region: For evaluation)

**Cotet Database:**
290~1220m in diameter craters, (measured manually)

**Learning**

- 1) Crop patches. Cropping size vary with crater diameter.
- 2) Resize all patches to same size.
- 3) Augmentation (rotation)

**Detection**

- Input image
- Probability image

**Datasets**

<table>
<thead>
<tr>
<th>Network architecture</th>
<th>Training</th>
<th>Test</th>
<th>label</th>
<th>Training Patches</th>
<th>Test Patches</th>
</tr>
</thead>
<tbody>
<tr>
<td>South, Center</td>
<td>North</td>
<td>non-crater</td>
<td>3204</td>
<td>411</td>
<td></td>
</tr>
<tr>
<td></td>
<td>non-crater</td>
<td>3204</td>
<td>411</td>
<td></td>
<td></td>
</tr>
<tr>
<td>North, South</td>
<td>Center</td>
<td>non-crater</td>
<td>3000</td>
<td>462</td>
<td></td>
</tr>
<tr>
<td></td>
<td>non-crater</td>
<td>3000</td>
<td>462</td>
<td></td>
<td></td>
</tr>
<tr>
<td>North, Center</td>
<td>South</td>
<td>non-crater</td>
<td>3492</td>
<td>339</td>
<td></td>
</tr>
<tr>
<td></td>
<td>crater</td>
<td>3492</td>
<td>339</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**3. Experiments**

**Patch size**
Prepared 2 patterns patches.
- 15pixel
- 25pixel

**Result of crater detection from TC ortho imagery**

**Feature of predicted craters:**
High contrast or shadow regions.

In result of p=25 model, miss labels were decreased than p=15 model.
- Context information is important for distinguishing non-craters from craters.

Multiscale craters detected by changing resolution of input images. (p=25 model)

3) Augmentation (rotation)

**4. Results**

**Result of patch based classification**

<table>
<thead>
<tr>
<th>Label</th>
<th>prediction</th>
<th>Negative</th>
<th>Positive</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>True Negative (TN)</td>
<td>False Negative (FN)</td>
<td>True Positive (TP)</td>
</tr>
</tbody>
</table>

Recall = \( \frac{TP}{TP + FN} \)
Precision = \( \frac{TP}{TP + FP} \)
F-measure = \( \frac{2 \cdot \text{Precision} \cdot \text{Recall}}{\text{Precision} + \text{Recall}} \)

In result of both patterns, F-measure is over 95 %.

**5. Conclusion**

- In patch based classification task, recall, precision, F-measure were over 90%.
- In detection task, CNN detected multiscale crater position. However, there were many miss labels.
- For applying crater chronology, crater diameter is important as well as position. So, we have to improve the method for estimating crater diameter.

**6. Future works**

- Considering supervised data.
- Suitable patch size and context area.
- Add other lunar data (e.g. DTM).
- Considering network architectures.
- Separate a network to 2 processes.