# COSMIC Semantic Segmentation Framework

(***C**ontent-based **O**bject **S**ummarization to **M**onitor **I**nfrequent **C**hange)* CL #18-4652

**Asher Trockman (Computer Science '19)**

*Advisor:* Dr. Don Roberts (University of Evansville) *Sponsor:* Dr. Lukas Mandrake (Jet Propulsion Laboratory, Caltech)

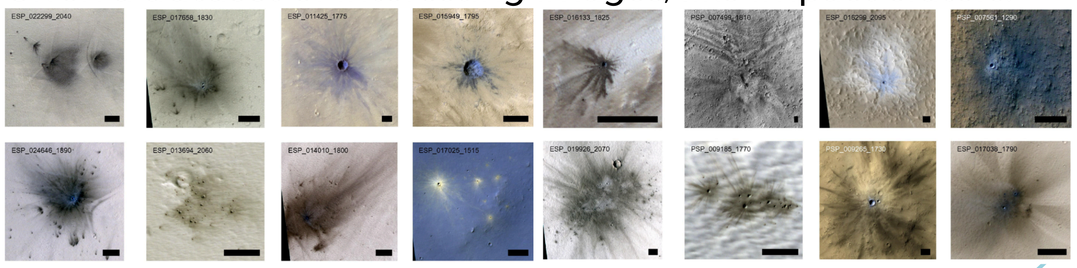
University of Evansville — Civic Mission…Sacred Trust — EST 1854

Jet Propulsion Laboratory, California Institute of Technology

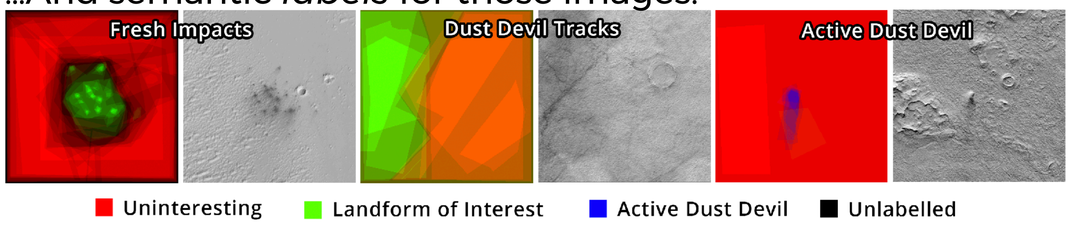
**Long-term Problem:** Deep space missions such as the Mars Reconnaissance Orbiter collect more data than can be sent back to Earth due to limited communications bandwidth.

**Long-term Solution:** *Machine learning algorithms* can be deployed on board orbiters to prioritize the downlink of scientifically interesting images to Earth, making better use of limited communications bandwidth.

Given a *dataset* of interesting images, like impact craters:



…And semantic *labels* for those images:

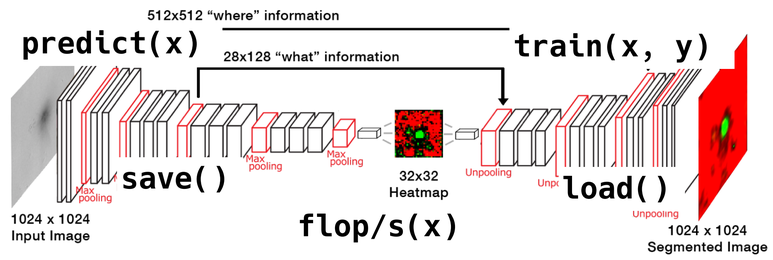



…We can train models to identify new, unseen, valuable images.

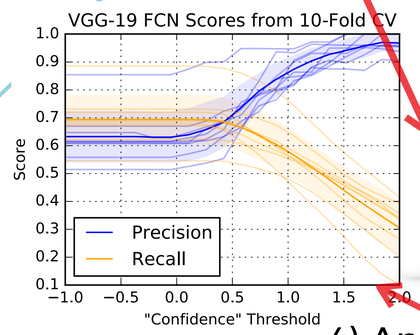
**Immediate Problem:** However, basic machine learning research is necessary to boost real-world performance on identifying these images, and numerous neural network architectures must be evaluated in terms of accuracy and compute requirements, which involves *software development challenges*.

**Immediate Solution:** A *framework* is designed to reduce redundant development, to standardize the algorithm testing process, and to allow developers to focus on the implementation details of novel machine learning algorithms.

1) *Observation*: ML **Models** have a common *interface*.




2) Using subsets of the **Dataset** of **LabeledImages** for training and testing…




3) …We can **Evaluate** models *consistently* after a *standardized* training **Curriculum.**


4) And **Log** the results in an easily-comparable format.

```
from cosmic import *
# 1. Easily modify or extend models
class VggNet19_Dice(Vgg19Model):
    name = 'VGG19/dice'
    def loss(self, x, y):
        yhat = self.heatmap(x)
        return dice_loss(y, yhat)

# 2. Test them in one line
Evaluator(VggNet19_Dice(),
    Dataset('./fresh_impacts'))

# ...many hours of computing
# 3. See results in ./VGG19/dice
```

**Conclusion:** The framework designed and the utility modules included will help researchers to efficiently test and compare new machine learning models with a torrent of newly-labeled data of the Martian surface.