DETECTING AND CHARACTERIZING COMPRESSION-RELATED ARTIFACTS IN MARS SCIENCE LABORATORY MASTCAM IMAGES.

H. R. Kerner[1], J. F. Bell III[1], H. Ben Amor[2] [1]Arizona State University School of Earth and Space Exploration, Tempe, AZ 85251 (hkerner@asu.edu); [2]Arizona State University School of Computing, Informatics, and Decision Systems Engineering, Tempe, AZ 85251.

Introduction: The Mastcam color imaging system on the Mars Science Laboratory Curiosity rover acquires images within Gale crater for a variety of geologic and atmospheric studies [1,2]. Images are often JPEG compressed onboard the rover before being downlinked to Earth. While critical for transmitting images on a low-bandwidth connection, this compression style can result in small image artifacts most noticeable as anomalous brightness or color changes within or near 8×8 JPEG compression block boundaries. In high-frequency detail regions of some images, for example in regions showing fine layering or lamination in sedimentary rocks, the image must be retransmitted losslessly to avoid introducing problems in the scientific interpretation of the data. The process of identifying which images have been adversely affected by such compression artifacts is performed manually by the Mastcam science team, costing significant expert human time and sometimes downlink volume.

In this work, we aim to automatically identify such problematic compression artifacts using a two-part machine learning solution that incorporated joint entropy as a measure of perceived information loss. Our proposed solution relies on: 1) a logistic regression model mapping the joint entropy between an uncompressed and compressed image to the probability of acceptance (by the scientist user); and 2) a convolutional neural network that learns to predict joint entropy from pixel information in the compressed image.

Background: Before being downlinked to Earth, Mastcam images are often compressed inside the camera's electronics using the JPEG compression algorithm. The JPEG compression algorithm splits an image into 8×8 pixel blocks, applies the discrete cosine transform, and quantizes the output in each block in order to reduce the footprint of the image while minimizing perceived loss of detail [3]. In high-detail regions of an image, this compression can result in visible artifacts at the block boundaries. Mastcam uses a Bayer RGB filter overlying the sensor [1], which can result in unnatural color changes along some block boundaries and magnified "zippering" when the image is compressed (Fig. 1). This loss of detail is prohibitive to study of some Mastcam images, such as in images used for measuring spacing and orientation of bed layers. Affected images are located manually in the image sequences and flagged for lossless re-transmission by the science team, expending significant expert human time and, potentially, significant downlink volume.

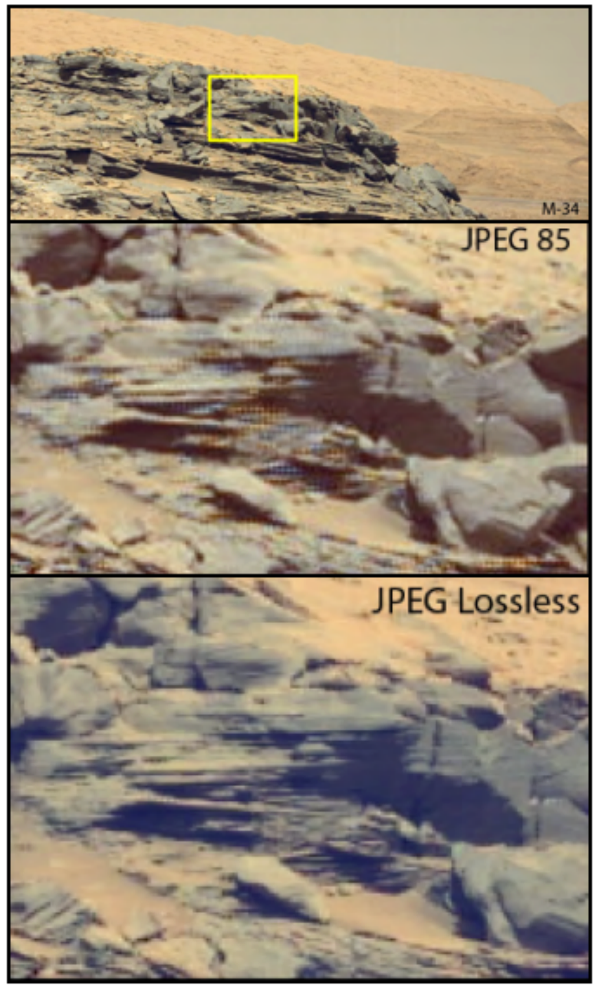


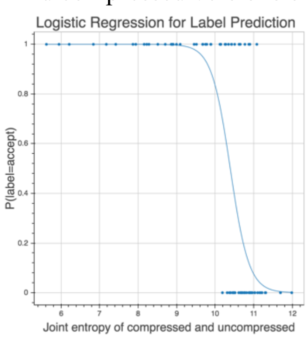
Figure 1 Mastcam M-34 image of finely-layered outcrop rocks acquired on Curiosity sol 1155 and sequence mcam05219. The middle inset shows a zoomed-in view of some of the layers in the originally-downlinked JPEG quality factor 85 image. The bottom image is an example of the same scene after re-downlinking the onboard image losslessly. (Adapted from [1])

Convolutional neural networks (ConvNets) have shown success in recent years learning complex structures in data. They are particularly well suited for image processing applications since they exploit local relationships in the data without restricting a relationship to a particular location in the image [4]. In traditional machine learning, a domain expert must determine what features are useful for discriminating between classes in the dataset, but ConvNets have the advantage of learning these features automatically. This is particularly useful when the features are not obvious or convenient to specify programmatically [4].

Methods: A supervised machine learning algorithm learns a mapping between features and classes by analyzing usually thousands to hundreds of thousands) of images labeled with the correct class, called a “training set”. Thus, there are two parts to any supervised machine learning problem: creating the training set, and then training/evaluating the model. Since labeling thousands of images is tedious and requires domain experts, we created a model for predicting the label for an unlimited number of images in training based on a smaller set of manually labeled images.

Training Set. The dataset was sourced from >7,000 images from the NASA PDS-released Mastcam database of uncompressed images called “Recovered Products” that were re-transmitted in the past. We create a batch for training by JPEG compressing one of these images to a random quality between 75 and 95. We slice the image into fragments of 100×100 pixels to “zoom in” to the level that artifacts are noticeable.

Automatic labeling. Rather than labeling the entire training set manually, we use a logistic regression model to predict the label of “acceptable” or “unacceptable” chosen by a human given the uncompressed and compressed versions of an image (Fig. 2). A single



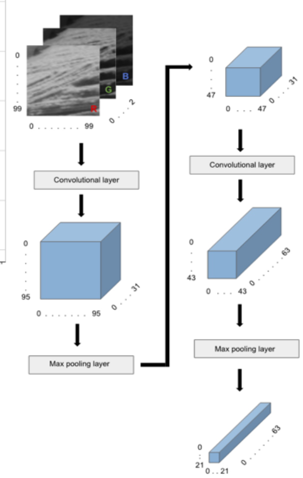
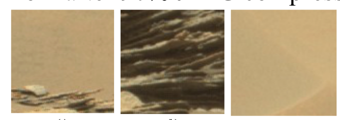


Figure 2: Top: Logistic regression model mapping joint entropy between two images to probability of image acceptability. **Right**: Convolutional stages of the neural network, computing image features and composing increasingly complex filters.

feature measuring information loss is used for classification: the joint entropy between an uncompressed and compressed version of an image. Joint entropy is a measure of statistical uncertainty between two distributions and has been used in image processing to represent the similarity (or lack thereof) between a pair of images [5]. This model is trained on 84 manually labeled images.

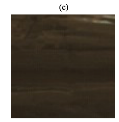
Network and training. We used TensorFlow to express the ConvNet architecture and perform training and testing. The input to the network is the 100×100 image fragment, segmented into three image color channels (red, green, blue). The network contains two convolutional layers, each followed by a max pooling layer to downsample the image in an effort to combat overfitting and reduce the number of features that must be computed. This is followed by a “dropout layer”, which randomly discards neurons and their weights to reduce overfitting, and finally the “readout” layer, which outputs the log odds of acceptability. We apply a softmax function to convert these log odds into probabilities.

Results: We trained our ConvNet on 7,690 image fragments. Fig. 3 shows the model output for a test on 4 image fragments and the image they were sliced from after 90% JPEG compression.



(a) (b) (c)

Image	Predicted entropy	P(accept)
(a)	10.765349	54%
(b)	12.412796	3.7%
(c)	10.276083	76%
(d)	10.258452	98%



(d)

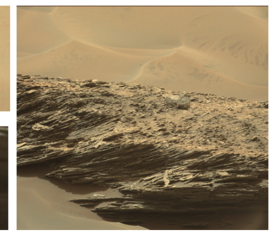


Figure 3 Predicted entropy and acceptance probability for image fragments from four different regions in the full image (right).

Discussion: The trained model correctly predicts highest information loss, and thus lowest acceptance probability, in regions of high detail (Fig. 3b) and vice versa for regions of low detail (Fig. 3c). In regions with moderate detail (Fig. 3a) the model is uncertain and gives a “hesitant yes”. In regions of shadow that are unlikely to be used in practice (Fig. 3d) the model confidently accepts quality.

Comparing the model’s predicted information loss and acceptance probabilities to the actual manual labeled model in Fig. 2, we see that the ConvNet has learned a good approximation for the user’s choices. Thus we have trained a model to reasonably predict the joint entropy between the compressed and uncompressed version of an image without ever actually seeing the uncompressed image or computing the entropy.

Future Work: While our ConvNet performs well according to the distribution it was trained with, the model used for labeling will be made more robust. We will investigate adding compression quality and the observation’s context to the feature space to improve the accuracy of the training set. We will investigate use of mutual information rather than joint entropy [5] as a measure of similarity between compressed and uncompressed images. Increasing the size of the image fragments may also improve generalization in labeling and in practice. After this tuning, we will do operational testing to potentially incorporate the system into the Mastcam operations pipeline.

References: [1] Bell *et al.* (2016) *Earth & Space Sci.,* http://molokai.sese.asu.edu/~jimbo/Mastcam_Calibration_2016.pdf, submitted. [2] Grotzinger J. P. et. al. (2012) *Space Science Reviews, 170(1),* 5-56. [3] Wallace G. K. (1991) *IEEE Trans. Consumer Electronics*. [4] LeCun Y et. al. (2015) *Nature, 521*, 436-444. [5] Pluim J. et. al. (2003) *IEEE Transactions on Medical Imaging*, 22(8), 986-1004.