Image Super-Resolution using a Multi Layer CNN

Ryan Baker, David Thibodeau

Overview

- Our project aims to take compressed or small resolution images and expand them to a larger size, and enhance or restore full size quality.
- A possible application would be to send images at a lower resolution and allow the recipient to be able to view a full quality image.
- Another application is that images taken with a lower quality camera could be improved
    - Security footage (License Plate Readers)
    - Older phones
    - Images with too high of an exposure

Our Dataset

- We found a dataset of 40,000 unique images of LEGO bricks rendered in blender
  - 50 Unique LEGO bricks, each rotated in 800 different orientations
- We chose this dataset because it contains a lot of simple shapes with very sharp edges and curves
- When compressed the image will lose the sharp edges, which we aimed to teach the model how to recreate
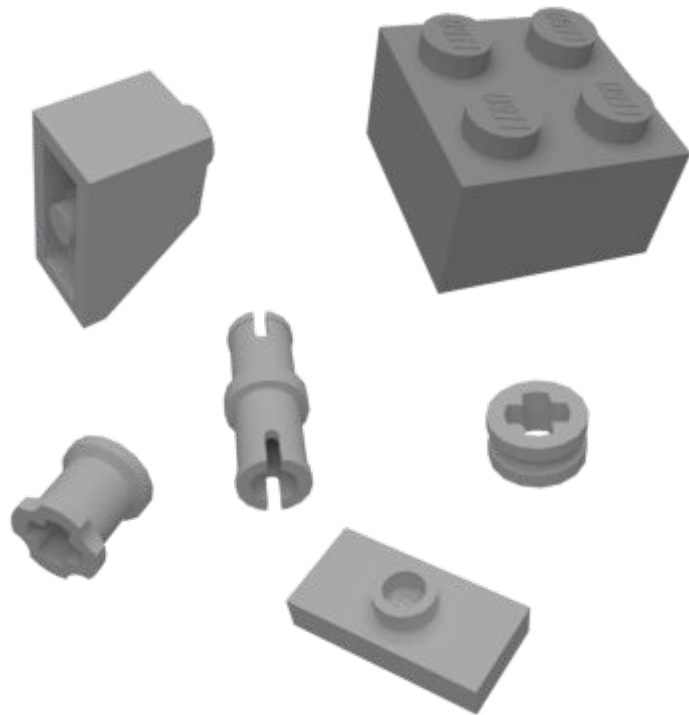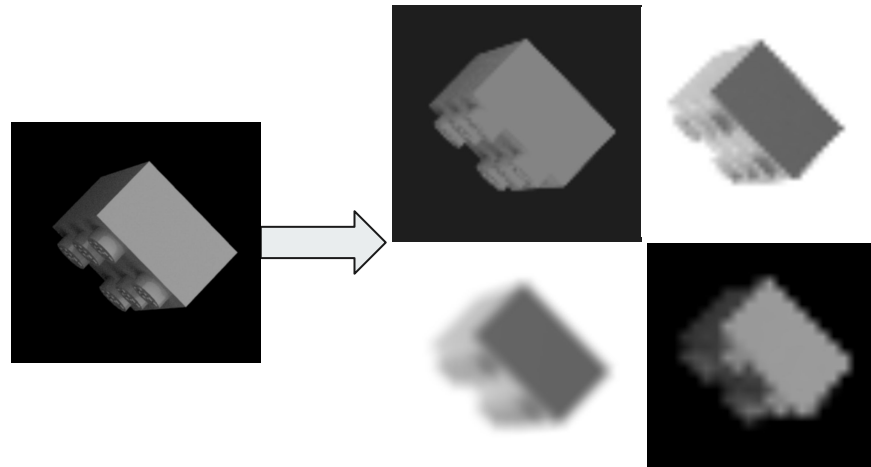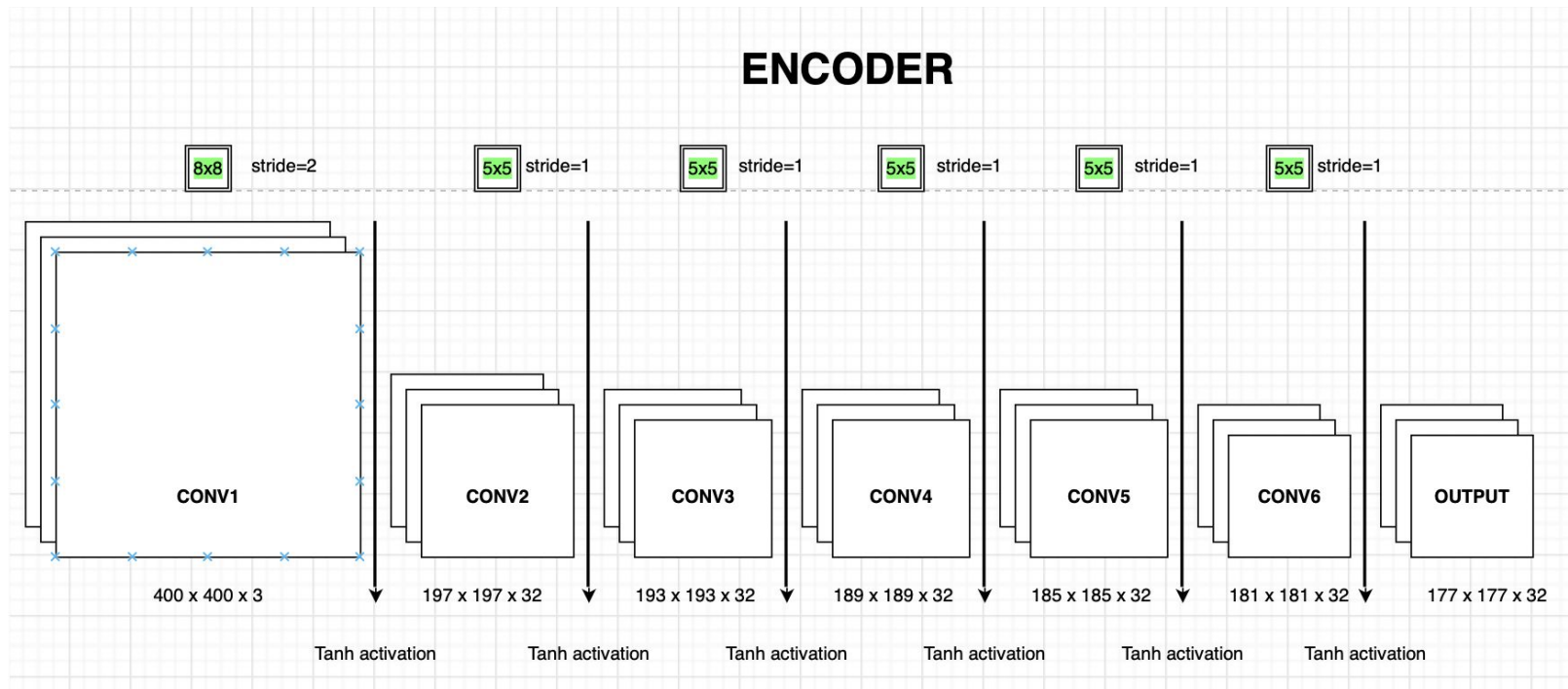
Image Preprocessing

- To create the input for our model, we applied 1 of 4 different transformations to the image
  - Apply a gaussian blur
  - Scale image down 12x then back up, then do one of the following:
    - Lighten the image to remove contrast
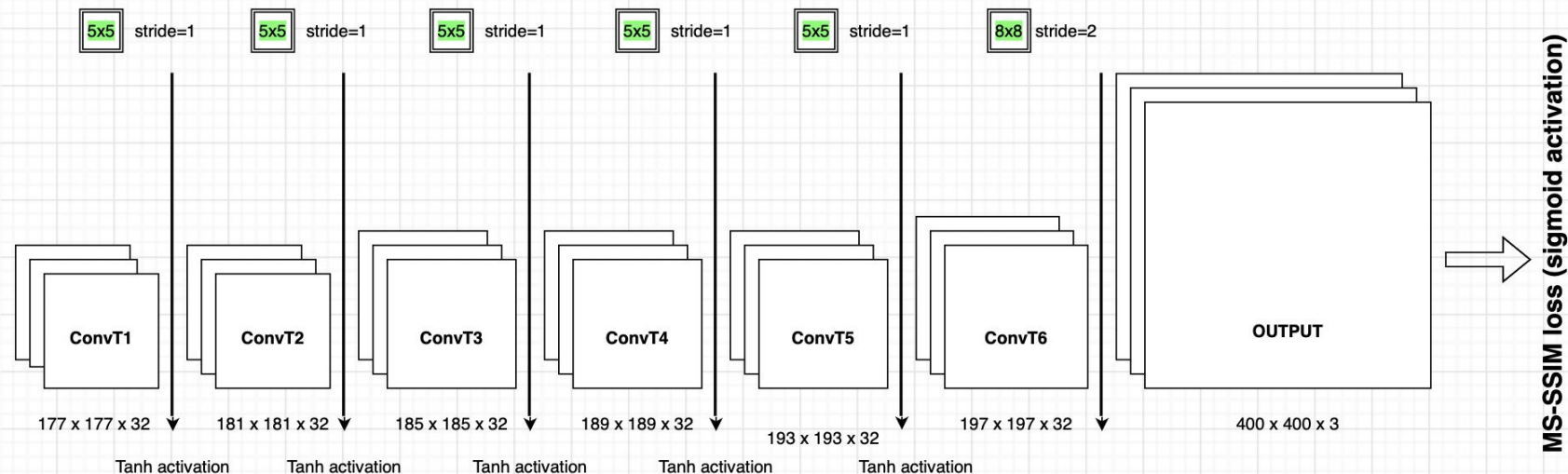   - Invert colors
    - Do nothing

Our Model's Architecture

ENCODER

| 8x8 stride=2 | 5x5 stride=1 | 5x5 stride=1 | 5x5 stride=1 | 5x5 stride=1 | 5x5 stride=1 |

CONV1   **CONV2**   **CONV3**   **CONV4**   **CONV5**   **CONV6**   **OUTPUT**

400 x 400 x 3   197 x 197 x 32   193 x 193 x 32   189 x 189 x 32   185 x 185 x 32   181 x 181 x 32   177 x 177 x 32

Tanh activation   Tanh activation   Tanh activation   Tanh activation   Tanh activation   Tanh activation

Our Model's Architecture

DECODER & LOSS

| 5x5 stride=1 | 5x5 stride=1 | 5x5 stride=1 | 5x5 stride=1 | 5x5 stride=1 | 8x8 stride=2 | |

ConvT1   ConvT2   ConvT3   ConvT4   ConvT5   ConvT6   OUTPUT

177 x 177 x 32   181 x 181 x 32   185 x 185 x 32   189 x 189 x 32   193 x 193 x 32   197 x 197 x 32   400 x 400 x 3

Tanh activation   Tanh activation   Tanh activation   Tanh activation   Tanh activation

MS-SSIM loss (sigmoid activation)

Loss Function

- We tried multiple different loss functions, and found Multi Scale-SSIM performed the best for our model
    - MSE was able to create a gray blob in the shape of the piece, but was unable to create any detail within the piece
- SSIM is the comparison of of luminance, contrast, and structure in a limited window for both images
- Using Multi Scale-SSIM, it performs the SSIM calculations on multiple differently sized windows.
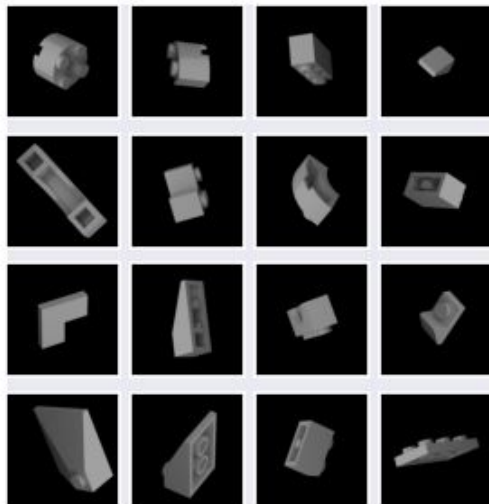
$$\text{SSIM}(x, y) = \frac{(2\mu_x \mu_y + c_1)(2\sigma_{xy} + c_2)}{(\mu_x^2 + \mu_y^2 + c_1)(\sigma_x^2 + \sigma_y^2 + c_2)}$$
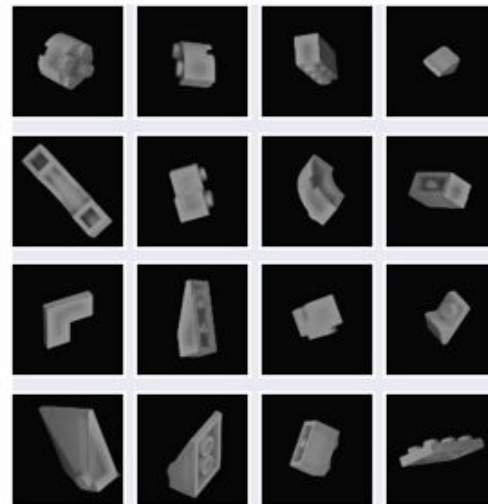
Results

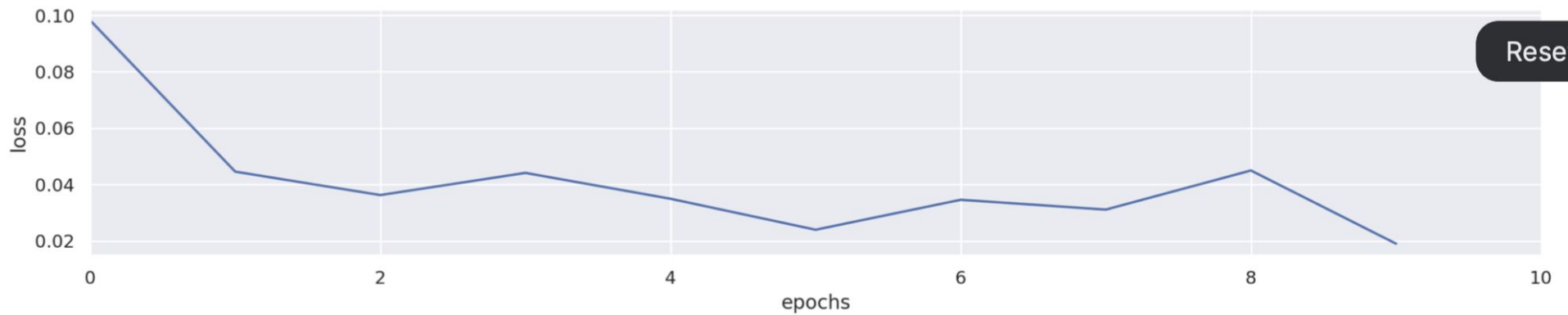compressed data (input)
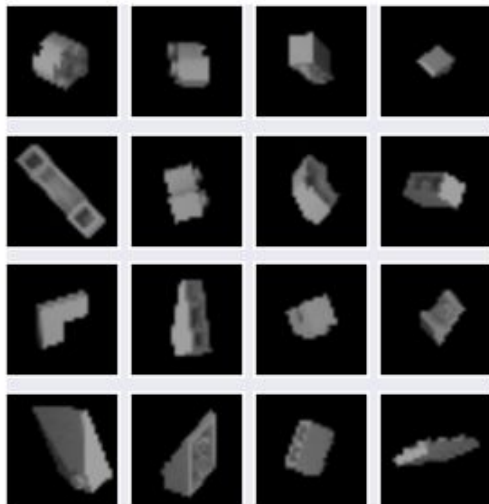
data (expected output)

reconstructed data (output)

Results (cont.)

Running the Model on Non-LEGO Images