Empirical Evaluation of Solving Jigsaw Puzzles Using Neural Networks

Jonathan Rodriguez[1], Vicente Ordonez[2]
[1]Tufts University, [2]University of Virginia
jonathan.rodriguez@tufts.edu, vicente@virginia.edu

1 Introduction

The subject of unsupervised learning brings forth an interesting problem in machine learning: How to learn from unannotated data. Most methods for training neural networks involve large amounts of data annotated by people, hence the name "supervised". Unsupervised learning requires no human input, making it more efficient, but in practice less accurate. In this project, we explore solving jigsaw puzzles, as a way of accomplishing unsupervised representation learning for visual recognition. In Jigsaw puzzles, the objective is to reconstruct an image that has been split into pieces by putting them together in the right order. We approach this problem by proposing a neural network model to perform this task. We experiment with several image resolutions, and number of pieces of the puzzle to see what settings work best for making the jigsaw puzzle a viable training method for neural networks.

2 Background

2.1 Convolutional Neural Networks

In machine learning, layers of fully connected neural networks are often used to train a machine to interpret data and perform various tasks. These fully connected layers are so called because each neuron in the new layer is connected to every neuron of the previous layer. For image analysis, this method could be very inefficient due to the size of the data input in each layer. Instead, we use convolutional neural networks, where the parameters for each neuron are its corresponding neuron in the previous layer and its neighbors. Due to the lower number of parameters, these networks are more efficient and easier to train than fully connected networks [1].

2.2 The Jigsaw Puzzle

A jigsaw puzzle was introduced in [2] as a novel way of performing unsupervised learning. The computer takes an image, divides it into parts, and shuffles those parts. However, while dividing, it chooses a slightly smaller piece from each tile to feed into the network. The machine then tries to reconstruct the image based on the cropped parts so that it cannot use the edges of each piece for reference.

In order to rearrange and reconstruct the pieces of the image, a list of permutations is created at the beginning. One of these is randomly chosen and the machine reorders the tiles following the order of the permutation. Then the goal is to correctly guess which permutation was chosen from the list (Figure 1 shows an illustration of this).

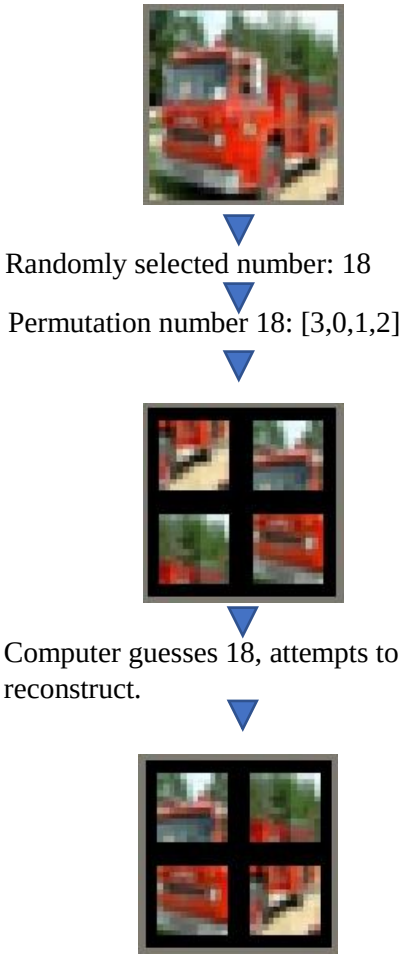

Figure 1: For each image, a random permutation is chosen from a previously saved list. The network attempts to guess the index of the permutation in the list and reconstruct based on the order of the guessed permutation. If it guesses correctly, the image is reconstructed.

1 Related Work

[2] presents the idea of using jigsaw puzzles as a method for unsupervised learning. However, this paper uses jigsaw puzzles as a pretext task, training a network for classification and detection. They also used techniques like color jittering and restricting the hamming distance in the permutations in order to further optimize the network. Here, the network is used is simpler, and the focus is on the jigsaw puzzle reconstruction as a task itself.

2 The Datasets

Two datasets were used in this project: CIFAR-10 and ImageNet. CIFAR-10 includes 50000 training images and 10000 test images, all 32x32. Due to the small size of the images, the whole image was used in the network, and they could only be used for the 2x2 jigsaw puzzle. Once divided into 16x16 tiles, a random 12x12 piece was cropped from each tile to be used in the model.

ImageNet consists of around 1.2 million training images and 50,000 validation images. For the 2x2 jigsaw puzzle, two different resolutions were used for comparison. The first was 32x32, and each image was used the same way as the CIFAR-10 images. Then, a 256x256 version of each image was used. A 224x224 image was cropped from the center and split into four 112x112 tiles. From these tiles, a random 96x96 piece was cropped and used in the model. For the 3x3 puzzle, the image was resized to 256 and a 225x225 part was taken from the center and split into nine 75x75 tiles. From each tile, a random 64x64 piece was used.

3 The Architecture

The network used (Figure 2) was roughly based on AlexNet [1] and the Context Free Network [2]. For the first part, each tile is introduced in some random order determined by the permutation, and the network runs on each piece individually. These convolutional layers have 96, 256, 384, 384, and 256 channels. ReLU is applied after each layer, as well as batch normalization and max pooling after the first, second, and fifth layers.

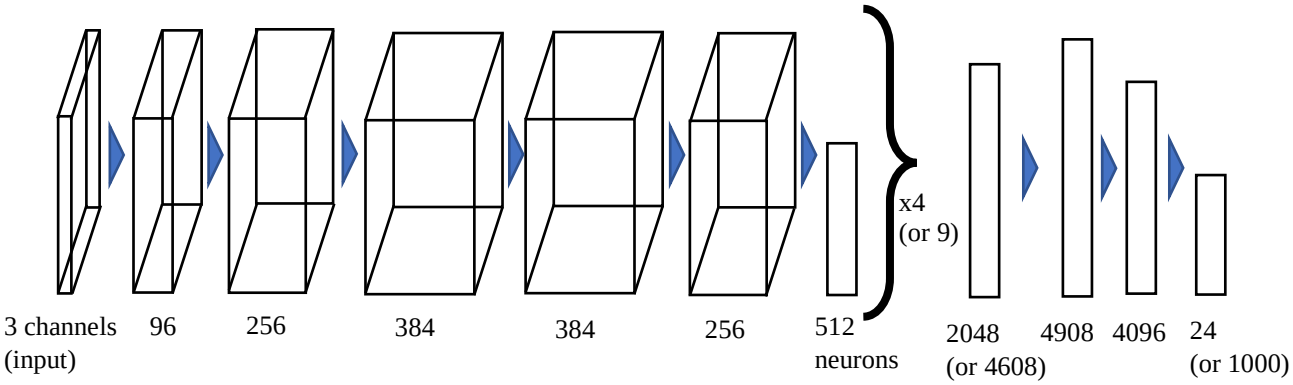

Figure 2: ***Neural Network Architecture*** *– The model takes a 32x32 or 224x224 image with three channels (RGB). It is then split into the appropriate number of pieces, and each piece goes through the first six layers individually. The number of channels in each layer are indicated in the figure. ReLU is applied after each layer, as well as max pooling and normalization after the first, second, and fifth. After the fifth layer, each piece is transformed into a one-dimensional tensor and a linear transformation is applied. The pieces are then stacked, and three more linear transformations are applied, with ReLU after the first two. The resulting tensor corresponds to the 24 (or 1000) possible permutations.*

The tensors are then rearranged into a 1d tensor and a linear transformation is applied with an output size of 512. These outputs are then stacked and fed into the second part of the network.

The stacked tensor goes through three more linear transformations of output sizes 4908, 4096, and 24 (or 1000 in the 3x3 jigsaw puzzle). After the first and second transformations, ReLU is applied. The final output indicates the permutation that was chosen for the shuffled image.

Class	Accuracy
Airplane	54.0%
Automobile	56.8%
Bird	59.2%
Cat	60.9%
Deer	63.2%
Dog	63.0%
Frog	65.5%
Horse	64.3%
Ship	65.2%
Truck	68.9%

Table 1: ***CIFAR-10 Accuracy per Class*** *– Best performance of the 2x2 Jigsaw Test in the CIFAR-10 dataset for each class.*

4 Results

4.1 Accuracy per Class in CIFAR-10

Overall, the model had the highest success rate reconstructing trucks, and the lowest when reconstructing planes. This could be due to the fairly uniform shape of trucks. Unlike most animals in the dataset, there is no drastic variation between types of trucks. They all have a similar front end, large wheels and windows, and usually a trailer. On the other hand, dogs and cats have several breeds with different proportions and cars have different models and types. Also, the shape is generally rectangular, and the truck cannot twist and turn like certain animals, which might make it harder to figure out the order of the pieces.

One factor that might contribute to the comparatively poor reconstruction accuracy of plane images is the wide range of angles

from which the image could have been taken. Since they can fly, pictures can be taken from any angle, not just the sides, front and back. In addition, images of planes in the air have less background information. With land-locked animals and vehicles, the background can give hints on the correct order of the pieces, but the computer cannot use it if it is completely blue. This might also be the case with birds, which also scored worse than most other categories.

4.2 2x2 Jigsaw Puzzle

In this test, I compared the accuracy scores of the 2x2 Jigsaw test on images of varying resolution. The 224x224 image set scored better than the lower-resolution images. Additionally, both 32x32 image sets had very similar performance. The CIFAR-10 set contains images from only 10 different classes, but ImageNet contains over one thousand, including several that are completely unrelated to those in CIFAR-10. This seems to suggest that the resolution plays a greater role in the reconstruction of the image than the class.

Dataset	Accuracy
CIFAR-10 (32x32)	58%
ImageNet (32x32)	58%
ImageNet (224x224)	81%

*Table 2: **2x2 Jigsaw Test Accuracy** – Best performance of the 2x2 Jigsaw Test in each dataset. The first is the CIFAR-10 dataset, which is already 32x32, second is ImageNet rescaled to a resolution of 32x32, and third is the 224x224 center of ImageNet images scaled to be 256x256.*

4.3 3x3 Jigsaw Puzzle

Overall, the 2x2 jigsaw puzzle performed better than the 3x3 one did. This result makes sense because a 2x2 jigsaw puzzle has only 24 possible permutations, while the 3x3 one had 1000 to choose from. However, it is still impressive how well it performed considering the size of the permutation set.

Dataset	Accuracy
ImageNet (2x2)	81%
ImageNet (3x3)	72%

*Table 3: 3x3 **Jigsaw Test Accuracy** – Best performance of 3x3 Jigsaw Test compared to 2x2 Jigsaw test. Both tests were on the 224x224 center of ImageNet images scaled to be 256x256.*

5 Future Work

In the future, it would be interesting to test more resolutions and see how they compare. Also, I would like to try training models using 4x4 and maybe even 5x5 jigsaw puzzles. Finally, I want to test models trained on differently sized jigsaw puzzles to solve other kinds of problems and see how the number of pieces impacts the adaptability of the model.

6 Acknowledgments

We would like to thank the UVA Dept. of Computer Science, in Partnership with DREU CRA-W, for hosting the internship, IAAMCS, and Access Computing for providing funding.

References

[1] A. Krizhevsky, I. Sutskever, & G. E. Hinton. ImageNet Classification with Deep Convolutional Neural Networks.

[2] M. Noroozi & P. Favaro. Unsupervised Learning of Visual Representations by Solving Jigsaw Puzzles.

[3] M. Noroozi, H. Pirsiavash & P. Favaro. Representation Learning by Learning to Count.