STA 208 Project Report

Members: Zachary Cosenza, Oliver Mikhail Gaerlan, Lahiru D. Chamain Hewa Gamage

Purpose

Faster and more accurate classification: Image classification in RGB or spatial domain has made tremendous progress over the past 15 years. In a cloud based image classification situation, a source device (a mobile phone) acquires an image X, sends it to the server to get the classification label **y**. When the source device sends X, it transforms (Red-Green-Blue) RGB/spatial domain information to other domains like discrete wavelet transform (DWT) or discrete cosine transform (DCT) with the purpose of compression to save bandwidth. So at the server end, X is transformed back to RGB domain before classification. We claim that this reconstruction step from DWT to RGB is unnecessary by training a classifier in DWT domain. Furthermore we show that faster and more accurate classification is possible with DWT domain based on the experiments on CIFAR-10 dataset.

Saving bandwidth with PCA: Image/text recognition requires a lot of data to training, so dimensionality reduction techniques principal component analysis (PCA) may be incorporated into preprocessing techniques before learners are used [2]. Additionally, how these dimensionality reduction techniques interface with data transform techniques such as DCT or DWT and their combined effect on test error of image classification is unknown. We show that in cloud based image classification, we can save bandwidth by transmitting only the projected PCA coefficients instead of the whole image to the server by training a classifier with images reconstructed with dimensionality reduced PCA coefficients of CIFAR-10 images.

Dataset

We used CIFAR-10 datasets for all the experiments related to this project. CIFAR-10 is a set of 60,000 RBG images of size 32 x 32 with ten classes (0-9) which will be used for image classification training and testing. The dataset is divided into a 50,000 training and 10,000 test set.

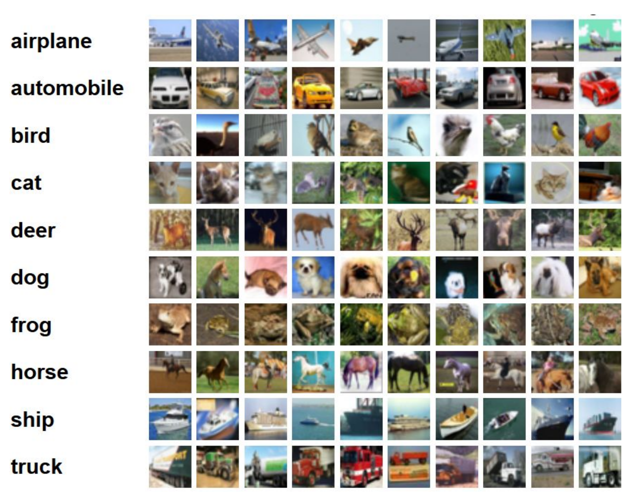


Figure 1 | CIFAR-10 Data Example

Methods

JPEG2000 Learning Pipeline: Assume a cloud based image classification scenario. Here source device (mobile phone) sends an inference image X over a bandlimited channel to the server to get the class label **y**. Server receives X and feeds it to a trained classifier to predict **y**. In order to conserve limited channel bandwidth and storage capacity, source devices often encode and compress **X** before transmitting to the cloud by utilizing standardized compression techniques such as JPEG2000 (Figure 2). Because most convolutional neural networks (CNNs) are designed to classify images in the RBG domain, the cloud currently receives and decodes the compressed j2k images back into the RGB domain before forwarding them to a CNN for further processing (note top part of Figure 2).

A natural question arises is to how to achieve faster training and inference with improved accuracy in a cloud based image classification under bandwidth, storage and computation constraints. We claim that the conventional use of image reconstruction is unnecessary for JPEG2000 encoded classification by constructing and training a deep CNN model with the DWT coefficients with CDF 9/7 wavelets (see the bottom part of Figure 2).

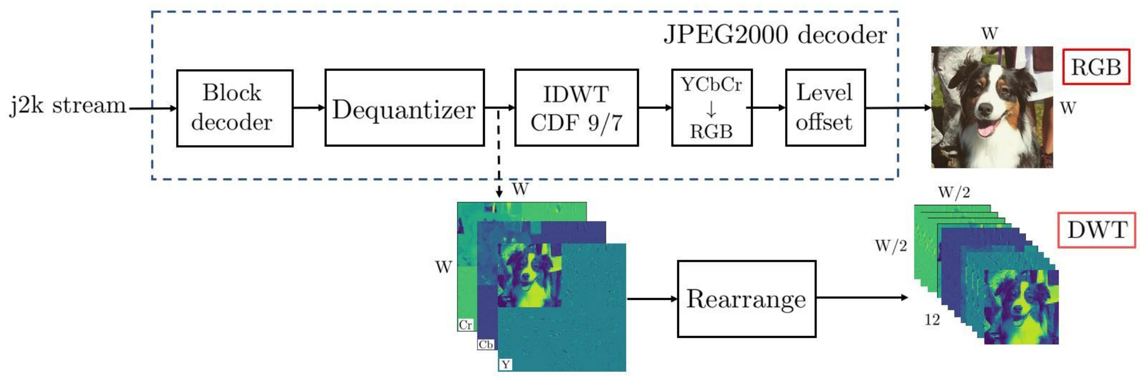


Figure 2 : Decoding pipeline of JPEG2000 decoder. We can extract the DWT coefficients in the middle of the decoder and feed it to the classifier

PCA: Briefly, PCA is used here as a dimensionality reduction technique in which a set of linearly uncorrelated principal components are constructed from an orthogonal transformation $T = XW$ where T is the transform of data X and eigenvector matrix W. We get fewer principal components from keeping fewer eigenvectors in W.

ResNet-8 Classification Algorithm: All classification is done using the well known ResNet-8 CNN with batch sizes of 32, 200 epochs of training 50,000 images and 10,000 test images. Even though the training set consists of 50,000 images, that is not enough to generalize the model. On the other hand we didn't use dropouts in the ResNet-8 to reduce overfitting. By randomly changing the images of each minibatch can help to solve the overfitting problem. For this, we used a custom data augmentation generator to randomly flip and shift the training images. We used an initial learning rate of 0.001 which is reduced progressively at 80, 120, 160 and 180 over 200 epochs. See the attached code for more details.

Bandwidth: To calculate the average bandwidth (BW) of a set of images we calculated the total number of bytes required for the loss less transmission of the set of images and calculated the average. For this, first we mapped the pixel/coefficient values of the dataset to a dictionary and calculated the frequency of each codeword. Then we generated binary strings to represent each codeword with Huffman encoding. To make sure that the codewords are integers, we quantized the pixel/coefficient values and used these quantized coefficients for later processes like reconstruction and classification. This way the effect of quantization is also reflected on the classification accuracy.

Experiments and Results

Experiment 1: Faster and accurate classification.

For all the experiments we used CIFAR-10 dataset. We trained a set of ResNet-8 models for CIFAR-10 dataset and following figures compare the test accuracy and speed for training and inference process.

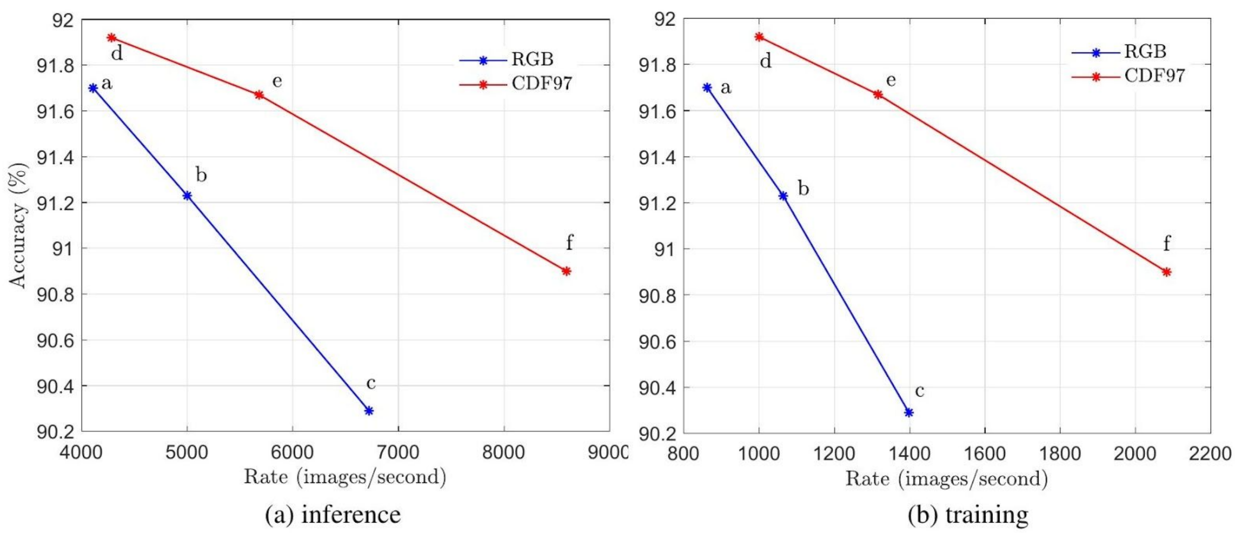


Figure: 3 Accuracy vs Inference and Training Speed

The above figure, (a) illustrates test accuracy vs inference speed for the CIFAR-10 data set. The blue lines represent results using RGB images. Red curve is the result using DWT coefficients with CDF 9/7 wavelets. (b) shows the Test error vs training speed/epoch. Here rate is the number of images that go through the model in each epoch. The proposed model delivers fast and accurate classification for both training and inference. The points a,b,c,d,e and f correspond to 6 different ResNet models. The following table summarizes these models.

Parameter/Domain	RGB			CDF 9/7		
Model	a	b	c	d	e	f
n	4	3	2	3	2	1
no of CONV layers	27	21	15	21	15	9
no of parameters (M)	0.37	0.27	0.18	1.79	1.17	0.55

Table 1: Parameters for ResNet-8 model.

Here ‘n’ is the number of residual blocks for each resnet stack in the network. Please see the code for more information.

Conclusion: We can see that DWT domain with CDF 9/7 wavelets facilitate faster classification with less number of convolution layers compared to RGB domain. But it needs more filters for each convolution layer demanding higher number of total parameters. This is because the DWT inputs are 16x16x12 compared to RGB 32x32x3. Since DWT inputs has more channels, it requires more filters for a convolution layer. But the DWT network is faster because the input image size is 16x16 compared to 32x32 RGB. All the rate/ speed calculation don’t take the time we save by not reconstructing the RGB domain into consideration. Together with this saving we can expect much faster cloud based image classification.

Experiment 2: Saving bandwidth with PCA.

In this section we describe 3 experiments to show that we can save bandwidth with PCA in 3 domains RGB, DCT and DWT.

Experiment 2.1 RGB Domain: In this experiment we take the original RGB domain and pass it through ResNet-8 using 200 epochs and batch sizes of 32. Additionally, we preserve 99% of the RGB image variance through *sklearn* PCA methods, reconstruct the image using the principle components, and use ResNet-8 to classify. BW was found for the full 32x32x3 image and PCA reduced image projection via Huffman encoding discussed in the Methods section. The PCA components can be plotted and the first few are shown in the RGB (Red) domain below in Figure 4. The same is done for the other domain and is in the code attached.

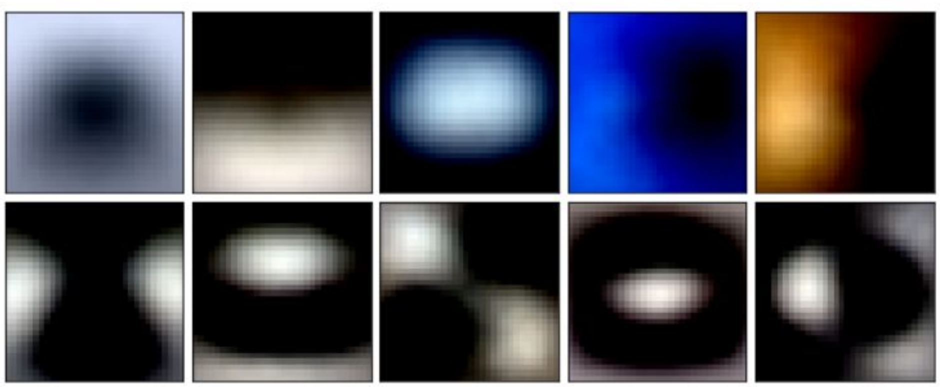


Figure: 4 PCA Components of RGB Domain

Conclusion: The BW (determined via Huffman encoding) for RGB domain is 2800 and 600 for full image and PCA coefficients respectively. This is to be expected as the full domain is larger than the PCA coefficients, suggesting PCA may be useful in improving efficiency of cloud image classification. The accuracy of the PCA reconstructed image is slightly lower than the full image using ResNet-8 (84.0% vs 88.0%) suggesting classification quality is not sacrificed due to image reduction. This can be seen visually in Figure 5, where the reconstructed images (32x32) look very similar to the unreconstructed images (32x32) with PCA retaining 99% of the image variance, showing why their classification accuracies are similar.

Figure 5: Visual Example of PCA Reconstructed and Original Images in CIFAR-10

Experiment 2.2 DCT Domain: This experiment will be similar to the previous experiment, except this time we will transform our image space using a discrete cosine transform. We will also be using comparing the bandwidth before and after PCA applied to the transformed image space. Each image is divided into 64 sub-images before applying the DCT. Each sub-image is 4x4 pixels each. Then the DCT is applied to each of the sub-images. The sub-images are combined to make an image of the original size. The figure below shows the PCA components after the DCT is applied.

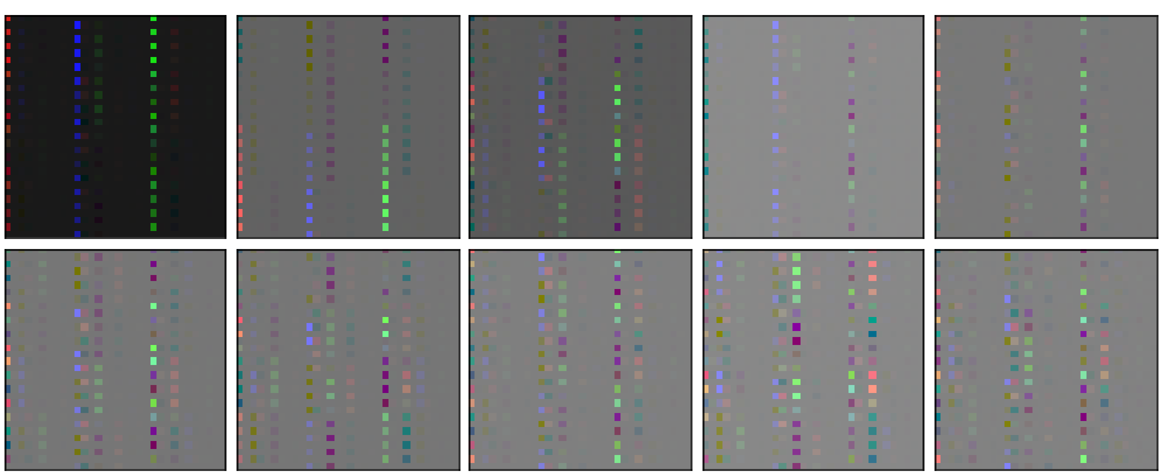


Figure 5: PCA components of DCT Domain

Conclusion: The PCA bandwidth savings were similar to the RGB domain. However, after 200 epochs of training, the ResNet-8 was only able to achieve a 65.9% accuracy on the test set. After PCA, the results were at approximately 64.8%. These results are better than the results from Fu and Guimaraes [2] but not as good as without the DCT.

Experiment 2.3 DWT Domain: In this experiment we start with Level 1 DWT coefficients with CDF 9/7 wavelet and explore methods to reduce the required BW in cloud based image classification. To create a baseline, we train a ResNet-8 with quantized CDF 9/7 DWT coefficients and measure the classification accuracy of CIFAR-10 dataset (say **a1**) and calculate the BW with Huffman encoding. (Say **BW1**). Then we perform a PCA on the vectorized dataset to reduce the dimensionality of data with the purpose of reducing bandwidth. We can calculate the average BW of these quantized PCA projections (say **BW2**). Then we reconstruct the wavelet coefficients to the original dimension and train a ResNet-8 to obtain **a2** accuracy. We observed that **BW2** is much smaller than **BW1** and **a1** and **a2** are considerably close.

We applied PCA to the vectorized DWT transformed images to preserve 99% of the variance. This reduced 3072 dimension to 687 values. Following are the first few principal components.

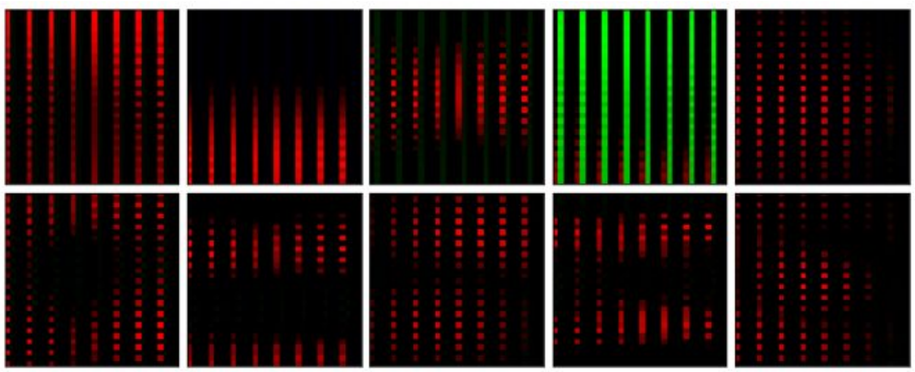
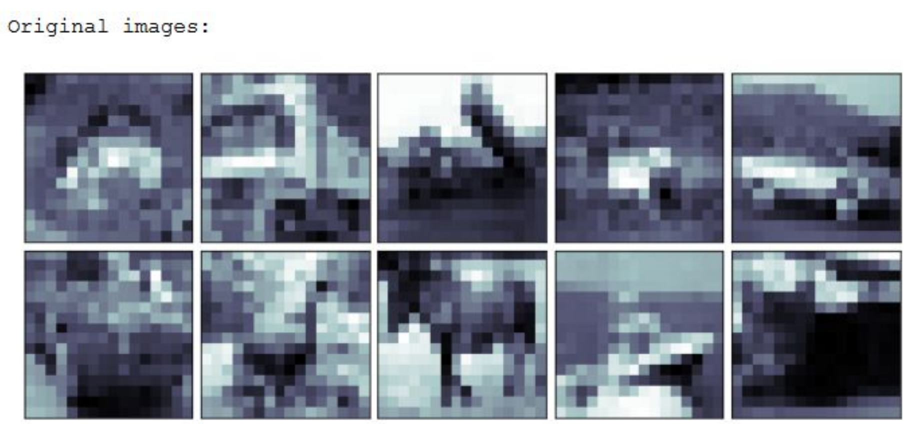


Figure 6: PCA Components of DWT Domain

Then we reconstructed the images from 687 values to see how good the reconstruction is after dimensionality reduction. Following are the original and reconstructed images. DWT transformed images of size (16x16x12) have 12 channels/subbands. Out of these 1st subband corresponds to the low frequency approximation of the original image in the Y channel. We plotted this subbands to visualize the reconstruction.



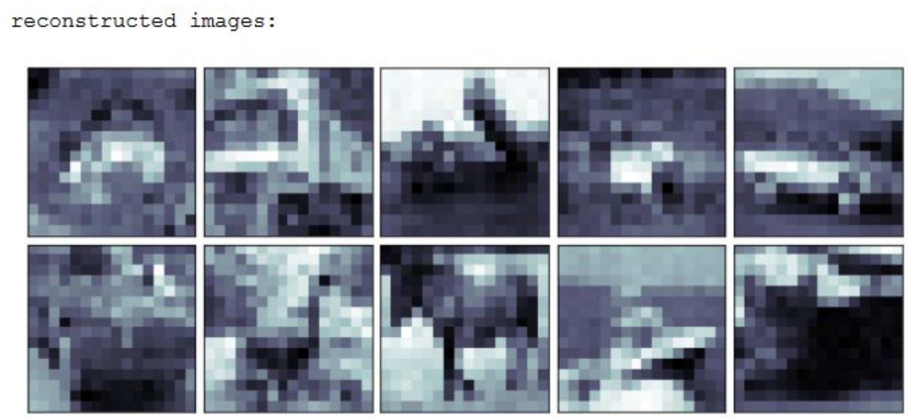


Figure 7: Original and reconstructed images in DWT domain.

We observed that the reconstructed images are almost the same as the original images images in DWT domain. this is only based on the 1st sub channel. One can transform the whole image to RGB and visualize to see exact quality of reconstruction. Then we trained a ResNet-8 model with original DWT quantized images and dimension reduced DWT quantized images and following are the results.

Method	Test accuracy (%)	Bandwidth (Bytes)
Original DWT quantized	90.0	1152
PCA DWT quantized	88.6	317

Table 2: Results for Accuracy and Bandwidth for DWT domain

Conclusion: This observation implies than we can save bandwidth by only sending the projected DWT coefficients rather than sending the complete image with a small accuracy loss. We can apply this in to practice like this. We can do a PAC analysis on a large dataset like ImageNet and store the principal components at the server. Source device can calculate the projections and transmit these projections consuming smaller BW. At the server end, high dimensional image can be reconstructed using the stored PCA coefficients and feed to a classifier.

Summary of Results for Experiment 2: We find that using the well known DWT transformation technique, we can achieve similar degrees of accuracy using the same CNN with a lower bandwidth image (both PCA and non-PCA variants) when compared to the standard RGB and DCT transform. This is encouraging as it indicates transformation techniques can be used in future image recognition pipelines without loss of learning capability and with improved efficiency, and that DWT transforms used in preprocessing may be the most efficient. This is especially true in mobile connected devices where image bandwidth is a design limitation on future CNN applications.

On the other hand sending the projections of the PCA components over the bandlimited network expecting reconstruction at the server end can save more than 75% of the bandwidth in RGB, DCT and DWT domains at the cost of small accuracy loss. We observe that image classification in the DCT domain is challenging with ResNet-8. This low classification accuracy in DCT compared to RGB and DWT can occur due to several reasons. One major reason can be the choice of the classification model. DCT represents spectral information of the image. But ResNets are designed for spatial inputs. Since RGB and DWT have spatial information, classification with sufficient accuracy is possible for these two domains. On the other hand, when we augment the training images, we use flips and shifts which are expected distortions in the spatial domain. These augmentations may not make sense in the spectral domain. This also can cause lower test accuracy.

Method	Test Accuracy (%)	Image Bandwidth (Bytes)
RGB • **Full Image** • **PCA**	 88.0 84.0	 2800 603
DCT • **Full Image** • **PCA**	 65.9 64.8	 2266 590
DWT • **Full Image** • **PCA**	 90.0 88.6	 1552 317

Table 3: Results for Accuracy and Bandwidth Experiments. 99% of PCA variance kept, epochs = 200 for ResNet - 8 classification CNN, batch size = 32

References:

[1] Abouelnaga, Yehya, et al. "Cifar-10: Knn-based ensemble of classifiers." *2016 International Conference on Computational Science and Computational Intelligence (CSCI)*. IEEE, 2016.

[2] Dan Fu, Gabriel Guimarães, et al. “Using Compression to Speed Up Image Classification in Artificial Neural Networks”