Combine Traditional Compression Method With Convolutional Neural Networks

Jianhua Hu
VimicroAI
Building 16,Hengqin Financial District,
Zhuhai city, Guangdong Provice, P.R.C
li.ming@zxelec.com

Ming Li,Changsheng Xia,Yundong Zhang
VimicroAI
Building 16,Hengqin Financial District,
Zhuhai city, Guangdong Provice, P.R.C

Abstract

Deep learning, e.g., convolutional neural networks (CNNs), has achieved great success in image processing and computer vision tasks like classification, detection and image compression. We propose a method by combining convolution neural networks and traditional compression method. The prepositive compression comes from the SVAC2(which is drafted and maintained by VimicroAI and China's Ministry of Public Security) video codec. We further improve the SVAC2 by adopting a recovering CNN network after the reconstruction. Our approach outperforms JPEG/JPEG2000/WebP standards, and is equivalent to BPG.

1. Introduction

Image/video comression is always a key technology in different industrial and commercial market[1]. In recent years, it attracts more interest because people founded that deep natural network the intrinsic power to map any transfer function, including image compress/decompress. So DNN has the potential to replace or improve the current image/video comression scheme. Image compression systems based on convolutional neural networks have become an active area of research recently. The aim of image compression is to reduce redundancy of an image in order to store or transmit the image at low bit rates. Although the end-to-end CNN auto-encoder may have competitive performance for the image compression, yet it has not exceeded the mainstream image compression standards like jpeg2000/webP. Our SVAC2 proposal does not only improve the traditional codec method, but we also use a CNN network to filter the image after SVAC2. Our framework takes advantage of SVAC2 and the CNN enhancement is optional, so it has much better performance and compatibility.

1.1. Previous Work

There are many image codec standards such as JPEG, WebP and BPG. WebP and BPG are based on the I frame codec of VP8 by GOOGLE and HEVC seperately. A usual image codec compress the image with one block as unit, with block size from 16x16 to 64x64. Each block use the same coding method including intra prediction, transform coding, quantization coding, de-locking filter, entropy coding and so on. BPG and WebP further use coding unit tree instead of simple macro-block used in JPEG. After reconstructing image, traditional image codec commonly improve the quality of decoded images by using post-processing techniques, which can be roughly categorized into de-blocking oriented and restoration oriented methods. The de-blocking oriented methods focus on removing blocking and ringing artifacts of the decoded images.

On the other side, auto-encoder has been used to reduce the dimensionality of images [2], convert images to compressed binary codes for retrieval [3], and to extract compact visual representations that can be used in other applications. Further, deep learning has been used both for lossy and lossless image compression and achieved competitive performance. For the lossy image compression Google proposed a fully deep learning architecture based on convolutional and de-convolutional LSTM recurrent networks.

2. The Proposed Compression Methods

In this section, we first introduce the architecture of the proposed method. Firstly the improved algorithms of SVAC2 in the traditional side and then the detailed CNNs architecture. With the raw input image in RGB channels, we transfer it into yuv420 format, and then encode to bitstreams with SAVC2 encoder. After reconstructing the yuv420 data by the SVAC2 decoder, we use a CNN network to filter only the Y data. Finally after interpolation chroma u/v, we get yuv444 data and transform it into the RGB data for PNG packing.

2.1. SVAC2 image codec introduction

As shown in Fig.1, SVAC2 I-frame encoder framework consists of intra prediction, transform coding, quantization coding, de-locking filter, entropy coding as same as other mainstream codec architecture. The codec can be split in two major parts: 1) encoder, 2) decoder. Codec a I frame follow the steps below:

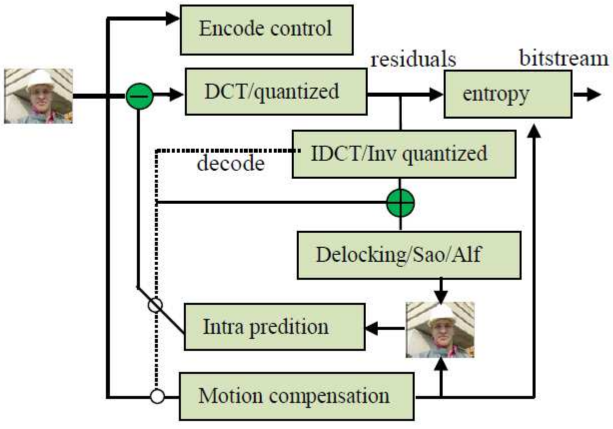

Figure 1. SVAC2 codec compression framework

Step 1. Image is partitioned in to blocks of different sizes and is known as coding tree unit(CTU).

Step 2. Each unit is predicted by intra-frame prediction. The result of prediction process is subtracted from the picture block[4].

Step 3. The residual is transformed mostly by Discrete Cosine Transform(DCT) and quantized. Finally, transformed output, prediction information, mode information and headers are encoded by entropy encoder.

Step 4. At decoder side every counter part of encoder blocks does the inverse operation to deliver the picture to the other end of the communication.

2.2. SVAC2 multi-scale residual encoding

We introduced a innovative encoding scheme in SVAC2, which has two encode modes : The normal mode, encode the whole image as a intra frame(I frame). The second mode called multi-scale residual encoding, as shown in Fig.2. This mode firstly encode a small image which is 1/2(1/4,1/8optional) down-scaled from the original one, as a I-frame. Then encode the original image as a inter frame(P frame), with the former small image up-scaled as the reference frame. So the second frame actually encode the residues between original image and up-scaled frame.

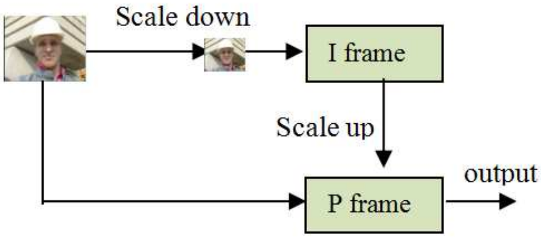

Figure 2. SVAC2's multi-scale residual encoding scheme

3. Reconstruction Frame Filtering With CNNs

Let's denote lumina of the SVAC2 decoded image as Y. The proposed CNN model focuses on learning the residuals between the decoded Y and the ground truth lumina X of source image. Our goal is to fit a mapping function X≈F(Y)+Y that reverses image degradation due to SVAC2 codec as much as possible. We wish to learn the F by training a CNN, which conceptually consists of two operations as shown in FIG.3. It has the feature extraction and image detail's reconstruction. The filter reconstruction frame model is a fully CNN network that consists of a set of convolution layers and non-linear layers cascades. To extract both the local and the global image features, all outputs of the hidden layers are concatenated at the end of feature extraction as skip connections from different layer domains. After concatenating all of the features, reconstruction par is used to reconstruct the image details. Input Y is fed into the network, residual is output from the second last layer, finally adding Y to form a F(Y)+Y function. The final addition is inspired by Res-Net. The model has totally 11 layers.

For colored images, we first transform RGB to YUV. And the reconstruction network is applied only on the luminance channel. The model is not specifically designed to be an end-to-end solution. On the contrary, the proposed optimizes an end-to-end mapping. It is faster at speed because of less layers and channels. It is not only a quantitatively superior method, but also a practically useful one.

3.1. Feature Extraction

The feature extraction part is responsible for extracting hidden features of the SVAC2's reconstructed image. It consists of 7 consecutive 2d-convolution layers. Each layer can be expressed as the equation:

$$E(Y) = W * Y + B$$

where W and B represent the filters and biases respectively, and * denotes the convolution operation. Here, W is a k×k×M×N matrix. k is convolution kernel size. M,N is the numbers of input/output channels. B is a vector of size N. E is the output feature maps of H×W×N dimen-

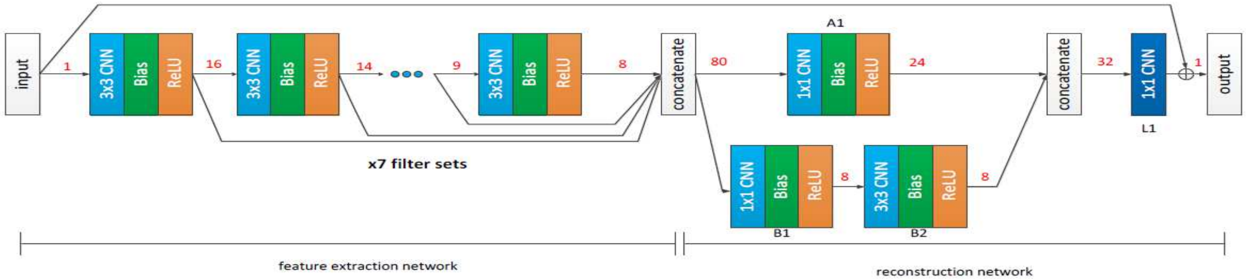

Figure 3. CNN filter structure

sion where H,W is the feature map's size. We use Relu as activation function for each layer.

We optimize the number of filters at each layer. The 7 filters with output feature num N are as follows: 16,14,12,11,10,9,8. All filter kernel size is k=3. The concatenation concatenates all layers' outputs, therefore the channels num in concatenation is 80.

3.2. Image Detail Reconstruction

Because of all of the hidden features are concatenated at the input layer of the reconstruction network, the dimension of input data is rather large. So we mainly use 1x1 CNNs to generate output residual pixels data, not only reduces the dimensions of the previous layer for faster computation with less information loss, but also adds more nonlinearity to enhance the potential representation of the network[5]. The reconstruction network have three CNN filters:A1, B1, B2,Fig.3. A1 use 1x1 CNNs and output channel number is 24, B1 use 1x1 CNNs and output channel number is 8 and B2 use 3x3 CNNs and output channel is 8. The cascades of B1/B2 is finally concatenates with output of A1, following a 1x1 CNN as final mapping. It can been concluded this branching method reach a compromise between size of CNN kernel and number of features. Branch A1 is of smaller 1x1 kernel but with more channels(24 here). Meanwhile branch B1/B2 contains larger kernel(1x1 and 3x3 here) but fewer channels(8 here). This arrangement balance the usage of different receptive fields and computation requirements.

The output residual is then added with the original input Y. Actually this is a typical residual learning networks, the model is made to focus on learning residual output and this greatly helps shorten the training time.

4. Training

The loss function we used is the MSE between input and output. The formula is as:

$$Loss = \frac{1}{n}\sum_{n=1}^{n} \|F(Y) + Y - X\|^2$$

where Y is the decoded luminance from SVAC2, F(Y)+Y is the whole CNNs function, X is the unencoded luminance and n is the training mini-batch size.

We constrain the size of the encoded CLIC2018 test images within the 13.3M limitation by qp48. So the training inputs are Y patches decoded by SVAC2 decoder using qp48. All patches are cropped from CLIC2018 training set plus BSD200[6], without overlapping. Patch size is 64x64. Using MSE as the loss function favors a high PSNR. The PSNR is a widely-used metric for quantitatively evaluating image restoration quality, and is at least partially related to the perceptual quality. We also found that more training samples do not improve the performance.

5. Result and Discussion

As for validation data of CLIC2018, after submitting the decoder, our PSNR result is 30.48 db. About 0.2 db is contributed by the later CNN network. We found that all images are PSNR boosted without exception while we use reasonable additional computation resources. We will use the similar techniques to replace or improve the SVAC2's multi-scale residual encoding(see section2.2) in the future.

References

[1] G. K. Wallace. *The jpeg still picture compression standard*, IEEE transactions on consumer electronics, vol. 38, no. 1, pp. xviiixxxiv, 1992.

[2] G. E. Hinton and R. R. Salakhutdinov. *Reducing the dimensionality of data with neural networks*, Science, 313(5786):504507, 2006. 1

[3] A. Krizhevsky and G. E. Hinton. *Using very deep autoencoders for content-based image retrieval*, In European Symposiumon Artificial Neural Networks, 2011. 1

[4] Dhruti Patel, Tarun Lad and Dharam Shah. *Review on Intra-prediction in High Efficiency Video Coding (HEVC) Standar*, International Journal of Computer Applications (0975 8887), 2015. 12

[5] Jin Yamanaka1, Shigesumi Kuwashima1 and Takio Kurita2. *Fast and Accurate Image Super Resolution by Deep CNN with Skip Connection and Network in Network.*, 24th International Conference On Neural Information Processing (ICONIP 2017), 2017. 9

[6] Arbelaez, P., Maire, M., Fowlkes, C., Malik, J. *Contour Detection and Hierarchical Image Segmentation*, IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 33, no. 5, pp. 898916 (2011)