CONVOLUTIONAL NEURAL NETWORK-BASED INVERTIBLE HALF-PIXEL INTERPOLATION FILTER FOR VIDEO CODING

Ning Yan¹, Dong Liu¹, Bin Li², Houqiang Li¹, Tong Xu¹, Feng Wu¹

¹CAS Key Laboratory of Technology in Geo-Spatial Information Processing and Application System, University of Science and Technology of China, Hefei 230027, China
nyan@mail.ustc.edu.cn, {dongeliu,lihq,tongxu,fengwu}@ustc.edu.cn
²Microsoft Research Asia, Beijing 100080, China, libin@microsoft.com

ABSTRACT

Fractional-pixel interpolation has been widely used in the modern video coding standards to improve the accuracy of motion compensated prediction. Traditional interpolation filters are designed based on the signal processing theory. However, video signal is non-stationary, making the traditional methods less effective. In this paper, we reveal that the interpolation filter cannot only generate the fractional pixels from the integer pixels, but also reconstruct the integer pixels from the fractional ones. This property is called invertibility. Inspired by the invertibility of fractional-pixel interpolation, we propose an end-to-end scheme based on convolutional neural network (CNN) to derive the invertible interpolation filter, termed CNNInvIF. CNNInvIF does not need the “ground-truth” of fractional pixels for training. Experimental results show that the proposed CNNInvIF can achieve up to 4.6% and on average 2.2% BD-rate reduction than HEVC under the low-delay P configuration.

Index Terms— Convolutional neural network, High Efficiency Video Coding, interpolation filter, invertibility.

1. INTRODUCTION

Fractional-pixel interpolation is widely adopted in the video coding standards to improve the accuracy of motion compensated prediction. The existing fractional-pixel interpolation filters can be divided into three categories: fixed-interpolation filter [1, 2, 3, 4], adaptive interpolation filter [5, 6] and learning-based interpolation filter [7]. The video coding standards mostly adopt fixed interpolation filters for the sake of low computational complexity and ease for hardware implementation. In earlier years, bilinear interpolation was usually adopted. Later on, more efficient filters have been investigated. For example, in MPEG-4 AVC/H.264, 6-tap filter is adopted to perform half-pel interpolation and the simple average filter is used for quarter-pel interpolation for luma component [2]. In the latest video coding standard HEVC, the DCT-based interpolation filter (DCTIF) is adopted [3]. The fixed interpolation filters are designed with the premise that the video signal is ideal low-passed. However, the video signal is actually not low-passed and not stationary. Further study has been conducted to design content dependent filters. In [5], an adaptive filter based motion compensation scheme is proposed, in which the filter coefficients are estimated for each frame, according to the content of the frame. Later on, a separable adaptive interpolation filter is proposed in [6], which can reduce the computational cost while maintaining the coding efficiency of non-separable adaptive filter.

Recently, convolutional neural network (CNN) has attracted numerous attention and achieves great success in many high-level computer vision tasks [8, 9]. Then CNN is introduced into some image processing tasks, such as image super-resolution [10] and artifact removal [11]. More recently, CNN has been applied into video coding [7, 12]. In our previous work [7], we investigated a CNN-based interpolation filter. To train a CNN, we proposed to blur a high-resolution image and then extract the pixels at odd and even locations as integer and half pixels, respectively. However, the generated “half pixels” by this manner were indeed not true. It remains a difficulty to train CNN-based interpolation filter because the “ground-truth” of fractional pixels is not available.

In this paper, we present a CNN based invertible fractional interpolation filter for video coding. We first reveal the invertibility inherent in the fractional interpolation problem, that is the integer pixels should also be recovered from the fractional pixels with the same interpolation filter. Inspired by the invertibility of fractional interpolation, we design an end-to-end scheme based on the popular convolutional neural network to derive the invertible interpolation filter, namely CNNInvIF. Different from [7], we do not need the fractional pixels as “ground-truth.” After training, the CNNInvIF is integrated into HEVC for evaluating its performance in video coding. In
this paper, the CNNInvIF is applied only to half-pixel samples of luma component. Experimental results show that the proposed CNNInvIF leads to up to 4.6% and on average 2.2% BD-rate reduction under low-delay P configuration.

2. PROPOSED METHOD

In this section, we will firstly introduce the invertibility inherent in interpolation problem, which inspires us to design the invertible interpolation filter. Then we will introduce the proposed end-to-end training scheme to derive the invertible interpolation filter.

2.1. Invertibility of half-pixel interpolation problem

Half-pixel interpolation is a process of generating half pixels from integer pixels. According to digital signal processing theory, integer and fractional pixels are all sampled from the original continuous signal. Typical methods of half-pixel interpolation are based on the signal processing theory, which generate a half pixel as a linear combination of its neighboring integer pixels, such as the DCTIF [13] adopted in HEVC. In this paper, we reveal an inherent property in the fractional interpolation problem, that is the interpolation filter should not only generate the fractional pixels from the integer pixels, but also recover the integer pixels from the fractional ones. Fig. 1 (a) illustrates the typical generating process of the horizontal half pixels, in which the blue points stand for the integer pixels and the red points stand for the fractional pixels. We can horizontally flip Fig. 1 (a) and obtain Fig. 1 (b). In Fig. 1 (b), we can regard the red points (half pixels in Fig. 1 (a)) as integer pixels and the blue points (integer pixels in Fig. 1 (a)) as half pixels. Therefore, if there exists an ideal interpolation filter which can generate the half pixels in Fig. 1 (a), the integer pixels can also be recovered by applying the same interpolation filter on the generated half pixels. We call this property invertibility of the fractional interpolation, which measures the ability of recovering the integer pixels from the generated fractional pixels.

2.2. The overall training framework

To fulfill the proposed CNN based invertible interpolation filter, namely CNNInvIF, we design an end-to-end training scheme, as illustrated in Fig. 2. As shown in Fig. 2, the proposed scheme consists of two parts: the first part generates the half pixels and the second part recovers the original pixels from the half pixels. A compressed picture is fed into the scheme, the first CNN plays the role of half-pixel generation. Then geometry flipping is performed on the generated half pixels. In this paper, the geometry flipping is applied horizontally, vertically, or diagonally, for the horizontal half pixels, vertical half pixels and diagonal half pixels, respectively. The picture consisting of half pixels after geometry flipping is fed into another CNN followed by the inverse geometry flipping, which recovers the original picture from the generated half pixels. Note that the two CNN modules in Fig. 2 share the same set of parameters. In practice, any fully convolutional neural network can be used as the CNN modules in Fig. 2. In this paper, we adopt the so-called Variable-filter-size Residue learning CNN (VRCNN) in [12], which was originally designed for post-processing. In the future, we will design more efficient network structures for fractional interpolation. Note that this scheme can play two roles: generating the half pixels and meanwhile reducing compression noise.

2.3. Objective function

We train the invertible interpolation filter based on CNN by jointly minimizing two loss functions that are known as invertible reconstruction loss and regularization loss, respectively.

2.3.1. Invertible reconstruction loss

Invertible reconstruction loss is used to measure the ability of recovering the integer pixels from the half pixels.

\[ L_{rec} = ||x - T^{-1}(F(F(x_c))))||_2^2 \]  

(1)

where \( x \) is the original picture, \( x_c \) is the compressed picture and \( x_f \) is the picture consisting of fractional pixels generated
2.3.2. Regularization loss

To produce a visually pleasant picture of half pixels, we consider a constraint on the output of the CNN for the invertible interpolation filter. Based on the assumption that the DCTIF is a good approximation of ideal low-pass filter [13], we design a regularization loss function. Let $DIF()$ be the DCTIF interpolation function, we define the regularization term as follows.

$$L_{reg} = \| x_f - DIF(x) \|_2^2 = \| F(x_c) - DIF(x) \|_2^2$$  (2)

2.3.3. Joint loss function

We combine (1) and (2) to achieve the final objective function with respect to the proposed scheme. The joint loss is:

$$L = (1 - \lambda) \cdot L_{rec} + \lambda \cdot L_{reg}$$  (3)

where $\lambda$ controls the relative weight of the regularization term. We empirically set $\lambda$ as 0.3 in our final results and we will investigate the effect of different values of $\lambda$ in our future work.

3. EXPERIMENTS

3.1. Training

We adopt the deep learning software Caffe [14] to train the invertible interpolated convolutional neural networks on an NVIDIA Tesla K80C graphical processing unit. We use the DIV2K dataset [15], which is a newly released high-quality image dataset for studying image super-resolution, to generate training data. The DIV2K dataset includes a collection of 800 high definition natural images. In this experiment, each image is compressed by HEVC intra coding at four different quantization parameters (QPs): 22, 27, 32 and 37. For each QP and each half-pixel position, a separate network is trained. Therefore, we finally have 12 convolutional neural networks. During the process of interpolation in video coding, a convolutional neural network will be selected according to the required half-pixel position and the corresponding QP of the frame. The nearest QP among 22, 27, 32 and 37 to the current frame QP will be selected. The DCTIF for half-pixel interpolation in HEVC is replaced by the CNNInvIF.

3.2. Evaluation of the invertibility performance

In this section, we conduct experiment to evaluate the invertibility of DCTIF and CNNInvIF. The first 10 pictures of the test sequences in Class D are used for evaluation. These pictures are firstly compressed with HEVC intra coding. The compressed pictures are used as input to the interpolation filter to generate the pictures of half pixels. The generated half-pixel pictures are then fed into the interpolation filter to reconstruct the original pictures. The above processes are performed iteratively for several times.

Fig. 3 gives the reconstruction PSNR at each iteration for each half-pixel position, when QP is 22. It can be seen that with the increase of iterations, the PSNR of both DCTIF and CNNInvIF declines. It reveals the fact that the interpolation operation will lose some information. Furthermore, the PSNR of the invertible reconstruction of CNNInvIF is obviously higher than that of DCTIF, yet the gap of the PSNR between DCTIF and CNNInvIF becomes less with the increase of iterations. Such results confirm that the trained CNNInvIF achieves better invertibility than DCTIF.

3.3. Results of video coding

The proposed CNNInvIF is implemented into the HEVC reference software HM version 16.7. Currently, only the half-pixel interpolation of luma component is replaced by CNNIF.
The low-delay P configuration is tested in the experiment under the HEVC common test conditions. BD-rate is used to measure the rate-distortion performance. The experimental results are summarized in Table 1. As can be seen from Table 1, the proposed CNNInvIF achieves on average 2.2% BD-rate reduction for the luma component, and up to 4.6% BD-rate reduction for the BQTerrace sequence.

In our previous work [7], we reported that CNN-based half-pixel interpolation filter achieved 0.9% BD-rate reduction on average. However, the “ground-truth” in [7] is not real as mentioned before. By contrast, in this paper we do not need the so-called ground-truth when training the CNNInvIF. From the experimental results, the proposed CNNInvIF achieves on average 2.2% BD-rate reduction, which outperforms the method in [7].

### 4. CONCLUSION

This paper presents a CNN based invertible interpolation filter. We firstly reveal the invertibility inherent in the fractional interpolation problem. Then we design an end-to-end training scheme to derive the invertible interpolation filter. We design two loss functions including the invertible reconstruction loss and the regularization loss for optimizing the scheme. Experimental results shows that the proposed method achieves remarkable performance improvement in both invertible reconstruction and compression efficiency compared with DC-TIF.

Our future work will proceed in three directions. First, we would like to study whether invertibility is helpful for training simple interpolation filters, like finite impulse response filters. Second, we want to seek a theoretical interpretation of the proposed joint loss function. Third, we will extend the proposed interpolation filter for other applications.
5. REFERENCES


