Decompressed Video Enhancement via Accurate Regression Prior

Tao Shen [1,2], Yulun Zhang [1,2], Yongbing Zhang [1], Xingzheng Wang [1], Haoqian Wang [1], and Qionghai Dai [1,2]

[1]*Graduate School at Shenzhen, Tsinghua University, Shenzhen 518055, China*

[2]*Department of Automation, Tsinghua University, Beijing 100084, China*

***Abstract*—There is an increasing need for high-quality multimedia applications based on block-based hybrid video coding. Inevitably, the frame will degrade during the process of block-wise intra/inter prediction, transformation, and quantization, especially when the bit rate is low. In this paper, we propose an efficient decompressed video enhancement algorithm based on the adjusted anchored neighborhood regression (A+) method. In our work, first, we learn offline linear regressors, i.e. projection matrices from the decompressed to original video frames in the training phase. For grouping anchored neighborhoods more accurately, we adopt MI-KSVD rather than KSVD to learn the dictionary. Moreover, we exploit the mutual coherence between dictionary atoms and training samples to find the nearest neighbors. Second, in the enhancement phase, we boost the quality of input decompressed videos offline by learned regression priors. To verify the robustness of our enhancement method, extensive experiments are conducted. As shown in our experimental results, the proposed enhancement method yields superior performance both objectively and subjectively.**

***Index Terms*—Dictionary learning, image enhancement, ridge regression, video coding standard, video decompression.**

I. Introduction

Video coding is an indispensable technique in image/video processing, due to its high desirable ability of reducing the huge data volume while maintaining the visual quality to the most extent. As for the human visual system, it is hard to perceive the visual degradation at high or moderate bit rates. However, since modern video coding standards employ some lossy modules, e.g., transformation and quantization, unpleasant visual artifacts are usually perceived when the bit rate is low or acute motions are contained in the input video. To reduce the distortion of the decompressed frames and improve the visual quality, it is desirable to develop some efficient enhancement algorithms. Many pioneering methods have been proposed to perform video enhancement, of which the most representative ones restored the decompressed image based on the sparse property of the natural images [1] [2]. Nevertheless, intensive computations are needed to finish the image enhancement, which is hard to implement in practical applications. Consequently, it is highly demanded to develop an efficient and fast enhancement method for decompressed frames.

Inspired by the superior performance of adjusted anchored neighborhood regression (A+) [3] in image Super-Resolution (SR), we make some improvements in A+ and introduce it into enhancing the quality of decompressed video sequences, which are degraded by nonlinear code. Instead of interpolating low resolution image to high resolution image, we focus on enhancing the decompressed sequence frame to high-quality one while keeping the original image size. Since there is no need to upscale the resolution of the frame, the enhancement of video decompressed sequences is less ill-posed than SR problem. It shows the technical feasibility.

Based on manifold learning method [4], the massive small patches from decompressed sequences and original counterparts lie on or near a locally linear patch of the manifold [5]. By this means, the high quality features can be reconstructed by a weighted average of local neighbors, which share the same weights trained from the low-quality feature set [5]. To reduce the high computational complexity caused by the large size of sample dictionaries, Yang et al. [6] proposed sparse coding by using a learned compact dictionary based on sparse signal representation. Zeyde et al. [7] used KSVD tool [8] for training the dictionary pair to reduce the execution time.

On the basis of these previous works, Timofte et al. proposed anchored neighborhood regression (ANR) [5] and developed it to adjusted anchored neighborhood regression (A+) [3] with great performance to SR problem. However, both ANR and A+ learned dictionary based on KSVD algorithm, neglecting the mutual incoherence among atoms in the dictionary. Further, Zhang et al. improved A+ by optimizing for grouping the nearest neighborhoods with faster and better results [9] [10].

Overall, the main work can be summarized in following aspects. First, we improve the accuracy of dictionary learning method by applying MI-KSVD. Then, instead of calculating the Euclidean distance in [3], we compute the mutual coherence between dictionary atoms and low-quality features extracted from degraded frames to group anchored neighborhoods with less training time and better enhancement quality. After learning regression priors in training phase, we restore the lost information in decompressed video sequences.

II. Regression Prior for Video Enhancement

In this section, we elaborate the training and enhancement process. In training phase, as shown in Fig. 1, we learn the dictionary via MI-KSVD and apply the mutual coherence to group anchored neighborhoods. Next, we train the offline prior models by using anchored neighborhoods based on ridge regression. Finally, we enhance the decompressed video sequences offline based on the prior models.

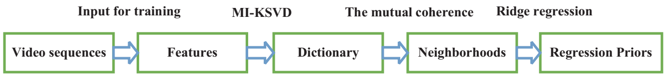

Fig. 1. Diagram of proposed enhancement method

A. Feature and Dictionary

We collect high-quality images from original video sequences and low-quality counterparts from decompressed video sequences. Then, we extract the patches from low- and high-quality images separately to organize two feature sets, i.e. the low-quality feature matrix $F_L = [f_{L,1}, f_{L,2}, \cdots, f_{L,N}] \in R^{S\times N}$ and high-quality counterpart $F_H = [f_{H,1}, f_{H,2}, \cdots, f_{H,N}] \in R^{S\times N}$. For reducing the computation complexity, we conduct Principal Component Analysis (PCA) with matrices Pca to reduce the dimensions of F_L while preserving 99.9% of information [9]. The dimensions of $F_L \in R^{T\times N}$ decline to T.

The sparse dictionary in our enhancement method is learned by using MI-KSVD algorithm proposed by Bo et al. [11] rather than by using KSVD in A+ [3]. We minimize the sum of reconstruction error and the mutual coherence in (1) to learn the corresponding sparse coefficients $X = [x_1, \cdots, x_N]$ as well as dictionary $D = [d_1, \cdots, d_M]$ as follows:

$$\begin{aligned} &\min_{D,X} ||F_L - D \cdot X||_F^2 + \lambda_0 \sum_{i=1}^{M} \sum_{j=1, j\neq i}^{M} |d_i^T d_j| \\ &s.t.\ \forall m,\ ||d_m||_2 = 1\ and\ \forall n,\ ||x_n||_0 \leq L, \end{aligned} \tag{1}$$

where $||\cdot||_F$ denotes the Frobenius norm, λ_0 is a weighting parameter, and L is the desired sparsity level. The solution of above optimization problem via using MI-KSVD leads to higher test accuracy [11] in enhancement process.

B. Grouping Anchored Neighborhoods

Zhang et al. proposed the mutual coherence method to improve the performance of grouping anchored neighborhoods between the dictionary atoms and the training low-quality feature set [12]. Different from A+ calculating the Euclidean distance to get the K nearest neighbors into a neighborhood, the mutual coherence coefficient computes matrix-vector products of dictionary atoms and low-quality features to measure the distance among low-quality features. Since the bigger matrix-vector product means the greater correlation between the dictionary atom and the corresponding low-quality feature.

Grouping anchored neighborhoods can be achieved by the improvement with less training time and better reconstruction quality [13]. Specifically, to group the neighborhood $N_{L,k}$ about the dictionary atom d_k, we compute the products c_k between d_k and each $f_{L,i}$ in F_L (2) respectively to sort the K nearest neighbors as follows:

$$\begin{aligned} c_k &= |(d_k)^T F_{L,i}|_{i=1}^N \\ &= [|(d_k)^T F_{L,1}|, \cdots, |(d_k)^T F_{L,N}|]. \end{aligned} \tag{2}$$

To find the K maximum values in product vector $c_k = [c_{k,1}, c_{k,2}, \cdots, c_{k,N}]$, we sort the vector and get the new coherence vector $c_k^{'} = [c_{k,1}^{'}, c_{k,2}^{'}, \cdots, c_{k,K}^{'}]$. Then, we exploit the original index of $c_{k,j}^{'}$ $(j = 1, 2, \cdots, K)$ in the vector c_k to group the corresponding $F_{L,j}^{'}$ $(j = 1, 2, \cdots, K)$ into a neighborhood $N_{L,k} = [f_{L,1}^{'}, f_{L,2}^{'}, \cdots, f_{L,K}^{'}]$. This method is much like K Nearest Neighborhoods (KNN), but we calculate matrix-vector multiplication instead of l_2-norm computation to find the nearest correlation. This improvement speeds up about 25% [13] in computational time compared to use the Euclidean distance in experiments.

The $N_{H,k}$ can be grouped by the same index of $N_{L,k} = [f_{L,1}^{'}, f_{L,2}^{'}, \cdots, f_{L,K}^{'}]$ in corresponding F_H. Thus, we get the feature neighborhood pairs $\{N_{L,k}, N_{H,k}\}_{k=1}^M$.

C. Offline Model with Ridge Regression

To obtain the projection matrix for mapping $f_{L,i}$ to $f_{H,i}$, first, we solve the l_2-norm regularized least squares regression, i.e. called ridge regression [14] as follows:

$$\min_{x_i} ||f_{L,i} - N_{L,k} \cdot x_i||_2^2 + \lambda ||x_i||_2^2, \tag{3}$$

where $f_{L,i} \in R^{T\times 1}$ is an arbitrary low-quality feature, $N_{L,k} \in R^{T\times K}$ is the low-quality feature neighborhood whose corresponding atom d_k is the closest to $f_{L,i}$, $x_i \in R^{K\times 1}$ is the sparse coefficient vector over $N_{L,k}$ to fitting $f_{L,i}$. The regularized parameter λ introduced into ridge regression effectively alleviates the singularity problems [15] and enables the formulation a closed solution

$$x_i = (N_{L,k}^T N_{L,k} + \lambda I)^{-1} N_{L,k}^T f_{L,i}. \tag{4}$$

Second, $f_{H,k}$ can be approximated by the same sparse representation x_i learned from low-quality dictionary:

$$f_{H,k} = N_{H,k}^T x_i = N_{H,k}^T (N_{L,k}^T N_{L,k} + \lambda I)^{-1} N_{L,k}^T f_{L,k}. \tag{5}$$

Third, noticing to the format above, as the $N_{H,k}$ and $N_{L,k}$ are known upon and λ is given in experiments, we can pre-calculate the regressors, i.e. projection matrices $P_k \in R^{S\times T}$ $(k = 1, 2, \cdots, M)$ via

$$P_k = N_{H,k}^T (N_{L,k}^T N_{L,k} + \lambda I)^{-1} N_{L,k}^T. \tag{6}$$

Finally, the solution of (3) can be simplified to

$$f_{H,k} = P_k f_{L,k}. \tag{7}$$

Based on this compact form, we can calculate high-quality features offline in enhancement phase by using the trained the regressors P_k $(k = 1, 2, \cdots, M)$.

D. Enhancement Process

The video enhancement can be achieved by the offline models $\{Pca, \{d_k, P_k\}_{k=1}^M\}$ learned from training process. We extract the low quality feature set F_L from decompressed frames by first- and second-order gradients in the horizontal and vertical directions. Then we reduce the dimensions of F_L using PCA matrices Pca. For each $F_{L,m}$ i.e. the columns of

F_L, we calculate the coherence vector c_m from the dictionary atoms $D = \{d_k\}_{k=1}^{M}$ and $F_{L,m}$ via

$$\begin{aligned} c_m &= |(d_k)^T F_{L,m}|_{k=1}^{M} \\ &= [|(d_1)^T F_{L,m}|, \cdots, |(d_M)^T F_{L,m}|] \\ &= [c_{m,1}, \cdots, c_{m,M}], \end{aligned} \tag{8}$$

where $c_{m,max}$ is the maximum coefficient in c_m and d_{max} is the corresponding dictionary atom. As d_{max} is found, the index max is exploited to match the projection matrix P_{max}. Once the corresponding projection matrix is fixed, it is convenient to calculate the high-quality features with format (7). At last, high-quality frames can be reconstructed by reshaping the features and averaging over the overlapping regions.

III. Experiment

In this section, a series of experiments are conducted to demonstrate the robustness of our proposed enhancement algorithm in the following.

A. Experimental Settings

In consideration there are lots of similar frames in a video sequence, we use 11 video sequences and extract each of 49 frames from 500 frames for training models. To prove the performance, we use 6 video sequences with each of 36 frames for testing. To demonstrate the performance of our proposed approach in various situations, first, we compress the video sequences to generate low-quality frames for training and enhancement both over H.264 and H.265 separately. Second, for obtaining diverse degraded frames, all the sequences are compressed under different quantization parameters (QP). Furthermore, in our work, three frame types of I-, P-, and B-frame extracted from the video sequences are all experimented respectively.

Peak signal-to-noise ratio (PSNR) are employed in our experiments to evaluate the quality of the enhancement results. Since the human visual system is more sensitive to luma than chroma [16], the evaluation standards are executed just between the luminance channel of all type of video sequence frames. We implement a set of experiments with the parameters M = 1024, K = 2048, λ = 0.1 [3].

B. Performance

In video coding, QP regulates how much spatial detail is saved after decompression. And there is less loss with a low QP. The video sequences we test are as follow, *BasketballDrillText_832×480* (BDT), *BasketballDrill_832×480* (BD), *FourPeople_1280×720* (FP), *KristenAndSara_1280×720* (KAS), *RaceHorses_416×240* (RH), *vidyo1_1280×720* (V1). We calculate the gain between decompressed (Input) and enhanced (Output) frame of each sequence and the average (Avg.) of all gains. As shown in TABLE I, our method produces high gains about I-frame on all testing video sequences in H.264. From TABLE II, three frame types are experimented on all decompressed video sequences in H.265. Except for some outliers, according to the two tables, we discover that all test sequences yield high gains and if the QP value on input sequence is bigger, the gain will be higher. This means that our approach can sufficiently recover more information even there are decompressed sequences with fewer details.

TABLE I

PSNR(dB) results of H.264 about I-frame.

QP	Test	BDT	BD	FP	KAS	RH	V1	Avg.
	Input	36.81	36.58	39.65	40.42	35.59	40.49	**38.26**
QP28	Output	36.97	36.75	39.83	40.53	35.69	40.60	**38.40**
	Gain	0.15	0.17	0.18	0.12	0.09	0.11	**0.14**
	Input	34.50	34.32	37.41	38.34	32.59	38.39	**35.93**
QP32	Output	34.71	34.55	37.63	38.51	32.75	38.53	**36.11**
	Gain	0.21	0.22	0.22	0.17	0.16	0.14	**0.19**
	Input	32.35	32.28	35.13	36.20	30.07	36.30	**33.72**
QP36	Output	32.57	32.52	35.34	36.37	30.25	36.47	**33.92**
	Gain	0.22	0.24	0.21	0.17	0.18	0.17	**0.20**
	Input	30.15	30.21	32.66	33.92	27.95	34.09	**31.50**
QP40	Output	30.36	30.42	32.85	34.06	28.12	34.25	**31.68**
	Gain	0.21	0.21	0.18	0.14	0.17	0.16	**0.18**

C. Visual Results

The performance also can be represented by visual comparison. We compare the result of decompressed frames and enhanced ones over H.264 in Fig. 2 and H.265 in Fig. 3. The regions illustrated by the red block magnify 3 times in the bottom right of figures. In Fig. 2, the enhanced frame presents sharper edges, and much clear structure. Also in the visual comparison over H.265 provided in Fig. 3, we can observe fewer artifacts in the enhanced frame. The result of both visual comparisons manifest that our models are able to restrain the blocking artifacts and enhance the frames to higher quality generally.

IV. Conclusion

During hybrid video coding operation, decompressed video sequences degraded are caused by several primary reasons. For producing high-quality videos, we propose a method based on A+ to figuring this matter. To tackle the nonlinear degradation caused by video coding, we learn the prior models between the original and decompressed sequences via solving ridge regression. MI-KSVD is introduced into training dictionary with higher test accuracy. And the mutual coherence is calculated for finding the nearest neighborhoods, which reduces the training time and improves the enhancement quality. Based on these priors, we enhance the input decompressed video frames offline under different quantization parameters (QP) and different frame types (I, P, and B). The results of various experiments manifest that our proposed approach remarkably improves the quality of decompressed frames in both PSNR and visual effect.

V. Acknowledgement

This work was supported in part by the National Natural Science Foundation of China under Grant 61571254, Grant U1301257, and the Guangdong Natural Science Foundation 2014A030313751.

References

[1] Xinfeng Zhang, Ruiqin Xiong, Xiaopeng Fan, Siwei Ma, and Wen Gao, "Compression artifact reduction by overlapped-block transform coefficient estimation with block similarity," *Image Processing, IEEE Transactions on*, vol. 22, no. 12, pp. 4613–4626, 2013.

[2] Jian Zhang, Debin Zhao, Ruiqin Xiong, Siwei Ma, and Wen Gao, "Image restoration using joint statistical modeling in a space-transform domain," *Circuits and Systems for Video Technology, IEEE Transactions on*, vol. 24, no. 6, pp. 915–928, 2014.

TABLE II

PSNR(dB) RESULTS OF H.265 WITH RESPECT TO DIFFERENT QP ABOUT ALL FRAME TYPES.

Frame	Test	QP28			QP32			QP36			QP40		
		Input	Output	Gain	Input	Output	Gain	Input	Output	Gain	Input	Output	Gain
I	BDT	38.04	38.18	0.14	35.64	35.84	0.20	33.42	33.64	0.22	31.17	31.37	0.20
	BD	37.92	38.06	0.14	35.55	35.75	0.20	33.39	33.61	0.22	31.22	31.43	0.21
	FP	40.71	40.85	0.14	38.39	38.58	0.19	35.92	36.13	0.21	33.29	33.47	0.18
	KAS	41.61	41.71	0.10	39.44	39.61	0.17	37.14	37.36	0.22	34.72	34.97	0.25
	RH	37.29	37.41	0.12	34.13	34.29	0.16	31.30	31.47	0.17	29.14	29.30	0.16
	V1	41.76	41.86	0.10	39.63	39.78	0.15	37.34	37.50	0.16	34.91	35.07	0.16
	Avg.	**39.55**	**39.68**	**0.12**	**37.13**	**37.31**	**0.18**	**34.75**	**34.95**	**0.20**	**32.41**	**32.60**	**0.19**
P	BDT	37.61	37.72	0.11	35.13	35.29	0.16	32.77	32.95	0.18	30.68	30.84	0.16
	BD	37.39	37.53	0.14	34.96	35.14	0.18	32.69	32.87	0.18	30.72	30.89	0.17
	FP	40.26	40.38	0.12	38.20	38.37	0.17	35.94	36.11	0.17	33.47	33.64	0.17
	KAS	41.03	41.10	0.07	39.14	39.27	0.13	37.04	37.19	0.15	34.80	34.99	0.19
	RH	35.70	35.83	0.13	32.54	32.69	0.15	29.87	30.03	0.15	27.91	28.06	0.15
	V1	41.31	41.43	0.12	39.36	39.51	0.15	37.22	37.36	0.14	34.98	35.12	0.14
	Avg.	**38.88**	**39.00**	**0.11**	**36.56**	**36.71**	**0.15**	**34.25**	**34.42**	**0.17**	**32.09**	**32.26**	**0.16**
B	BDT	38.13	38.24	0.11	35.66	35.83	0.17	33.38	33.58	0.20	31.32	31.52	0.20
	BD	37.89	38.02	0.13	35.47	35.65	0.18	33.25	33.45	0.20	31.34	31.54	0.20
	FP	41.03	41.10	0.07	39.19	39.32	0.13	37.02	37.18	0.16	34.58	34.76	0.18
	KAS	41.74	41.76	0.02	40.05	40.12	0.07	38.03	38.15	0.12	35.81	36.00	0.19
	RH	36.00	36.12	0.12	33.00	33.15	0.15	30.33	30.49	0.16	28.29	28.43	0.14
	V1	41.99	42.05	0.06	40.20	40.32	0.12	38.17	38.32	0.15	35.90	36.05	0.15
	Avg.	**39.46**	**39.55**	**0.08**	**37.26**	**37.40**	**0.14**	**35.03**	**35.20**	**0.16**	**32.87**	**33.05**	**0.18**

(a) The ground truth frame (b) The decompressed frame (c) The enhanced frame

Fig. 2. Visual quality comparisons between the decompressed frames and the enhanced frames under QP of 40 over H.264 (a) the ground truth frame of $BasketballDrill$_832x480. (b) the decompressed frame of $BasketballDrill$_832x480. (c) the enhanced frame of $BasketballDrill$_832x480.

(a) The ground truth frame (b) The decompressed frame (c) The enhanced frame

Fig. 3. Visual quality comparisons between the decompressed frames and the enhanced frames under QP of 40 over H.265 (a) the ground truth frame of $RaceHorses$_416x240. (b) the decompressed frame of $RaceHorses$_416x240. (c) the enhanced frame of $RaceHorses$_416x240.

[3] Radu Timofte, Vincent De Smet, and Luc Van Gool, "A+: Adjusted anchored neighborhood regression for fast super-resolution," in *ACCV*, pp. 111–126. Springer, 2014.

[4] Sam T Roweis and Lawrence K Saul, "Nonlinear dimensionality reduction by locally linear embedding," *Science*, vol. 290, no. 5500, pp. 2323–2326, 2000.

[5] Radu Timofte, Vivek De, and Luc Van Gool, "Anchored neighborhood regression for fast example-based super-resolution," in *ICCV*. 2013, pp. 1920–1927, IEEE.

[6] Jianchao Yang, John Wright, Thomas S Huang, and Yi Ma, "Image super-resolution via sparse representation," *Image Processing, IEEE Transactions on*, vol. 19, no. 11, pp. 2861–2873, 2010.

[7] Roman Zeyde, Michael Elad, and Matan Protter, "On single image scale-up using sparse-representations," in *Curves and Surfaces*, pp. 711–730. Springer, 2012.

[8] Michal Aharon, Michael Elad, and Alfred Bruckstein, "K-svd: An algorithm for designing overcomplete dictionaries for sparse representation," *Signal Processing, IEEE Transactions on*, vol. 54, no. 11, pp. 4311, 2006.

[9] Yulun Zhang, Kaiyu Gu, Yongbing Zhang, Jian Zhang, and Qionghai Dai, "Image super-resolution based on dictionary learning and anchored neighborhood regression with mutual incoherence," in *ICIP*. 2015, pp. 591–595, IEEE.

[10] Yongbing Zhang, Yulun Zhang, Jian Zhang, and Qiongdai Dai, "Ccr: clustering and collaborative representation for fast single image super-resolution," *Multimedia, IEEE Transactions on*, vol. 18, no. 3, pp. 405–417, Mar. 2016.

[11] Liefeng Bo, Xiaofeng Ren, and Dieter Fox, "Multipath sparse coding using hierarchical matching pursuit," in *CVPR*. 2013, pp. 660–667, IEEE.

[12] Yulun Zhang, Yongbing Zhang, Jian Zhang, Haoqian Wang, Xingzheng Wang, and Qiongdai Dai, "Adaptive local nonparametric regression for fast single image super-resolution," in *VCIP*, Dec. 2015, pp. 1–4.

[13] Yulun Zhang, Yongbing Zhang, Jian Zhang, Haoqian Wang, and Qionghai Dai, "Single image super-resolution via iterative collaborative representation," in *PCM*. 2015, pp. 63–73, Springer.

[14] Lei Zhang, Meng Yang, and Xiangchu Feng, "Sparse representation or collaborative representation: Which helps face recognition?," in *ICCV*. 2011, pp. 471–478, IEEE.

[15] Andre Nikolaevich Tikhonov and Vasili IAkovlevich Arsenin, *Solutions of ill-posed problems*, Vh Winston, 1977.

[16] H ITU-T RECOMMENDATION, "264 advanced video coding for generic audiovisual services," *ISO/IEC*, vol. 14496, 2003.