MATEC Web of Conferences **63**, 04037 (2016) DOI: 10.1051/matecconf/20166304037
MMME 2016

Fast Image Super-resolution with Sparse Coding

Zhi-chao Yuan [1,a], Ben-tu Li [1]

[1]*Shandong Vocational College of Science and Technology, Weifang, 261053, China;*

Abstract. In this paper, we introduce a novel fast image reconstruction method for super-resolution (SR) base on sparse coding. This method combine online dictionary learning and a fast sparse coding way, both of which can improve the efficiency of the reconstruction process and ensure the image visual quality. The new online optimization algorithm for dictionary learning based on stochastic approximations, which can drastically advance the learning speed, especially on millions of training samples. Meanwhile, we trained a neural network to speed up the reconstruction process, which based on iterative shrinkage-thresholding algorithm (ISTA), we called learned iterative shrinkage-thresholding algorithm (LISTA). It would produce best approximation sparse code with some fixed depth. We demonstrate that our approach can simultaneously improve the image fidelity and cost less computation.

Keywords: SRIR; sparse coding; super-resolution; fast image super-resolution.

0. Introduction

High-resolution image is valuable in many social areas. Such as reconstruct the high-resolution image from low-resolution image which take in Criminal Investigation, get high-accurate CT, MIR, Ultrasonic wave to help doctor diagnose disease. Additionally, this is also play an important role in HDTV, Video Rip and Military occasion.

The main contribution of this paper is combining the online dictionary learning and approximating sparse coefficient together to reconstruct the High-resolution image, so that the reconstruction rate is splendidly speedup. Learning the dictionary by online method process one element of the millions of training samples at a time, which based on stochastic approximations(Aharon & Elad, 2008). Instead of optimize the l1-norm optimization for accurate sparse coefficients, we employ a feed-forward neural network to construct approximated sparse coefficients(Gregor & LeCun, 2010) .

1. Fast image super-resolution

This section describe the two key component of SRIR—learn the over-complete dictionary and find sparse coefficient. Online dictionary learning proposed an iterative algorithm that solves eqn. by efficiently minimizing at each step a quadratic surrogate function of the empirical cost over the set of constraints (Mairal, Bach, Ponce, & Sapiro, 2009). We find the approximation sparse coefficient by train a non-linear, feed-forward neural network instead of find the accurate solution(Gregor & LeCun, 2010).

1.1 Online dictionary learning

According to we talked above, the dictionary learning can be get by optimize the energy function:

$$f_n(\tilde{\mathrm{D}}) \triangleq \frac{1}{n}\sum_{i=1}^{n} l(\tilde{y}_i,\tilde{\mathrm{D}}),\ \ where\ \ l(\tilde{y}_i,\tilde{\mathrm{D}}) \triangleq \min_{\alpha\in R^k} \frac{1}{2}\left\|\tilde{y}_i - \tilde{D}\,\alpha_i\right\|_2^2 + \lambda\left\|\alpha_i\right\|_1 \quad (1.1)$$

As point by Bettou and Bousquet(Bousquet & Bottou, 2007), maybe its not important in a perfect minimization of energy function $f_n(\mathrm{D})$, but interested in expected minimization cost:

$$f\left(\tilde{D}\right) \triangleq E_x\left[l(\tilde{y}_i,\tilde{\mathrm{D}})\right] = \lim_{n\to\infty} f_n\left(\tilde{D}\right) \quad (1.2)$$

The outline of our algorithm is described in algorithm 1

Algorithm1 Online dictionary learning

Require: $x \in R^m$: $\mathrm{p}(x)$ (random variable and an algorithm to draw i.i.d samples of p), $\lambda \in R$ (regularization parameter), $D_0 \in R^{m\times k}$ (initial dictionary). T (number of iterations)

1: $A_0 \leftarrow 0$, $B_0 \leftarrow 0$ (reset the "past" information); 2: For t=1 to T do

3: Draw xt from p(x). 4: Sparse coding: compute using LARS

$$\alpha_t = \arg\min_{\alpha\in R^k} \frac{1}{2}\left\|x_t - D_{t-1}\alpha\right\|_2^2 + \lambda\left\|\alpha\right\|_1 \quad (1.3)$$

5: $A_t \leftarrow A_{t-1} + \alpha_t\alpha_t^T$; 6: $B_t \leftarrow B_{t-1} + x_t\alpha_t^T$.

7: Compute Dt using Algorithm 2, with D_{t-1} as warm restart. So that

$$D_t \triangleq \arg\min_{D\in C} \frac{1}{t}\sum_{i=1}^{t}\frac{1}{2}\left\|x_i - D\alpha_i\right\|_2^2 + \lambda\left\|\alpha_i\right\|_1 = \arg\min_{D\in C}\frac{1}{t}\left(\frac{1}{2}Tr\left(D^TDA_t\right) - Tr\left(D^TB_t\right)\right) \quad (1.4)$$

8: End for; 9: Return DT (learned dictionary).

We supposed the training set is composed of i.i.d samples of a distribution p(x), every loop it draws one element from the training set. It alternatively solve α by classical sparse coding and D by minimizing the following function:

$$\bar{f}_t(D) @ \frac{1}{t}\sum_{i=1}^{t}\frac{1}{2}\left\|x_i - D\alpha_i\right\|_2^2 + \left\|\alpha_i\right\|_1 \quad (1.5)$$

The quadratic function $f_t(\bar{D})$ is a surrogate. This function aggregates the previous information by the process of computing α. The convergence analysis has shown that $\bar{f}_t(D)$ and f_t convergence almost to the same limit(Mairal et al., 2009) . Because of $\bar{f}_t(D)$ is close to f_t , so it can be efficiently computed via D_{t-1} as warm restart.

Algorithm 2 Dictionary update

Require: $D = [d_1 \quad d_2 \quad \mathrm{K} \quad d_{k-1} \quad d_k] \in R^{m\times k}$ (input dictionary),

$$A = [a_1 \quad a_2 \quad \mathrm{K} \quad a_k] \in R^{k\times k} = \sum\nolimits_{i=1}^{t}\alpha_i\alpha_i^T \quad , B = [b_1 \quad b_2 \quad \mathrm{K} \quad b_k] \in R^{m\times k} = \sum\nolimits_{i=1}^{t}x_i\alpha_i^T \quad ,$$

1: Repeat;2: For j=1 to k do

3: Update the j-1 column to optimize for eqn. :

$$u_j \leftarrow \frac{1}{A_{jj}}\left(b_j - D\alpha_j\right) + d_j, \quad d_j \leftarrow \frac{1}{\max(\left\|u_j\right\|_2, 1)}u_j \quad (1.6)$$

4: End for;5: Until convergence;6: Return D (update dictionary).

The algorithm 2 uses block-coordinate descent with warm restart to update the dictionary. The benefits of this method is parameter-free and does not require any learning rate tuning. It sequentially update D column by column when update the j-th column, when set the other ones fixed under constraint $d_jd_j^T \le 1$. Recent works shows that separable constraint could make the function convergence to a global optimum. Since we use Dt-1 to warm restart Dt , so the block-coordinate descent will more efficient.

1.2 Approximation sparse coefficient

In this section, we employ the same neural network structure in paper(Gregor & LeCun, 2010) to reconstruct the HR patches.

In paper(Gregor & LeCun, 2010) , the approach of train a neural network is inspired by the iterative shrinkage-thresholding algorithm (ISTA), The outline of ISTA is described in algorithm 3

Algorithm 3 ISTA

Require: L>largest eigenvalue of $W_d^T W_d$.

Initialize: Z=0.

1:Repeat $Z = h_{(\alpha/L)}\left(Z - \frac{1}{L} W_d^T \left(W_d Z - X \right) \right)$ （1.7）

2:Until convergence; 3:End function.

The elements of the eqn.are defined as:Filter matrix: $W_e = \frac{1}{L} D_y^T$.

Inhibition matrix: $S = I - \frac{1}{L} D_y^T D_y$

Shrinkage function: $h_\theta(z) = sign(z)(|z| - \theta)_+$.

Here, L is a constant, which is larger than the largest eigenvalue of $D_y^T D_y$, function $h_\theta(z)$ is a component-wise shrinkage function and θ is usually set as λ / L . In practice, because of this algorithm have to solve thousands of optimization and take tens of iterations to convergence, it’s too slow for practical applications. We use for reference paper (26), proposed a new idea which only need a fixed number of iteration T, they do not use fixed parameter in ISTA. The algorithm through minimize the following energy function to get the parameter W = (We,S, θ):

$$E(\mathrm{w}) = \frac{1}{N} \sum_{i=1}^{N} C(\mathrm{W}, \mathrm{y}_i), \text{with } \mathrm{C}(\mathrm{W}, \mathrm{y}_i) = \frac{1}{2} \left\| z_i - f_e\left(W, y_i\right) \right\|_2^2 \quad (1.8)$$

Here, zi is the accurate sparse coefficient of yi which from traditional sparse method. And $f_e(W, y_i)$ is the transformation of eqn. .

Extensively, in our super-resolution algorithm we rewrite the energy function as:

$$C(\mathrm{W}, \mathrm{y}_i, \mathrm{x}_i) = \frac{1}{2} \left\| z_i - f_e\left(W, y_i\right) \right\|_2^2 + \frac{\nu}{2} \left\| D_x z_i - x_i \right\|_2^2 \quad (1.9)$$

This will work better to predict the HR image. The outline of LISTA is describe in algorithm 4 and algorithm 5.

Algorithm 4

Require ： We, X.Initialize: B=WeX; Z(0)=h θ (B).Arguments are passed by reference.Variables Z(t), C(t) and B are saved for bprop.

1:For t=1 to T do; 2:C(t) = B+SZ(t-1); 3:Z(t) = h θ (C(t)); 4:End for;5:Return Z=Z(T)

Algorithm 5

Require: Z* (accurate sparse coefficient from traditional methods); Z(t), C(t), B (saved in fprop).

Initialize:

$\delta B = 0; \delta S = 0; \delta\theta = 0; \delta Z(\mathrm{T}) = (\mathrm{Z(T)} - Z^*)$

1:for t=T down to 1 do;

2: $\delta C(t) = h_\theta'(C(t)) \cdot \delta Z(t)$;

3: $\delta\theta = \delta\theta - sign(\mathrm{C(t)}) \cdot \delta \mathrm{Z}(t)$

4: $\delta B = \delta B + \delta C(t)$;

5: $\delta S = \delta S + \delta C(t) Z(t-1)^T$;

6: $\delta(t-1) = S^T \delta C(t)$; 7:end for

8: $\delta B = \delta B + h_\theta'(B) \cdot \delta Z(0)$;

9: $\delta\theta - sign(\mathrm{B}) \cdot \mathrm{h}_\theta'(\mathrm{B}) \delta \mathrm{Z}(0)$;

10: $\delta W_e = \delta B X^T; \delta X = W_e \delta B$

2.Experiment results

In the following, we first compare the SRIR image with Bicubic interpolation method(Yang et al., 2010), which is a splendid approach. Then we discuss the efficiency with other splendid approaches.

2.1 Comparison with joint dictionary learning

We compare our algorithm with Bicubic interpolation in visual result and other index. In order to make the fair comparison, we adopt the same training set and initial D for both methods.

a) Visual result: fig. 1(a)-(d) compared the results of Bicubic interpolation method with our fast image super-resolution four test images. From the left to right are the original HR images, the reconstructed images by Bicubic interpolation method and the reconstructed images by fast super-resolution method. From the results, we can easily see that our recovery are closed to the original images, the edge is more sharper than the Bicubic recovery, such as the moustaches of the cat, the texture of the freckles of the girl，the texture of the leaves and the hair of lena.

Fig. 1 from left to right: original image, Bicubic interpolation recovery, our recovery

b) Numerical result: now we compute the Peak signal-to-noise ratio (PSNR) of the fast super-resolution image. PSNR is most easily defined via the mean squared error (MSE), MSE is defined as:

$$MSE = \frac{1}{MN}\sum_{m=0}^{M-1}\sum_{n=0}^{N-1}[\mathrm{I(m,n)} - \mathrm{f(m,n)}]^2 \qquad (2.1)$$

Here, I and K are respectively represent the original image and our recovery. The PSNR is defined as:

$$PSNR = 10\log_{10}\left(\frac{MAX_I^2}{MSE}\right) = 20\log_{10}\left(\frac{MAX_I}{\sqrt{MSE}}\right) = 20\log_{10}\left(MAX_I\right) - 10\log_{10}\left(\sqrt{MSE}\right) \qquad (2.2)$$

Here, MAXI is the maximum possible pixel value of the image. When the pixels are represented using 8 bits per sample, this is 255. Typical values for the PSNR in lossy image and video compression are between 30 and 50 dB, where higher is betters.

DOI: 10.1051/matecconf/20166304037

images	Bicubic	Fast recovery	
		`T = 3	T=5
girl	31.699140db	33.324861db	33.336294db
cat	28.3597db	28.8539db	28.8669db
leave	37.197505db	38.060257db	38.060146db
lena	32.794678db	35.02261db	35.022206db

Table I the PSNR of Bicubic interpolation recovery and our recovery

Table 1 shows the PSNR of test images. From table I we can see that our recovery outperforms the Bicubic interpolation recovery well with high PSNR. When the iteration times is fixed as 3, the fast sparse coding is converging to optimal.

2.2 Algorithm efficiency

In this subsection, we evaluate the efficiency of our algorithm and other state-of-art algorithms (Glasner, Bagon, & Irani, 2009; Yang et al., 2010). All of the experiments are conducted in Matlab 8.1. Table II show the consumed times of different method.

iamge	mesuares	Bicubic	Yang at el.	Fast recovery
Girl	PSNR(db)	31.699140	33.34565	33.336294
254×256	Time(s)	0.012371	285.496759	34.6597
Cat	PSNR(db)	28.3597	28.8658	28.8669
326×298	Time(s)	0.017776	446.296511	48.6235
Leave	PSNR(db)	37.197505	38.0611	38.060146
328×170	Time(s)	0.012091	246.902859	41.3629
statuary	PSNR(db)	26.0253	26.3064	26.2923
256×162	Time(s)	0.007996	180.382172	23.26

Table II processing time of different SR method

From table II we can see that our algorithm speedup the reconstruction process significantly. In addition, we can ensure both of the PSNR and the cost of compute time, since our algorithm adopt the approximate sparse coefficients and train the online dictionary. From fig. 2 for “statuary” image, Yang’ s algorithm has the higher PSNR than our algorithm, but both of two images almost have the same visual quality.

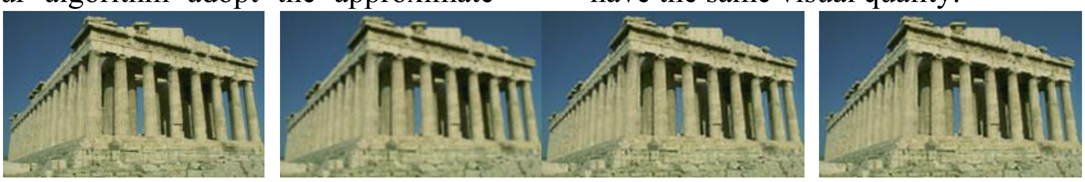

Fig. 2 top: original, Bicubic; bottom: Yang at el. Fast recovery.

3. Conclusions

In this paper, we propose a novel approach to reconstruct the high-resolution image based on online dictionary learning and approximate sparse coefficient.The preliminary experiment prove that our algorithm not only ensure the visual quality but faster than other state-of-art approaches. However, our algorithm is not have the optimal recovery than other SR approaches, so, our future research is to train accurate dictionary to represent the sparse coefficients.

References

[1] Aharon, M., & Elad, M. (2008). Sparse and Redundant Modeling of Image Content Using an Image-Signature-Dictionary. Siam Journal on Imaging Sciences, 1(3), 228-247.

[2] Beck, A., & Teboulle, M. (2009). A Fast Iterative Shrinkage-Thresholding Algorithm for Linear Inverse Problems. Siam Journal on Imaging Sciences, 2(1), 183-202.

DOI: 10.1051/matecconf/20166304037

[3] Bousquet, O., & Bottou, L. (2007). The Tradeoffs of Large Scale Learning. Paper presented at the Advances in Neural Information Processing Systems.

[4] Candes, E. J., Romberg, J. K., & Tao, T. (2006). Stable signal recovery from incomplete and inaccurate measurements. Communications on Pure And Applied Mathematics, 59(8), 1207-1223. doi: Doi 10.1002/Cpa.20124

[5] Daubechies, I., Defrise, M., & De Mol, C. (2004). An iterative thresholding algorithm for linear inverse problems with a sparsity constraint. Communications on Pure And Applied Mathematics, 57(11), 1413-1457. doi: Doi 10.1002/Cpa.20042