IMAGE COMPRESSION USING GMM MODEL OPTIMIZATION

Jianjun Sun   Yan Zhao   Shigang Wang

College of Communication Engineering, Jilin University, Changchun, China
sunjj17@mails.jlu.edu.cn, zhao_y@jlu.edu.cn, wangshigang@vip.sina.com

ABSTRACT

A Gaussian Mixture Model (GMM)-based framework for image compression is proposed in this paper. The image is predicted using GMM whose parameters are estimated using common Expectation-Maximization (EM) algorithm and encoded with fixed length bits. We introduce a new GMM Model Optimization (GMO) measure to select the optimal number of models and avoid local optimum of EM at the same time. The encoding cost of the residual and parameters are considered in GMO which is demonstrated to be near concave and effective. A parameter dictionary is designed to utilize the correlation of the parameters to improve the coding efficiency. The residual between the original image and the GMM image is encoded using High Efficiency Video Coding (HEVC) intra coding. Experimental results show that our method performs better than HEVC.

Index Terms—Gaussian Mixture Model, GMM Model Optimization, Expectation-Maximization, Parameter Dictionary, Image Compression

1. INTRODUCTION

With the explosion growth of the image and video data today, it is more significant to design a more efficient framework for image coding. JPEG [1] uses Discrete Cosine Transform (DCT) and JPEG2000 [2] uses wavelet transform to utilize the correlation in the image. The intra standard of HEVC [3] predicts the blocks with 35 different modes to encode an image.

Studies have shown that GMM can fit any kind of probability density function (pdf) well with enough number of models [4]. Likewise, GMM can fit the correlation of the pixel information, which is the reason we try to utilize GMM in our coding scheme. GMM has been used to classify different kinds of photograph and distinguish the background [5]. But few use GMM to fit the pixel information itself to encode images.

Our work is motivated by the efforts of using Steered Mixture-of-Experts (SMOE) in image compression [6], which put forward the idea of using 3-D Gaussian steering kernels with global support. It has better performance when comparing with JPEG at low bitrates. The authors of [6] also utilize GMM for color image coding in [7].

This work is supported by the National Natural Science Foundation of China (No.61631009, No.61771220), the Fundamental Research Funds for the Central Universities (No.2017TD-19), and the National Key R&D Program of China (No.2017YFB1002900, No.2017YFB0404800).

There are many technical difficulties when using GMM to compress images. Firstly, GMM itself is a clustering method without supervision [8]. For unsupervised clustering, the certain number of models is not easy to determine in the initialization stage. Secondly, the common EM algorithm which is widely used in GMM may suffer from local optimum. Thirdly, GMM utilized in a coding approach should take the cost of residual and parameters into consideration when selecting the optimal number of models. To achieve an efficient coding, GMO needs to be able to determine the optimal number of models and avoid local optimum of EM algorithm.

The purpose of our work is to design an optimized image compression approach based on GMM.

2. PROPOSED COMPRESSION SCHEME

The block diagram of the proposed method is shown in Fig.1.

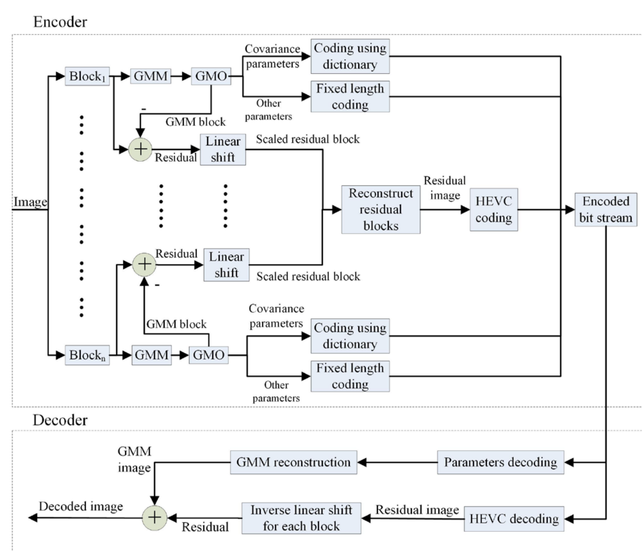


Fig. 1. The block diagram of the encoder and decoder.

In our design, the image is firstly transformed into YUV 4:2:0 formats and processed into blocks. The processing of YUV channels is same. The blocks are predicted using GMM. A common EM algorithm initialized using K-means algorithm is used to get the parameters of the models. And then GMO is used to select the optimal number of models. After that, a parameter dictionary is introduced to reduce the encoding cost of parameters. The residual will be scaled and defined as residual block. Finally all residual blocks are reconstructed and encoded by HEVC intra coding.

2.1. Prediction using Gaussian Mixture Model

In multivariate normal distribution, a random vector can be well predicted based on a known random vector when the two random vectors are dependent. Therefore, Gaussian Mixture Model (GMM) can be used to model a data set. The parameters of these Gaussian models can be estimated using EM algorithm. We consider the two location variables (the row coordinate and the column coordinate of a pixel) and the gray value of the pixel of an image as a three-dimensional random vector, which is denoted by $\boldsymbol{data}_i$. Assume the training data $Data = \{\boldsymbol{data}_i\}_{i=1}^{\mathrm{T}}$ has joint probability density as follows.

$$p_{LG} = \sum_{j=1}^{\mathrm{K}} \alpha_j N(\boldsymbol{\mu}_j, \boldsymbol{Cov}_j) = \sum_{j=1}^{\mathrm{K}} \alpha_j \varphi_j \quad (1)$$

and $\boldsymbol{data}_i = \begin{bmatrix} \boldsymbol{L}_i \\ G_i \end{bmatrix}$, $\boldsymbol{L}_i = \begin{bmatrix} Row_i \\ Col_i \end{bmatrix}$, $\boldsymbol{Cov}_j = \begin{bmatrix} \boldsymbol{Cov}_{\boldsymbol{L}_j\boldsymbol{L}_j} & \boldsymbol{Cov}_{\boldsymbol{L}_jG_j} \\ \boldsymbol{Cov}_{G_j\boldsymbol{L}_j} & Cov_{G_jG_j} \end{bmatrix}$,

$\sum_{j=1}^{\mathrm{K}} \alpha_j = 1$, $\boldsymbol{\mu}_j = \begin{bmatrix} \boldsymbol{\mu}_{\boldsymbol{L}_j} \\ \mu_{G_j} \end{bmatrix}$

where T is the number of pixels in the image, Row_i is the row coordinate of the ith pixel while Col_i is the column coordinate, K is the number of models, G_i is the gray value of the ith pixel, $\boldsymbol{\mu}_{\boldsymbol{L}_j}$ is the mean of all $\boldsymbol{L}_i$ in the jth model, μ_{G_j} is the mean of all G_i in the jth model. And $\boldsymbol{Cov}_j$ is the covariance matrix of the jth model. α_j is the ratio of the pixels belonging to the jth Gaussian model, which is calculated by dividing the number of pixels in the jth Gaussian model by the number of all pixels in the image.

In multivariate normal distribution, the probability density function of a multivariate with dimension $p+q$ can be factorized as:

$$N_{p+q}\left(\begin{bmatrix} \boldsymbol{\mu_L} \\ \mu_G \end{bmatrix}, \boldsymbol{\sigma}^2\right) = N_q(\mu_{G|\boldsymbol{L}}, \sigma_G^2) N_p(\boldsymbol{\mu_L}, \boldsymbol{Cov_{LL}}) \quad (2)$$

where $\mu_{G|\boldsymbol{L}}$ is the conditional mean.

Accordingly, equation (1) can also be written as:

$$p_{\mathrm{LG}} = \sum_{j=1}^{\mathrm{K}} \alpha_j \varphi_{G|\boldsymbol{L}_j}(est_j(\boldsymbol{L}), \sigma_j^2) \varphi_{\boldsymbol{L}_j}(\boldsymbol{\mu}_{\boldsymbol{L}_j}, \boldsymbol{Cov}_{\boldsymbol{L}_j\boldsymbol{L}_j}) \quad (3)$$

where $est_j(\boldsymbol{L})$ is the conditional mean and σ_j^2 is the conditional variance. And we can get the conditional mean and the conditional variance using the knowledge of mathematical statistics [9].

$$est_j(\boldsymbol{L}) = \mu_{G_j} + \boldsymbol{Cov}_{G_j\boldsymbol{L}_j} \boldsymbol{Cov}^{-1}_{\boldsymbol{L}_j\boldsymbol{L}_j} (\boldsymbol{L} - \boldsymbol{\mu}_{\boldsymbol{L}_j}) \quad (4)$$

$$\sigma_j^2 = Cov_{G_jG_j} - \boldsymbol{Cov}_{G_j\boldsymbol{L}_j} \boldsymbol{Cov}^{-1}_{\boldsymbol{L}_j\boldsymbol{L}_j} \boldsymbol{Cov}_{\boldsymbol{L}_jG_j} \quad (5)$$

where $est_j(\boldsymbol{L})$ is also the estimated gray value assuming that it belongs to the jth model, and $\boldsymbol{L}$ is the location of the pixel to be estimated.

Equation (4) is used to predict the gray value of the pixel with the known location for single Gaussian model. To reconstruct the image, we need two mean values in $\boldsymbol{\mu}_{\boldsymbol{L}_j}$, the mean value in μ_{G_j}, four covariance values in $\boldsymbol{Cov}_{\boldsymbol{L}_j\boldsymbol{L}_j}$, two covariance values in $\boldsymbol{Cov}_{\boldsymbol{L}_jG_j}$, and one covariance value in $Cov_{G_jG_j}$ for one Gaussian model. Generally there are more than one Gaussian model for using GMM to predict an image. Every Gaussian model has its main "territory", and it can affect the main territories of other Gaussian models.

GMM assumes that every model influences the gray value of the pixel at different level which can be determined by a weight coefficient. The products of the estimated values and weight coefficients are sum up to get the gray value of a pixel as follows.

$$\hat{G}(\boldsymbol{L}) = est(\boldsymbol{L}) = \sum_{j=1}^{\mathrm{K}} w_j(\boldsymbol{L}) \times est_j(\boldsymbol{L}) \quad (6)$$

where $w_j(\boldsymbol{L}) = \dfrac{\alpha_j N_j(\boldsymbol{\mu}_{\boldsymbol{L}_j}, \boldsymbol{Cov}_{\boldsymbol{L}_j\boldsymbol{L}_j})}{\sum_{l=1}^{\mathrm{K}} \alpha_l N_l(\boldsymbol{\mu}_{\boldsymbol{L}_l}, \boldsymbol{Cov}_{\boldsymbol{L}_l\boldsymbol{L}_l})}$.

And $w_j(\boldsymbol{L})$ is the weight coefficient which shows the influence ratio of the jth model to all models.

In this way, we can get the whole image reconstructed by GMM parameters.

2.2. GMM Model Optimization

The proposed GMM Model Optimization (GMO) is designed for determining the optimal number of models and avoiding local optimum of EM algorithm.

The image block can be predicted using GMM with different number of models, which may take the range from 1 to 32 in general. GMO is performed after the prediction step.

GMO is given by:

$$\arg\max_k \{F(k)\} \quad (1 \le k \le 32) \quad (7)$$

where $F(k)$ is defined by:

$$F(k) = SSIM''_k - [fm(k) + fr(k)] \quad (8)$$

$SSIM''_k$, $fm(k)$ and $fr(k)$ are defined as follows.

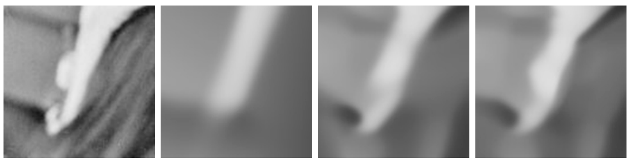
$$SSIM''_k = \frac{SSIM_k}{SSIM_{\max}} \quad (9)$$

$$fm(k) = \frac{k}{\beta_{\max}} \quad (10)$$

$$fr(k) = \frac{1}{M \times N} \sum_{i=1}^{\mathrm{T}} \frac{\left|\hat{G}_k(L_i) - G_i\right|}{G_i} \quad (11)$$

where $\beta_{\max} = \dfrac{M \times N}{32 \times 32} \times 12$. $SSIM_k$ is the structural similarity index (SSIM) [10] value when k models are used to predict the block and $SSIM_{\max}$ is the maximal $SSIM_k$. $SSIM''_k$ represents the normalized $SSIM_k$ in equation (8). $SSIM_k$ is always smaller than 1 in case that GMM cannot reconstruct details which are too complex. Thus, a block of size 32×32 can be adequately predicted with at most 12 models as the extreme case according to a large number of experiments. $\beta_{\max}$ denotes the maximal number of models for blocks with different sizes of the extreme case which increases with the growth of $M \times N$. $M \times N$ denotes the size of the block we used. T is the number of pixels and G_i is the gray value of the *ith* pixel. $\hat{G}_k(L_i)$ is the predicted gray value which can be calculated using equation (6).

The three parts of equation (8) are designed to be normalized meaningfully to the range from 0 to 1. $SSIM_k''$ is considered as the gain in GMO while other parts represent the possible cost in further coding. $fm(k)$ denotes the cost of the parameter encoding related to k and it is considered as 1 when $\beta_{\max}$ models are used to predict the block. In addition, $fr(k)$ denotes the average of the normalized difference of the pixels between the original block and GMM block which indicates the possible cost of residual encoding. Fig.2 and Table 1 are given to show the rationality of GMO.



(a) (b) *k*=4 (c) *k*=18 (d) *k*=22

Fig. 2. An example of the GMM block: (a) original block (b) (c) (d) GMM block (*k* is the number of models).

Fig.2 is a block at the lower right corner of the test image Lena to show the details of the reconstructed GMM block. The size of the image block is 128×128 here to make the difference of GMM blocks more obvious. We can see from Fig.2 (b) (c) (d) that the most similar GMM block comparing with the original block is Fig.2 (d). However, from Table 1, we can see that GMO proposed here chooses Fig.2 (c) as the best situation with the maximal $F(k)$ instead of Fig.2 (d) in case that the cost of the extra parameters has been considered.

Table 1. The comparison of the GMO components with different number of models.

k \ Items	$F(k)$	$SSIM_k''$	$fm(k)$	$fr(k)$
4	0.8333	0.8909	0.0208	0.0368
10	0.8960	0.9570	0.0521	0.0089
14	0.9001	0.9799	0.0729	0.0069
16	0.8747	0.9662	0.0833	0.0082
18	0.9008	0.9978	0.0938	0.0032
22	0.8823	1.0000	0.1146	0.0031
28	0.8429	0.9918	0.1458	0.0031
32	0.8313	0.9999	0.1667	0.0019

It can be seen from Table 1, $F(k)$ is near concave when k is regarded as the independent variable. $k=18$ is the extreme point for this block. It is worth noting that $F(16)$ is smaller than $F(14)$ abnormally. $SSIM_k''$ becomes smaller even more models are used in prediction and $fr(k)$ becomes larger for suffering from local optimum. Thus GMO can recognize and avoid local optimum by decreasing $F(k)$.

2.3. Coding of parameters

For each block, 5 bits are used to encode the number of models since the maximum number of models is 32. We use fixed length coding to encode parameters of each model, in which 7 bits for α_j and 8 bits for the mean gray value. And the two mean location values will be encoded with several bits according to the size of the blocks which will be indicated at the beginning of the bit stream. For example, the row coordinate will be encoded with 5 bits if the height of the blocks is not larger than 32. The covariance matrix is symmetrical as shown in Fig.3. Therefore, there are only six numbers to be encoded. The covariance parameters are reserved as integers. We encode the covariance matrix with 8 bits for C_{00}, 8 bits for C_{01}, 9 bits for C_{02}, 8 bits for C_{11}, 10 bits for C_{12} and 12 bits for C_{22} according to the numerical range of these parameters. Thus we need at least 80 bits for each model.

$$\begin{bmatrix} C_{00} & C_{01} & C_{02} \\ C_{01} & C_{11} & C_{12} \\ C_{02} & C_{12} & C_{22} \end{bmatrix}$$

Fig. 3. The covariance matrix.

2.4. Parameter dictionary

A parameter dictionary is used in our method to save the bits for parameters. Thus, the encoding of covariance parameters can be achieved by encoding the index of the dictionary.

The single Gaussian model in GMMs may generate similar GMM blocks with fixed mean gray value and location of center. In this case, these corresponding Gaussian models can be considered the same. Therefore, in order to improve the coding efficiency, we intend to utilize the correlation between Gaussian models. These GMM blocks generated by single model are called model blocks here which can be tools to compare the Gaussian models.

In the proposed method, we predict every block with GMM. All the GMM blocks and model blocks will be gotten firstly. The model blocks are compared using SSIM. It is assumed that when the SSIM is larger than 0.9, the model blocks are considered the same. Their corresponding covariance parameters can be replaced by each other.

For the first image block, all the model blocks are compared. The parameters corresponding to the model blocks which are not repeated are collected into the covariance parameter dictionary. For the following image blocks, every model block is compared with the ones generated by the parameters in the dictionary. If one model block can find a similar one generated by the parameters in the dictionary, its covariance parameters need not to be encoded and only one index is needed. If the model block cannot match a similar one, the parameters are considered as a new dictionary tuple. After the complete iteration, we can get the parameter dictionary. The covariance parameters in the dictionary are encoded using fixed length coding as described in Section 2.3.

The size of the dictionary is often not larger than 32. In this case, each model needs only 5 bits to encode the index of the dictionary instead of 55 bits for the covariance parameters.

2.5. Coding of the residual image

The residual between the original block and the GMM block with optimal parameters of models may have negative values. In order to encode the residual using HEVC intra coding, the residual is scaled to values from 0 to 255. The minimum value of the residual is encoded with 8 bits in order to recover the non-shift residual in the decoder. The scaled residual blocks of the whole image are reconstructed to get the residual image, which is encoded by HEVC intra coding.

2.6. The components of bit stream

The components of the bit stream is shown in Fig.4.

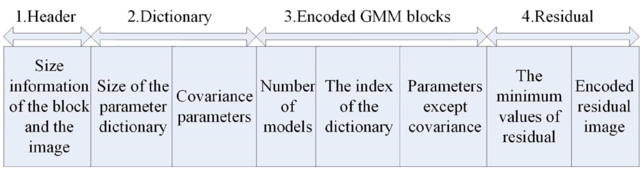


Fig. 4. The components of the bit stream.

In summary, the encoded bit stream has mainly four parts.

The first part is the header of the bit stream, in which 7 bits for the width of the block, 7 bits for the height of the block, 6 bits for the number of blocks in a column of the image and 6 bits for the number of blocks in a row. In this way, an image with the resolution of 8192×8192 at most can be processed by the proposed algorithm.

The second part is the parameter dictionary. In this part, the size of the parameter dictionary is encoded first with 5 bits. With the certain size of the parameter dictionary, the following parameters encoded here can be decoded correctly. And the covariance parameters are encode with 55 bits as described in Section 2.3 for each tuple of the dictionary.

The third part is the information of encoded GMM blocks. The number of models should be encoded with 5 bits first. Every model needs another 5 bits to encode the index of the dictionary. The encoded parameters except covariance are also in this part.

The last part includes the encoded residual image and the encoded minima of the non-shifted residual which are encoded with 8 bits.

3. EXPERIMENTS

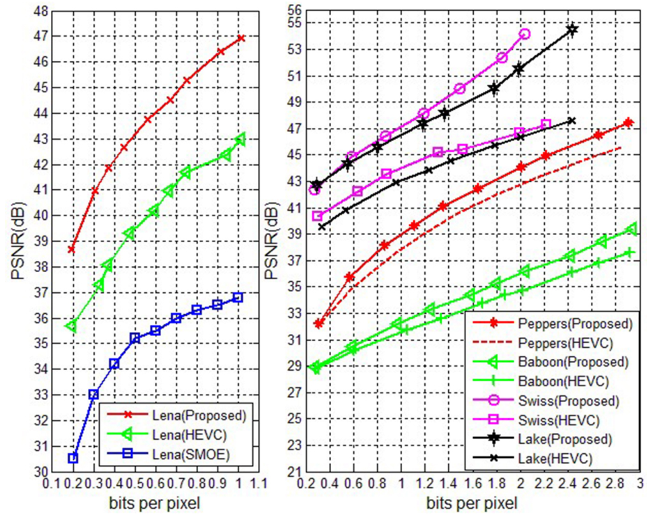
We compare the rate-distortion curves of the proposed method, SMOE and HEVC intra coding. For comparison with SMOE, the size of Lena is 512×512. Other test images has the resolution of 256×256. The images are processed into blocks of size 64×64 in the experiments. The size of the block can be arbitrary in fact. All images are transformed into YUV 4:2:0 formats.

The calculation of the bits per pixel (bpp) of the results is based on the 24-bit depth setting. PSNR is the average of the three channels of the decoded image.

Fig.5 shows the rate-distortion curves of the three methods. The bpp range of the images takes from about 0.2 to 3 while PSNR is controlled lower than 55dB except Lena. The bpp range for Lena is the same as that in [7].

The coding efficiency of HEVC is much better than SMOE in case that there is no extra coding method used which can be seen from Fig.5 (a). And it can be seen from both graphs, the proposed method has a comparable efficiency to HEVC or SMOE. It can also be concluded that the results of the images with large area of background like Swiss and Lake are better than the others.

Fig.6 shows the image and its partial enlargement of the output of decoding compared with HEVC for the image "Lena". Our method can reproduce texture and color correctly compared with HEVC which can be seen from the partial enlargement of the area showed by a red rectangular box.



(a) Lena 512×512 (b) Other images 256×256

Fig. 5. The rate-distortion curves comparing SMOE in [7], HEVC and the proposed method.

(a) Original image (b) Proposed (c) HEVC

(d) Original image (e) Proposed (f) HEVC

Fig. 6. The comparison of decoded images between the proposed method and HEVC.

4. CONCLUSIONS AND FUTURE WORK

In this paper, we proposed an image compression scheme based on GMM. Our method introduced GMO to determine the optimal number of models used in GMM and avoid local optimum of EM algorithm. In this way, the optimal GMM parameters with best coding efficiency can be determined. Meanwhile, the parameter dictionary can obviously reduce the bits used to encode the parameters.

So far, our work has focused on the problems when GMM used in image compression. The variable-length encoding can be considered to improve the coding efficiency of the parameters in the future work.

5. REFERENCES

[1] N. N. Ponomarenko, V. V. Lukin, K. O. Egiazarian, et al. "Adaptive visually lossless JPEG-based color image compression," in *SIViP*, vol. 7, no. 3, pp. 437-452, 2013.

[2] C. Christopoulos, A. Skodras, and T. Ebrahimi, "The JPEG2000 still image coding system: An overview," *IEEE Transactions on Consumer Electronics*, vol. 46, no. 4, pp. 1103-1127, Nov 2000.

[3] F. Belghith, H. Kibeya, M. A. B. Ayed, N. Masmoudi, "Fast coding unit partitioning method based on edge detection for HEVC intra-coding," in *SIViP*, vol. 10, no. 5, pp. 811–818, 2016.

[4] S. Cherifa and R. Messaoud, "New technique to use the GMM in speaker recognition system (SRS)," in *ICCAT*, Sousse, pp. 1-5, 2013.

[5] Dongxiang Zhou and Hong Zhang, "Modified GMM background modeling and optical flow for detection of moving objects," in *ICSMC*, vol. 3, pp. 2224-2229, 2005.

[6] R. Verhack, T. Sikora, L. Lange, G. Van Wallendael, and P. Lambert, "A universal image coding approach using sparse steered Mixture-of-Experts regression," in *ICIP*, Phoenix, AZ, pp. 2142-2146, 2016.

[7] R. Verhack, S. Van De Keer, G. Van Wallendael, T. Sikora, and P. Lambert, "Color prediction in image coding using Steered Mixture-of-Experts," in *ICASSP*, New Orleans, LA, pp. 1288-1292, 2017.

[8] B. Kolewe, A. Haghani, R. Beckmann, and T. Jeinsch, "Gaussian mixture regression and local linear network model for data-driven estimation of air mass," *IET Control Theory & Applications*, vol. 9, no. 7, pp. 1083-1092, April 2015.

[9] J. H. Zhou, C. K. Pang, and W. Yan, "Gaussian mixture model for new fault categories diagnosis," in *ETFA*, Limassol, pp. 1-6, 2017.

[10] Zhou Wang, A. C. Bovik, H. R. Sheikh, and E. P. Simoncelli, "Image quality assessment: from error visibility to structural similarity," *IEEE Transactions on Image Processing*, vol. 13, no. 4, pp. 600-612, April 2004.