AN IMAGE CODING APPROACH BASED ON MIXTURE-OF-EXPERTS REGRESSION USING EPANECHNIKOV KERNEL

Boning Liu[1], Yan Zhao[1], Xiaomeng Jiang[2]*, and Shigang Wang[1]*

[1]College of Communication Engineering, Jilin University, Changchun, 130012, P. R. China.
[2]College of Mathematics, Jilin University, Changchun, 130012, P. R. China.

ABSTRACT

In this paper, we propose an optimal modeling framework for image compression using EMM (Epanechnikov Mixture Model). Epanechnikov Kernel and its correlated statistics are basement of our Epanechnikov Mixture Regression (EMR). In our scheme, the stochastic processes of the pixel values are modelled as an EMM with K experts in three-dimensional space and then we use EMR to search for the optimal solution, whose parameters are determined through EM (Expectation-Maximization) algorithm. In the process of regression, the conditional density is the regression kernel function. Experimental results show that the proposed scheme is effective especially for the image with complex texture without consuming extra bits compared to Gaussian Mixture Regression (GMR).

Index Terms— Epanechnikov Mixture Regression, Epanechnikov Mixture Models, Epanechnikov Kernel, Gaussian Mixture Regression, Image coding, Kernel Density Estimation

1. INTRODUCTION

Image compression has been an intense field with a lot of advances coming up these years. However, as for kernel based statistical modeling methods of image coding, not much progress has been made. Kernel regression is a state-of-the-art theory widely applied in image processing [1], and our purpose is to encode images with the kernel-based mixture of experts (ME) model for linear regression [2]. Therefore, kernel function plays a significant role in our scheme. Though Gaussian kernel function is frequently used in Bayesian framework [3], there are also several existing kernel functions such as epanechnikov, triangular, uniform, biweight, triweight and cosine [4]. Other than Gaussian kernel, Epanechnikov kernel has a discontinuous distribution as well as a rapid slope [5]. Therefore, in our paper we select Epanechnikov kernel function to accomplish the proposed coding scheme. Our coding strategy is motived by sparse *steered mixture-of-experts regression* (SMoE) by Verhack R [6], which assumes that image pixel values are instantiations of non-linear or non-stationary random process that can be modelled by spatially piecewise stationary Gaussian processes. As for reconstructed image, a major disadvantage of the GMM based compression is that it cannot well model the images with complex texture. The goal in the paper is to exploit Epanechnikov kernel function in mixture-model image coding in order to fill the gaps of the space-continuous *Gaussian Mixture Model* (GMM) [7].

Epanechnikov kernel has already been used in mean-shift algorithm [8][9][10], 3D point cloud robust registration algorithm [11] and neural network [12], etc. We consult from mathematical statistics and kernel density estimation to expand the theory of Epanechnikov kernel [13][14]. In this paper, we provide an accurate functional expression of Epanechnikov kernel, meanwhile we figure out the marginal distribution, conditional distribution and the conditional mean value of Epanechnikov kernel in three-dimensional space. Under the Bayesian framework, image pixels are assumed as local experts through Epanechnikov process with global support. The encoder modeling task thus involves estimating the parameters of the model. For our approach, we originally put the three-dimensional *Epanechnikov Mixture Model* (EMM) into image statistical modeling, which estimates the component centroid, variances, together with the weight of each "expert". The parameters can be estimated from the training data by the *Expectation-Maximization* (EM) algorithm [15]. All parameters are then passed to the *Epanechnikov Mixture Regression* (EMR) that steers regression with the kernel function and achieves the maximum probability of amplitude towards each pixel location. The regression kernel function in our method is the conditional probability distribution of Epanechnikov kernel.

The rest of the paper is organized as follows: the specific theory of Epanechnikov kernel and the statistics of Epanechnikov Mixture Regression are given in Section 2. Coding approach is presented in Section 3. Experimental results are shown and analyzed in Section 4. Finally, we draw a conclusion is in Section 5.

This work is supported by the National Natural Science Foundation of China (No.61631009, No. 61771220) and the National Key R&D Program of China (2017YFB1002900, 2017YFB0404800) (E-mail: liuboning_jlu@sina.com, zhao_y@jlu.edu.cn, jxmlucy@sina.com)

2. MIXTURE-OF-EXPERTS REGRESSION USING EPANECHNIKOV KERNEL

2.1. Epanechnikov Kernel Density Estimation

As for an image, *Kernel Density Estimation* (KDE) works at pixel level. According to KDE, we can estimate the probability density of a random distribution at any point:

$$P(\boldsymbol{x_t}) = \frac{1}{N}\sum_{i=1}^{N} K(\boldsymbol{x_t} - \boldsymbol{x_i}) \quad (1)$$

where $K(\bullet)$ is a multivariate kernel function satisfying the condition $K(t) \geq 0$ and $\int K(t)dt = 1$ .There are N data points $\boldsymbol{x_i} \in \{\boldsymbol{x_1}, \boldsymbol{x_2}, \cdots, \boldsymbol{x_N}\}$ and $\boldsymbol{x_t}$ is the observed variable.

Then, the probability density distribution of three-dimensional Epanechnikov kernel can be estimated by:

$$P(x_i) = \frac{15}{8\pi N h^3}\sum_{i=1}^{N}\prod_{j=1}^{3}(1 - (\frac{x_{tj} - x_{ij}}{h})^2) \quad (2)$$

where h is the univariate coefficient.

In our scheme, the probability density must be expressed by covariance matrix as follows:

$$p_{xyz}(\boldsymbol{\varphi}) = \frac{15}{8\pi\sqrt{7^3|\boldsymbol{\Sigma}|}}\left[1 - \frac{1}{7}(\boldsymbol{\varphi} - \boldsymbol{\mu})^T \boldsymbol{\Sigma}^{-1}(\boldsymbol{\varphi} - \boldsymbol{\mu})\right] \quad (3)$$

where $\boldsymbol{\varphi}$ is a three-dimensional random variable $(x, y, z)^T$ and $\boldsymbol{\mu} = (\mu_x, \mu_y, \mu_z)^T$ is the mean value. Σ is a 3×3 covariance matrix.

In order to pave the way for conditional probability distribution, it is necessary to deduce the marginal probability distribution of $p_{XYZ}(\boldsymbol{\varphi})$ which, in our model, is the joint distribution of (X, Y):

$$p_{xy}(\boldsymbol{\delta}) = \frac{5}{14\pi\sqrt{\boldsymbol{R}}}\left[1 - \frac{1}{7}(\boldsymbol{\delta} - \hat{\boldsymbol{\mu}})^T \boldsymbol{R}^{-1}(\boldsymbol{\delta} - \hat{\boldsymbol{\mu}})\right]^{\frac{3}{2}} \quad (4)$$

where $\boldsymbol{\delta}$ is the two-dimensional random variable $(x, y)^T$ and $\hat{\boldsymbol{\mu}} = (\mu_x, \mu_y)^T$ is mean value. $\boldsymbol{R}$ is a 2×2 covariance matrix.

In particular, the formulas presented are all among the intervals of $\frac{1}{7}(\boldsymbol{\varphi} - \boldsymbol{\mu})^T \boldsymbol{\Sigma}^{-1}(\boldsymbol{\varphi} - \boldsymbol{\mu}) \leq 1$ and $\frac{1}{7}(\boldsymbol{\delta} - \hat{\boldsymbol{\mu}})^T \boldsymbol{R}^{-1}(\boldsymbol{\delta} - \hat{\boldsymbol{\mu}}) \leq 1$, while the function value is 0 otherwise.

2.2. Epanechnikov Mixture Regression

The principle of our regression is to optimize a vector z according to a vector $(X, Y) \in \mathbb{R}^2$. In this paper, we define x as the horizontal axis of an image as well as y as the vertical axis, and z is the grey value at location $\boldsymbol{\delta}$ and here comes the joint pdf $p_{XYZ}(\boldsymbol{\varphi}) \in \mathbb{R}^3$.

$$p_{XYZ}(\boldsymbol{\varphi}) = \sum_{j=1}^{K}\alpha_j p_{xyz}(\boldsymbol{\varphi})\Big|_{(\mu_j, \Sigma_j)} \quad (5)$$

and $\sum_{j=1}^{K}\alpha_j = 1$, $\boldsymbol{\mu_j} = \begin{bmatrix}\mu_{X_j}\\ \mu_{Y_j}\\ \mu_{Z_j}\end{bmatrix}$, $\hat{\boldsymbol{\mu}}_{\boldsymbol{j}} = \begin{bmatrix}\mu_{X_j}\\ \mu_{Y_j}\end{bmatrix}$ $\boldsymbol{\Sigma_j} = \begin{bmatrix}\Sigma_{X_jX_j} & \Sigma_{X_jY_j} & \Sigma_{X_jZ_j}\\ \Sigma_{Y_jX_j} & \Sigma_{Y_jY_j} & \Sigma_{Y_jZ_j}\\ \Sigma_{Z_jX_j} & \Sigma_{Z_jY_j} & \Sigma_{Z_jZ_j}\end{bmatrix}$,
$\boldsymbol{R_j} = \begin{bmatrix}\Sigma_{X_jX_j} & \Sigma_{X_jY_j}\\ \Sigma_{Y_jX_j} & \Sigma_{Y_jY_j}\end{bmatrix}$.

The parameters set of each model is $\Phi_j = (\alpha_j, \boldsymbol{\mu_j}, \boldsymbol{\Sigma_j})$ where α_j, $\boldsymbol{\mu_j}$ and $\boldsymbol{\Sigma_j}$ represent the weight, center and covariance of the component j respectively.

Follow on, we work out the conditional probability distribution:

$$p_Z(Z|(X, Y) = \boldsymbol{\delta}) = \frac{p_{xyz}(\boldsymbol{\varphi})}{p_{xy}(\boldsymbol{\delta})}\Big|_{(\mu_j, \Sigma_j)} \quad (6)$$

$$p_Z(Z|(X, Y) = \boldsymbol{\delta}) = \frac{3}{4}\sqrt{\frac{|\boldsymbol{R}|}{7|\boldsymbol{\Sigma}|}}\frac{1 - \frac{1}{7}(\boldsymbol{\varphi} - \boldsymbol{\mu})^T \boldsymbol{\Sigma}^{-1}(\boldsymbol{\varphi} - \boldsymbol{\mu})}{\left[1 - \frac{1}{7}(\boldsymbol{\delta} - \hat{\boldsymbol{\mu}})^T \boldsymbol{R}^{-1}(\boldsymbol{\delta} - \hat{\boldsymbol{\mu}})\right]^{\frac{3}{2}}}\Big|_{(\mu_j, \Sigma_j)} \quad (7)$$

Eq.7 is deduced by Eq.6. Regressing our Epanechnikov Mixture Models means finding the optimal gray value at (x, y) through the conditional probability distribution (Eq.7). The parameters can be estimated from the training data using the Expectation-Maximization (EM) algorithm.

Finally, the conditional mean must be completed, which indicates the intensity of gray value signal around the center of a component:

$$m_j(x, y) = \mu_{Z_j} + \frac{(\Sigma_{X_jX_j}\Sigma_{Y_jZ_j} - \Sigma_{X_jY_j}\Sigma_{X_jZ_j})y + (\Sigma_{X_jZ_j}\Sigma_{Y_jY_j} - \Sigma_{X_jY_j}\Sigma_{Y_jZ_j})x}{|\boldsymbol{R_j}|} \quad (8)$$

Mixing weights is the softmax function widely used in artificial neural networks. It defines how important the role each $m_j(x, y)$ plays:

$$w_j(x, y) = \frac{\alpha_j p_{xy}(\boldsymbol{\delta})}{\sum_{j=1}^{K}\alpha_j p_{xy}(\boldsymbol{\delta})}\Big|_{(\mu_j, \Sigma_j)} \quad (9)$$

We can find from Eq.10 that the regression kernel function $m(x, y)$ is with global support:

$$\hat{Z} = m(x, y) = \sum_{j=1}^{K} m_j(x, y) w_j(x, y) \quad (10)$$

The idea of our EMR is that the gray value at location (x, y) is approximated from the weighted sum over all K mixture components (Eq.10). After finding every optimum parameter set Φ_j, we put it into $m_j(x, y)$ (Eq.8) and $w_j(x, y)$ (Eq.9), which are further used to recover the estimated image by Eq.10.

3. PROPOSED CODING APPROACH

The image is divided into blocks in our experiment. As for each block, the first step is to get the image energy entropy, through which we determine the number of models for certain image. *Image Energy Entropy* reflects the amount of average information in an image. The observed variable is gray value and we calculate the energy entropy H of it [16]:

$$H = \sum_{i=0}^{255} p_i \log p_i \quad (11)$$

where p_i represents the percent of pixels whose gray value is i. In 16×16 block, the piecewise sections of H are [0,2.5], [2.5,2.8], [2.8,3.2] and [3.2,4]. As for 32×32 block, they are [0,1.1], [1.1,1.5], [1.5,2], [2,2.4], [2.4,3] and [3,4]. We define more models for the sections with larger H and in addition we allocate different groups of model-numbers towards the sets of H-sections on experiments at different bitrates.

Then, we step into Epanechnikov Mixture Regression for each block. The Expectation-Maximization (EM) algorithm, which is initialized by k-means++, is used to estimate the parameters set $\Phi_j = (\alpha_j, \boldsymbol{\mu}_j, \boldsymbol{\Sigma}_j)$ for every component j . After obtaining the optimal parameters, we calculate $\hat{z}$ to reconstruct the modeled block. The reconstructed image can be obtained by putting the modeled blocks together.

The parameters that need to be encoded are α_j , three parameters in $\boldsymbol{\mu}_j = \left(\mu_{X_j}, \mu_{Y_j}, \mu_{Z_j}\right)^T$ and six parameters of $\Sigma_{X_jX_j}$, $\Sigma_{X_jY_j}$, $\Sigma_{X_jZ_j}$, $\Sigma_{Y_jY_j}$, $\Sigma_{Y_jZ_j}$, $\Sigma_{Z_jZ_j}$ Thus, each component has ten parameters for encoding. Blocksizes of our experiments are 16×16 or 32×32. Table.1 gives the allocation of bits for these parameters.

Table 1: The range of bits consumed in each parameter

bits/model		α_j	$\boldsymbol{\mu_j}$	$\boldsymbol{\Sigma_j}$	Φ_j
16×16	EMM	7	[4,8]	[6,13]	77
	GMM	7	[4,9]	[6,14]	79
32×32	EMM	7	[5,9]	[8,13]	88
	GMM	7	[5,9]	[8,13]	87

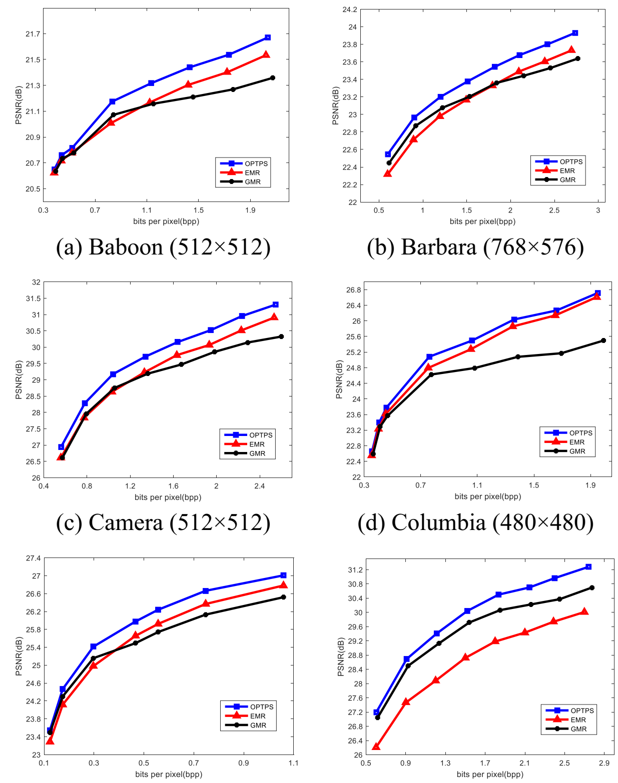
4. EXPERIMENTAL RESULTS AND ANALYSIS

4.1 Experimental results

In our experiments, we perform 7 iterations and set the threshold value 10^{-6} in EM algorithm for the regression of the conditional probability function (Eq.7). We implement the framework using Gaussian Mixture Regression as well in order for comparison. In addition, we also consider a mixture method, which is referred to the *Optimal PSNR Scheme* (OPTPS). In OPTPS, each image block is encoded utilizing both EMM and GMM respectively. Then we can get two modeled results, between which we choose the block with better PSNR. In this way, a relative increase of image quality is achieved.

Grayscale images Baboon, Barbara, Camera, Columbia, Lena and Peppers are used in the test. The relationship between the bit consumption and the reconstructed image quality of three coding schemes, EMR, GMR, and OPTPS is shown in Fig.1, from which we can easily observe the improvement of the coding efficiency based on our Epanechnikov Mixture Model. What is more considerable is the absolute predominance of OPTPS for all test images. From the observation on the overall trend of six test figures, we can see that at low bitrates, the number of models for encoding a block is not sufficient enough to show any details, and both regression methods suffer from lack of components. As a result, there is similarity between EMR and GMR at low bitrates. What's more, the curve trend mainly indicates that a considerable compression gain is achieved especially for high bitrates.

Fig.2, Fig.3 and Fig.4 depict the contrast of details among EMR, GMR and OPTPS with key areas noted. Fig.2 shows that EMR outperforms GMR in dealing with image with complex details. *Baboon* has wild beard, as for which, more messy details have been reconstructed by EMR. In Fig.3, from the reconstruction of the pillars' color change, we can know that EMR can show a dominant change towards image regions that having distinct span of gray scale while GMR ignores the transition.



(a) Baboon (512×512) (b) Barbara (768×576)
(c) Camera (512×512) (d) Columbia (480×480)
(e) Lena (512×512) (f) Peppers (512×512)

Fig. 1. The bpp-PSNR map comparing GMR, proposed EMR and OPTPS

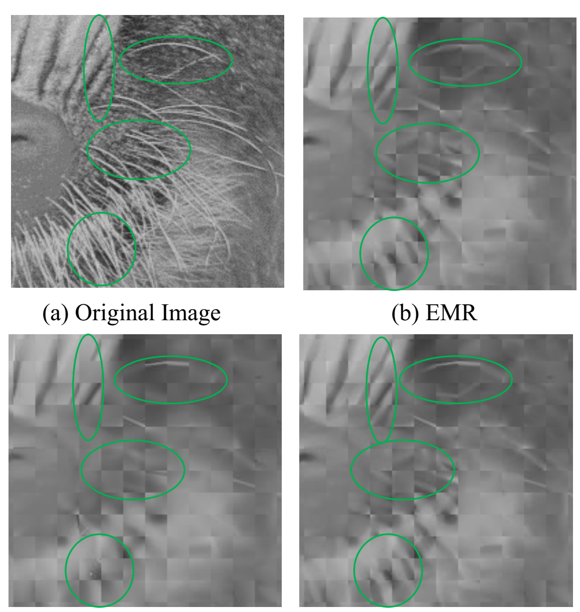
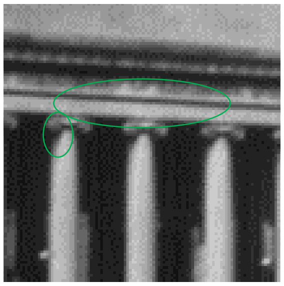
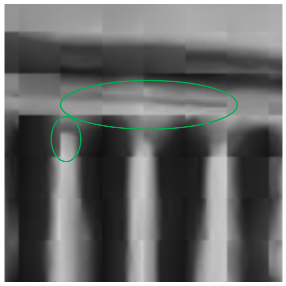
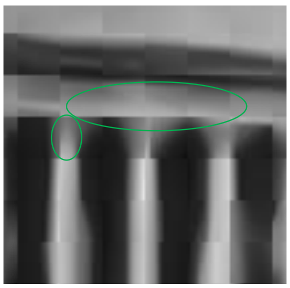

(a) Original Image (b) EMR
(c) GMR (d) OPTPS

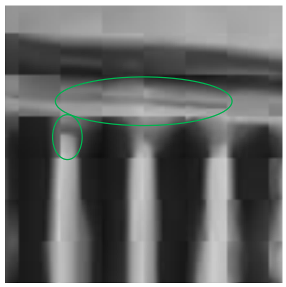
Fig.2. Experimental results on *Baboon* at 1.41 bpp





(a) Original Image (b) EMR

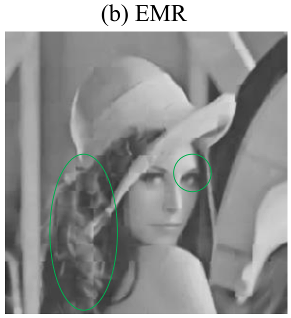




(c) GMR (d) OPTPS

Fig.3. Experimental results on *Columbia* at 0.75 bpp

(a) Original Image (b) EMR



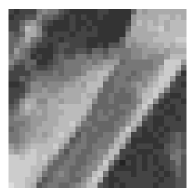
(c) GMR (d) OPTPS

Fig.4. Experimental results on *Lena* at 0.56 bpp

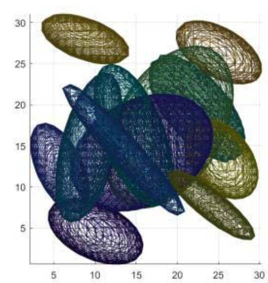
As a representative image for human, *Lena* is a wonderful example as shown in Fig.4, from which we can know our scheme can also be a good approach for reconstructing figure details. In Fig.4, we can easily see the difference between (b) and (c) in eye region and hair region. Above all, we can find the excellent performance of EMR on non-flat images, especially complex texture. As for the complexity of EMR, the running time of EMR is about 1.2 to 1.7 times longer than that of GMR. The OPTPS we present shows the most comprehensive advantage over both EMR and GMR coding. We mainly focus on improving the model and consider less about coding optimization after modeling in this paper, therefore the whole image coding performance is not yet comparable to the state-of-the-art image coding frameworks, such as JPEG and HEVC.

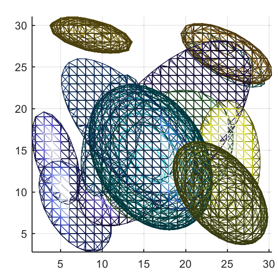
4.2 Model analysis

The model-based coding method we proposed has its particular advantage, which results from the original Epanechnikov kernel. In this part, we preliminarily explore the correlation between the coding regression and the three-dimensional Epanechnikov kernel. Fig.5 illustrates that how is the proposed model influence the image quality by modeling a 32×32 pixel block of Lena at 1.0 bpp in comparison to Gaussian Mixture Model at the same rate. Fig.5(b) and Fig.5(c) derive from the modeling with 12 models, from which we can see the better effect for EMM. Fig.5(d) and (e) are the top views of three-dimensional mixture models. From the two figures, we can see that three-dimensional Epanechnikov kernel has a tighter distribution than three-dimensional Gaussian kernel under the same data set. The concentrated-distribution ellipsoids of EMM can enhance the correlation of regression, while GMM has a more incompact distribution which may not present a better representation of data changes



(a) Original (b) EMR (c) GMR





(d) Top view of EMM (e) Top view of GMM

Fig.5. Mixture model analysis

3. CONCLUSION

In this paper, we explore the possibility of using an Epanechnikov-kernel based regression on image compression. Ahead of all, we enrich the theoretical basis of Epanechnikov kernel and then make good use of it in image coding. At higher bitrates, the reconstructed image quality of the proposed method is superior to GMR based coding, and as we notice, EMR coding has the advantage that it can reconstruct at least the baseplate for texture-complicated image. With the advantages of EMR and GMR coding, our proposed OPTPS can get the maximization of the image quality.

5. REFERENCES

[1] H. Takeda, S. Farsiu, and P. Milanfar, "Kernel Regression for Image Processing and Reconstruction," *IEEE Trans on Image Processing*, vol. 16, no. 2, pp. 349–366, 2007.

[2] J. Santarcangelo, and X. P. Zhang, "Kernel-based mixture of experts models for linear regression," *IEEE International Symposium on Circuits and Systems*, pp. 1526-1529, 2015.

[3] M. R. Morelande and B. Ristic, "Bayesian analysis of finite Gaussian mixtures," *IEEE International Conference on Acoustics Speech and Signal Processing*, pp. 3962-3965, 2010.

[4] S. C. Gustafson, "Bayesian Threshold Estimation," *IEEE Trans*, vol. 52, no. 3, pp. 400-403, 2009.

[5] M. Zhang, L. Xu, "Constructing re-substitution entropy estimator with discontinuous kernels," *IEEE International Conference on Wavelet Analysis and Pattern Recognition*, pp. 128-135, 2013

[6] R. Verhack, T. Sikora and L. Lange, "A universal image coding approach using sparse steered Mixture-of-Experts regression," *IEEE International Conference on Image Processing*, pp. 2142-2146, 2016.

[7] J. Yang and X. Liao, "Compressive Sense by Learning a Gaussian Mixture Model," *IEEE Trans on Image Processing*, vol. 24, no.1, pp. 106-119, 2015.

[8] J. Zhang and J. Fang, "Mean-shift algorithm integrating with SURF for tracking," *IEEE International Conference on Natural Computation*, vol.2, pp. 960-963, 2011.

[9] Q. Guo, X. Chang, and H. Chu, "Mean-Shift of Variable Window Based on the Epanechnikov Kernel," *IEEE International Conference on Mechatronics and Automation*, pp. 2314-2319, 2007.

[10] D. Li, L. Xu, and Y. Wu, "Improved CAMShift object tracking based on Epanechnikov Kernel Density Estimation and Kalman filter," *IEEE Control and Decision Conference*, pp. 3120-3126, 2017.

[11] H. Lin and B. Liu, "Research on a novel 3D point cloud robust registration algorithm," *IEEE International Conference on Computer Design and Applications*, vol.5, pp. 441-445, 2010.

[12] M. Ohiro and T. Nishimura, "US image improvement using fuzzy Neural Network with Epanechnikov kernel," *IEEE Industrial Electronics, pp.* 2130-2135, 2009.

[13] G. R. Terrell and D. W. Scott, "Variable Kernel Density Estimation," *Annals of Statistics*, vol.20, no.3, pp.1236-1265, 1992.

[14] R. J. Elliott, "Stochastic Calculus and Applications, " Springer, 1982.

[15] T. Moon, "The Expectation-Maximization Algorithm," *IEEE Signal Processing Magazine*, vol. 13, no. 6, pp.47–60, 1996.

[16] Chen, Cunjian, and J. Zhang, "Wavelet Energy Entropy as a New Feature Extractor for Face Recognition," *IEEE International Conference on Image and Graphics*, pp. 616-619, 2007.