International Journal of Engineering Technology and Applied Science
(ISSN: 2395 3853), Vol. 1 Issue 6 December 2015

Estimation Analysis of Probabilistic Density Function of Improved Gaussian Mixture Model for Motion Sensing based on Least Square Cross Validation and Gaussian PSO with Gaussian Jump

Poonam Tripathi[1], Sarla Singh[2], Anil Mishra[3]
M-tech scholar[1], Asst. Prof. JNCT REWA[2], HOD JNCT REWA[3]
engineerpoonam@gmail.com[1], sarlasingh_mits@rediffmail.com[2], anilmishraec@gmail.com[3]

ABSTRACT:- This paper discusses in detail the Gaussian mixture model (GMM) is always used to estimate the underlying density function in many real applications. In this pa- per, we develop an improved Gaussian mixture model (IGMM) based on least-squares cross-validation (LSCV) and Gaussian PSO with Gaussian jump (GPSOGJ). According to least-squares cross-validation, a new error measure criterion is derived which is used to evaluate the estimation error between the true density function and the estimated density function. Then, GPSOGJ is used to find the optimal parameters that can make the estimation error reach the minimum. In our experiments, we com- pare IGMM with two existing methods as GMM with Parzen window (PGMM) and GMM based on particle swarm optimization (PSOGMM) on four probability distributions: Uniform density, Normal density, Exponential density, and Rayleigh density. The experimental results demonstrate that our strategy can get good estimation performance when the corresponding parameters are optimized with GPSOGJ.

KEYWORDS:- Gaussian mixture model, Gaussian PSO with Gaussian jump

INTRODUCTION

It is very important to estimate the underlying probability density function in many practical applications. Gaussian Mixture Model (GMM) is the most popular parametric probability density estimator which represents the unknown density function with the weighted sum of Gaussian probability density functions. The estimated density function can be written as the following form under the framework of GMM. For the sake of simplicity, we only expand our discussion based on the one-dimensional dataset. Let (x) in Eq. (1.1) denote the estimated density function:

$$\hat{f}(x) = \sum_{i=1}^{N} \omega_i \, G(x-\mu_i,\sigma_i^2) \quad \text{Eqn.1.1}$$

Where, N is the number of given one dimensional dataset $X = \{x_1, x_2, x_3, \ldots, x_n\}$, μ_i and σ_i are the mean and variance of the i-th Gaussian component, $G(x-\mu_i,\sigma_i^2) = 1/\sqrt{2\pi}\,\sigma_i \exp\left[-\frac{(x-\mu_i)^2}{2\sigma_i^2}\right]$, the weight ω_i should satisfy the following constraint:

$$\sum_{i=1}^{N} \omega_i = 1, \forall\, \omega_i > 0.$$

When the parameter ω_i, μ_i, and (i=1, 2, 3……., N) in GMM are determined, the underlying density function can be obtained. The main method used to determine the parameters is Expectation-Maximization Algorithm (EMA). The published literatures all report that their EMA based algorithms can improve the estimation performance of GMM. However, the computational complexity of GMM with EMA is extraordinarily high, and this point is always neglected by the re- searchers. In our study, we try to reduce the time cost for solving GMM and at the same time maintain a satisfied estimation performance. In order to achieve this goal, the Particle Swarm Optimization (PSO) method is adopted. Therefore, in this paper, we develop an improved Gaussian mixture model (IGMM) based on least-squares cross-validation (LSCV) and Gaussian PSO with Gaussian jump (GPSOGJ). According to least-squares cross-validation, a new error measure criterion is derived which is used to evaluate the estimation error between the true density function and the estimated density function. Then, GPSOGJ is used to find the optimal parameters that can make the estimation error reach the minimum. In our experiments, we compare IGMM with the existing methods- GMM with Parzen window (PGMM) and GMM based on particle swarm optimization (PSOGMM) on four probability distributions: Uniform density, Exponential density, and Rayleigh density. The experimental results demonstrate that our strategy can get good estimation performance when the corresponding parameters are optimized with GPSOGJ.

LEAST SQURES CROSS VALIDATION (LSCV)

There are three kinds of parameters which should be determined in GMM: the weight ω_i, the mean μ_i, and the variance σ_i of the Gaussian density function G (x- μ_i, σ_i^2), (i=1, 2, …..,N). The setup of parameter μ_i is enlightened by parzen window method which computes the underlying density according to Eqn. 1.2.

$$\widehat{f}(x) = \frac{1}{Nh}\sum_{i=1}^{N} k\left(\frac{x-x_i}{h}\right)\frac{1}{Nh}\sum_{i=1}^{N}\exp\left[-\frac{1}{2}\left(\frac{x-x_i}{h}\right)^2\right.$$

Eqn. 1.2

Where N is the number of samples in the dataset, h is the band-width parameter, and K is the kernel function. The bandwidth parameter h is determined by N which meets the following requirements: $\lim_{N\to+\infty} h(N) = 0$ and $\lim_{N\to+\infty}[N \times h(N)] = \infty$. The kernel is a real valued, non-negative, and integrable function which satisfies the following conditions: $\int_{-\infty}^{+\infty} K(u) = 1$ and $\forall\, u, K(u) = K(-u)$. Let μ_i = x_i, (i=1, 2, …., N), then Eqn.1.1 could be as rewritten Eqn. 1.3.

$$\widehat{f}(x) = \sum_{i=1}^{N}\frac{\omega_i}{\sqrt{2\pi}\sigma_i}\exp\left[-\frac{(x-x_i)^2}{2\sigma_i^2}\right]$$

Eqn. 1.3

Then two parameters need to be fixed, i.e. the weight ω_i and the variance σ_i. We want to select the parameters that can make the estimation error between the true density f(x) and the estimated density $\hat{f}(x)$ reach the minimum. The estimation error can be represented as Eqn. 1.4.

$$E_{IGMM} = \int[f(x) - \hat{f}(x)]^2 \qquad \text{Eqn. 1.4}$$

From the Eqn. 1.4, we can get

$$E_{IGMM} = \int[\hat{f}(x)]^2 dx - 2\hat{f}(x)f(x)dx + \int[f(x)]^2 dx \qquad \text{Eqn. 1.5}$$

From the above Eqn. 1.5, we can easily find that the third term $\int[f(x)]^2 dx$ is not related with the parameters that need to be determined, thus can be neglected. The error criterion E_{IGMM} is then changed as Eqn. 1.6.

$$E_{IGMM}^{*} = \int[\hat{f}(x)]^2 dx - 2\hat{f}(x)f(x)dx = A - B$$

Eqn. 1.6

Firstly,

$$A = \int[\hat{f}(x)]^2 dx = \int\left\{\sum_{i=1}^{N}\frac{\omega_i}{\sqrt{2\pi}\sigma_i}exp\left[-\frac{(x-x_i)^2}{2\sigma_i^2}\right]\right\}^2 dx$$

$$= \int\left[\sum_{i=1}^{N}\hat{f}_i^2(x)\right]dx + \int\left[\sum_{i=1}^{N}\sum_{j\neq i}^{N}2\hat{f}_i(x)\hat{f}_j(x)\right]dx$$

$$= \sum_{i=1}^{N}\int\hat{f}_i^2(x)\,dx + 2\sum_{i=1}^{N}\sum_{j\neq i}^{N}\int\hat{f}_i(x)\hat{f}_j(x)dx$$

$$= \sum_{i=1}^{N}A_{1i} + 2\sum_{i=1}^{N}\sum_{j\neq i}^{N}A_{2ij} \qquad Eqn. 1.7$$

Where, $\hat{f}_i(x) = \frac{\omega_i}{\sqrt{2\pi}\sigma_i}\exp\left[-\frac{(x-x_i)^2}{2\sigma_i^2}\right]$.

Now, we will determine the components A_{1i} and A_{2ij}, (i=1, 2, ……,N; j=1,2,……,N; $j \neq i$):

$$A_{1i} = \int\hat{f}_i^2(x)dx = \frac{\omega_i^2}{2\pi\sigma_i}\int\exp\left[-\left(\frac{x-x_i}{\sigma_i}\right)^2\right]d\left(\frac{x-x_i}{\sigma_i}\right)$$

$$v = \frac{x-x_i}{\sigma_i}\frac{\omega_i^2}{2\pi\sigma_i}\int\exp(-v^2)\,dv$$

$$\int\exp(-v^2)\,dv = \frac{\sqrt{\pi}\omega_i^2}{2\sqrt{\pi}\sigma_i}$$

And,

$$A_{2ij} = \int\hat{f}_i(x)\hat{f}_j(x)dx$$

$$= \int\{\frac{\omega_i}{\sqrt{2\pi}\sigma_i}\exp[-\frac{(x-x_i)^2}{2\sigma_i^2}]\frac{\omega_j}{\sqrt{2\pi}\sigma_j}\exp[-\frac{(x-x_j)^2}{2\sigma_j^2}]\}dx$$

$$= \frac{\omega_i\omega_j}{2\pi\sigma_i\sigma_j}\int\{\exp[-\frac{(x-x_i)^2}{2\sigma_i^2}-\frac{(x-x_j)^2}{2\sigma_j^2}]\}dx$$

$$= \frac{\omega_i\omega_j}{2\pi\sigma_i\sigma_j}\int\{\exp[-\frac{\sigma_j^2(x-x_i)^2+\sigma_i^2(x-x_i)^2}{2\sigma_i^2\sigma_j^2}]\}dx$$

$$= \frac{\omega_i\omega_j}{2\pi\sigma_i\sigma_j} \times \int\{\exp[-\frac{(\sigma_i^2+\sigma_i^2)x^2-2(\sigma_i^2x_j+\sigma_j^2x_i)x+(\sigma_i^2x_j^2+\sigma_j^2x_i^2)}{2\sigma_i^2\sigma_j^2}]\}dx$$

$$= \frac{\omega_i\omega_j}{2\pi\sigma_i\sigma_j} \times \int\{\exp[-\frac{ax^2-2bx+c}{2d}]\}dx$$

$$= \frac{\omega_i\omega_j}{2\pi\sigma_i\sigma_j}\exp\left(\frac{b-c}{a}\right)\int\exp\left\{-\left[\sqrt{\frac{a}{2d}}\left(x-\frac{b}{a}\right)^2\right]\right\}dx$$

$$= \frac{\omega_i\omega_j}{\sqrt{2\pi}\sigma_i\sigma_j}\sqrt{\frac{d}{a}}\exp\left(\frac{b-c}{a}\right)$$

$$= \frac{\omega_i\omega_j}{\sqrt{2\pi}}\sqrt{\frac{1}{\sigma_i^2\sigma_j^2}}\exp\left(\frac{\sigma_i^2x_j+\sigma_j^2x_i-\sigma_i^2x_j^2-\sigma_j^2x_i^2}{\sigma_i^2+\sigma_j^2}\right)$$

Where,

$a = \sigma_i^2 + \sigma_j^2, b = \sigma_i^2 x_j + \sigma_j^2 x_i, c = \sigma_i^2 x_j^2 + \sigma_i^2 x_i^2, and\ d = \sigma_i^2\sigma_j^2.$

Thus, we can get the formulation (i.e., Eqn. 1.8) of A by applying A_{1i} and A_{2ij} to Eqn.1.7.

$$A = \int[\hat{f}(x)]^2\,dx = \sum_{i=1}^{N}\frac{\omega_i^2}{2\sqrt{\pi}\sigma_i} + 2\sum_{i=1}^{N}\sum_{j\neq1}^{N}[\frac{\omega_i\omega_j}{\sqrt{2\pi}}\sqrt{\frac{1}{\sigma_i^2+\sigma_j^2}}\exp(\frac{\sigma_i^2x_j+\sigma_j^2x_i-\sigma_i^2x_j^2-\sigma_j^2x_i^2}{\sigma_i^2+\sigma_j^2})] \quad \text{Eqn.1.8}$$

Next, we will give the expression of B:

$$B = 2\int\hat{f}(x)f(x)dx = 2E[\hat{f}_{-i}(x_i)] = \frac{2}{N}\sum_{i=1}^{N}\hat{f}_{-i}(x_i) = \frac{2}{N} \quad \text{Eqn. 1.9}$$

Where $\hat{f}_{-i}(x_i)$ denotes the kernel density estimation of (x_i), $(i = 1,2,\dots\dots\dots,N)$ based on the other (N-1) samples, and this is why we call this manner cross validation. Based on the above analysis, the new error criterion E^*_{IGMM} of obtained by combining Eqn. 1.8 and Eqn. 1.9. So, the problem of finding the optimal parameters of GMM can be transformed into the following optimization formula as shown in Eqn. 1.10

$$\min_{\substack{\omega_1,\omega_2,\dots\omega_N\\ \sigma_1,\sigma_2,\dots\dots\sigma_N}} E^*_{IGMM} = \min_{\substack{\omega_1,\omega_2,\dots\omega_N\\ \sigma_1,\sigma_2,\dots\dots\sigma_N}} \{\sum_{i=1}^{N}\frac{\omega_i^2}{2\sqrt{\pi}\sigma_i} + 2\sum_{i=1}^{N}\sum_{j\neq i}^{N}\left[\frac{\omega_i\omega_j}{\sqrt{2\pi}}\sqrt{\frac{1}{\sigma_i^2+\sigma_j^2}}\exp\left(\frac{\sigma_i^2x_j+\sigma_j^2x_i-\sigma_i^2x_j^2-\sigma_j^2x_i^2}{\sigma_i^2+\sigma_j^2}\right)\right] - \frac{2}{N}\sum_{i=1}^{N}\sum_{j\neq i}^{N}\frac{\omega_i}{\sqrt{2\pi}\sigma_j}\exp\left[-\frac{(x_i-x_j)^2}{2\sigma_j^2}\right]\} \quad Eqn. 1.10$$

$$s.t.\sum_{i=1}^{N}\omega_i = 1, \quad \forall i \in \{1,2,\dots,N\}, \omega_i \ni 0.$$

GAUSSIAN PSO WITH GAUSSIAN JUMP-GPSOGJ

PSO initialize the candidate solutions as a population of particles which is associated with the random particle keeps tracking its coordinates which are associated with the best fitness it has achieved so far, i.e., pbest. Another best value tracked by global version of the particle swarm optimizer is the overall best value, i.e., gbesf, and its location is obtained so far by any particle in the current population. It is worth nothing that PSO may stuck into locally optimal, where the fitness of the particle will have no improvement after certain number of iteration. In order to deal with this problem, the Gaussian PSO with jump is proposed, in which the "jump" mechanism is introduced to help PSO escaping the locally optimal. The procedures of $GPSO_{GJ}$ algorithm discussed later in proposed methodology section.

PROPOSED WORK

Algorithm showing the sequence of step for determination of probabilistic density function of improved Gaussian mixture model for motion sensing based least square cross validation and Gaussian PSO with Gaussian jump, For each particle x, in population P

$$v_i = \underline{v} + (\overline{v} - \underline{v}) \times U_i(0,1);$$

∬ Use uniform probability distribution to initialize the velocity of particle x_i, Where $\overline{v}$ and $\underline{v}$ denotes the upper and lower boundary of velocity

$$x_i = \underline{x} + (\overline{x} - \underline{x}) \times U_i,(0,1);$$

∬ Use uniform probability distribution to initialize the position of particle x_i, Where $\overline{x}$ and $\underline{x}$ denote the upper and lower boundary of search space

$$pbest_i = x_i;$$

∬The personal best position of all particle

END FOR

$$gbest_i = \min_{x_i \in i}[f(x_i)];$$

∬The global best position of all particle

DO

FOR each particle x_i in population P

$$IF\ wait\ [i] \leq \max\ \_waitTHEN$$

∬*wait[i]* records the number of no improvement of fitness of particle x_i. if wait [i] achieve the maximum number of no improvement of fitness *max_wait*, it indicates that the jump operation is needed,

$$v_i \leftarrow |R_1|(pbest_i - x_i) + |R_2|(gbest_i - x_i);$$

∬ R_1 and R_2 are the random number generated from the standard Normal distribution N (0, 1)

$$x_i \leftarrow x_i + v_i;$$

ELSE

$$x_i \leftarrow x_i + \eta . N_i(0,1);$$

∬The Gaussian jump ∬The parameter η is in interval $[0.01(\overline{x} - \underline{x}), 0.1(\overline{x} - \underline{x})]$

COMPUTE f (x_i);

END IF

$$\text{IF } f(x_i) € f(pbest_i)\ THEN$$

$$pbest_i = x_i;$$

∬Update the personal best position of each particle

COMPUTE f(x_i);

$$wait[i] = 0;$$

ELSE

$$wait[i] ++;$$

∬If there is no improvement of fitness of particle x_i, wait[i] increases by one in every iteration

END IF

IF f (x_i) € f (gbest) THEN gbest = pbest,;

END IF

∬Update the global best position of all particles

END FOR

UNTIL the termination condition is met

OUTPUT gbest;

In order to test the estimation performance of GMM with PSG algorithm, four different types of one dimensional artificial datasets are randomly generated as: Uniform dataset (UniD), Normal dataset (NorD), Exponential dataset (ExpD) and Rayleigh dataset (RayD). The probability density functions of these four distributions are as follows;

$$\underset{Uniform}{f(x)} = 1, x \in [0,1];$$

$$\underset{Normal}{f(x)} = \frac{1}{\sqrt{2\pi}} \exp\left(-\frac{x^2}{2}\right), -\infty < x < +\infty;$$

$$\underset{Exponential}{f(x)} = \frac{1}{0.2} \exp\left(-\frac{x}{0.2}\right), x > 0;$$

$$\underset{rayleigh}{f(x)} = x \exp\left(-\frac{x^2}{2}\right), x \geq 0.$$

In our experiment, all datasets can be randomly generated by MATLAB instructions. These instructions have been listed in Table 1 where N denotes the number of samples generated.

Matlab Instructions

Probability density	MATLAB implementation
Uniform density	X = unifrnd (0, 1, N)
Normal density	X= normrnd (0, 1, N)
Exponential density	X = exprnd (0.2, N)
Rayleigh density	X = raylrnd (1, N)

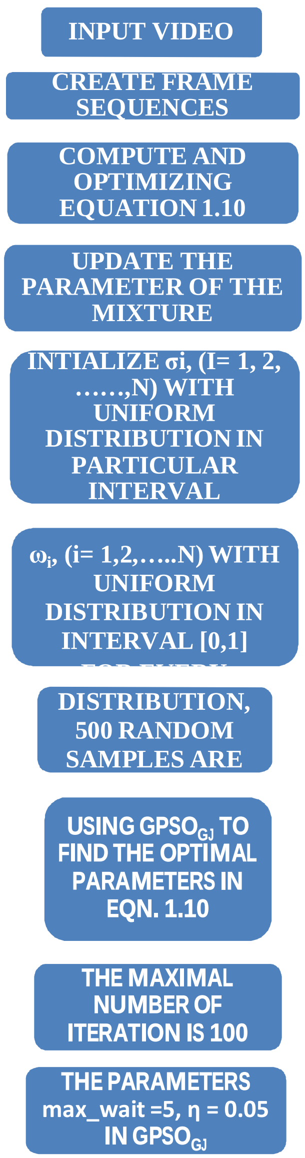


Fig. Flow Block Diagram of Optimization and Computing Process of PDF and MSE

Algorithm 1: Optimizing Eqn. 1.10 with GPSOGJ

1: Initialize σ$_i$, (i= 1, 2, ……,N) with uniform distribution in interval [0, $(\frac{270}{70\sqrt{\pi}})^{1/5} \times s$] and ω$_i$, (i= 1,2,…..N) with uniform distribution in interval [0,1];

2: For every distribution, 500 random samples are generated;

3: Using GPSO$_{GJ}$ to find the optimal parameters in Eqn. 1.10. The maximal number of iteration is 100. The parameters max_wait =5, η = 0.05 in GPSO$_{GJ.}$
Implementation Detail Of Estimation Of Mse

Two Parameters need to be optimized by PSO. We give the following two strategies to initialize the population:

1. First for the variance σ$_i$, $(i = 1, 2, \ldots\ldots, N)$, We let $0 < \sigma_i \le \left(\frac{270}{70\sqrt{\pi}}\right)^{\frac{1}{5}} \times \boldsymbol{s}$, Where s is the standard derivation of training dataset $X = \{x_1, x_2, \ldots\ldots, x_N\}$.

The upper bound of σ_i can be obtained according to the following rule: In parzen window method, $\hat{f}(x)$ is the function of smoothing parameter and training samples X. The h can affect the estimation performance a lot. A smaller h will give a too detailed curve hence leads to small bais and large variance, while a larger h will lead to low variance at the expense of increased bias. So, the upper bound of h is limited, which is$\left(\frac{270}{70\sqrt{\pi}}\right)^{\frac{1}{5}} \times \boldsymbol{s}$. By observing Eqn. 1.6 and 1.7, we find that GMM may treated as the generalized form of parzen window method. GMM is variation parameter estimation model; and parzen window method is a fixation parameter estimation model. So, we let this upper bound of h as the upper bound as the same time, that is to say;

$$0 < \sigma_i \le \left(\frac{\mathbf{270}}{\mathbf{70}\sqrt{\boldsymbol{\pi}}}\right)^{\frac{1}{5}} \times \boldsymbol{s}, (\boldsymbol{i} = \mathbf{1}, \mathbf{2}, \ldots\ldots, \boldsymbol{N})$$

We initialize the population of $\sigma_i, (i = 1, 2, \ldots\ldots, N)$ with the random numbers which obey the uniform density distribution in interval $[0, (\frac{270}{70\sqrt{\pi}})^{1/5} \times s]$.

2. Second, we initialize the population of $\omega_i, (i = 1, 2, \ldots\ldots, N)$with the random number which obey the uniform density distribution in interval [0, 1] and guarantee that $\sum_{i=1}^{N} \omega_i = 1$. Our experiment is carried out according to the algorithm ……:

For each probability density, we use $GPSO_{GJ}$ algorithm to search the optimal parameters based on four different types of univariate artificial dataset. Every type of dataset is generated 100 times randomly. The averaged result based on these 100 datasets is summarized for some distributions. The mean square error (MSE) is used to evaluate the estimation performance.

Parameter for Simulation for motion sensing

The parameters that have been used in simulation are mentioned and briefly discussed below;

1) Number of Gaussian Densities (K): It represents the number of Gaussian densities used that are used to compute the PDF. Calculations have been done for K=3 and K=4.
2) Background Threshold (λ): A threshold λ is applied to the cumulative sum of weights to find the set {1...B} of Gaussians modelling the background.
3) Covariance (σ): Covariance matrix which is used in calculation of initial pdf.
4) Component Threshold: Normally taken as 10.

Simulation Results

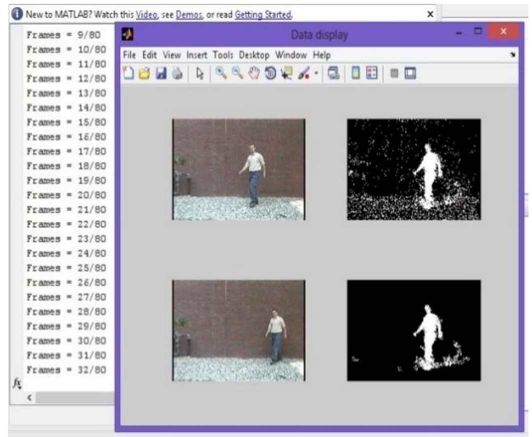
Results on the Zdenek Kalal Database. WALKING PERSON VIDEO.

Video Details.

1) 440 frame Video.
2) 3 fps.
3) Background: Stable.
4) Illumination Change: Partial.
5) Objects to track: Single.

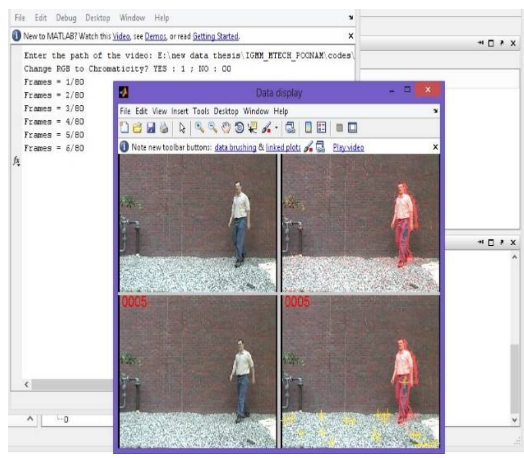
Input walking person Video from ZK Database

Input frame of this is dull and background has also poor lighting so we extracted this frame into best background frame as we can see from figure.



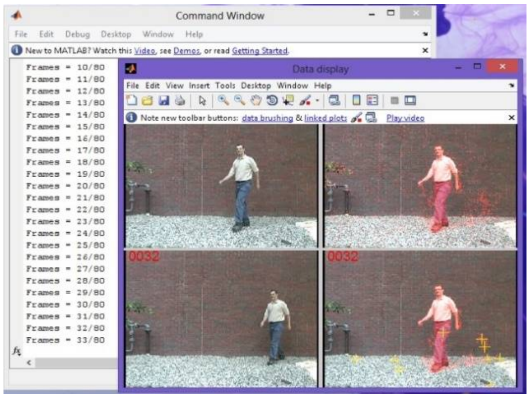
Extracted Best Background Image

Again we can see the best background subtraction and track the object after image subtraction as we can see in the Figure.



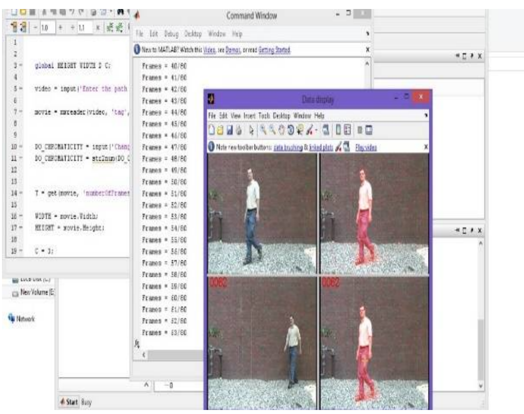
Tracked object Image after Subtraction

False detection is more prominently visible in the initial learning stage that should be removing after using some applications. As we can see in the figure. When some false detection occurred in tracked object image after subtraction frame then we updated the mixture parameter, and the object is traced successfully with few false detection being removed by Morphological operations and filtering. As we can see in results of Figure.

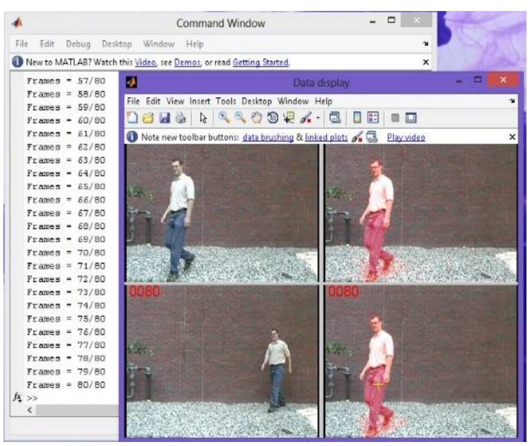


Removed False Detection after Morphological Operations and Filtering

Applying Hue to track the object after removing the false detection in the frame and see results in the Figure.



After applying Hue to Tracked object



Applied Markers on Tracked Object

As we can see from the result that the object has been tracked successfully apart from a few false detections. Specifically gives the images only after background subtraction has been done. The follow up images Figures depict images after application of hue and depicted markers. The False detections are more prominently visible in the initial learning stage. As the mixture parameters are updated, the objects are tracked successfully with a few false detection being removed by morphological operations and filtering. To track the multiple objects we have to extract the best background of this input video and this extracted image is extracted best background image. False detection is more prominently visible in the initial learning stage that should be removing after using some applications. When some false detection occurred in tracked object image after subtraction frame then we updated the mixture parameter, and the object is traced successfully with few false detection being removed by filtering After filtering of image frame of initial learning phase apply hue to detected area for tracking the object. Applying Hue to detected area we have to show the object so we marked the object and traced the object successfully. The video consists of multiple objects that are required to be tracked. The system efficiently tracks both the moving car and the pedestrian. It locks on to moving man once the car is stationary. However the initial learning phase was slightly slower than previous videos owing to the initial visibility in this video is very poor as the illumination change is significant and the camera is at a significant distance away from the object.

EXPERIMENTAL DATA

In order to test the estimation performance of GMM with PSG algorithm, four different types of one dimensional artificial datasets are randomly generated as: Uniform dataset (UniD), Normal dataset (NorD), Exponential dataset (ExpD) and Rayleigh dataset (RayD). The probability density functions of these four distributions are as follows;

$$\underset{Uniform}{f(x)} = 1, x \in [0,1];$$

$$\underset{Normal}{f(x)} = \frac{1}{\sqrt{2\pi}} \exp\left(-\frac{x^2}{2}\right), -\infty < x < +\infty;$$

$$\underset{Exponential}{f(x)} = \frac{1}{0.2} \exp\left(-\frac{x}{0.2}\right), x > 0;$$

$$\underset{rayleigh}{f(x)} = x \exp\left(-\frac{x^2}{2}\right), x \geq 0.$$

IMPLEMENTATIN DETAILS OF GMMPSO$_{GJ}$

Two Parameters need to be optimized by PSO. We give the following two strategies to initialize the population:

1. First for the variance σ_i, $(i = 1,2, \ldots \ldots, N)$, We let $0 < \sigma_i \leq \left(\frac{270}{70\sqrt{\pi}}\right)^{\frac{1}{5}} \times \boldsymbol{s}$, Where s is the standard derivation of training dataset $X = \{x_1, x_2, \ldots \ldots, x_N\}$.

   The upper bound of σ_i can be obtained according to the following rule: In parzen window method, $\hat{f}(x)$ is the function of smoothing parameter and training samples X. The h can affect the estimation performance a lot. A smaller h will give a too detailed curve hence leads to small bais and large variance, while a larger h will lead to low variance at the expense of increased bias. So, the upper bound of h is limited, which is$\left(\frac{270}{70\sqrt{\pi}}\right)^{\frac{1}{5}} \times \boldsymbol{s}.$

   By observing Eqn. 1.6 and 1.7, we find that GMM may treated as the generalized form of parzen window method. GMM is variation parameter estimation model; and parzen window method is a fixation parameter estimation model. So, we let this upper bound of h as the upper bound as the same time, that is to say;

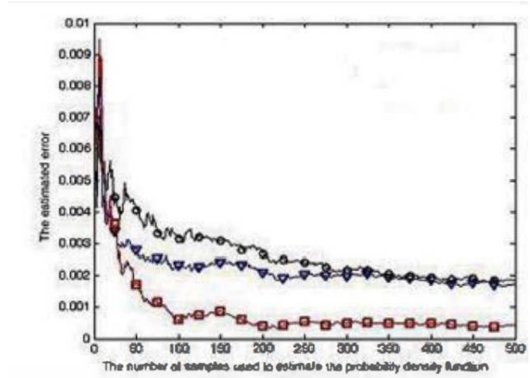
$$0 < \sigma_i \leq \left(\frac{\mathbf{270}}{\mathbf{70}\sqrt{\boldsymbol{\pi}}}\right)^{\frac{1}{5}} \times \boldsymbol{s}, (\boldsymbol{i} = \mathbf{1}, \mathbf{2}, \ldots \ldots, \boldsymbol{N})$$

We initialize the population of $\sigma_i, (i = 1, 2, \ldots \ldots, N)$ with the random numbers which obey the uniform density distribution in interval $[\mathbf{0}, (\frac{\mathbf{270}}{\mathbf{70}\sqrt{\boldsymbol{\pi}}})^{\mathbf{1/5}} \times \boldsymbol{s}]$.

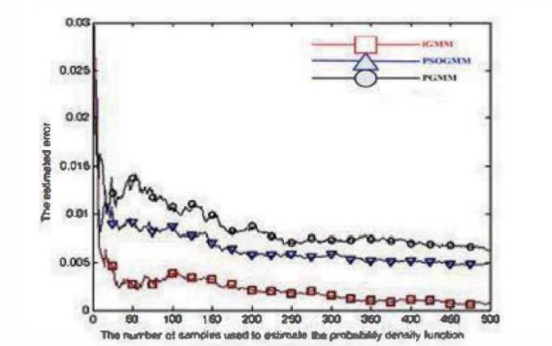
2. Second, we initialize the population of $\omega_i, (i = 1,2, \ldots, N)$with the random number which obey the uniform density distribution in interval [0, 1] and guarantee that $\sum_{i=1}^{N} \omega_i = 1$. our experiment is carried out according to the algorithm ……:

For each probability density, we use $GPSO_{GJ}$ algorithm to search the optimal parameters based on four different types of univariate artificial dataset. Every type of dataset is generated 100 times randomly. The averaged result based on these 100 datasets is summarized for some distributions. The mean square error (MSE) is used to evaluate the estimation performance.

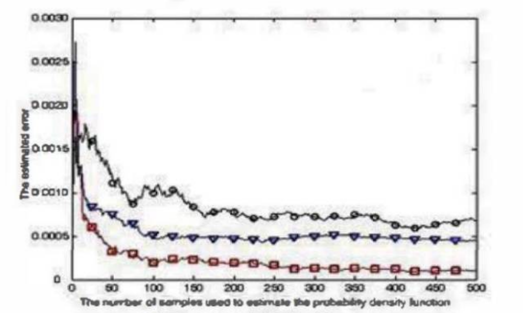
COMPARATIVE RESULTS GRAPH OF MSE IN PDF



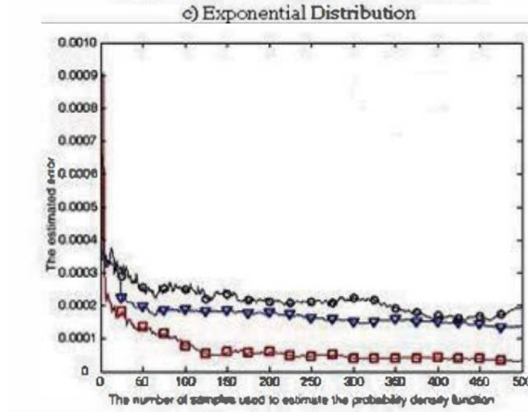
b) Normal Distribution



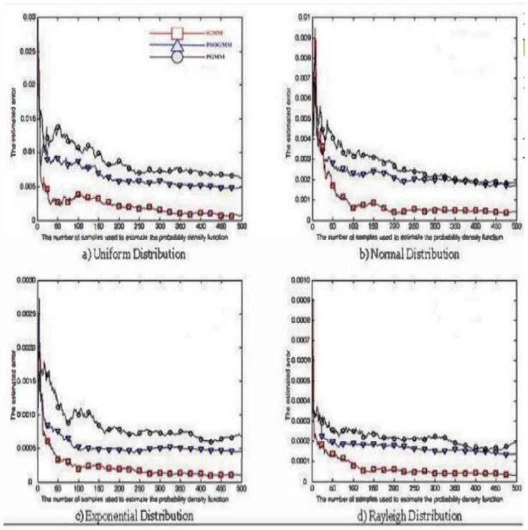
a) Uniform Distribution



c) Exponential Distribution



d) Rayleigh Distribution



The detailed comparative results are listed in fig… From the experimental results, we can get the following three observations,

1. With increase of iteration the MSE (Mean Squared Error) decrease gradually, and when the optimal parameters are searched, MSE becomes steadily.
2. The estimation performances of PGMM are worst among all the competitive density estimation algorithms, where the estimation errors of PGMM are higher than the other two.
3. IGMM obtains the best estimation performance, since it can find the stable and robust parameters for the probability density estimation application.

CONCLUSION

This paper has presented, an improved Gaussian mixture model (IGMM) based on least-squares cross-validation (LSCV) and Gaussian PSO with Gaussian jump (GPSOGJ) is developed. In order to measure the estimated error between the true density function and the estimated density function, a new error measure criterion is derived based on the least-squares cross-validation. Then, GPSOGJ is used to find the optimal parameters that can make the estimation error reach the minimum. Finally, in the experiments, we compare the performance of IGMM with two existing methods, i.e., GMM with Parzen window (PGMM) and GMM based on particle swarm optimization (PSOGMM), on four probability distributions: Uniform density, Normal density, Exponential density, and Rayleigh density. The experimental results show that the proposed IGMM strategy is feasible and effective, which can achieve a better estimation performance by applying the optimized parameters.This work has also presented a detailed account on the state of the art in the field of Motion Detection through Computer Vision.

REFERENCES

[1]. Kalal Z, Mikolajczyk K, Matas, J., "Tracking-Learning-Detection", IEEE Transactions on Pattern Analysis and Machine Intelligence (Volume: 34, Issue: 7 Page(s):1409 – 1422) July 2012.

[2]. Berthold K. Horn ; Brian G. Schunck, "Determining Optical Flow", Proc. SPIE 028 Conference on Techniques and Applications of Image Understanding, Washington, April 21, 1981.

[3]. Antonio Fernández, Caballero A, José Carlos Castillo a, Javier Martínez-Cantos c,, "Optical flow or image subtraction in human detection from infrared camera on mobile robot"s, Robotics and Autonomous Systems , Pp 1273–1281, Science Direct (2010).

[4]. Jaesik Choi, "Realtime On-Road Vehicle Detection with Optical Flows and Haar-Like Feature Detectors", F. T. Luk, ed., SPIE-The International Society for Optical Engg. Proceedings 2563,SPIE, Washington, DC, pp. 314-325, 1995.

[5]. S. S. Beauchemin , J. L. Barron, "The computation of optical flow", Journal of ACM Computing Surveys (CSUR) Volume 27 Issue 3, , Pages 433-466, Sept. 1995.

[6]. O. Barnich and M. Van Droogenbroeck. Vi Be: A Universal Background Subtraction Algorithm for Video Sequences. IEEE Transactions on Image Processing, 20(6):1709–1724, June 2011.

[7]. Ahmed Elgammal, David Harwood, and Larry Davis. Non-parametric model for background subtraction. In FRAME-RATE WORKSHOP, IEEE, pages 751–767, 2000.

[8]. K Toyama, J Krumm, B Brumitt, and B Meyers. Wallflower: principles and practice of background maintenance. Proceedings of the Seventh IEEE International Conference on Computer Vision, 1(c):255–261, 1999.

[9]. L. Maddalena and A. Petrosino. A Self-Organizing Approach to Background Subtraction for Visual Surveillance Applications. IEEE Transactions on Image Processing, 17(7):1168–1177, 2008.

[10]. Ismail Haritaoglu, David Harwood, and Larry S. Davis. W4: Real-time surveillance of people and their activities. IEEE Transactions on Pattern Analysis

and Machine Intelligence, 22:809–830, 2000.

[11]. DA Migliore, M Matteucci, M Naccari – "A revaluation of frame difference in fast and robust motion detection" Proceedings of the 4th ACM,dl.acm.org.- 2006.

[12]. Chris Stauffer, W. Eric, and W. Eric L. Grimson. Learning patterns of activity using real-time Tracking. IEEE Transactions on Pattern Analysis and Machine Intelligence, 22:747–757, 2000.

[13]. Alan J. Lipton, Hironobu Fuji yoshi, and Raju S. Patil. Moving target classification and tracking from real-time video. In Proceedings of the 4th IEEE Workshop on Applications of Computer.

[14]. Douglas Reynolds ,Gaussian Mixture Models, MIT Lincoln Laboratory,1998.

[15]. Gray, R. Vector Quantization. IEEE ASSP Magazine pages: 4–29 (1984).

[16]. Reynolds, D.A. A Gaussian Mixture Modelling Approach to Text-Independent Speaker Identification. PhD thesis, Georgia Institute of Technology (1992).

[17]. Reynolds, D.A., Rose, R.C.: Robust Text-Independent Speaker Identification using Gaussian Mixture Speaker Models. IEEE Transactions on Acoustics, Speech, and Signal Processing 3(1)72–83 (1995).

[18]. C. Stauffer and W. E. L. Grimson, "Adaptive back ground mixture models for real-time tracking," in Proc. of IEEE Conference on Computer Vision and Pattern Recognition, vol. 2, pp. 246–252, 1999.

[19]. N. McFarlane and C. Schofield, "Segmentation and tracking of piglets in images," Machine Vision and Applications, 8(3), pp. 187-193, 1995.

[20]. I. Haritaoglu, D. Harwood, and L. Davis, W4, "Real-time surveillance of people and their activities," IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI), vol. 22, no. 8, 2000.

[21]. R.Collins, A.Lipton, and T.Kanade, "A system for video surveillance and monitoring," in Proceedings of American Nuclear Society (ANS) Eighth International Topical Meeting on Robotics and Remote Systems, pp. 25-29, April 1999.

[22]. A. Elgammal, R. Duraiswami, D. Harwood, and L. Davis, "Background and foreground modeling using nonparametric kernel density estimation for visual surveillance," in Proceedings of the IEEE, pp. 1151-1163, July 2002.

[23]. N. Friedman and S. Russell., Image segmentation in video sequences: A probabilistic approach," in Proceedings of Thirteenth Conference on Uncertainty in Artificial Intelligence, pp. 175-181, 1997.

[24]. F. Porikli and O. Tuzel, "Human body tracking by adaptive background models and mean-shift analysis," in IEEE International Workshop on Performance Evaluation of Tracking and Surveillance, March 2003.

[25]. C. Wren, A. Azar bayejani, "T. Darrell, and A. Pent land: Real-time tracking of the human body," IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI), vol. 19, pp. 780-785, July 1997.

[26]. K. Toyama, J. Krumm, B. Brumitt, and B. Meyers, "Principles and practice of background maintenance," in Proceedings of IEEE International Conference on Computer Vision (ICCV), pp. 255-261, 1999.

[27]. M. M. A. J. Manoj S. Nagmode, Ashok M. Sapkal, "A novel approach to detect and track moving object using Partitioning and Normalized Cross Correlation," ICGST-GVIP vol.9, 2009.

[28]. Robert T. Collins, Alan J. Lipton, Takeo Kanade, "A System for Video Surveillance and Monitoring," The Robotics Institute, Carnegie Mellon University, Pittsburgh PA, 2012.

[29]. P. KaewTraKul Pong, R. Bowden, "An Improved Adaptive Background Mixture Model for Real-time Tracking with Shadow Detection 2 Background Modelling," Online(Weston, Conn.), vol. 1, p. 1, 2001.

[30]. R. Cucchiara, C. Grana, M. Piccardi, and A. Prati, "Detecting moving objects, ghosts, and shadows in video streams," IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI), vol. 25, October 2003.

[31]. S. J. McKenna, S. Jabri, Z. Duric, and A. Rosenfeld, "Tracking groups of people," Computer Vision and Image Understanding (CVIU), no. 80, pp. 42-56, 2000.

[32]. Y. Yang and M. Levine, "The background primal sketch an approach for tracking moving objects," Machine Vision and Applications, vol. 5, pp. 17-34, 1992.

[33]. S.-C. S. Cheung and C. Kamath, "Robust techniques for background subtraction in urban tracking video," in Proceedings of SPIE, Visual Communications and Image Processing 2004, vol. 5308, pp. 881-892, January 2004.

[34]. P. Kaew TraKul Pong, R. Bowden, "An Improved Adaptive Background Mixture Model for Real-time Tracking with Shadow Detection 2 Background

Modelling," Online (Weston, Conn.), vol. 1, p. 1, 2001.

[35]. D. H. Thanarat Horprasert, Larry S. Davis, "A Statistical Approach for Real-time Robust Background Subtraction and Shadow Detection," presented at the IEEE ICCV'99FRAME-RATE WORKSHOP, 1999.