CLOSED-FORM CAUCHY-SCHWARZ PDF DIVERGENCE FOR MIXTURE OF GAUSSIANS

Kittipat Kampa, Erion Hasanbelliu and Jose C. Principe

***Abstract*— This paper presents an efficient approach to calculate the difference between two probability density functions (pdfs), each of which is a mixture of Gaussians (MoG). Unlike Kullback-Leibler divergence (D_{KL}), the authors propose that the Cauchy-Schwarz (CS) pdf divergence measure (D_{CS}) can give an analytic, closed-form expression for MoG. This property of the D_{CS} makes fast and efficient calculations possible, which is tremendously desired in real-world applications where the dimensionality of the data/features is very high. We show that D_{CS} follows similar trends to D_{KL}, but can be computed much faster, especially when the dimensionality is high. Moreover, the proposed method is shown to significantly outperform D_{KL} in classifying real-world 2D and 3D objects, and static hand posture recognition based on distances alone.**

I. INTRODUCTION

THE Gaussian mixture model has been a very useful probability model for a variety of applications due to the fact that the number of parameters used in a mixture of Gaussians (MoG) is very small and due to its flexibility to model distributions whose parametric forms are unknown. In many applications, one would like to compare two pdfs, each of which is a MoG, by measuring the difference between the two pdfs using various types of available divergences or distance measures. However, not all divergences are equally useful for the MoG model because most well known divergences, including the Kullback-Leibler divergence (D_{KL}), do not yield an analytic closed-form expression for MoG.

To work around this problem, a few approaches are used to estimate the D_{KL} in practice, such as numerical integration (NI) and stochastic integration (SI) [1], [2]. In NI, the whole feature space is uniformly gridded, then each gridded cell is used in the calculation of the D_{KL}. Thus, the accuracy will highly depend on the resolution of the grid. The smaller the grid size, the better accuracy obtained, but this comes with the cost of a larger memory size being used to store those cells. This is the trade-off between memory and accuracy when using NI. In addition, the size of memory used in NI grows exponentially with the dimensionality of the data/feature vector (the curse of dimensionality). Another severe drawback of NI is that it sometimes misses narrow peaks of MoG components [3].

Stochastic integration techniques have been proposed to mitigate this problem by sampling directly from the MoG of interest. This reduces the chance of missing narrow peaks. However, this approach also cannot avoid the dimensionality curse. When the dimensionality increases, the number of samples has to increase in order to keep up with the additional details of the MoG. In addition, there is no theoretical criterion to relate the number of samples to the dimensionality. Consequently, a closed-form expression for divergence of a MoG model is desired.

The rest of the paper looks into criteria to reach closed-form expressions and identifies Cauchy-Schwarz divergence as one of the measures that yields closed-form solutions for MoG. The results show that D_{CS} performs in a similar manner to D_{KL}, however D_{CS} is more efficient. This could lead to real-time application of this measure in machine learning on mobile devices.

The organization of this paper is as follows. In section II, the authors investigate why D_{KL} does not yield a closed-form expression for MoG. In section III, the authors show that the Cauchy-Schwarz (CS) pdf divergence measure yields an analytic closed-form expression for MoG. Numerical examples are shown in section IV. In section V, the proposed expression is tested in real-world 2D and 3D object classifications. Finally, in section VI, the proposed method is applied to the real-world hand posture recognition problem where classification accuracy and the run-time are reported.

II. CLOSED-FORM EXPRESSION OF D_{KL} FOR MOG?

In this section, we demonstrate that an analytic closed-form expression of D_{KL} for MoG is not possible. Let $q(x)$ and $p(x)$ denote two distributions each of which is a mixture of Gaussians with different parameters and number of clusters:

$$q(x) = \sum_{m=1}^{M} \pi_m \mathcal{N}(x|\mu_m, \Lambda_m^{-1})$$

and

$$p(x) = \sum_{k=1}^{K} \tau_k \mathcal{N}(x|\nu_k, \Omega_k^{-1})$$

where M and K denote the number of Gaussian components in $q(x)$ and $p(x)$ respectively. Let π_i , μ_i and Λ_i denote the mixture coefficient, the mean, and the precision matrix of the i^{th} component of $q(x)$, and τ_k, ν_k, and Ω_k denote the respective terms of the k^{th} component of $p(x)$. The multivariate Gaussian distribution is given by

$$\mathcal{N}(x|\mu_i, \Lambda_i^{-1}) = \frac{|\Lambda_i|^{1/2}}{(2\pi)^{D/2}} \exp\left(-\frac{1}{2}(x-\mu_i)^\top \Lambda_i (x-\mu_i)\right)$$

where $x \in R^D$.

Kittipat Kampa, Erion Hasanbelliu and Jose C. Principe are with the Department of Electrical and Computer Engineering, University of Florida, Gainesville, FL 32611, USA (email: kittipat@gmail.com, erioni@cnel.ufl.edu, principe@cnel.ufl.edu).

This work is partially funded by ONR N00014-10-1-0375.

In the Kullback-Leibler divergence,

$$\begin{aligned} D_{KL}(q||p) &= \int q(x)\log\frac{q(x)}{p(x)}dx \\ &= \int q(x)\log\left[\sum_{m=1}^{M}\pi_m\mathcal{N}(x|\mu_m,\Lambda_m^{-1})\right]dx \\ &\quad -\int q(x)\log\left[\sum_{k=1}^{K}\tau_k\mathcal{N}(x|\nu_k,\Omega_k^{-1})\right]dx. \end{aligned}$$

the integral and the summations cannot be interchanged due to the logarithm operator. The functional form of KL-divergence contains the log operator inside the integral. In addition, an MoG, as shown above, is the weighted summation of Gaussians (terms as expanded in 2^{nd} and 3^{rd} rows above). Since, the integral and summation are separated by the log, x cannot be marginalized out by the integral. This prevents the solution from being an analytic closed-form. (Note that as long as x appears in the final expression, it will not result in an analytic closed-form solution).

The authors also investigated other divergence measures which have the integral inside the log in the hope that the integral would be distributed inside the summation and, thus, x will be marginalized out. The α-divergence [4] is one well-known measure, however, it failed to give a closed form expression despite having the integral inside the log. That is because there is no clear insight into how to calculate an x-independent closed-form expression from the inverse of a MoG generated by a minus-sign power $(1-\alpha)$ in the α-divergence.

From the failures learned from these two divergences and based on Gaussian identities, search domain for a distance/divergence measure can be narrowed significantly. The authors found that a closed-form expression for MoG can be derived from some divergences, such as the Cauchy-Schwarz pdf divergence measure (D_{CS}) [5], Jensen-Renyi divergence (D_{JR}) [6], and corresponding concordance [7]. In this paper, we restrict our discussion to D_{CS} as its behavior is similar to D_{KL} [8], and it is simple to implement and understand due to its relation to the Cauchy-Schwarz inequality.

III. Closed-form expression for D_{CS}

Inspired by the renowed Cauchy-Schwarz inequality, the Cauchy-Schwarz PDF divergence measure [5] is given by:

$$D_{CS}(q,p) = -\log\left(\frac{\int q(x)p(x)dx}{\sqrt{\int q(x)^2dx\int p(x)^2dx}}\right). \quad (1)$$

This is a symmetric measure for any two pdfs q and p, such that $0 \le D_{CS} < \infty$, where the minimum is obtained if and only if $q(x) = p(x)$. The measure plays important roles in information theoretic learning (ITL), non-parametric density estimation [9], graph theory, Mercer kernel theorey and spectral theory [5]. Before we present its derivation, it is important to understand why D_{CS} is considered in the MoG case. First, it is obvious that the integral can be distributed into the weighted summation of Gaussian components because these terms appear inside the log operator. Moreover, we know that the integral of the product of two MoGs is a MoG in the space of the mean parameters μ.

The closed-form expression for D_{CS} of a pair of MoGs can be derived by rewritting (1) as

$$\begin{aligned} D_{CS}(q,p) = &-\log\left(\int q(x)p(x)dx\right) \\ &+\frac{1}{2}\log\left(\int q(x)^2dx\right)+\frac{1}{2}\log\left(\int p(x)^2dx\right). \end{aligned} \quad (2)$$

By distributing the integral into the summation, and using the Gaussian identity,

$$\mathcal{N}(x|\mu_1,\Lambda_1^{-1})\mathcal{N}(x|\mu_2,\Lambda_2^{-1}) = z_{12}\mathcal{N}(x|\mu_{12},\Lambda_{12}^{-1})$$

where $\Lambda_{12} = \Lambda_1+\Lambda_2$, $\mu_{12} = \Lambda_{12}^{-1}(\Lambda_1\mu_1+\Lambda_2\mu_2)$ and $z_{12} = \mathcal{N}(\mu_1|\mu_2,(\Lambda_1^{-1}+\Lambda_2^{-1}))$, the first term on the r.h.s. of (2), $\log\left(\int q(x)p(x)dx\right)$, can be written in a closed-form expression independent of x:

$$\begin{aligned} &\log\left(\int\sum_{m=1}^{M}\sum_{k=1}^{K}\pi_m\tau_k\mathcal{N}(x|\mu_m,\Lambda_m^{-1})\mathcal{N}(x|\nu_k,\Omega_k^{-1})dx\right) \\ &= \log\left(\sum_{m=1}^{M}\sum_{k=1}^{K}\pi_m\tau_k\int\mathcal{N}(x|\mu_m,\Lambda_m^{-1})\mathcal{N}(x|\nu_k,\Omega_k^{-1})dx\right) \\ &= \log\left(\sum_{m=1}^{M}\sum_{k=1}^{K}\pi_m\tau_k z_{mk}\right). \end{aligned}$$

Applying the same trick to the second and third terms, the closed-form expression is given by:

$$\begin{aligned} &D_{CS}(q,p) = \\ &-\log\left(\sum_{m=1}^{M}\sum_{k=1}^{K}\pi_m\tau_k z_{mk}\right) \\ &+\frac{1}{2}\log\left(\sum_{m=1}^{M}\frac{\pi_m^2|\Lambda_m|^{1/2}}{(2\pi)^{D/2}}+2\sum_{m=1}^{M}\sum_{m'<m}\pi_m\pi_{m'}z_{mm'}\right) \\ &+\frac{1}{2}\log\left(\sum_{k=1}^{K}\frac{\tau_k^2|\Omega_k|^{1/2}}{(2\pi)^{D/2}}+2\sum_{k=1}^{K}\sum_{k'<k}\tau_k\tau_{k'}z_{kk'}\right) \end{aligned} \quad (3)$$

where

$$\begin{aligned} z_{mk} &= \mathcal{N}(\mu_m|\nu_k,(\Lambda_m^{-1}+\Omega_k^{-1})) \\ z_{mm'} &= \mathcal{N}(\mu_m|\mu_{m'},(\Lambda_m^{-1}+\Lambda_{m'}^{-1})) \\ z_{kk'} &= \mathcal{N}(\nu_k|\nu_{k'},(\Omega_k^{-1}+\Omega_{k'}^{-1})) \end{aligned}$$

are the integrals of product of two corresponding Gaussian pdfs. The expression has a complexity of order $O(M^2)$ when $M \ge K$, which is much smaller than that of NI and SI whose complexities depend on the dimensionality of the data D and the number of samples (in general, $N \gg M^2$) respectively.

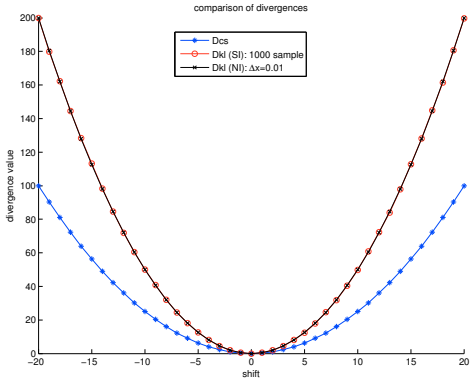


Fig. 1. Comparison of divergences evaluated from D_{CS}, $D_{KL}^{(NI)}$ and $D_{KL}^{(SI)}$ in numerical experiment.

TABLE I
AVERAGE RUN-TIME (SECONDS) IN NUMERICAL EXPERIMENT

dimensionality (D)	D_{CS}	$D_{KL}^{(NI)}$	$D_{KL}^{(SI)}$
2	0.002	26	4.3
3	0.002	37.2	4.3

IV. NUMERICAL EXAMPLES

In this experiment, the behavior of D_{CS} and that of D_{KL} are compared in terms of their values in several circumstances and their run-time when the feature vectors are all 2D. The parameters for $q(x) = q([x_1\, x_2]^\top)$, a mixture of 3 Gaussian pdfs, are given by

$$(\pi_1, \mu_1, \Sigma_1) = (0.3, [0\ 0]^\top, I)$$
$$(\pi_2, \mu_2, \Sigma_2) = (0.3, [3\ 0]^\top, I)$$
$$(\pi_3, \mu_3, \Sigma_3) = (0.4, [8\ 0]^\top, I)$$

For illustrative purposes, the distribution $p(x)$ is picked to be an x_2-shifted version of $q(x)$, that is $p(x) = q([x_1\, x_2 + \Delta x_2]^\top)$ where the shifts are $\Delta x_2 \in \{-20, -19, ..., 19, 20\}$. In this experiment, we calculate divergences using 3 approaches: 1) D_{CS}, 2) KL-divergence using NI ($D_{KL}^{(NI)}$) and 3) KL-divergence using SI ($D_{KL}^{(SI)}$). The D_{CS} is calculated by (3). The $D_{KL}^{(NI)}$ is evaluated on the region of interest (x_1, x_2): $x_1 = [-15, 23]$ and $x_2 = [-15, 15]$ with equal resolution of 0.01 on both the x_1 and x_2 axis. The $D_{KL}^{(SI)}$ is evaluated by sampling (as in Monte Carlo method) from the distributions $q(x)$ using number of sample $N = 1000$ samples. The results and run-time are shown in Fig. 1 and Table I respectively.

The results in Fig. 1 show that the divergence $D_{CS} = 0$ when $q(x)$ and $p(x)$ are the same distribution, and the value of D_{CS} increasing when the distributions move away from each other. In addition, the curves depict the similarity in behavior between D_{CS} and D_{KL} in terms of performance, where both reach the minimum when the distributions are identical and as the difference between the distributions increases so does the divergence value.

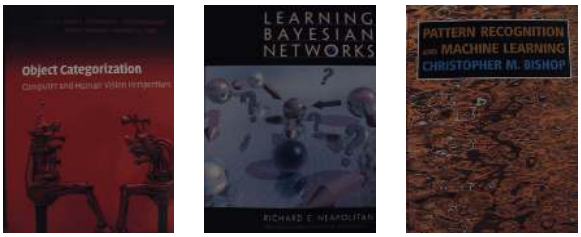
The average run-time is shown in Table I. The table illustrates the run-time performance for the experiment in 2D and 3D vector space. We found that when the dimensionality of the data increases, the run-time of $D_{KL}^{(NI)}$ increases significantly, whereas that of D_{CS} remains almost the same. In the table, you will notice that the run-time of $D_{KL}^{(SI)}$ also remained the same, however this is not the case when dimensionality increases significantly. In the next section, the divergences are compared on real-world 2D and 3D object classification.

V. OBJECT CLASSIFICATION

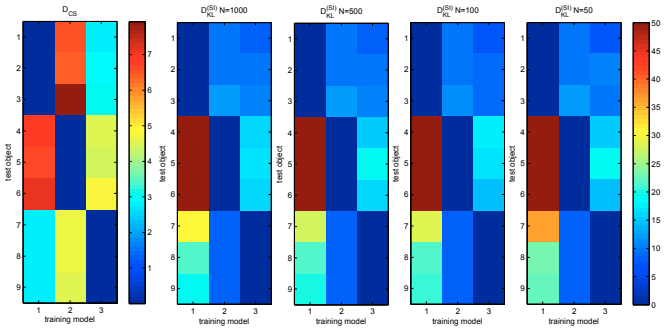
Object classification has been an active area of research for years. Nowadays, the area of object classification is more interesting and challenging because richer and more informative feature vectors can be obtained by novel sensor technology, which in turn makes it possible to classify objects of more complicated shapes in many real-world applications. Several approaches have been proposed. Pdf-based classification [8], [10], [11] is among the well-known methods, as the pdf of features can be viewed as a very informative descriptor of an object [8]. In this paper, we take advantage of having extracted features as MoGs to exploit pdf-distance-based classification, i.e. the model/class whose divergence to the test sample is smallest will be assigned to the sample. Let $\mathcal{C}$ denote the set of models, $c_i \in \mathcal{C}$ denote the i^{th} model, s_j denote the j^{th} (test) sample, $q(x|s_j)$ and $p(x|c_i)$ denote the pdf of extracted features for s_j and c_i respectively. Then the class $c^\star$ will be assigned to the sample s_j if $c^\star = \arg\min_{c_i \in \mathcal{C}} D(q(x|s_j)||p(x|c_i))$. In this section, we present the performance of the proposed algorithms in 2D and 3D object recognition.

A. Experiment1: Object classification in 2D

In this experiment, we have 3 types/classes of images (front cover of 3 books), A, B and C as shown in Fig. 2 (a). The features are extracted from each image by converting the RGB value of each pixel to CIELuv [12], which is a very powerful feature in image recognition because the Euclidean distance between two sets of color coordinates approximates the human perception of color difference. Therefore, each image is modeled by a 3D MoG with 2 Gaussian components decided using BIC. The probability model of each type is built using 4 sample images of the same book with different scales and orientations. The performance of D_{CS} is tested against that of $D_{KL}^{(SI)}$ (with varying number of samples N = 1000, 500, 100, 50) on the test dataset, which comprise 3 types of front covers, each of which having 3 sample images resulting in 9 test images total. The objective is to classify the front cover of the 3 books based on the pdf distance mentioned earlier. The divergences evaluated in each case and their computational times are shown in Fig. 2 (b) and Table II respectively. The results will be discussed at the end of this section.



(a) Sample of type A, B and C from left to right respectively



(b) (left-most) The results from using D_{CS}. Next are $D_{KL}^{(SI)}$ using number of sample N= 1000, 500, 100, 50 respectively.

Fig. 2. Data samples and results from Experiment1.

TABLE II
RUN-TIME OF EXPERIMENT1

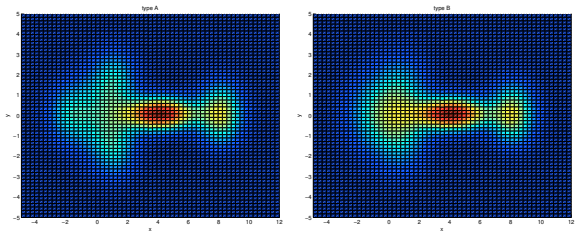
	D_{CS}	$D_{KL}^{(SI)}$			
number of sample N	-	1000	500	100	50
runtime (sec)	0.006	3.7	1.9	0.37	0.19

B. Experiment2: Object classification in 3D

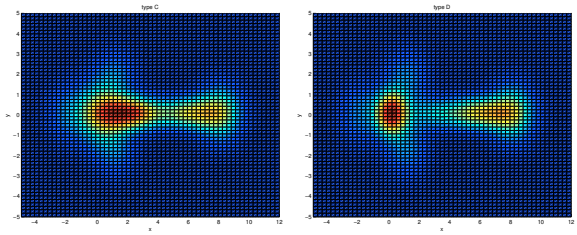
In this experiment, we have 4 types/classes of synthesized 3D objects A, B, C and D whose 2D footprints are shown in Fig. 3 (a), (b), (c) and (d), each of which is modeled by 3D-MoG with 6, 6, 7 and 7 Gaussian components respectively. The inputs of this experiment are MoG parameters learned from the object point cloud whose footprints are shown in Fig. 4 (a), (b), (c) and (d). We test the performance of D_{CS} against $D_{KL}^{(SI)}$ (with varying number of samples N = 1000, 500, 200, 100) on 3 datasets; Dataset_ns1, ns2 and ns3 where the number represents the noise level from low to high. Each dataset contains 4 object types and each type has 10 samples, resulting in 40 samples total in each dataset. Our goal is to classify/label each sample based on minimum divergence criteria mentioned earlier. The performances of D_{CS} and $D_{KL}^{(SI)}$ are shown in the total accuracy matrix in Table III.

C. Discussion of object classification

In Experiment 1, by the nature of the dataset the complexity of distributions is not very high, so it can be fitted appropriately using an MoG with 2 components. In Fig. 2 (b), the y-axis represents the 9 test images. The x-axis represents classes A, B and C from left to right respectively. The test

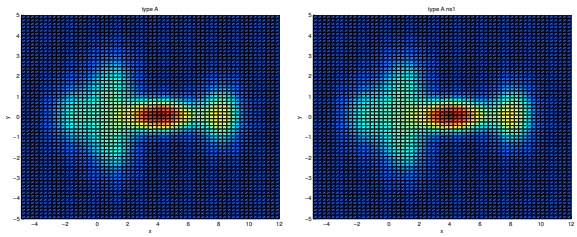


(a) Type A (b) Type B

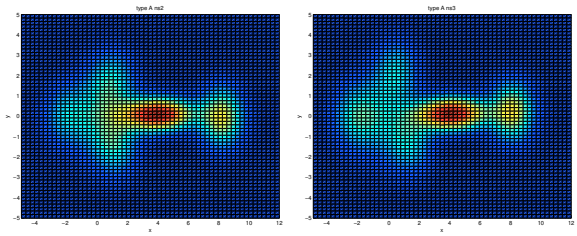


(c) Type C (d) Type D

Fig. 3. 2D footprints of objects in Experiment2. (a)-(d) illustrate ideal footprint of object of Type A-D respectively.



(a) Type A (b) Type A, noise level 1



(c) Type A, noise level 2 (d) Type A, noise level 3

Fig. 4. Random samples of noisy footprint of object of Type A in Experiment2 from level 1 (small) to level 3 (big).

images 1-3, 4-6 and 7-9 are from types A, B and C respectively. The resulting figure shows that for each image in 1-3, 4-6, 7-9 the minimum divergence is reached on the correct type A, B and C respectively. That yields 100% classification accuracy for all the approaches. However, looking further at the divergence results (the color codes), we notice that the divergence values (the color) of the correct type compared to the incorrect types for D_{CS} are more distinct in the color spectrum compared to the values of $D_{KL}^{(SI)}$. This is easily noticed on test images 1-3 where the results of the incorrect types B and C for D_{CS} are very far in the spectrum compared to those of type A, however the results for $D_{KL}^{(SI)}$ are very close. In addition, D_{CS} tends to be more time-efficient than

TABLE III

TOTAL PERCENT ACCURACY MATRIX OF EXPERIMENT2

	noise level			
	ns1	ns2	ns3	run-time (sec)
D_{CS}	100%	100%	82.5%	0.037
$D_{KL}^{(SI)}$, $N = 1000$	100%	87.5%	65%	5.82
$D_{KL}^{(SI)}$, $N = 500$	100%	83.75%	60%	2.92
$D_{KL}^{(SI)}$, $N = 200$	98.33%	80.83%	55%	1.15
$D_{KL}^{(SI)}$, $N = 100$	96.87%	78.12%	56.88%	0.58

$D_{KL}^{(SI)}$ as shown in the run-times in Table II.

In Experiment 2, the results indicate excellent performance of both approaches in the low noise environment (ns1) when the sample number is large enough. In the intermediate noise level (ns2), the D_{CS} still performs flawlessly, however $D_{KL}^{(SI)}$ performance decreases significantly. This degradation continues as the number of samples decreases. When the noise level is high (ns3), the performance of D_{CS} also drops. However, the $D_{KL}^{(SI)}$ performance is considerably worse. In all the cases, D_{CS} outperforms D_{KL}, and its run time is a fraction of that of D_{KL} regardless of the number of samples used.

The results of Experiment 1 and 2 show that the closed-form expression D_{CS} outperforms and computationally outruns the solution of D_{KL} significantly in both applications. That is because the number of parameters used by an MoG is only $M(\frac{D}{2}+1)(D+1)$, which is much less than the number of samples N needed in order to maintain a good estimator of distribution. Furthermore, when the number of dimensions D becomes very high, the sample size N in D_{KL} will grow exponentially with the dimension. The results also illustrate similar behavior of D_{CS} and $D_{KL}^{(SI)}$ which implies that replacing $D_{KL}^{(SI)}$ with D_{CS} is possible in many applications especially when the input is given in terms of MoG, or when fast computation is required, or when there are limitations in computational resources and power consumption which usually happens when working in modern mobile or hand-held devices.

VI. APPLICATION: STATIC HAND POSTURE RECOGNITION

Hand gesture recognition has been an active area of research for years. There are mainly two types of hand gestures: 1) dynamic hand gesture which involves the hand movement, hence the recognition of this type must be done on sequence of images or video, and 2) static hand posture recognition which does not involve hand movement, hence can be done using still images. Our approach is applied on the latter type. In this section, the minimum divergence measure classifier is applied to the static hand posture recognition. Additionally, we compare the performance and the runtime of our proposed method against KL-divergence using stochastic integration with various numbers of samples N_S.

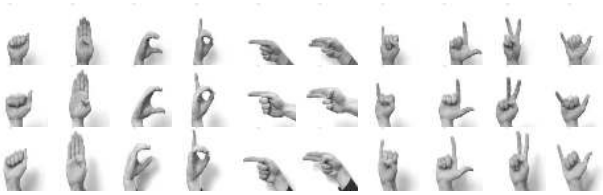


Fig. 5. Sample images of 3 people in the database.

A. Static hand posture recognition database

In this experiment, we use Jochen Triesch Static Hand Posture Database [13], which is available on the Internet[1]. The database is composed of gray-scale 128×128 images taken from 10 hand postures forming the alphabetic letters: (A, B, C, D, G, H, I, L, V, and Y) for 24 persons on 3 different backgrounds: (light, dark, complex). Details can be found from the paper [13]. In our experiments, we use images with light background only because we want to restrict our attention to the hand posture recognition alone without focusing on the background. There are 16-20 images for each hand posture (168 images total) remaining after visually screening and applying background-removing algorithms to the images in the original database. A sample of hand posture images for each letter is shown in Fig. 5.

B. Preprocessing and feature extraction

Due to the positional inconsistences of hand appearance on images in this database, the rectangular bounding box of hand is created in each image, and all the pixels outside the box are removed. In addition, the lower part of wrist is removed because of its size inconsistence. Each bounding box is then standardized such that the most top-left pixel and the most bottom-right pixel are coordinates (0,0) and (1,1) respectively which enables the algorithm to be more invariant to the size of the hand.

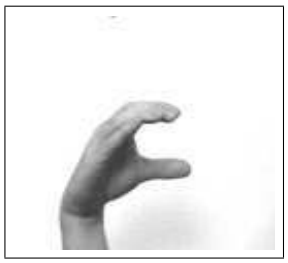
At this point the feature vector of each image pixel site $s \in S$ is encoded by $[x(s), y(s), i(s)]^\top$ where S denotes a set of image site indices; the $x(s)$ and $y(s)$ denote standardized x-y location respectively at the pixel site s; and $i(s)$ denotes intensity at the pixel site s. In fact, it is interesting to view the intensity as an arbitrary nonnegative function of x-y location, so we write $i([x, y])$ instead $i(s)$. The function $i([x, y])$ can be approximated by a Gaussian mixture:

$$i([x, y]) = \alpha q([x, y])$$

$$q([x, y]) = \sum_{c=1}^{C} \pi_c \mathcal{N}([x, y]; \mu_c, \Lambda_c^{-1})$$

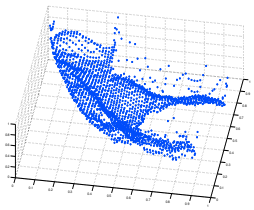
where α denotes the scale constant converting an MoG $q([x, y])$ to the arbitrary function $i([x, y])$, and other parameters are the same as ones in previous sections. This requires a special type of EM algorithm similar to that proposed in

[1]http://www.idiap.ch/resource/gestures/

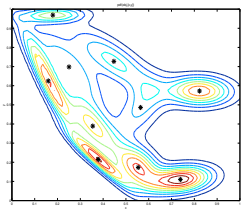


(a) A sample image of hand posture 'C'

(b) Preprocessed image of hand posture 'C'



(c) The intensity function



(d) A mixture of 10 Gaussian components

Fig. 6. (a) A sample image of hand posture 'C'. (b) Processed image of hand posture 'C'. (c) The intensity function $i([x, y])$ after preprocessing. (d) Approximate $i([x, y])$ with a mixture of 10 Gaussian components $q([x, y])$.

[14]. In this experiment, we pick the number of Gaussian components $C = 10$ so as to collect significant details of the intensity function. After the EM algorithm converges, we discard the constant α, but keep the MoG $q([x, y])$ of its corresponding image. Consequently, we will have a Gaussian mixture of 10 components for each hand posture image in our database as shown in Fig. 6.

C. Similarity measure

We first compare the performance of our proposed method D_{CS} with D_{KL} calculated by stochastic integration using 100 samples to create the similarity matrix of the database. Ten sample images are randomly picked from each posture, 100 images total, resulting in the similarity matrix of size 100×100. The (i, j) element of the similarity matrix is calculated using the divergence measure between the MoG $q_i([x, y])$ and $q_j([x, y])$ from image i and image j respectively. The values of members in both similarity matrices are normalized so that the maximum and minimum take the value of 1 and 0 respectively. The results are shown in Fig. 7.

From the results, shown in Fig. 7, both divergence measures behave as expected. The similarity within the same posture images is higher than similarity between different posture samples as shown from the low divergence values gathered around the diagonal elements (same-class boxes). But, we can also spot some different posture pairs whose divergence measures are also low, for instance A and I, or G and H, whose shapes look very similar. Nevertheless, when comparing the performance of D_{CS} against D_{KL}, the similarity matrices visually depict that D_{CS} performs better than D_{KL} on distinguishing different postures. When looking at both similarity matrices, we can visually notice that similarity values of positions from different classes for D_{CS} are distinguishably larger than those of same-class positions compared to D_{KL}.

D. Hand posture recognition

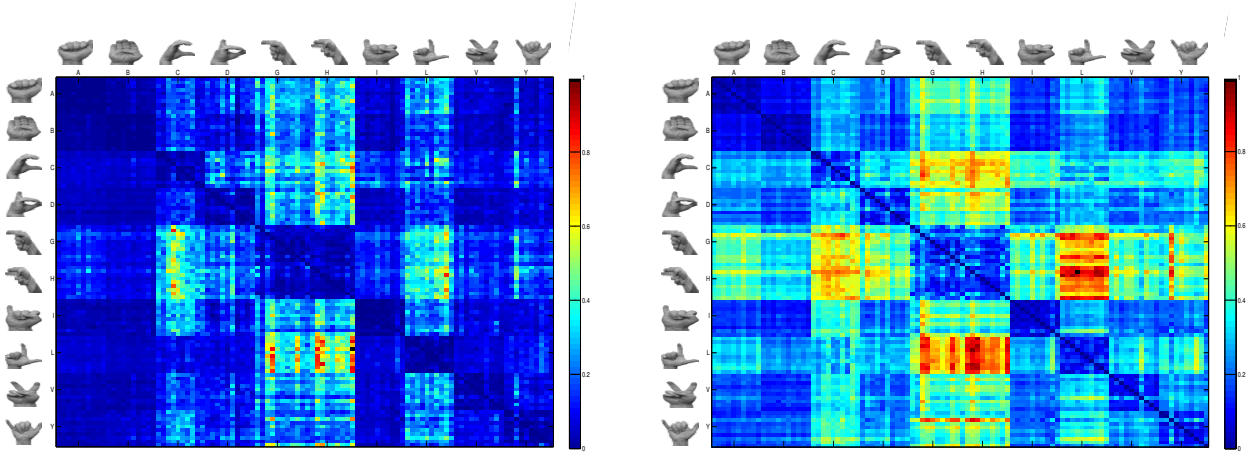
In this experiment, we exploit the D_{CS} capability to recognize static hand posture on the aforementioned database and compare the results to those of using D_{KL} with various number of samples N_S = 10, 25, 50, 75, 100, 125, 150, 175, 200, 225 and 250. Minimum divergence measure classifier and 5-fold cross-validation are used to evaluate the performances of D_{CS} and the different settings of D_{KL}. The classification accuracy of this experiment is represented by the confusion matrices shown in Fig. 8, and the summary of the accuracy and computational run-time are illustrated in Fig. 9. The classification accuracy of D_{KL} increases as the number of samples used increases and gives its best at 92.3% accuracy with $N_S = 175$ and averaged run-time 0.45 sec per run. Nevertheless, that is still outperformed by D_{CS} whose classification accuracy is 95.2% with averaged run-time 0.03 sec per run. Unlike D_{KL}, the accuracy of D_{CS} does not depend on the number of samples, but instead depends on how well the MoG can approximate the arbitrary function, and note that the run-time of D_{KL} depends on only the number of the Gaussian components in the MoG. (This experiment was performed using MATLAB r2010a on an Intel(R) Core(TM)2 Duo CPU E8400 @ 3.00GHz machine with Ubuntu operating system.)

E. Discussion on hand posture recognition

Both D_{CS} and D_{KL} are applied to the real-world problem of hand posture recognition. In this experiment, we assume that divergence calculated from two similar postures is greater than that obtained from two different postures, therefore we use minimum divergence measure based classifier for this application despite of the fact that there are many possibilities for alternative classifier. Even though we know from the previous section that D_{CS} and D_{KL} share similar behaviors, D_{CS} outperforms D_{KL} in this application in similarity visualization, classification accuracy and computational run-time.

Regarding similarity visualization, both D_{CS} and D_{KL} perform well on grouping same postures together, but D_{CS} seems to outperform D_{KL} when discriminating different groups of postures. The similarity matrices are normalized so that its members are ranged from 0 to 1 before visually comparison. Due to the obvious difference between the two matrices, it is sufficient in this experiment to evaluate the performance of two divergence measures using eyeball estimation instead of resorting on entropy-based measurement. In addition, unlike D_{KL}, D_{CS} offers symmetry measure which is more intuitive when comparing 2 postures in general.

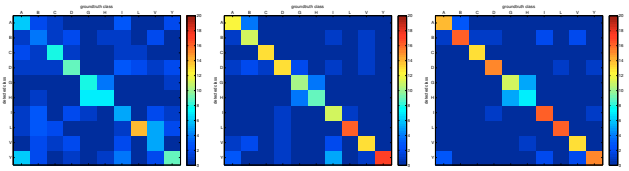
In hand posture recognition, D_{CS} outperforms D_{KL} in both classification accuracy and computational run-time. Since D_{KL} does not provide closed-form expression for



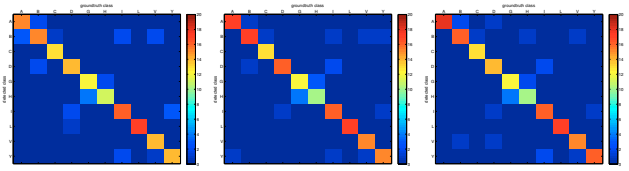
(a) Similarity matrix calculated by D_{KL}

(b) Similarity matrix calculated by D_{CS}

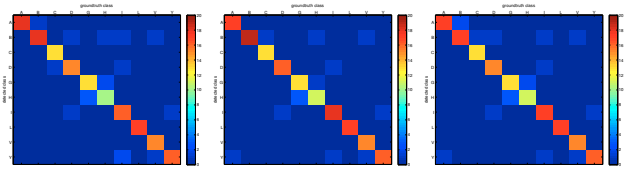
Fig. 7. Similarity matrices of D_{KL} and D_{CS} on 100 hand posture images from 10 hand postures (10 samples from each posture).



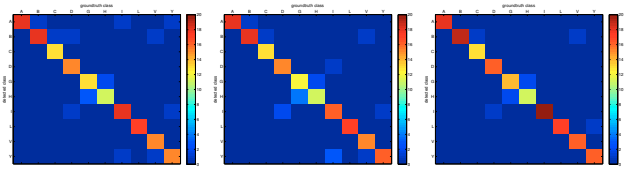
(a) D_{KL}, $N_S = 10$ (b) D_{KL}, $N_S = 25$ (c) D_{KL}, $N_S = 50$



(d) D_{KL}, $N_S = 75$ (e) D_{KL}, $N_S = 100$ (f) D_{KL}, $N_S = 125$



(g) D_{KL}, $N_S = 150$ (h) D_{KL}, $N_S = 175$ (i) D_{KL}, $N_S = 200$



(j) D_{KL}, $N_S = 225$ (k) D_{KL}, $N_S = 250$ (l) D_{CS}

Fig. 8. The confusion matrices. Detected class and groundtruth class are corresponding to row and column respectively. Each confusion matrix is plotted with the same color scale with 20 at maximum and the confusion matrix summed to 168, the number of total samples in the database.

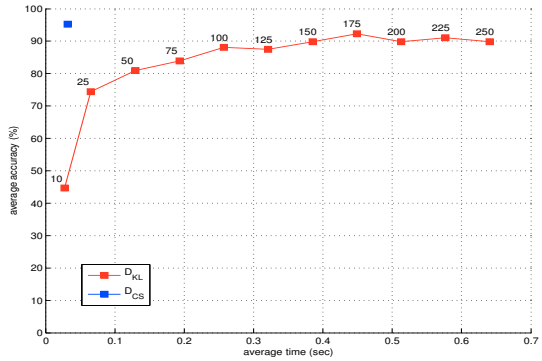


Fig. 9. The percentage of classification accuracy vs the averaged run-time per run (sec). The number over each read square represents the number of samples N_S. The D_{KL} curve is below that of D_{CS}, which indicates that the latter outperforms the former even with big number of N_S.

MoG, stochastic integration is a good way to calculate the quantity. The more samples used in the process, the more accurate the estimated divergence measure is. However, when comparing the run-time of the two divergence measures for the same accuracy, D_{CS} is significantly more computationally efficient than D_{KL}. We did not use numerical integration to calculate D_{KL} in this application because the method requires a relatively great amount of resources which is less efficient than stochastic integration approach for large databases.

This experiment emphasizes that the divergence of two MoGs can be computed more efficiently using D_{CS}. To benefit this approach even further, more efficient ways of extracting MoG from the input data/feature is required. In some cases, producing MoG can be a bottleneck to the entire process. Therefore finding the right application and setting model parameters (e.g. number of Gaussian components) to apply this approach is still an open problem.

VII. Conclusion and Future Work

In this paper, we illustrate why D_{KL} and α-divergence do not give a closed-form expression for MoG. From this observation, we come up with some preliminary criteria to

search for divergences that provide closed-form expressions. We then restrict our attention to the Cauchy-Schwarz pdf divergence measure. Using the Gaussian multiplication identity, we come up with a closed-form expression for D_{CS} which does not depend on x. This reduces the complexity to $O(M^2)$, which is much smaller than that of NI and SI whose complexities depend on the number of samples $N \gg M^2$ in general.

We also show that D_{CS} outperforms and outruns $D_{KL}^{(SI)}$ significantly in real-world object classification in both 2D and 3D, and also in the static hand posture recognition problem. Additionally, similar trends between both approaches suggest the possibility to replace D_{KL} with D_{CS} in many real-world applications where the form of input is appropriate. In future work, we will further pursue more general criteria to pinpoint such divergences in the hope that the criteria will lead to the way to construct divergences that give a closed-form expression for MoG.

REFERENCES

[1] Jesper Hjvang, Jensen Daniel, P. W. Ellis, Mads G. Christensen, and Sren Holdt Jensen, "Evaluation of distance measures between gaussian mixture models of mfccs," 2007.

[2] Elias Pampalk, "Speeding up music similarity," in *in Proceedings of the MIREX Annual Music Information Retrieval eXchange*, 2005.

[3] David Mackay, *Information Theory, Inference, and Learning Algorithms*, Cambridge University Press, 2005.

[4] Alfred O. Hero, Bing Ma, Olivier Michel, and John Gorman, "Alpha-divergence for classification, indexing and retrieval," Tech. Rep., Communication and Signal Processing Laboratory, Technical Report CSPL-328, U. of Mich., 2001.

[5] R. Jenssen, D. Erdogmus, K. Hild, J. Principe, and T. Eltoft, "Optimizing the Cauchy-Schwarz PDF distance for information theoretic, non-parametric clustering," in *Energy Minimization Methods in Computer Vision and Pattern Recognition*. Springer, 2005, pp. 34–45.

[6] A. Ben Hamza and H. Krim, "Jensen-renyi divergence measure: theoretical and computational perspectives," jun. 2003, pp. 257 – 257.

[7] Surajit Ray, *Distance-based Model-Selection with application to Analysis of Gene Expression Data*, Ph.D. thesis, 2003.

[8] Jose C. Principe, *Information theoretic learning: Renyi's entropy and Kernel perspectives*, Springer Verlag, 2010.

[9] Sudhir Rao, Allan de Medeiros Martins, and Jose C. Principe, "Mean shift: An information theoretic perspective," *Pattern Recogn. Lett.*, vol. 30, no. 3, pp. 222–230, 2009.

[10] Rui Liao, Christoph Guetter, Chenyang Xu, Yiyong Sun, Ali Khamene, and Frank Sauer, "Learning-based 2d/3d rigid registration using jensen-shannon divergence for image-guided surgery," in *Medical Imaging and Augmented Reality*, Guang-Zhong Yang, Tianzi Jiang, Dinggang Shen, Lixu Gu, and Jie Yang, Eds., vol. 4091 of *Lecture Notes in Computer Science*, pp. 228–235. Springer Berlin / Heidelberg, 2006.

[11] A. Mastin, J. Kepner, and J. Fisher, "Automatic registration of lidar and optical images of urban scenes," *Computer Vision and Pattern Recognition, IEEE Computer Society Conference on*, vol. 0, pp. 2639–2646, 2009.

[12] Günther Wyszecki and W. S. Stiles, *Color Science: Concepts and Methods, Quantitative Data and Formulae (Wiley Series in Pure and Applied Optics)*, Wiley-Interscience, 2 edition, August.

[13] J. Triesch and C. Von Der Malsburg, "Robust classification of hand postures against complex backgrounds," in *Automatic Face and Gesture Recognition, 1996., Proceedings of the Second International Conference on*. IEEE, 1996, pp. 170–175.

[14] J. Tory Cobb, K.C. Slatton, and G.J. Dobeck, "A parametric model for characterizing seabed textures in synthetic aperture sonar images," *Oceanic Engineering, IEEE Journal of*, vol. 35, no. 2, pp. 250–266, 2010.