

Article

Measure of Similarity between GMMs Based on Geometry-Aware Dimensionality Reduction

Branislav Popović ^{1,*} , Marko Janev ², Lidija Krstanović ¹, Nikola Simić ¹  and Vlado Delić ¹¹ Faculty of Technical Sciences, University of Novi Sad, Trg Dositeja Obradovića 6, 21000 Novi Sad, Serbia² Institute of Mathematics, Serbian Academy of Sciences and Arts, Kneza Mihaila 36, 11000 Belgrade, Serbia

* Correspondence: bpopovic@uns.ac.rs

Abstract: Gaussian Mixture Models (GMMs) are used in many traditional expert systems and modern artificial intelligence tasks such as automatic speech recognition, image recognition and retrieval, pattern recognition, speaker recognition and verification, financial forecasting applications and others, as simple statistical representations of underlying data. Those representations typically require many high-dimensional GMM components that consume large computing resources and increase computation time. On the other hand, real-time applications require computationally efficient algorithms and for that reason, various GMM similarity measures and dimensionality reduction techniques have been examined to reduce the computational complexity. In this paper, a novel GMM similarity measure is proposed. The measure is based on a recently presented nonlinear geometry-aware dimensionality reduction algorithm for the manifold of Symmetric Positive Definite (SPD) matrices. The algorithm is applied over SPD representations of the original data. The local neighborhood information from the original high-dimensional parameter space is preserved by preserving distance to the local mean. Instead of dealing with high-dimensional parameter space, the method operates on much lower-dimensional space of transformed parameters. Resolving the distance between such representations is reduced to calculating the distance among lower-dimensional matrices. The method was tested within a texture recognition task where superior state-of-the-art performance in terms of the trade-off between recognition accuracy and computational complexity has been achieved in comparison with all baseline GMM similarity measures.

Keywords: Gaussian Mixture Models; similarity measures; dimensionality reduction; texture recognition**MSC:** 68T01; 97R40

Citation: Popović, B.; Janev, M.; Krstanović, L.; Simić, N.; Delić, V. Measure of Similarity between GMMs Based on Geometry-Aware Dimensionality Reduction.

Mathematics **2023**, *11*, 175. <https://doi.org/10.3390/math11010175>

Academic Editor: Marko Petković

Received: 28 October 2022

Revised: 7 December 2022

Accepted: 11 December 2022

Published: 29 December 2022



Copyright: © 2022 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

Gaussian Mixture Models play an important role in various artificial intelligence and machine learning tasks, such as computer vision, natural language processing, data clustering, classification, and recognition, due to their simplicity and capability to model any probability density function knowing the exact number of modes. GMMs are successfully used in automatic speech recognition [1], speaker verification and recognition [2], image retrieval [3], pattern recognition [4], genre classification [5], age and gender recognition [6] as well as economy [7], mechanics [8], robotics [9], and numerous other research areas. They are also exceptionally popular as input and/or output data representations in deep learning [10]. At the same time, complex machine learning systems often comprise high-dimensional feature representations. Finding efficient and precise similarity measures between GMMs involving proper dimensionality reduction techniques to reduce computational complexity and consequently, execution times, became an imperative.

Numerous GMM similarity measures have been proposed in the literature. Informational distances among probability distributions, such as Chernoff, Bhattacharyya, or Matusita [11], have been thoroughly analyzed and explored. Nevertheless, the Kullback–Leibler (KL) divergence [12] emerged as the most natural and effective informational

distance measure between two probability distributions. A solution for the KL divergence among two Gaussian components exists in the analytic, i.e., closed form. However, a closed-form solution for the KL divergence between arbitrary GMMs cannot be analytically expressed, nor does any computationally efficient algorithm exist. Consequently, various more or less computationally expensive approximation functions have been proposed in recent years. The Monte Carlo sampling method [13] offers an arbitrary accurate, computationally expensive solution, often inappropriate for real-time classification and/or recognition tasks. Other approximations of the KL divergence among two GMMs have also been proposed [14], such as the variational approximation and the variational upper bound, an approximation based on the unscented transform, or the matched bound approximation, performing poorly on GMMs comprising a few low-probability components. In [15], the so-called Gaussian Quadratic Form Distance (GQFD) with a closed-form solution for GMMs of diagonal covariance matrices is presented. A multivariate online Kernel Density Estimation (KDE) has been proposed in [16]. The KDE enables building PDFs from data by observing only a single data point at a time. In [17], a metric on the space of multivariate Gaussian distributions based on the fundamental idea of parametrizing the space as the Riemannian symmetric space is proposed. In [18], a robust and efficient sparse representation-based Earth Mover's Distance (EMD) is presented. The EMD uses an effective pair-wise-based method to learn EMD metrics among GMMs, along with two ground distances between Gaussian components based on the information geometry, obtained by embedding the space of Gaussians into a Lie group or regarding it as the product of Lie groups, to measure the intrinsic distance between Gaussians in the underlying Riemannian manifold. The method is additionally improved and extended in [19], also including a study on various image features for GMM matching, such as the Gabor filter, Local Binary Pattern descriptor, SIFT, covariance descriptor and high-level features extracted by deep convolution networks. Computational efficiency, as well as the accuracy of these approximations, have been confirmed in most cases by experiments on both real and synthetic data.

Defining the proper feature space dimensionality reduction technique to resolve computational complexity and cope with the problems of data sparsity and the curse of dimensionality is another challenging issue. In miscellaneous natural language processing, speech recognition and synthesis, emotion recognition, image recognition and retrieval and other machine learning tasks, feature vectors contain hundreds or even thousands of features. Diverse dimensionality reduction techniques have been designed to reduce computational complexity, aiming to keep the same or at least highly comparable accuracy. For instance, the Principal Geodesic Analysis (PGA) tries to map SPD matrices into a tangent by maximizing the variability of the mapped data points [20]. Nonlinear dimensionality reduction algorithms such as Locally Linear Embedding (LLE) and Laplacian Eigenmaps (LE) provide embeddings into lower dimensional space based on Riemannian geometry [21]. The LE method uses the connection between the Laplace Beltrami operator and the graph Laplacian to construct representation with locally preserving properties [22]. Locality Preserving Projections (LPP), an approach based on the LE, was developed as an alternative to the Principal Component Analysis (PCA). The LPP tends to learn linear projective maps by solving a variational problem that optimally preserves the local neighborhood structure of the original dataset in the transformed space [23]. On the other hand, kernel approaches, like those presented in [24], try to embed feature matrices in a Reproducing Kernel Hilbert Space. Dimensionality reduction is then performed using various kernel-based methods. Finally, the idea behind manifold learning techniques is to increase discrimination of the transformed features by projecting those features to a lower dimensional manifold embedded in a subspace of the original high dimensional feature space [25]. Supervised methods, such as Linear Discriminant Analyses (LDA) or the Maximum Margin Criterion (MMC) as well as unsupervised methods, like the Principal Component Analysis, are some of the most popular manifold learning representatives. Unlike the PCA which aims to preserve the global structure of the data, the LPP tends to preserve the local structure of the data. Therefore, it may keep more discriminating information, assuming that samples

from the same class are close to each other in the input space. Neighborhood Preserving Embedding (NPE) is yet another methodology aiming to preserve the local neighborhood structure on a data manifold. The NPE is able to learn not only the projective matrix but also the weights extracting the neighborhood information in the original feature space [26].

Two different dimensionality reduction methods have been analyzed and compared in our previous work. The first one was based on an LPP-like projection of the parameter space [27]. The LLP-based GMM similarity measure was used to calculate the linear transformation matrix that projects the vectorized parameters of Gaussian components of arbitrary GMMs into a low-dimensional Euclidean space. At the same time, the distinctiveness of the original feature space is preserved in lower-dimensional space. Both the symmetric and the nonsymmetric version of the LPP-based GMM similarity measure has been developed, utilizing the symmetric or the one-sided KL divergence between Gaussian components corresponding to GMMs. The other one assumes that the parameters of full covariance Gaussians lie close to each other in a lower-dimensional surface embedded in the cone of positive definite matrices. This is contrary to the assumption that data themselves lie on the lower-dimensional manifold embedded in the feature space [28]. The NPE-based idea has been employed to evaluate the projection matrix. The matrix is then applied to the parameter space of Gaussian components, by projecting the parameters of Gaussian components into lower-dimensional Euclidean vectors.

Recently, a novel geometry-aware dimensionality reduction technique has been presented [29]. This technique tends to preserve the local structure of the data by Distance Preservation to the Local Mean (DPLM), considering the geometry of the SPD matrices. Based on this approach, a novel GMM similarity measure is proposed in the paper. The method utilizes the fact that the space of multivariate Gaussians is a Riemannian manifold that can be embedded into the cone of SPD matrices. Both the supervised and unsupervised version of the DPLM algorithm has been employed. Baseline KL-based GMM similarity measures are then applied over low-dimensional feature matrices, i.e., GMM projections, preserving the locality induced by the manifold structure from the original parameter space, while achieving significantly lower computational cost. A much better trade-off between the recognition accuracy and the computational complexity has been achieved in comparison to KL-based distance approximations calculated between GMMs from the original parameter space. Experiments are conducted within a texture recognition task, but the proposed method is suitable for any big-data artificial intelligence system using a large number of GMMs as well as high dimensional features.

The paper is organized as follows. In Section 2, we start with a review of baseline KL-based GMM similarity measures presented in the literature. We then propose a novel GMM similarity measure motivated by the geometry-aware dimensionality reduction algorithm presented in [29], projecting the original feature space into a low-dimensional feature space. Computational complexities in the recognition phase are also estimated. In Section 3, we compare and discuss the results obtained using the proposed DPLM-based and baseline KL-based similarity measures within a texture recognition task conducted on three publicly available image databases (UIUC [30], KTH-TIPS [31], and UMD [32] database). In all examined cases, the results obtained using the proposed method were highly superior concerning the trade-off between accuracy and computational complexity compared to all baseline methods. The paper is summarized in Section 4. Author contributions, funding and data availability statement are provided at the end of the manuscript.

2. Materials and Methods

In the following section, we will discuss baseline GMM similarity measures based on some of the most popular KL divergence approximations presented in the literature and propose a novel geometry-aware GMM similarity measure constructed using a nonlinear geometry-aware dimensionality reduction algorithm for the manifold of SPD matrices. At the end of the section, we will estimate the computational complexities of the proposed and baseline GMM similarity measures.

2.1. KL-Based GMM Similarity Measures

The KL divergence measures how much one probability distribution differs from another probability distribution [33]. Although there are other loss functions, the KL divergence is used as a fundamental equation in information theory and the most natural solution in many machine learning tasks dealing with probability distributions. For two probability distributions p and q , the measure is defined as $KL(p||q) = \int_{\mathbb{R}^d} p(x) \log \frac{p(x)}{q(x)} dx$. For two simple Gaussians \hat{p} and \hat{q} , it can be computed easily using an intuitive closed-form solution given by

$$KL(\hat{p}||\hat{q}) = \frac{1}{2}[\log \frac{|\Sigma_{\hat{q}}|}{|\Sigma_{\hat{p}}|} + Tr[\Sigma_{\hat{q}}^{-1}\Sigma_{\hat{p}}] + (\mu_{\hat{p}} - \mu_{\hat{q}})^T \Sigma_{\hat{q}}^{-1}(\mu_{\hat{p}} - \mu_{\hat{q}}) - d], \tag{1}$$

where d is the dimensionality of Gaussians \hat{p} and \hat{q} , μ and Σ are their mean vectors and covariance matrices, and Tr is the trace function, i.e., the sum of elements on the main diagonal. On the other hand, there is no closed-form solution for the KL divergence between two GMMs.

The Monte Carlo (MC) sampling [14] is a straightforward and the most accurate, although computationally extremely expensive solution for the KL divergence between two different GMMs. The idea is to sample the probability distribution p using independent and identically distributed (i.i.d.) random samples $x_i, i = 1, \dots, N$, so that $KL(p||q) = E_p[\ln \frac{p(x)}{q(x)}]$. Using N samples, we obtain

$$KL_{MC}(p||q) = \frac{1}{N} \sum_{i=1}^N \ln \frac{p(x_i)}{q(x_i)} \rightarrow KL(p||q), \tag{2}$$

as $N \rightarrow \infty$. The variance of the estimation error is now computed as $\frac{1}{n} Var_p[\log \frac{p}{q}]$. Unfortunately, the solution is unacceptably time-consuming and expensive for most real-world and big-data applications, which is why various approximations of the KL divergence are proposed for estimating the KL divergence between two GMMs accurately and efficiently.

The roughest approximation is based on the convexity of the KL divergence. The upper bound of the KL divergence [34] between two GMMs $p = \sum_{i=1}^n \alpha_i p_i$ and $q = \sum_{j=1}^m \beta_j q_j$ is given by

$$KL(p||q) \leq \sum_{i,j} \alpha_i \beta_j KL(p_i||q_j),$$

where $p_i = \mathcal{N}(\Sigma_i, \mu_i)$ and $q_j = \mathcal{N}(\Sigma_j, \mu_j)$ are Gaussian components of the corresponding mixtures, $\alpha_i > 0$ and $\beta_j > 0$ are the corresponding weights, satisfying $\sum_i \alpha_i = 1, \sum_j \beta_j = 1$, and $KL(p_i||q_j)$ can be computed using (1), yielding the Weighted Average (WA) approximation

$$KL_{WA}(p||q) \approx \sum_{i,j} \alpha_i \beta_j KL(p_i||q_j). \tag{3}$$

KL_{WA} approximation is computationally much more efficient than KL_{MC} approximation. However, this approximation is too crude in cases when each mixture density is composed of unimodal distributions and the modes are far apart [34].

Various other approximations of the KL divergence between two GMMs have been proposed and applied in several machine learning tasks, such as speech recognition, image retrieval and speaker identification [34–36]. The Matching-Based (MB) Approximation [34] given by

$$KL_{MB}(p||q) \approx \sum_i \alpha_i \min_j \left[KL(p_i||q_j) + \log \left(\frac{\alpha_i}{\beta_j} \right) \right] \tag{4}$$

is based on the assumption that the element q_j that is most proximal to p_i dominates the integral $\int p_i \log q$. A more efficient matching based approximation has also been proposed, given by

$$KL_{MBS}(p||q) \approx \sum_i \alpha_i \min_j KL(p_i||q_j). \tag{5}$$

This approximation provides good performances when the Gaussian components of p and q are mostly far apart. However, KL_{MBS} is inappropriate in cases when there is significant overlapping among Gaussians from p and q . The Unscented Transform based approximation which uses the unscented estimator similar to the Monte Carlo approximation, except the samples are chosen deterministically, provides a way to deal with GMMs overlapping.

The Unscented Transform (UC) is a mathematical function used to estimate the results of applying a nonlinear transformation to a probability distribution characterized in terms of a finite set of statistics [35]. Assuming that $KL(p||q) = \int_{\mathbb{R}^d} p \log p - \int_{\mathbb{R}^d} p \log q$, the unscented transform approach tends to approximate the integral $\int_{\mathbb{R}^d} p_i \log q$ as

$$\int_{\mathbb{R}^d} p_i \log q \approx \frac{1}{2d} \sum_{k=1}^{2d} \log q(x_{i,k}), \tag{6}$$

$$x_{i,k} = \mu_i + (\sqrt{\Sigma_i})_k, \quad k = 1, \dots, d, \tag{7}$$

$$x_{i,d+k} = \mu_i - (\sqrt{\Sigma_i})_k, \quad k = 1, \dots, d, \tag{8}$$

$(\sqrt{\Sigma_i})_k$ is the k th column of the matrix square root of Σ_i . Approximating the integral $\int_{\mathbb{R}^d} p \log q$, we obtain

$$\int_{\mathbb{R}^d} p \log q \approx \frac{1}{2d} \sum_{i=1}^n \alpha_i \sum_{k=1}^{2d} \log q(x_{i,k}). \tag{9}$$

Approximating the second integral $\int_{\mathbb{R}^d} p \log p$ in similar manner, the $KL_{UC}(p||q)$ is obtained.

The Variational Approximation (VA) [14,36] is given by

$$KL_{VA}(p||q) = \sum_i \alpha_i \frac{\sum_{i'} \alpha_{i'} e^{-KL(p_i||p_{i'})}}{\sum_j \beta_j e^{-KL(p_i||q_j)}} \tag{10}$$

This approximation utilizes the KL divergence between the Gaussian components in order to obtain an approximate KL divergence between the full GMMs p and q . This is a simple, closed-form expression.

2.2. DPLM-Based GMM Similarity Measure

To construct a novel DPLM-based GMM similarity measure and decrease the computational cost, we propose the following procedure:

1. The original set of Gaussian mixture components is embedded into the cone of SPD matrices;
2. A nonlinear geometry-aware dimensionality reduction algorithm for the manifold of SPD matrices [29] is applied to obtain a projection matrix, providing a low dimensional representation of the manifold by preserving the distance to the local mean (DPLM);
3. The embeddings are projected into a lower dimensional space using the projection matrix, where baseline KL-based measures can now be used to measure how much one GMM differs from another in a cost-efficient way.

A Riemannian manifold is a real, smooth manifold with a Riemannian metric, equipped with a positive-definite inner product on the tangent space at each point that varies smoothly from point to point, providing local notions of angle, length of curves, surface area and volume [37]. A set of $d \times d$ SPD matrices denoted by $Sym_+(d)$ is a differentiable manifold with a natural Riemannian structure. To obtain the projection matrix using the proposed nonlinear geometry-aware dimensionality reduction algorithm for the manifold of SPD matrices [29], we use the fact that a set of multivariate Gaussians is a Riemannian manifold and that any d -dimensional multivariate Gaussian $g = \mathcal{N}(\mu, \Sigma)$ can be embedded into the cone of SPD matrices $Sym_+(d + 1)$ in the following way

$$g \hookrightarrow P = |\Sigma|^{-\frac{1}{d+1}} \begin{bmatrix} \Sigma + \mu\mu^T & \mu \\ \mu^T & 1 \end{bmatrix} \tag{11}$$

where $|\Sigma| > 0$ denotes the determinant of the covariance matrix of Gaussian component g [18]. All information regarding the particular Gaussian component $\mathcal{N}(\mu, \Sigma)$ of a given GMM is now contained in a single positive definite matrix P , an element of $Sym_+(d + 1)$.

The aim is to find a projection matrix W , mapping the above-mentioned embeddings into $Sym_+(l)$, where $l < d$. The local structure is preserved in lower dimensional representation by preserving distance to the local mean, i.e., by calculating the Riemannian mean of the K -nearest neighbors of each embedding in order to find a projection matrix that preserves distances between nearest neighbors and their means.

Let us assume we have obtain a set of N $(d + 1)$ -dimensional embeddings $\{(P_1, c_1), (P_2, c_2), \dots, (P_N, c_N)\}$ where $P_i \in Sim_+(d + 1)$, c_i is a corresponding class label, and $N_i = \{P_{i,1}, P_{i,2}, \dots, P_{i,K}\}, 1 \leq i \leq N$, is the set of K nearest neighbors of P_i . In the case of the supervised version of the algorithm, K -nearest neighbors are selected only among embeddings which have the same class label as the current embedding. The Riemannian mean of each set N_i denoted by \hat{N}_i is calculated using equation

$$\hat{N}_i = \operatorname{argmin}_{P \in Sim_+(d+1)} \sum_{k=1}^K \delta_g^2(P, P_{i,k}), \tag{12}$$

where δ_g^2 is Affine Invariant Riemannian Metric [38], given by

$$\delta_g^2(P, Q) = \|\log(PQ^{-1})\|_F^2. \tag{13}$$

The projection matrix W can now be calculated by solving optimization problem given by

$$\min_{W \in \mathbb{R}^{(d+1) \times l}} H(W), W^T \times W = I_l, \tag{14}$$

where I_l is an $l \times l$ identity matrix, and

$$H(W) = \sum_{i=1}^N \sum_{k=1}^K |\delta_{I_d}^2(P_{i,k}, \hat{N}_i) - \delta_{I_d}^2(W^T P_{i,k} W, W^T \hat{N}_i W)|, \tag{15}$$

where

$$\delta_{I_d}^2(P, Q) = \sqrt{J(P, Q)}, \tag{16}$$

and $J(P, Q)$ is the Jensen–Bregman LogDet Divergence [39], given by

$$J(P, Q) = \logdet\left(\frac{P + Q}{2}\right) - \frac{1}{2} \logdet(PQ). \tag{17}$$

The gradient of $H(W)$ with respect to W can be computed as

$$\frac{\partial H(W)}{\partial W} = - \sum_{i=1}^N \sum_{k=1}^K \left[\operatorname{sgn}(\delta_{id}^2(P_{i,k}, \hat{N}_i) - \delta_{id}^2(W^T P_{i,k} W, W^T \hat{N}_i W)) \times \left((P_{i,k} + \hat{N}_i) W (W^T \frac{P_{i,k} + \hat{N}_i}{2} W)^{-1} - P_{i,k} W (W^T P_{i,k} W)^{-1} - \hat{N}_i W (W^T \hat{N}_i W)^{-1} \right) \right], \tag{18}$$

where sgn is the sign function, using the prior knowledge

$$\frac{\delta_{id}^2(W^T P W, W^T Q W)}{\partial W} = (P + Q) W (W^T \frac{P+Q}{2} W)^{-1} - P W (W^T P W)^{-1} - Q W (W^T Q W)^{-1}. \tag{19}$$

The lower dimensional projection of $P_i \in \operatorname{Sim}_+(d + 1)$ can now be computed as

$$P'_i = W^T P_i W \in \operatorname{Sim}_+(l), \tag{20}$$

and used as the original GMM representative inside Equations (3), (4) and (10), providing computationally efficient DPLM-based approximations $sDPLM_{WA}$, $sDPLM_{MB}$ and $sDPLM_{VA}$ for the supervised, and $uDPLM_{WA}$, $uDPLM_{MB}$ and $uDPLM_{VA}$ for the unsupervised version of the above-mentioned algorithm. Namely, the GMMs p and q in Formulas (3), (4) and (10) are obtained as lower-dimensional projections of the original $(d + 1)$ -dimensional embeddings given in the form of symmetric positive definite matrices obtained by (11), using the projection matrix W , calculated by solving the optimization problem given by (14) and introduced into the expression (20). As previously explained, for $sDPLM_{WA}$, $sDPLM_{MB}$ and $sDPLM_{VA}$, K nearest neighbors (see expression (12)) are selected only among embeddings which have the same class label as the current embedding. In the case of $uDPLM_{WA}$, $uDPLM_{MB}$ and $uDPLM_{VA}$, no such requirement has been applied.

2.3. Computational Complexity

In the following subsection, we'll define the computational cost of the above-mentioned algorithms in the testing phase, bearing in mind that computational cost in the training phase is not crucial for employment.

Let us assume, without loss of generality, that GMMs p and q have the same number of components m , represented using full covariance matrices, and d is the dimension of the original feature space.

The computational complexity of the Monte Carlo approximation KL_{MC} is estimated as $O(Nmd^3)$, where N is the number of samples. However, to obtain an efficient KL divergence approximation, the number of i.i.d. samples N must be very large, i.e., $N \gg m$.

The complexity of the KL-based measures KL_{WA} , KL_{MB} and KL_{VA} is roughly equivalent and estimated as $O(m^2d^3)$. The complexity of calculating the KL divergence between two d -variate Gaussians is of order $O(d^3)$. This complexity is approximately equal to the complexity of calculating the inversion of a $d \times d$ matrix. Since there are m^2 such inversions, we obtain the previous estimate.

According to [40], the complexity of multiplications between $(d \times d)$ -dimensional embeddings P_i and $(d \times l)$ -dimensional projection matrix W is estimated as $O(ld^2) + O(dl^2)$ based on expression (20) (both left and right multiplications). There are m such multiplications, so the complexity is calculated as $O(mld^2) + O(mdl^2)$. The complexity of the KL-based measures applied over $(l \times l)$ -dimensional projections is roughly estimated as $O(m^2l^3)$, as previously explained. Therefore, the complexity of the proposed DPLM-based solutions, namely, the $sDPLM_{WA}$, $sDPLM_{MB}$, $sDPLM_{VA}$, $uDPLM_{WA}$, $uDPLM_{MB}$ and $uDPLM_{VA}$ is now estimated as $O(m^2l^3) + O(mld^2) + O(mdl^2)$, where l is the dimension of the transformed feature space.

3. Results and Discussion

In this section, we present the results obtained using novel DPLM-based GMM similarity measures proposed in Section 2.2 with baseline KL-based GMM similarity measures described in Section 2.1. The algorithms were evaluated in a texture recognition task. The system was trained using data extracted from three publicly available corpora—the UIUC texture database (named after the University of Illinois Urbana-Champaign), the KTH-TIPS image database (KTH is an abbreviation of the Royal Institute of Technology while TIPS stands for Textures under varying Illumination, Pose and Scale), and the UMD texture dataset (named after the University of Maryland). Concerning the UIUC database, 5 classes have been extracted (wood, water, granite, marble, and floor), taken at 640×480 pixels. In the case of KTH-TIPS, we also took 5 classes (aluminum foil, brown bread, corduroy, cotton, and cracker), and the images were cropped at 200×200 pixels. Finally, for the third database, we used sample images from classes 2 (paint cans), 3 (stones), 8 (brick walls), 9 (apples) and 12 (textile patterns), sampled at 1280×960 pixels. Selected samples from all three databases are shown in Figure 1.

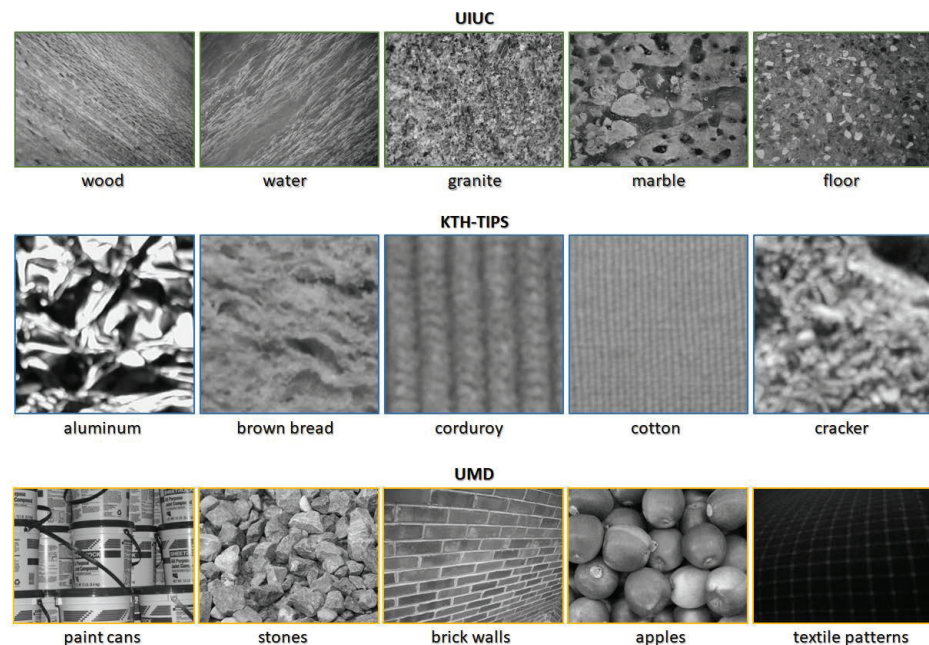


Figure 1. Samples extracted from UIUC, KTH-TIPS and UMD databases.

For the purposes of experiments, KL_{WA} , KL_{MB} and KL_{VA} , defined by Equations (3), (4) and (10), were selected as baseline GMM similarity measures. Both the supervised ($sDPLM_{WA}$, $sDPLM_{MB}$, $sDPLM_{VA}$) and the unsupervised ($uDPLM_{WA}$, $uDPLM_{MB}$, $uDPLM_{VA}$) versions of DPLM-based GMM similarity measures were applied. Compared to all baseline measures, a significantly better trade-off between computational cost and accuracy was obtained for the proposed DPLM-based GMM similarity measures.

Region covariance descriptors proposed in [41] were used as texture features since they have already shown good performance in various texture recognition tasks. They were formed in the following way. For any given image, patches of size 128×128 for the UIUC (step 16), 40×40 for the KTH-TIPS (step 5), or 256×256 for the UMD database (step 32) were extracted. For every pixel positioned at (x, y) , features were calculated in a form $[I, |I_x|, |I_y|, |I_{xx}|, |I_{yy}|]$, where I represents illumination, I_x and I_y are the first, and I_{xx} and I_{yy} are the second-order derivatives (meaning that the actual dimension of feature vectors was $\hat{d} = 5$). Covariance matrices were calculated using these features and additionally vectorized by aligning their upper triangular values into $d = \hat{d}(\hat{d} + 1)/2 = 15$ -dimensional feature vectors. The parameters of GMMs were then estimated using the Expectation Maximization (EM) algorithm [42] applied over the pool of feature vectors obtained as

previously described. Every sample image was uniformly divided into four sub-images represented by four GMMs, which were then used for training and testing purposes. Every GMM (or its low-dimensional projection in the case of DPLM-based algorithms) was compared to all other GMMs in the train set and its label was determined as a majority vote over 5 nearest neighbors, using the K-Nearest Neighbors algorithm (KNN), and in the case of DPLM-based algorithms, the GMMs were embedded into the cone of SPD matrices $Sym+(16)$.

In Tables 1–3, the recognition accuracies are presented for the proposed DPLM-based measures as well as baseline KL-based measures. Each GMM was represented using one to ten Gaussian components ($m \in \{1, 5, 10\}$). For the case of DPLM-based measures, the number of nearest neighbors of any current embedding was set to $K = 3$, and the projection dimension was set to $l \in \{5, 7\}$, providing projection matrices of size 16×5 and 16×7 , respectively, i.e., less than a half of the original feature space dimension.

Table 1. Recognition accuracies for DPLM-based and KL-based measures on the UIUC database.

GMM Sim. Meas.	Accuracy					
	m = 1		m = 5		m = 10	
KL-MB	0.82		0.80		0.80	
KL-WA	0.82		0.82		0.82	
KL-VA	0.82		0.82		0.82	
	l = 5	l = 7	l = 5	l = 7	l = 5	l = 7
uDPLM-MB	0.72	0.81	0.73	0.74	0.79	0.79
uDPLM-WA	0.72	0.81	0.73	0.74	0.80	0.80
uDPLM-VA	0.72	0.81	0.73	0.74	0.80	0.80
sDPLM-MB	0.73	0.80	0.73	0.74	0.79	0.72
sDPLM-WA	0.73	0.80	0.73	0.74	0.80	0.73
sDPLM-VA	0.73	0.80	0.73	0.74	0.80	0.73

Table 2. Recognition accuracies for DPLM-based and KL-based measures on the KTH-TIPS database.

GMM Sim. Meas.	Accuracy					
	m = 1		m = 5		m = 10	
KL-MB	0.78		0.74		0.75	
KL-WA	0.78		0.78		0.78	
KL-VA	0.78		0.78		0.78	
	l = 5	l = 7	l = 5	l = 7	l = 5	l = 7
uDPLM-MB	0.57	0.73	0.69	0.71	0.63	0.72
uDPLM-WA	0.57	0.73	0.72	0.75	0.64	0.75
uDPLM-VA	0.57	0.73	0.72	0.75	0.63	0.75
sDPLM-MB	0.56	0.74	0.66	0.71	0.63	0.71
sDPLM-WA	0.56	0.74	0.70	0.75	0.64	0.73
sDPLM-VA	0.56	0.74	0.69	0.75	0.63	0.73

For each class, a fixed number of samples was randomly selected, keeping the rest for testing. For $l = 7$, the accuracies obtained using the baseline KL-based measures and the full covariance matrices were in most cases only slightly, i.e., less than 5% better than the results obtained using the proposed DPLM-based measures and the reduced-sized representatives. The difference was somewhat more significant for $l = 5$, but on the other hand, the recognition time was reduced by more than third in comparison to all baseline measures. No significant difference has been observed between the supervised and the unsupervised version of the proposed algorithm, probably because we used a relatively small value for K , which is why most nearest neighbors belonged to the same

class even for the unsupervised version, but this will be additionally examined in the future. Recall that computational complexities of the proposed DPLM-based algorithms were roughly estimated as $O(m^2l^3) + O(mld^2) + O(mdl^2)$. At the same time, computational complexities of baseline KL-based measures were estimated as $O(m^2d^3)$. The ratio between the computational complexity of the proposed DPLM-based and any mentioned baseline KL-based measures is therefore largely in favor of the proposed DPLM-based measures.

Table 3. Recognition accuracies for DPLM-based and KL-based measures on the UMD database.

GMM Sim. Meas.	Accuracy					
	m = 1		m = 5		m = 10	
KL-MB	0.75		0.73		0.72	
KL-WA	0.75		0.75		0.75	
KL-VA	0.75		0.75		0.75	
	l = 5	l = 7	l = 5	l = 7	l = 5	l = 7
uDPLM-MB	0.73	0.74	0.72	0.72	0.70	0.72
uDPLM-WA	0.73	0.74	0.73	0.74	0.71	0.75
uDPLM-VA	0.73	0.74	0.73	0.74	0.71	0.75
sDPLM-MB	0.75	0.75	0.73	0.72	0.71	0.71
sDPLM-WA	0.75	0.75	0.71	0.73	0.72	0.74
sDPLM-VA	0.75	0.75	0.71	0.73	0.72	0.74

In Figures 2–4, CPU processing times during the test phase are presented for the proposed DPLM-based and baseline KL-based algorithms ($DPLM_{WA}$ vs. KL_{WA} , $DPLM_{MB}$ vs. KL_{MB} , and $DPLM_{VA}$ vs. KL_{VA}), for the UIUC database. Note that the CPU times include not only the given measures but the whole testing procedure, i.e., they also comprise the final voting, meaning that the results would otherwise be even more in favor of the proposed DPLM-based algorithms vs. the KL-based algorithms. In Figures 5–7, the same results are presented for the experiments conducted using the KTH-TIPS database. The results of experiments conducted using the UMD database are presented in Figures 8–10. It can be concluded that the proposed measures provide significantly lower CPU processing times in comparison to all baseline measures due to a significant reduction in the dimensionality of the original feature space.

The experiments on UMD and UIUC databases were conducted on a workstation equipped with AMD Ryzen™ 7 5800H processor, 3.20 GHz, 8 cores, 16 threads, 16 MB cache, 16 GB (2 × 8 GB) DDR4 3200 MHz RAM. The experiments on the KTH-TIPS database were conducted on a workstation equipped with Intel® Core™ i5-4690 processor, 3.50 GHz, 4 cores, 4 threads, 6 MB cache and 16 GB (2 × 8 GB) DDR3 1600 MHz RAM. Differences in execution times among repeated experiments for the same configuration were statistically negligible, i.e., the measurements are reproducible and consistent. Bearing in mind the purpose of these experiments, we did not care about total execution times, i.e., their absolute values, as long as all experiments are conducted on the same hardware for a single database. In fact, we only cared about statistical differences, i.e., ratios between the baseline and the proposed algorithms for different configurations used in the paper.

The proposed methodology could also be applied in various other tasks and learning methodologies comprising models trained during a learning phase, such as the one presented in [43]. The transformation matrix is formed during the training. As a consequence, the process does not consume CPU time during the employment phase. By reducing the dimension of the original feature space, significantly better results have been obtained concerning trade-offs between speed and accuracy in all our experiments.

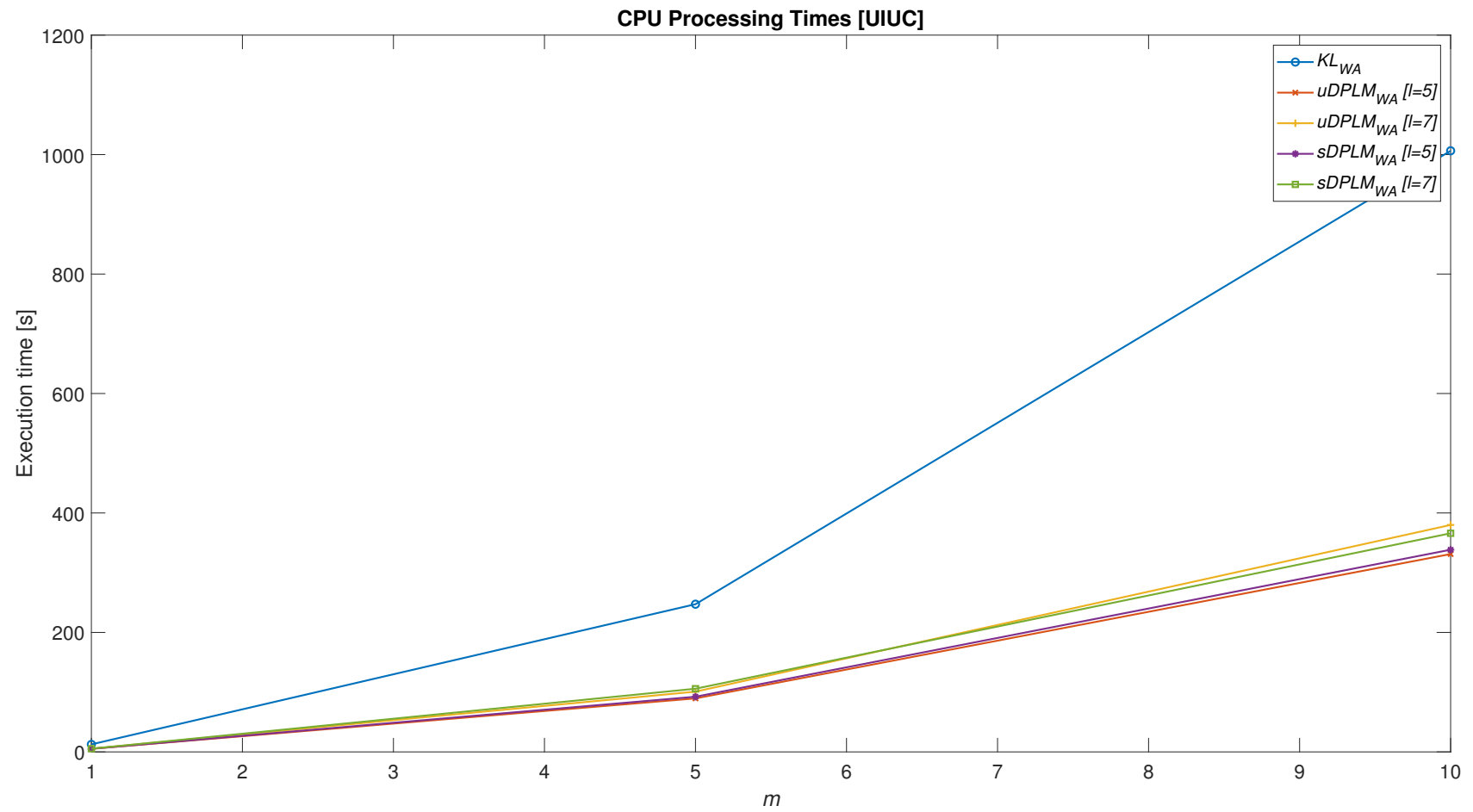


Figure 2. CPU processing times for the UIUC database, $DPLM_{WA}$ vs. KL_{WA} .

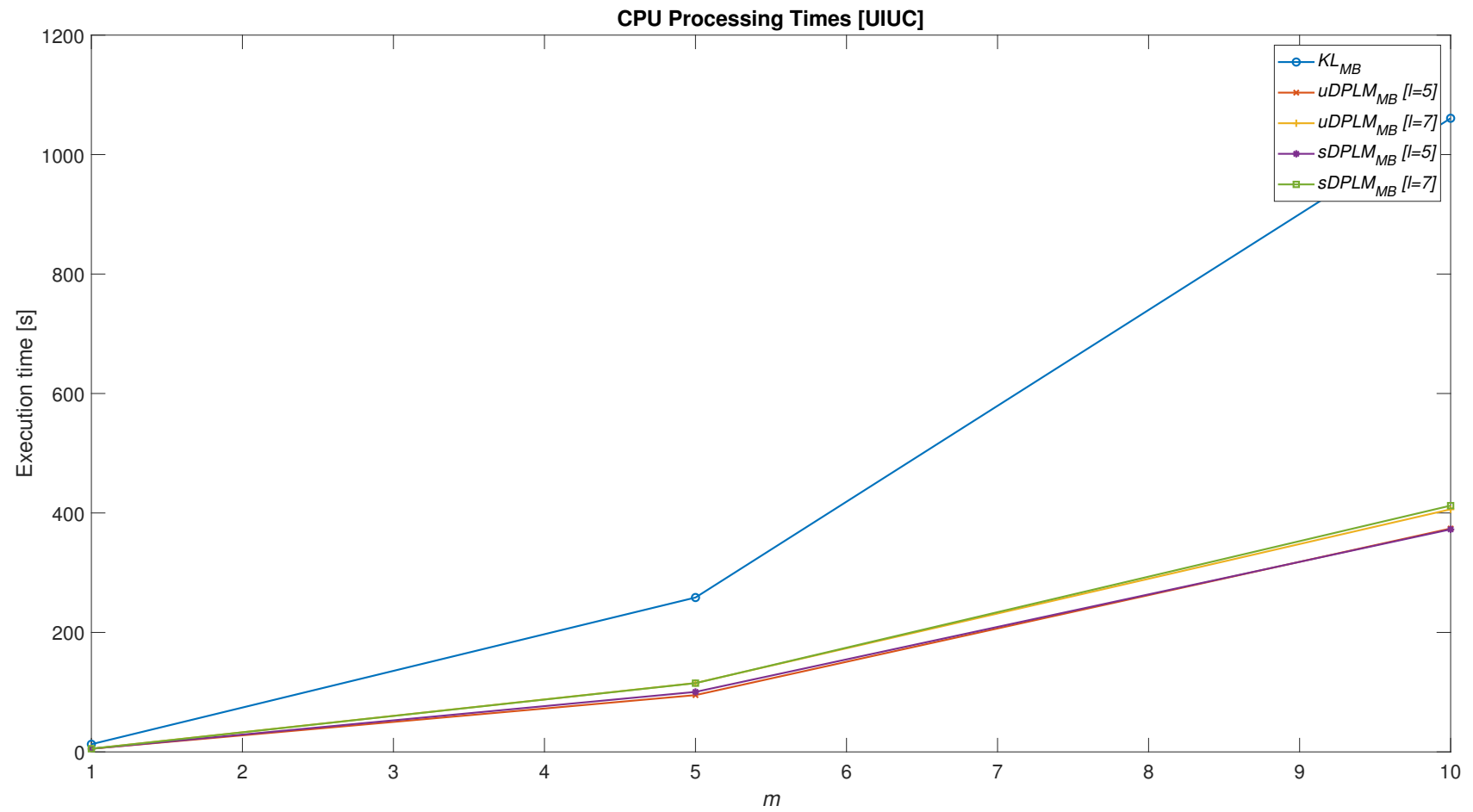


Figure 3. CPU processing times for the UIUC database, $DPLM_{MB}$ vs. KL_{MB} .

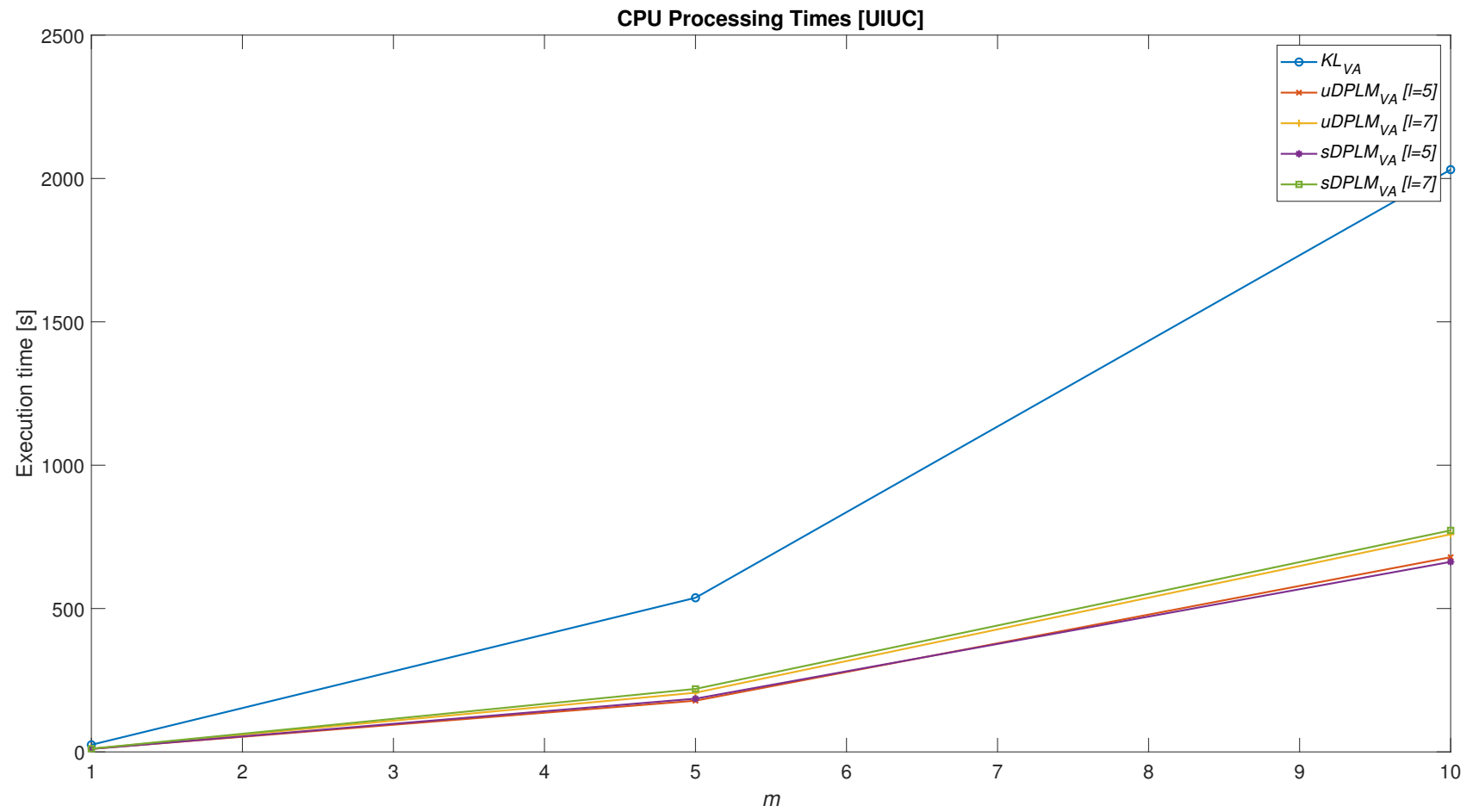


Figure 4. CPU processing times for the UIUC database, $DPLM_{VA}$ vs. KL_{VA} .

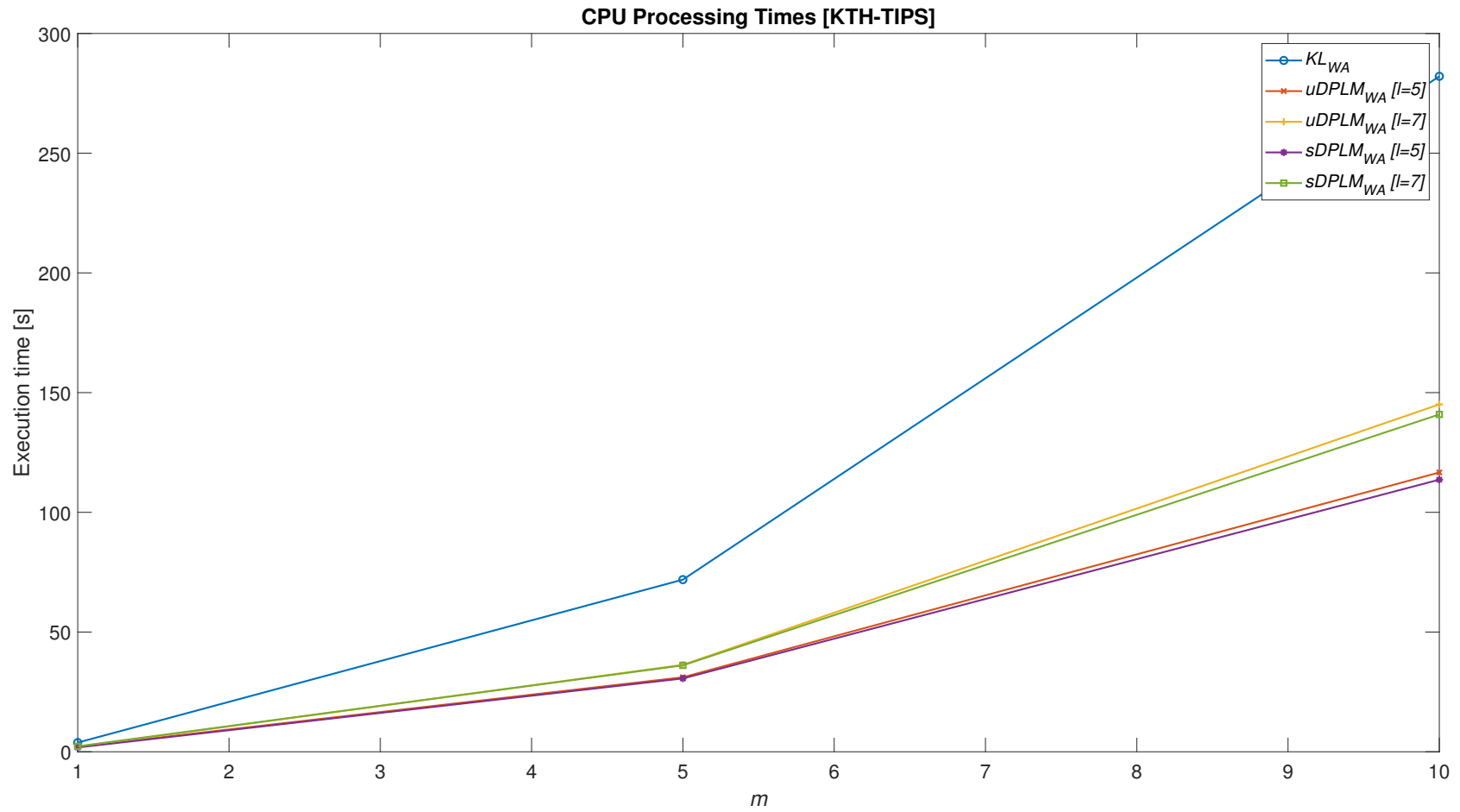


Figure 5. CPU processing times for the KTH-TIPS database, $DPLM_{WA}$ vs. KL_{WA} .

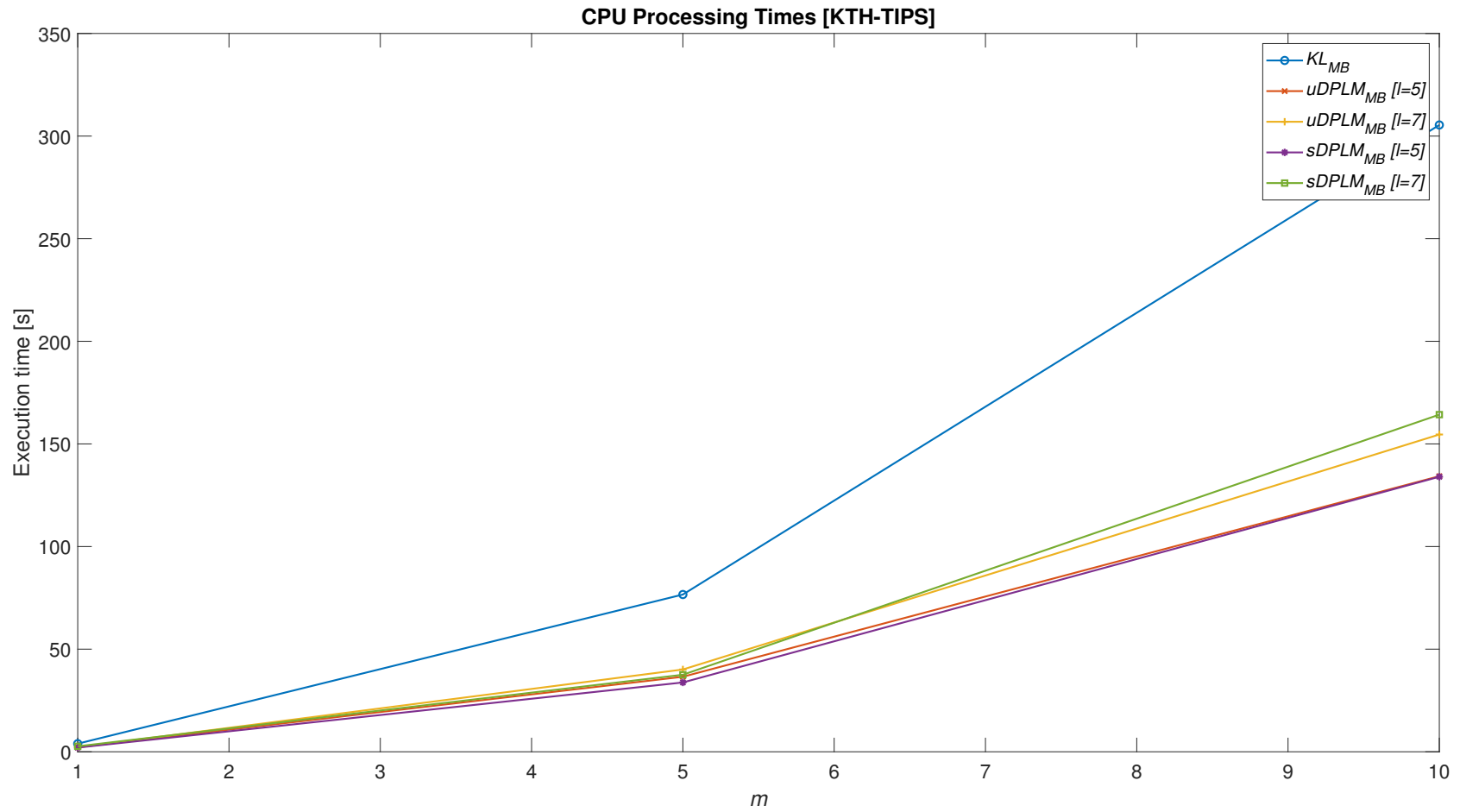


Figure 6. CPU processing times for the KTH-TIPS database, $DPLM_{MB}$ vs. KL_{MB} .

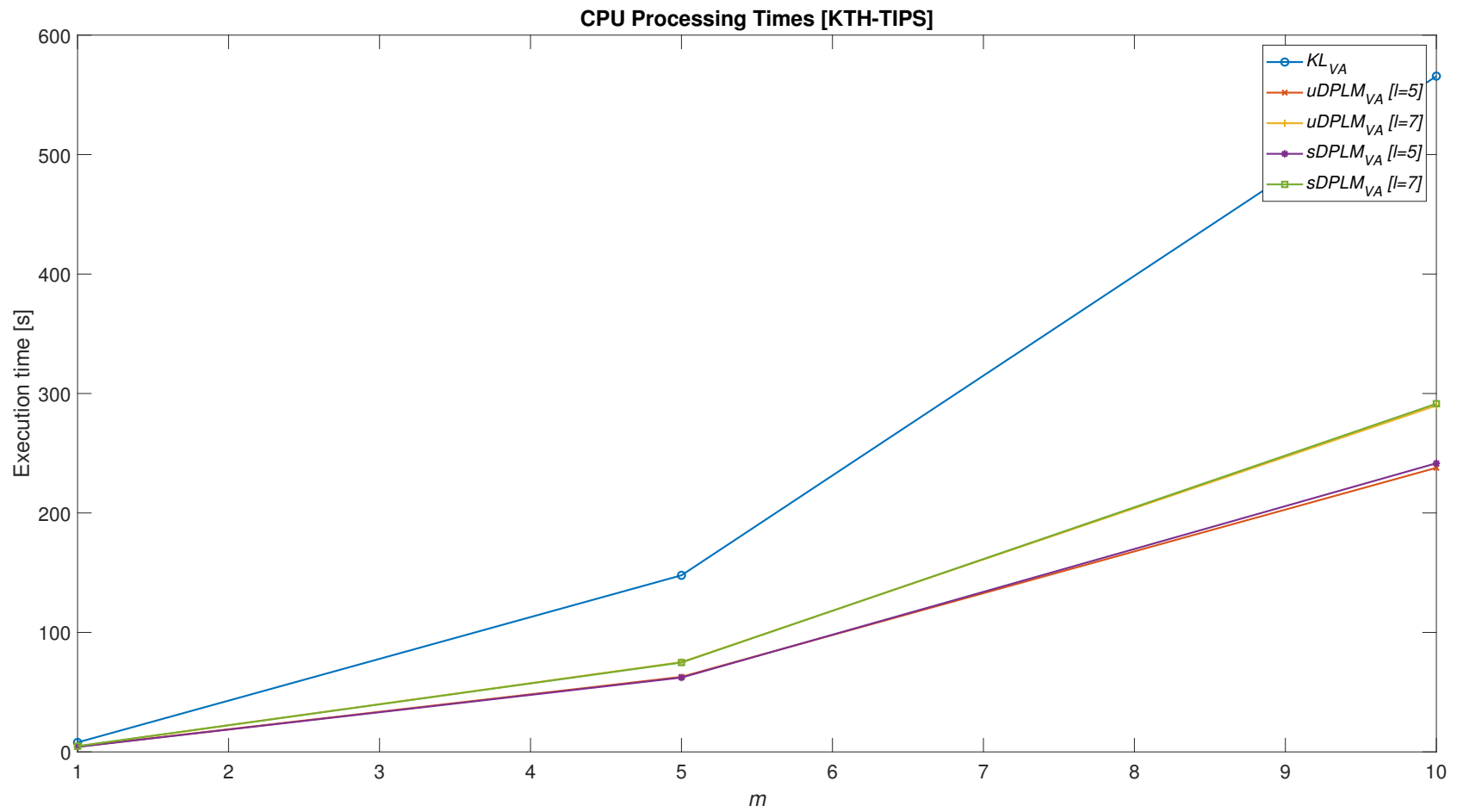


Figure 7. CPU processing times for the KTH-TIPS database, $DPLM_{VA}$ vs. KL_{VA} .

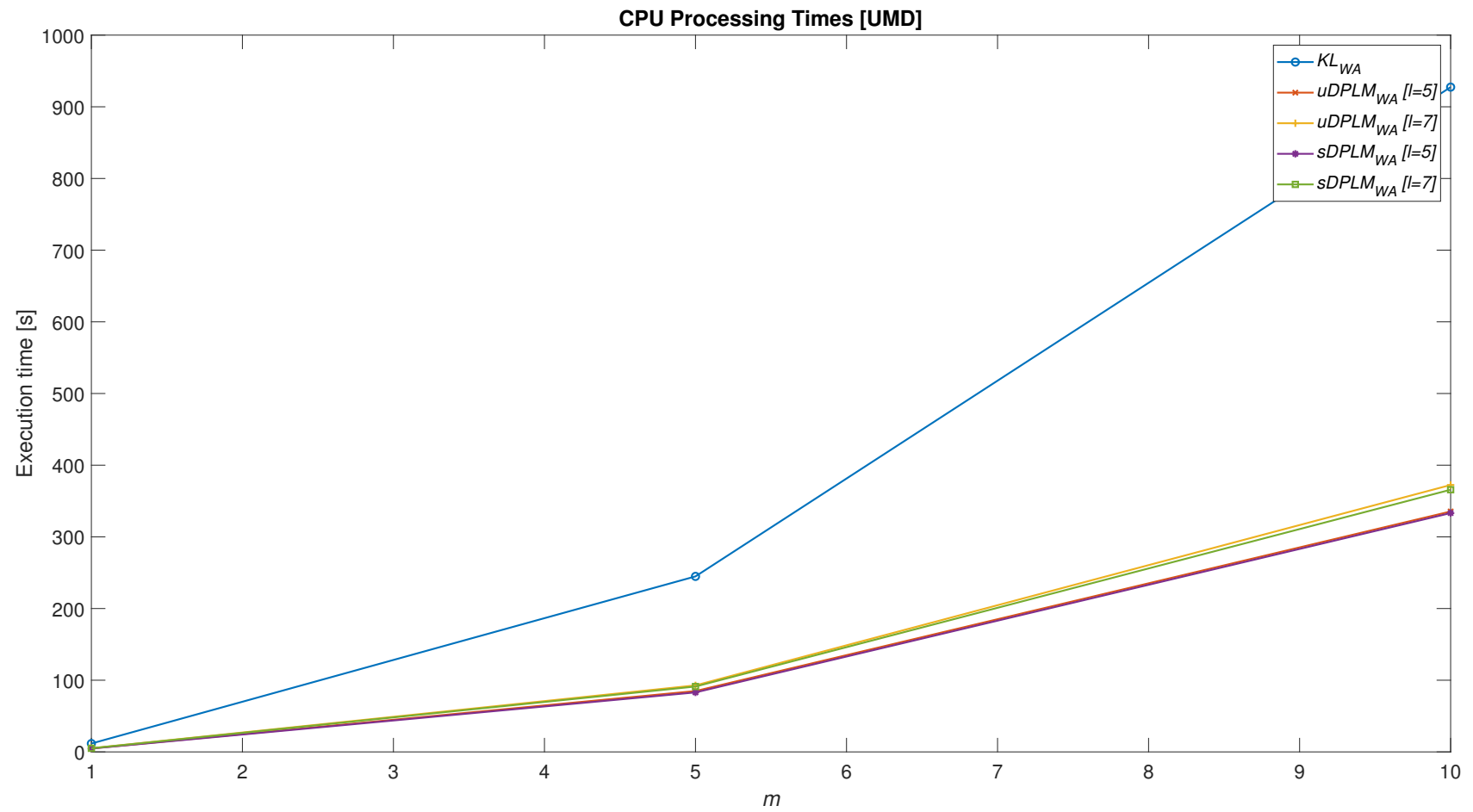


Figure 8. CPU processing times for the UMD database, $DPLM_{WA}$ vs. KL_{WA} .

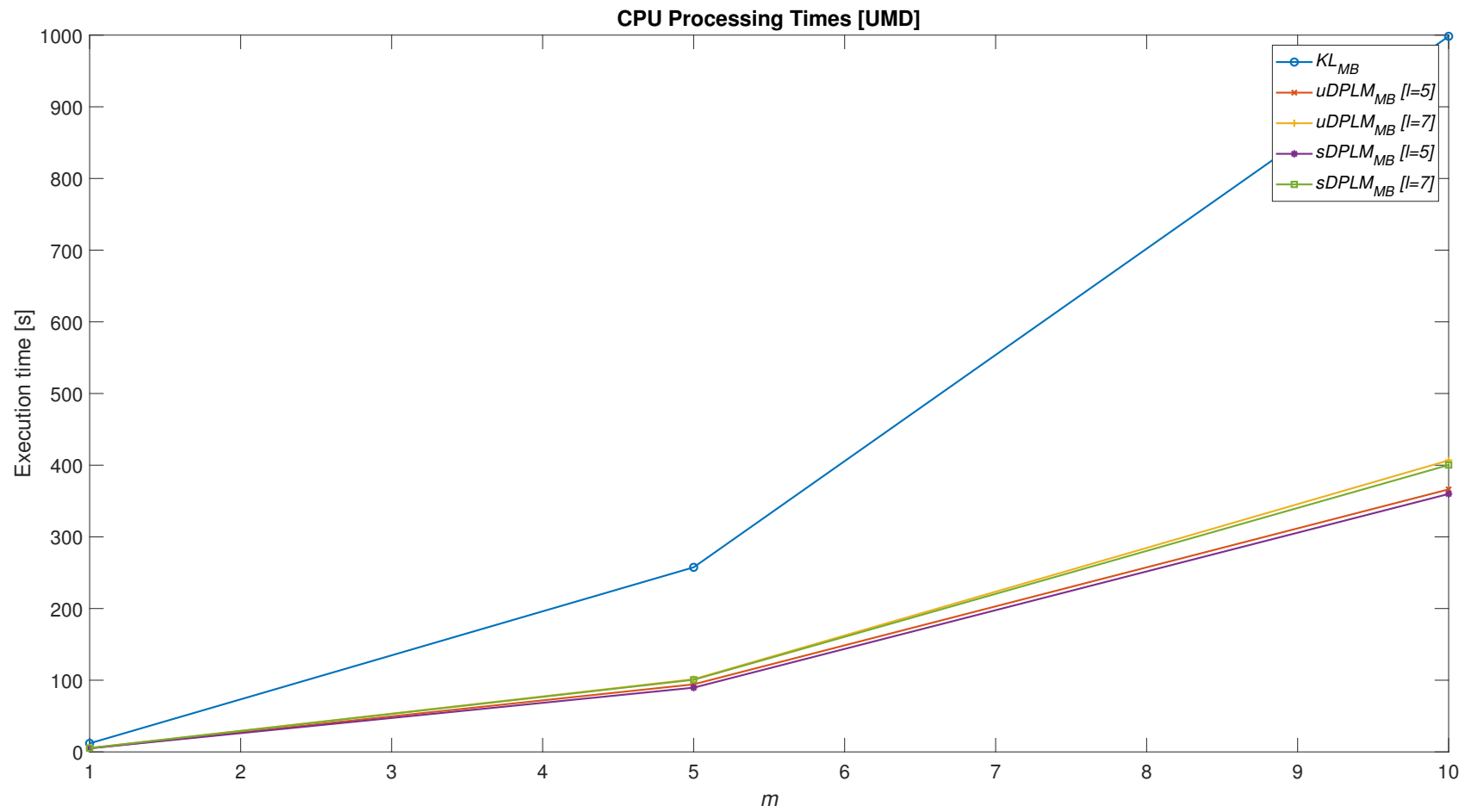


Figure 9. CPU processing times for the UMD database, $DPLM_{MB}$ vs. KL_{MB} .

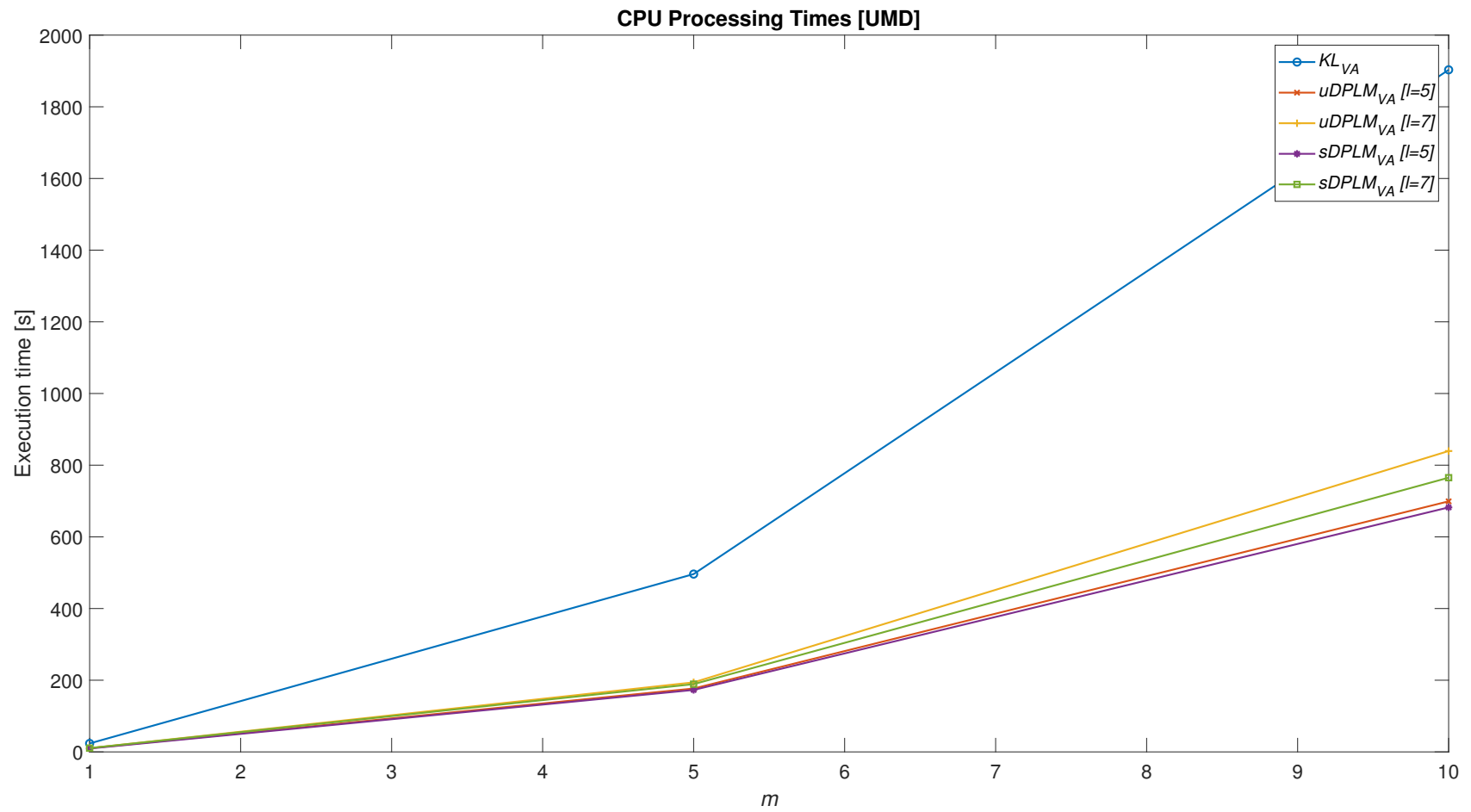


Figure 10. CPU processing times for the UMD database, $DPLM_{VA}$ vs. KL_{VA} .

4. Conclusions

A novel measure of similarity between GMMs based on a geometry-aware dimensionality reduction algorithm applied over SPD representations of the original data is proposed in the paper. The original feature space is projected into a low-dimensional feature space, therefore reducing the computational complexity. The method was successfully evaluated within various texture recognition tasks. Superior results have been achieved in terms of the trade-off between speed and accuracy in comparison to all baseline measures. We believe that the proposed method could also be successfully applied to other classification and recognition tasks dealing with high-dimensional Gaussian representations of underlying data.

Author Contributions: Conceptualization, M.J. and L.K.; methodology, B.P. and M.J.; software, B.P.; validation, B.P. and M.J.; formal analysis, L.K. and M.J.; investigation, B.P.; resources, B.P. and V.D.; data curation, B.P.; writing—original draft preparation, B.P. and N.S.; writing—review and editing, M.J., L.K., V.D. and N.S.; visualization, B.P.; supervision, M.J.; project administration, V.D. and B.P.; funding acquisition, V.D. All authors have read and agreed to the published version of the manuscript.

Funding: This research was supported by the Science Fund of the Republic of Serbia, #6524560, AI-S-ADAPT, and by the Serbian Ministry of Education, Science and Technological Development through the project no. 45103-68/2020-14/200156: “Innovative Scientific and Artistic Research from the Faculty of Technical Sciences Activity Domain”.

Data Availability Statement: Publicly available datasets were analyzed in this study. UIUC data description can be found in [30], and data can be found here: (http://www.cvr.ai.uiuc.edu/ponce_grp, accessed on 1 September 2016). KTH-TIPS data description can be found in [31], and data can be found here: (<https://www.csc.kth.se/cvap/databases/kth-tips>, accessed on 1 October 2022). UMD data description can be found in [32], and data can be found here: (<http://users.umiaccs.umd.edu/~fer/website-texture/texture.htm>, accessed on 3 December 2022).

Conflicts of Interest: The authors declare no conflict of interest. The funders had no role in the design of the study; in the collection, analyses or interpretation of data; in the writing of the manuscript; or in the decision to publish the results.

Abbreviations

The following abbreviations are used in this manuscript:

DPLM	Distance Preservation to the Local Mean
EM	Expectation Maximization
EMD	Earth Mover’s Distance
GMM	Gaussian Mixture Model
GQFD	Gaussian Quadratic Form Distance
KDE	Kernel Density Estimation
KL	Kullback–Leibler
KNN	K-Nearest Neighbors
KTH	Royal Institute of Technology in Stockholm
LDA	Linear Discriminant Analyses
LE	Laplacian Eigenmaps
LLE	Locally Linear Embedding
LPP	Locality Preserving Projections
MB	Matching-Based
MC	Monte Carlo
MMC	Maximum Margin Criterion
NPE	Neighborhood Preserving Embedding
PCA	Principal Component Analysis
PGA	Principal Geodesic Analysis
SPD	Symmetric Positive Definite
TIPS	Textures under varying Illumination, Pose and Scale
UC	Unscented Transform
UIUC	University of Illinois Urbana-Champaign
UMD	University of Maryland

VA	Variational Approximation
WA	Weighted Average

References

1. Kaur, A.; Sachdeva, R.; Singh, A. Classification Approaches for Automatic Speech Recognition System. In *Artificial Intelligence and Speech Technology*; CRC Press: Boca Raton, FL, USA, 2021; pp. 1–7, ISBN 1-00-315066-7.
2. Demir, K.E.; Eskil, M.T. Improved Microphone Array Design with Statistical Speaker Verification. *Appl. Acoust.* **2021**, *175*, 107813. [[CrossRef](#)]
3. Gangodkar, D. A Novel Image Retrieval Technique Based on Semi Supervised Clustering. *Multimed. Tools Appl.* **2021**, *80*, 35741–35769.
4. Asheri, H.; Hosseini, R.; Araabi, B.N. A New EM Algorithm for Flexibly Tied GMMs with Large Number of Components. *Pattern Recognit.* **2021**, *114*, 107836. [[CrossRef](#)]
5. Scavone, G.; Smith, J.O. A Landmark Article on Nonlinear Time-Domain Modeling in Musical Acoustics. *J. Acoust. Soc. Am.* **2021**, *150*, R3–R4. [[CrossRef](#)]
6. Yücesoy, E. Two-Level Classification in Determining the Age and Gender Group of a Speaker. *Int. Arab J. Inf. Technol.* **2021**, *18*, 663–670. [[CrossRef](#)]
7. Salamzadeh, A.; Ebrahimi, P.; Soleimani, M.; Fekete-Farkas, M. Grocery Apps and Consumer Purchase Behavior: Application of Gaussian Mixture Model and Multi-Layer Perceptron Algorithm. *J. Risk Financ. Manag.* **2022**, *15*, 424. [[CrossRef](#)]
8. Xu, J.; Zhang, Y.; Wang, D.; Dai, H. A Novel Structural Reliability Method on the Basis of Gaussian Mixture and Scaled Unscented Transformation. *J. Eng. Mech.* **2021**, *147*, 04021110. [[CrossRef](#)]
9. Lou, J. Crawling Robot Manipulator Tracking Based on Gaussian Mixture Model of Machine Vision. *Neural Comput. Appl.* **2022**, *34*, 6683–6693. [[CrossRef](#)]
10. Narasimhan, H.; Vinayakumar, R.; Mohammad, N. Unsupervised Deep Learning Approach for In-Vehicle Intrusion Detection System. *IEEE Consum. Electron. Mag.* **2021**. [[CrossRef](#)]
11. Vangi, E.; D’Amico, G.; Francini, S.; Giannetti, F.; Lasserre, B.; Marchetti, M.; Chirici, G. The New Hyperspectral Satellite PRISMA: Imagery for Forest Types Discrimination. *Sensors* **2021**, *21*, 1182. [[CrossRef](#)]
12. Kim, T.; Oh, J.; Kim, N.; Cho, S.; Yun, S.-Y. Comparing Kullback-Leibler Divergence and Mean Squared Error Loss in Knowledge Distillation. *arXiv* **2021**, arXiv:2105.08919.
13. Shapiro, A. Monte Carlo Sampling Methods. *Handbooks Oper. Res. Manag. Sci.* **2003**, *10*, 353–425.
14. Hershey, J.R.; Olsen, P.A. Approximating the Kullback Leibler Divergence between Gaussian Mixture Models. In Proceedings of the 2007 IEEE International Conference on Acoustics, Speech and Signal Processing-ICASSP’07, Honolulu, HI, USA, 15–20 April 2007; Volume 4, pp. IV-317–IV-320.
15. Beecks, C.; Ivanescu, A.M.; Kirchhoff, S.; Seidl, T. Modeling Image Similarity by Gaussian Mixture Models and the Signature Quadratic Form Distance. In Proceedings of the 2011 International Conference on Computer Vision, Barcelona, Spain, 6–13 November 2011; pp. 1754–1761.
16. Kristan, M.; Leonardis, A. Multivariate Online Kernel Density Estimation. In *Computer Vision Winter Workshop*; Verlag der Technischen Universität Graz: Graz, Österreich, 2010; pp. 77–86.
17. Lovrić, M.; Min-Oo, M.; Ruh, E.A. Multivariate Normal Distributions Parametrized as a Riemannian Symmetric Space. *J. Multivar. Anal.* **2000**, *74*, 36–48. [[CrossRef](#)]
18. Li, P.; Wang, Q.; Zhang, L. A Novel Earth Mover’s Distance Methodology for Image Matching with Gaussian Mixture Models. In Proceedings of the IEEE International Conference on Computer Vision, Sydney, NSW, Australia, 1–8 December 2013; pp. 1689–1696.
19. Hao, H.; Wang, Q.; Li, P.; Zhang, L. Evaluation of Ground Distances and Features in EMD-Based GMM Matching for Texture Classification. *Pattern Recognit.* **2016**, *57*, 152–163. [[CrossRef](#)]
20. Pont, M.; Vidal, J.; Tierny, J. Principal Geodesic Analysis of Merge Trees (and Persistence Diagrams). *IEEE Trans. Vis. Comput. Graph.* **2022**, 1–16. [[CrossRef](#)]
21. Singh, S. Topological Clustering on Riemannian Manifold. Ph.D. Thesis, Indian Institute of Science Education and Research, Pune, India, 2022.
22. Chen, B.; Gao, Y.; Wu, S.; Pan, J.; Liu, J.; Fan, Y. Soft Adaptive Loss Based Laplacian Eigenmaps. *Appl. Intell.* **2022**, *52*, 321–338. [[CrossRef](#)]
23. Lu, X.; Long, J.; Wen, J.; Fei, L.; Zhang, B.; Xu, Y. Locality Preserving Projection with Symmetric Graph Embedding for Unsupervised Dimensionality Reduction. *Pattern Recognit.* **2022**, *131*, 108844. [[CrossRef](#)]
24. Chu, L.; Wang, R.; Wu, X.-J. Collaborative Representation for SPD Matrices with Application to Image-Set Classification. *arXiv* **2022**, arXiv:2201.08962.
25. Izenman, A.J. Introduction to Manifold Learning. *Wiley Interdiscip. Rev. Comput. Stat.* **2012**, *4*, 439–446. [[CrossRef](#)]
26. He, X.; Cai, D.; Yan, S.; Zhang, H.-J. Neighborhood Preserving Embedding. In Proceedings of the Tenth IEEE International Conference on Computer Vision (ICCV’05), Beijing, China, 17–21 October 2005; Volume 2, pp. 1208–1213.
27. Krstanović, L.; Ralević, N.M.; Zlokolica, V.; Obradović, R.; Mišković, D.; Janev, M.; Popović, B. GMMs Similarity Measure Based on LPP-like Projection of the Parameter Space. *Expert Syst. Appl.* **2016**, *66*, 136–148. [[CrossRef](#)]

28. Popović, B.; Cepova, L.; Cep, R.; Janev, M.; Krstanović, L. Measure of Similarity between GMMs by Embedding of the Parameter Space That Preserves KL Divergence. *Mathematics* **2021**, *9*, 957. [[CrossRef](#)]
29. Davoudi, A.; Ghidary, S.S.; Sadatnejad, K. Dimensionality Reduction Based on Distance Preservation to Local Mean for Symmetric Positive Definite Matrices and Its Application in Brain–Computer Interfaces. *J. Neural Eng.* **2017**, *14*, 036019. [[CrossRef](#)] [[PubMed](#)]
30. Lazebnik, S.; Schmid, C.; Ponce, J. A Sparse Texture Representation Using Local Affine Regions. *IEEE Trans. Pattern Anal. Mach. Intell.* **2005**, *27*, 1265–1278. [[CrossRef](#)] [[PubMed](#)]
31. Fritz, M.; Hayman, E.; Caputo, B.; Eklundh, J.-O. The Kth-Tips Database. 2004. Available online: <https://www.csc.kth.se/cvap/databases/kth-tips/doc/> (accessed on 1 October 2022).
32. Xu, Y.; Ji, H.; Fermüller, C. Viewpoint Invariant Texture Description Using Fractal Analysis. *Int. J. Comput. Vis.* **2009**, *83*, 85–100. [[CrossRef](#)]
33. Joyce, J.M. Kullback-Leibler Divergence. In *International Encyclopedia of Statistical Science*; Springer: Berlin/Heidelberg, Germany, 2011; pp. 720–722.
34. Goldberger, J.; Gordon, S.; Greenspan, H. An Efficient Image Similarity Measure Based on Approximations of KL-Divergence Between Two Gaussian Mixtures. In *Proceedings of the ICCV, Nice, France, 13–16 October 2003*; IEEE Computer Society: Washington, DC, USA, 2003; Volume 3, pp. 487–493.
35. Goldberger, J.; Aronowitz, H. A Distance Measure between GMMs Based on the Unscented Transform and Its Application to Speaker Recognition. In *Proceedings of the INTERSPEECH, Lisbon, Portugal, 4–8 September 2005*; pp. 1985–1988.
36. Durrieu, J.-L.; Thiran, J.-P.; Kelly, F. Lower and Upper Bounds for Approximation of the Kullback-Leibler Divergence between Gaussian Mixture Models. In *Proceedings of the 2012 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), Kyoto, Japan, 25–30 March 2012*; pp. 4833–4836.
37. Lee, J.M. *Introduction to Riemannian Manifolds*; Springer: Cham, Switzerland, 2018; Volume 176.
38. Pennec, X.; Fillard, P.; Ayache, N. A Riemannian Framework for Tensor Computing. *Int. J. Comput. Vis.* **2006**, *66*, 41–66. [[CrossRef](#)]
39. Cherian, A.; Sra, S.; Banerjee, A.; Papanikolopoulos, N. Efficient Similarity Search for Covariance Matrices via the Jensen-Bregman LogDet Divergence. In *Proceedings of the 2011 International Conference on Computer Vision, Barcelona, Spain, 6–13 November 2011*; pp. 2399–2406.
40. Probert, R.L. On the Complexity of Matrix Multiplication. Ph.D. Thesis, University of Waterloo, Dept. of Applied Analysis and Computer Science, Waterloo, ON, Canada, 1973.
41. Sivalingam, R.; Boley, D.; Morellas, V.; Papanikolopoulos, N. Tensor Sparse Coding for Region Covariances. In *European Conference on Computer Vision*; Springer: Berlin/Heidelberg, Germany, 2010; pp. 722–735.
42. Webb, A.R. *Statistical Pattern Recognition*; John Wiley & Sons: Hoboken, NJ, USA, 2003; ISBN 0-470-85478-2.
43. Stai, E.; Kafetzoglou, S.; Tsiropoulou, E.E.; Papavassiliou, S. A Holistic Approach for Personalization, Relevance Feedback & Recommendation in Enriched Multimedia Content. *Multimed. Tools Appl.* **2018**, *77*, 283–326.

Disclaimer/Publisher’s Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.