



Article Data-Weighted Multivariate Generalized Gaussian Mixture Model: Application to Point Cloud Robust Registration

Bingwei Ge, Fatma Najar and Nizar Bouguila *D

Concordia Institute for Information Systems Engineering, Concordia University, 1515 St. Catherine Street West, Montreal, QC H3G 2W1, Canada; bi_ge@encs.concordia.ca (B.G.); f_najar@encs.concordia.ca (F.N.) * Correspondence: nizar.bouguila@concordia.ca

Abstract: In this paper, a weighted multivariate generalized Gaussian mixture model combined with stochastic optimization is proposed for point cloud registration. The mixture model parameters of the target scene and the scene to be registered are updated iteratively by the fixed point method under the framework of the EM algorithm, and the number of components is determined based on the minimum message length criterion (MML). The KL divergence between these two mixture models is utilized as the loss function for stochastic optimization to find the optimal parameters of the transformation model. The self-built point clouds are used to evaluate the performance of the proposed algorithm on rigid registration. Experiments demonstrate that the algorithm dramatically reduces the impact of noise and outliers and effectively extracts the key features of the data-intensive regions.

Keywords: multivariate generalized Gaussian; weighted-data clustering; minimum message length; point set robust registration; KL divergence; stochastic optimization



Citation: Ge, B.; Najar, F.; Bouguila, N. Data-Weighted Multivariate Generalized Gaussian Mixture Model: Application to Point Cloud Robust Registration. *J. Imaging* 2023, *9*, 179. https://doi.org/10.3390/ jimaging9090179

Academic Editor: Antonio Fernández-Caballero

Received: 31 July 2023 Revised: 24 August 2023 Accepted: 28 August 2023 Published: 31 August 2023



Copyright: © 2023 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

The purpose of point cloud registration is to extract the key points or features corresponding to the target point set and the point set to be registered as well as find the transformation mapping relationship between two point sets [1–5]. This task involving image processing, data analysis and computer vision has essential applications in many practical scenarios.

For instance, point cloud registration is essential for driverless technology. Indeed, various hardware sensors, such as lidars, short-wave radars and depth-of-field cameras, could be mounted on the crew-less vehicle and point cloud registration technology can be used to fuse data collected from multiple sensors [6-8] to provide fundamental functions such as scene stitching, vehicle positioning [9], and typical scene recognition and matching for vehicle control strategies. For example, the authors in [10] proposed a framework used for unmanned vehicles based on end-to-end point cloud registration deep networks. They obtained the corresponding relationship by learned matching probabilities (LMP) among a group of candidate points related to static characteristics instead of using existing points. Point cloud registration is also applied in medical imaging [11,12]. Indeed, to facilitate the diagnosis of the disease, several medical images from different instruments, such as Positron Emission Tomography (PET), Computed Tomography (CT) and Magnetic Resonance Imaging (MRI), need to be combined [13]. For example, authors in [14] improved the popular iterative Closest Point (ICP) algorithm by combining 3D scale-invariant feature transform to register 3D free-form closed surfaces (human skull models). In another work, the authors in [15] used the Gaussian mixture model (GMM) with a semi-supervised EM algorithm and geometric constraints to achieve retinal image registration. Moreover, 3D reconstruction also makes extensive use of point cloud registration technology [16,17]. For large buildings, for example, general scanning equipment cannot complete the whole scanning process at one time because the scanning range is limited. It requires scanning

multiple parts and then splicing point clouds together [18]. In other cases, the objects to be observed may be dynamic or have complex surface characteristics. The accuracy of these features plays a crucial role in modelling and analysis; repeated scanning to build a fusion model can improve details [19].

Considering the point set's acquisition perspective, noise and outliers generated in the acquisition process, as well as the deformation and missing parts of the point set caused by other factors, point cloud registration is a challenging task [1]. Various methods are proposed to enhance the robustness and accuracy of point cloud registration. In terms of pairwise registration, considering only two point sets, there are three main registration categories: distance-based methods including ICP [20], Graph Matching (GM) [21], filter-based methods and probability-based methods [22].

However, most point-to-point methods are prone to fall into local optima, especially if there are some similar point structure blocks in the point set. To improve this situation, registration based on mixed models (most are GMM-based) has proven effective [22–25]. The core idea is to model and describe the probability distribution of the point set using a parameterized mixed model and find the closest response of the mixture model to determine the corresponding relationship between point sets. These models perform well even if the sampling rate of two point clouds is not the same.

Nonetheless, there are two evident deficiencies when using the GMM. First, the GMM cannot effectively describe certain non-Gaussian distributions, such as the typical peak-trailing distributions in signal processing [26–29]. In the point cloud, intuitively speaking, space with dense data will carry more information. These high-density areas may represent the crucial feature structures in the point cloud, yet the GMM cannot effectively fit these high-density blocks, and its results tend to be relatively average. Secondly, GMM is easy to be disturbed by noise. Different noise levels will result in divergent response parameters in the mixture model, which could compromise the final registration accuracy [30].

The goal of this paper is to propose a point cloud registration method based on the weighted multivariate generalized Gaussian mixture model (WMGGMM) that we develop in this paper to address the difficulties above. The generalized Gaussian distribution (GGD) belongs to the family of elliptic distributions. Due to the addition of a shape parameter, it has a strong ability to describe various data distributions, particularly considering the data peak [31,32]. Its special cases contain Gaussian and Laplace distributions, and therefore it is widely used in feature extraction [33–35] and texture retrieval [36–38]. The mixture model of GGD has also been applied in several applications such as image processing and segmentation [39–41] and human movement recognition [42,43].

We show that the generalized Gaussian mixture model (GGMM) is an alternative worthy choice for point cloud registration when real-time analysis is not required. Although most parameters of GGMM have no closed solutions, its offers high registration accuracy and robustness. In addition, we introduce weights to reinforce the ability to pay attention to dense areas and reduce the influence of noise and outliers on the parameter estimation process. After obtaining the GGMM models for the target scene and the scene to be registered, the approximate Kullback–Leibler divergence (KLD) is computed to measure the models' difference. This will be used as a loss function to find the optimal registration parameters through the stochastic optimization algorithm.

The paper is organized as follows. Section 2 will tackle the WMGGMM parameter estimation. This section will also give the complete learning algorithm. Section 3 presents some experimental results conducted on a synthetic data set to verify the algorithm's performance and robustness. It is also devoted to our approach to obtain the optimal registration parameters using stochastic optimization, and the final registration effect performed on rigid transformation is presented. Conclusions and future works are finally reported in Section 4.

2. Weighted Multivariate Generalized Gaussian Mixture Model

In the majority of existing works, the feature independency assumption is used to simplify modeling high-dimensional data, which results in a distribution which is a product of one-dimensional generalized Gaussians [41,42]. Unlike these works, we use here the PDF as defined in [31]:

$$p(x;\mu,\Sigma,\beta,m) = \frac{\Gamma\left(\frac{d}{2}\right)\beta}{\Gamma\left(\frac{d}{2\beta}\right)\pi^{\frac{d}{2}}2^{\frac{d}{2\beta}}m^{\frac{d}{2}}|\Sigma|^{\frac{1}{2}}}e^{-\frac{1}{2m^{\beta}}\left((x-\mu)^{T}\Sigma^{-1}(x-\mu)\right)^{\beta}}$$
(1)

where $\Gamma(\cdot)$ denotes the Gamma function; $x, \mu \in \mathbb{R}^d$ and μ are the mean vectors; Σ is a $d \times d$ real symmetrical positive definite matrix called the scatter matrix; and m and $\beta > 0$ are the scale and shape parameters of the MGGD. The shape parameter controls the peak's sharpness and tail extension in the probability density function. It is worth noting that when $\beta = 1$, the MGGD becomes the multivariate Gaussian distribution. If $\beta < 1$, the distribution will have a sharper peak and a larger tail. However, when β tends to infinity, MGGD is close to a multivariate uniform distribution [31].

2.1. Data Weighting

Recent research has shown that data weighting could improve modeling capabilities (see, for example, [44] and references therein). Here, we follow these works via an extension to GGMM-based modeling. As proposed in [44], for example, each sample obtains a corresponding weight *w* greater than 0. If we consider the likelihood for a specific sample *x*, we can write it as $p(x; \mu, \Sigma, \beta, m)^w$. This is not a probability distribution because the integral is not equal to one. However, we notice that $p(x; \mu, \Sigma, \beta, m)^w \propto p(x; \mu, \Sigma, \beta, mw^{-1/\beta})$, and thus we can then obtain the PDF with weight as a parameter:

$$\hat{p}(x;\theta,w) = p(x;\mu,\Sigma,\beta,mw^{-1/\beta})$$
⁽²⁾

where $\theta = {\mu, \Sigma, \beta, m}$. It can be seen that individual weight directly influences *m* in the PDF here. Still, in parameter estimation, the weights will simultaneously affect the scale and shape in the final mixture. Having the new distribution in Equation (2) in hand, we can obtain the *K*-component mixture:

$$\widetilde{p}(x;\Theta,w) = \sum_{k=1}^{K} \pi_k p(x;\mu_k,\Sigma_k,\beta_k,m_k w^{-1/\beta_k})$$
(3)

where $\Theta = \{\pi_1, ..., \pi_K, \theta_1, ..., \theta_K\}$ denotes the parameter set of the model, and $(\pi_1, ..., \pi_K)$ are the mixing coefficients which satisfy $\pi_k > 0$ and $\sum_{k=1}^K \pi_k = 1$. $\theta_k = (\mu_k, \sum_k, \beta_k, m_k)$ are the parameters of the k^{th} component. Let $X = \{x_1, ..., x_N\}$ represent the whole data set and $W = \{w_1, ..., w_N\}$ is the weight set, then the log-likelihood function is given by:

$$L(X|\Theta, W) = \sum_{i=1}^{N} \ln \left[\sum_{k=1}^{K} \pi_k p(x_i; \mu_k, \Sigma_k, \beta_k, m_k w_i^{-1/\beta_k}) \right]$$
(4)

2.2. MGGD with Fixed Weights

For maximum likelihood estimation, the missing variables $Z = \{z_1, ..., z_N\}$ are introduced. If x_i is generated by the *k*th component, then $z_i = k$. We assume that the weights are already known by prior knowledge, so the expected complete data log-likelihood (*Q* function) can be written as:

$$Q_{c}\left(\Theta,\Theta^{(r)}\right) = \mathbb{E}_{p\left(Z|X;W,Q^{(r)}\right)}\left[\ln P(X,Z;W,\Theta)\right]$$

$$\stackrel{\Theta}{=} \sum_{i=1}^{N} \sum_{k=1}^{K} \eta_{ik} \left[\ln \pi_{k} + \ln \beta_{k} + \frac{d}{2\beta_{k}}\ln w_{i} - \ln \Gamma\left(\frac{d}{2\beta_{k}}\right) - \frac{d}{2\beta_{k}}\ln 2\right]$$

$$- \frac{d}{2}\ln m_{k} - \frac{1}{2}\ln|\Sigma_{k}| - \frac{w_{i}\left[\left(x_{i} - \mu_{k}\right)^{T}\Sigma_{k}^{-1}\left(x_{i} - \mu_{k}\right)\right]^{\beta_{k}}}{2m_{k}^{\beta_{k}}}\right]$$
(5)

where $E_P[\cdot]$ is the expectation with respect to the distribution P and $\stackrel{\Theta}{=}$ represents items only related to Θ . Subsequently, the optimal parameters Θ^* can be obtained by the EM algorithm. In the expectation step, the posteriors are updated with:

$$\eta_{ik} = p(z_i = k \mid x_i; \Theta, w_i) = \frac{\hat{p}(x_i; \theta_k, w_i)}{\sum_{k=1}^{K} \pi_k \hat{p}(x_i; \theta_k, w_i)}$$
(6)

By taking the derivatives of the complete data log-likelihood with respect to the parameters and making the results equal to zero, we can obtain the parameter update formulas:

$$\pi_k = \frac{1}{N} \sum_{i=1}^N \eta_{ik} \tag{7}$$

$$m_{k} = \left[\frac{\beta_{k}}{d\sum_{i=1}^{N}\eta_{ik}}\sum_{i=1}^{N}\eta_{ik}w_{i}\left[(x_{i}-\mu_{k})^{T}\Sigma_{k}^{-1}(x_{i}-\mu_{k})\right]^{\beta_{k}}\right]^{\frac{1}{\beta_{k}}}$$
(8)

$$\mu_{k} = \frac{\sum_{i=1}^{N} \eta_{ik} w_{i} \Big[(x_{i} - \mu_{k})^{T} \Sigma_{k}^{-1} (x_{i} - \mu_{k}) \Big]^{\beta_{k} - 1} x_{i}}{\sum_{i=1}^{N} \eta_{ik} w_{i} \Big[(x_{i} - \mu_{k})^{T} \Sigma_{k}^{-1} (x_{i} - \mu_{k}) \Big]^{\beta_{k} - 1}}$$
(9)

$$\Sigma_{k} = \frac{\beta_{k}}{m_{k}^{\beta_{k}} \sum_{i=1}^{N} \eta_{ik}} \sum_{i=1}^{N} \left[\frac{\eta_{ik} w_{i} (x_{i} - \mu_{k}) (x_{i} - \mu_{k})^{T}}{\left[(x_{i} - \mu_{k})^{T} \Sigma_{k}^{-1} (x_{i} - \mu_{k}) \right]^{1 - \beta_{k}}} \right]$$
(10)

After replacing m_k in Equation (10) with the previous result in Equation (8), we can find that Σ_k is the independent form m_k . If we let $y_i = (x_i - \mu_k)^T \Sigma_k^{-1} (x_i - \mu_k)$, we have:

$$\Sigma_{k} = \frac{d}{\sum_{i=1}^{N} \left(\eta_{ik} w_{i} y_{i}^{\beta_{k}}\right)} \sum_{i=1}^{N} \left[\frac{\eta_{ik} w_{i} (x_{i} - \mu_{k}) (x_{i} - \mu_{k})^{T}}{y_{i}^{1 - \beta_{k}}}\right]$$
(11)

Furthermore, it is worth noting that μ_k and Σ_k do not have closed solutions; they are both solved through the fixed point (FP) method. According to Banach's fixed point theorem [45], if (S, d) is a non-empty complete metric space with a contraction mapping $T: S \to S$, then there exists a unique fixed point S^* in S. The authors in [31] proved the convergence of Σ_k and explained the existence and uniqueness of the fixed point, based on the fact that $\beta \in (0, 1]$ and Σ is a positive definite real symmetric matrix. This is also consistent with our assumptions about β . We introduce the Frobenius norm defined in Equation (12) to measure the difference between S_n and S_{n-1} in the fixed point iteration. The process stops when the approximate solution satisfies the preset precision. Furthermore, to ensure the convergence of the FP equation, the value range of weights should also be between zero and one.

$$\|S_n - S_{n-1}\|_F = \sqrt{\sum_{i=1}^m \sum_{j=1}^n \left|s_n^{ij} - s_{n-1}^{ij}\right|^2}$$
(12)

Then parameter β can be estimated using Newton–Raphson iterations [42–44]:

$$\beta_k^{(t+1)} = \beta_k^{(t)} - \xi \frac{f\left(\beta_k^{(t)}\right)}{f'\left(\beta_k^{(t)}\right)}$$
(13)

where ξ is the learning rate used to prevent the oscillation and overflow in the iterative process, and it is usually around 0.1. If necessary, the method of exponential decay can be applied to make the convergence more stable. $f(\beta_k^{(t)})$ and $f'(\beta_k^{(t)})$ are given as follows:

$$f(\beta_{k}) = \frac{d\sum_{i=1}^{N} \eta_{ik}}{2\sum_{i=1}^{N} (\eta_{ik}w_{i}y_{j}^{\beta_{k}})} \sum_{i=1}^{N} \left[\eta_{ik}w_{i}y_{i}^{\beta_{k}}\ln(y_{i}) \right] \\ + \frac{d\sum_{i=1}^{N} (\eta_{ik}\ln w_{i})}{2\beta_{k}} - \frac{d\sum_{i=1}^{N} \eta_{ik} \left[\Psi\left(\frac{d}{2\beta_{k}}\right) + \ln 2 \right]}{2\beta_{k}}$$
(14)
$$- \sum_{i=1}^{N} \eta_{ik} - \frac{d\sum_{i=1}^{N} \eta_{ik}}{2\beta_{k}} \ln \left[\frac{\beta_{k}}{d\sum_{i=1}^{N} \eta_{ik}} \sum_{i=1}^{N} \left(\eta_{ik}w_{i}y_{i}^{\beta_{k}} \right) \right] \\f'(\beta_{k}) = \frac{d\sum_{i=1}^{N} \eta_{ik}}{2\left(\sum_{i=1}^{N} \eta_{ik}w_{i}y_{i}^{\beta_{k}}\right)^{2}} \left[\left(\sum_{i=1}^{N} \eta_{ik}w_{i}y_{i}^{\beta_{k}} \right) \right] \\\left(\sum_{i=1}^{N} \eta_{ik}w_{i}y_{i}^{\beta_{k}}(\ln y_{i})^{2} \right) - \left(\sum_{i=1}^{N} \eta_{ik}w_{i}y_{i}^{\beta_{k}}\ln y_{i} \right)^{2} \right] \\- \frac{d\sum_{i=1}^{N} (\eta_{ik}\ln w_{i})}{2\beta_{k}^{2}} + \frac{d\sum_{i=1}^{N} \eta_{ik}}{2\beta_{k}^{2}} \left[\Psi\left(\frac{d}{2\beta_{k}}\right) + \ln 2 \right] \\+ \frac{d^{2}\sum_{i=1}^{N} \eta_{ik}}{4\beta_{k}^{3}} \Psi'\left(\frac{d}{2\beta_{k}}\right) + \frac{d\sum_{i=1}^{N} \eta_{ik}}{2\beta_{k}^{2}} \left[\ln\left(\frac{\beta_{k}}{dN}\right) \\+ \ln\left(\sum_{i=1}^{N} \eta_{ik}w_{i}y_{i}^{\beta_{k}}\right) - 1 - \beta_{k} \frac{\sum_{i=1}^{N} \eta_{ik}w_{i}y_{i}^{\beta_{k}}\ln y_{i}}{\sum_{i=1}^{N} \eta_{ik}w_{i}y_{i}^{\beta_{k}}} \right]$$

where $\Psi(\cdot)$ is the digamma function.

2.3. Weights Considered as Random Variables

Above, we have derived the WMGGMM parameter updating formulas with fixed weights. However, it is pointed out in [44] that the Bayesian formalism is more inclined to treat parameters as random variables and update the posterior of parameters by combining the parameters prior with the observed sample. Under this framework, the limitation of insufficient prior knowledge on the accuracy of weights is reduced, and the inference process of weights will also be more explanatory. As mentioned before, the generalized Gaussian distribution belongs to the family of elliptic distributions, and thus we select the same prior distribution, i.e., Gamma distribution, as in [44]. At this point, the prior and posterior of the parameters have the same form. The advantage is that when we make a new observation, we do not have to re-calculate the whole process but only directly obtain the posterior distribution through the parameters, which undoubtedly simplifies the updating process of weight parameters. Then, the posterior will become the prior in the next calculation, Therefore, we can obtain:

$$p(w;\phi) = G(w;a,b) = \Gamma(a)^{-1}b^{a}w^{a-1}e^{-bw}$$
(16)

where G(w; a, b) is the Gamma distribution, and $\phi = \{a, b\}$ are the parameters of the prior distribution of w. The mean and variance of random variable w are given by:

$$E[w] = a/b \tag{17}$$

$$Var[w] = a/b^2 \tag{18}$$

Due to the addition of prior parameters, the log-likelihood of the complete data becomes the following form:

$$Q_{c}\left(\Theta,\Theta^{(r)}\right) = \mathrm{E}_{p\left(Z,W|X;\Theta^{(r)},\Phi\right)}\left[\ln P(X,Z,W;\Theta,\Phi)\right]$$
(19)

where $\Phi = {\phi_1, ..., \phi_N}$ and $\phi_i = {a_i, b_i}$. The posterior distribution factorizes on *i* as follows:

$$P(Z, W \mid X; \Theta^{(r)}, \Phi) = \prod_{i=1}^{N} p(z_i, w_i \mid x_i; \Theta^{(r)}, \phi_i)$$
(20)

Each of these product terms can be expressed as two-factor expressions:

$$p(z_i, w_i \mid x_i; \Theta^{(r)}, \phi_i) = p(w_i \mid z_i, x_i; \Theta^{(r)}, \phi_i) p(z_i \mid x_i; \Theta^{(r)}, \phi_i)$$
(21)

According to the above formula, the expectation step in the EM algorithm is divided into two parts (E-Z step and E-W step). In the E-Z step, the marginal posterior distribution of z_i is obtained by integrating over w_i :

$$\eta_{ik} = \int p\left(z_i = k, w_i \mid x_i; \Theta^{(r)}, \phi_i\right) dw_i$$

$$\stackrel{k}{\propto} \int \pi_k p\left(x_i \mid z_i = k, w_i; \Theta^{(r)}\right) p(w_i; \phi_i) dw_i$$

$$= \int \pi_k \hat{p}(x_i; \theta_k, w_i) G(w_i; a_i, b_i) dw_i$$

$$\propto \pi_k \bar{p}(x_i; \mu_k, \Sigma_k, \beta_k, m_k, a_i, b_i)$$
(22)

where $\bar{p}(x; \mu, \Sigma, \beta, m, a, b)$ is given as:

$$\bar{p}(x;\mu,\Sigma,\beta,m,a,b) = \frac{\beta\Gamma\left(\frac{d}{2}\right)\Gamma\left(a+\frac{d}{2\beta}\right)}{(m\pi)^{\frac{d}{2}}(2b)^{\frac{d}{2\beta}}|\Sigma|^{\frac{1}{2}}\Gamma(a)\Gamma\left(\frac{d}{2\beta}\right)} \left[\frac{\left((x-\mu)^{T}\Sigma^{-1}(x-\mu)\right)^{\beta}}{2bm^{\beta}} + 1\right]^{-\left(a+\frac{d}{2\beta}\right)}$$
(23)

In step E-W, according to the property of conjugate distribution, we can obtain:

$$p(w_{i} \mid z_{i} = k, x_{i}; \Theta, \phi_{i}) \stackrel{w_{i}}{\propto} p(x_{i} \mid z_{i} = k, w_{i}; \Theta) p(w_{i}; \phi_{i})$$

= $\hat{p}(x_{i}; \theta_{k}, w_{i}) G(w_{i}; a_{i}, b_{i}) = G\left(w_{i}; a_{i}^{(r+1)}, b_{i}^{(r+1)}\right)$ (24)

Thus, the updating formulas of prior parameters can be obtained:

$$a_{ik}^{(r+1)} = a_i^{(0)} + \frac{d}{2\beta_k}$$
(25)

$$b_{ik}^{(r+1)} = b_i^{(0)} + \frac{\left[(x_i - \mu_k)^T \Sigma_k^{-1} (x_i - \mu_k) \right]^{\beta_k}}{2m_k^{\beta_k}}$$
(26)

$$\bar{w}_{ik} = \mathbf{E}_{P(w_i|z_i=k, x_i, \Theta^{(r)}, \phi_i)}[w_i] = \frac{a_{ik}^{(r+1)}}{b_{ik}^{(r+1)}}$$
(27)

This explains the outliers shielding feature of the weighted algorithm. As an outlier is by definition far from the center of all components, it has a low posterior weight \bar{w}_{ik} for each component and then a low mean posterior probability \bar{w}_i . By expanding Equation (19), we can obtain a result similar to the Q function with fixed weights:

$$Q_{c}\left(\Theta,\Theta^{(r)}\right) = \sum_{i=1}^{N} \sum_{k=1}^{K} \int_{w_{i}} \eta_{ik} \ln \pi_{k} \hat{p}(x_{i};\theta_{k},w_{i}) \times p\left(w_{i} \mid x_{i},z_{i}=k,\Theta^{(r)},\phi_{i}\right) dw_{i}$$

$$\stackrel{\Theta}{=} \sum_{i=1}^{N} \sum_{k=1}^{K} \eta_{ik} \left[\ln \pi_{k} + \ln \beta_{k} + \frac{d}{2\beta_{k}} \ln \bar{w}_{ik}^{(r+1)} - \ln \Gamma\left(\frac{d}{2\beta_{k}}\right) - \frac{d}{2\beta_{k}} \ln 2 \right]$$

$$- \frac{d}{2} \ln m_{k} - \frac{1}{2} \ln |\Sigma_{k}| - \frac{\bar{w}_{ik}^{(r+1)} \left[(x_{i}-\mu_{k})^{T} \Sigma_{k}^{-1} (x_{i}-\mu_{k})\right]^{\beta_{k}}}{2m_{k}^{\beta_{k}}} \right]$$
(28)

Therefore, w_i will be replaced with \bar{w}_{ik} in all the parameter updates formulas of the mixture model. Since the equations are very similar, they are not repeated here.

2.4. Automatic Determination of the Number of Components

The model selection problem is tackled using the minimum message length (MML) criterion as proposed in [46]:

$$\Theta_{MML} = \underset{\Theta}{\arg\min} \left\{ -\log P(\Theta) - Q_R \left(\Theta, \Theta^{(r+1)}\right) + \frac{1}{2} \log |I_c(\Theta)| + \frac{D(\Theta)}{2} \left(1 + \log \frac{1}{12}\right) \right\}$$
(29)

where $I_C(\Theta)$ denotes the expected complete Fisher information matrix (FIM) and $D(\Theta)$ is the dimensionality of the model. Using a similar process as in [46], we can show that

$$\Theta_{MML} = \underset{\Theta}{\arg\min} \left\{ \frac{M}{2} \sum_{k \in \mathbf{K}^+} \log \pi_k - Q_R \left(\Theta, \Theta^{(r+1)}\right) + \frac{K^+(M+1)}{2} \left(1 + \log \frac{n}{12}\right) \right\}$$
(30)

where \mathbf{K}^+ is the set of non-empty components and K^+ is the number of elements in it. Moreover, we can rewrite the formula for calculating π_k in the maximization step of the EM algorithm:

$$\pi_{k} = \frac{\max\left(0, \sum_{i=1}^{N} \eta_{ik} - MK^{+}/2\right)}{\sum_{l=1}^{K} \max\left(0, \sum_{i=1}^{N} \eta_{il} - MK^{+}/2\right)}$$
(31)

The threshold for minimum support is high when the number of non-empty components is large at the beginning. Some components can be removed quickly. In the process of updating components one by one, the threshold gradually approaches the situation in [46] as the number of non-empty components decreases.

2.5. Complete Algorithm

The complete steps are outlined in Algorithm 1. We use k-means to initialize π_k , μ_k , Σ_k . The initial parameter β_k is specified as 0.5, and the parameter m_k is calculated according to the pre-clustering results and the formula in [31]:

$$m_{k} = \left\{ \frac{\beta_{k}}{dN_{k}} \sum_{i=1}^{N_{k}} \left[(x_{i} - \mu_{k})^{T} \Sigma_{k}^{-1} (x_{i} - \mu_{k}) \right]^{\beta_{k}} \right\}^{\frac{1}{\beta_{k}}}$$
(32)

where $x_i \in X_k$ is the *i*th sample for the *k*th cluster, and N_k is the number of samples.

We adopt the same data similarity measurement method based on the Gaussian kernel as in [44] for the initialization of weights. However, due to the constraint of the weight range, we modify it as follows:

$$w_i = \frac{1}{q} \sum_{j \in S_i^q} exp\left[-\frac{d^2(x_i, x_j)}{\sigma}\right]$$
(33)

where $d^2(x_i, x_j)$ denotes the Euclidean distance, S_i^q is the set containing q nearest neighbors of x_i , and σ is a positive scale. The default setting for q is 20. We can then calculate the initialization of the prior parameters of the weight through Equations (17) and (18), i.e., $a_i = w_i^2$ and $b_i = w_i$.

Figure 1 shows the impact of different σ on the weights; its significance is to change the degree of differentiation of the weights. When there is a small σ , the difference in the weights between dense and sparse areas becomes more pronounced. Conversely, the distribution of weights will be relatively close. In other words, a smaller σ is more beneficial at removing outliers and noise. However, it is important to note that when the σ is too small, the weight operation is equivalent to removing most of the points that are away from the clustering center in the data distribution. The actual points involved in parameter identification are reduced, leading to inadequate support of the components. Furthermore, the resulting mixed model parameters will have a large deviation from the original data distribution. Therefore, the selection of σ is a balance between model robustness and model accuracy. We set it to 25 in our case.



Figure 1. The influence of different Gaussian kernel parameter σ on the weights. (a) $\sigma = 1$; (b) $\sigma = 5$; (c) $\sigma = 10$; (d) $\sigma = 50$.

Algorithm 1 Proposed WMGGMM algorithm with component-wise EM procedure.

Input: $X = \{x_i\}_{i=1}^N$; $\Phi^{(0)} = \{a_i^{(0)}, b_i^{(0)}\}_{i=1}^N$; K_{max} $\Theta^{(0)} = \{\pi_k^{(0)}, \mu_k^{(0)}, \Sigma_k^{(0)}, \beta_k^{(0)}, m_k^{(0)}\}_{k=1}^{K_{max}};$ **Output:** Optimal mixture model parameters: Θ^* Set: r = 0, $MML = +\infty$ repeat for k = 1 To K_{max} do E-Z step using (22): $\eta_{ik}^{(r+1)} = \frac{\pi_k^{(r)} \overline{p}(x_{i};\mu_k^{(r)},\Sigma_k^{(r)},\beta_k^{(r)},m_k^{(r)},a_{ik}^{(r)},b_{ik}^{(r)})}{\sum_{l=1}^{k_{max}} \pi_l^{(r)} \overline{p}(x_{i};\mu_l^{(r)},\Sigma_l^{(r)},\beta_l^{(r)},m_l^{(r)},a_{il}^{(r)},b_{il}^{(r)})}$ Compute the # of non-empty components: K^+ E-W step using (25)–(27): $a_{ik}^{(r+1)} = a_i^{(0)} + \frac{d}{2\beta_k}$ $b_{ik}^{(r+1)} = b_i^{(0)} + \frac{[(x_i - \mu_k)^T \Sigma_k^{-1} (x_i - \mu_k)]^{\beta_k}}{2m_k^{\beta_k}}$ $\overline{w}_{ik} = a_{ik}^{(r+1)} / b_{ik}^{(r+1)}$ I step: M step: $\pi_{k}^{(r+1)} = \frac{\max(0,\eta_{ik}^{(r+1)} - K^{+}M/2)}{\sum_{l=1}^{K_{max}} \max(0,\eta_{il}^{(r+1)} - K^{+}M/2)}$ if $\pi_{k}^{(r+1)} > 0$ then Üpdate μ_k using (9): repeat $\mu_k^{new} = T(\mu_k^{old})$ until $\|\mu_k^{new} - \mu_k^{old}\|_F < \epsilon$ $\mu_k^{(r+1)} = \mu_k^{new}$ Update Σ_k using (11): repeat
$$\begin{split} & \Sigma_k^{new} = T(\Sigma_k^{old}) \\ & \text{until } \|\Sigma_k^{new} - \Sigma_k^{old}\|_F < \epsilon \end{split}$$
 $\Sigma_k^{(r+1)} = \Sigma_k^{new}$ Update β_k using (13)–(15): repeat $\beta_k^{new} = \beta_k^{old} - \zeta \frac{f(\beta_k^{old})}{f'(\beta_k^{old})}$ **until** $|\mu_k^{new} - \mu_k^{old}| < \epsilon$ $\beta_k^{(r+1)} = \beta_k^{new}$ Update $m_k^{(r+1)}$ using (8) end if end for Compute $MML^{(r+1)}$ using (30) r = r + 1until $|\Delta MML^{(r)}| < \epsilon$ Return the parameters Θ of non-empty components as optimal mixture model parameters Θ^*

3. Experimental Results

3.1. Synthetic Data

First, we demonstrate the performance of the WMGGMM model through synthetic data. The method for generating data points that follow an MGGD comes from [31]:

$$\mathbf{x} \in \mathbb{R}^d \cong \tau C^{1/2} \mathbf{u} \tag{34}$$

where **x** is a random vector that follows an MGGD with scatter matrix $C = m\Sigma$ and shape parameter β , and \cong denotes equivalence on the distribution; **x** is a random vector uniformly distributed on a unit sphere and τ is a positive scalar random variable that satisfies $\tau^{2\beta} \sim \Gamma(d/(2\beta), 2)$.

Tables 1 and 2 show the parameter estimation results for two generated two-dimensional data sets from the 4- and 3-component mixture models, respectively, where $\rho_k = cov(X, Y) / \sqrt{var(X)var(Y)}$ represents the correlation coefficient used to measure the slope of the scatter matrix.

Default Mixture Model Parameters						
π_k	μ_k	β_k	C_k	$ ho_k$		
0.25 (N = 300)	[1 1]	0.85	$\begin{bmatrix} 3 & 1 \\ 1 & 5 \end{bmatrix}$	0.35		
0.25 (N = 300)	[15 2]	0.85	$\begin{bmatrix} 2 & 0 \\ 0 & 2 \end{bmatrix}$	0.00		
0.25 (N = 300)	[1 18]	0.85	$\begin{bmatrix} 3 & -2 \\ -2 & 4 \end{bmatrix}$	-0.58		
0.25 (N = 300)	[16 16]	0.85	$\begin{bmatrix} 3 & -1 \\ -1 & 3 \end{bmatrix}$	-0.33		
Estimated mixture model parameters						
π_k	μ_k	eta_k	C_k	$ ho_k$		
0.2444	[0.98 0.90]	0.72	$\begin{bmatrix} 1.43 & 0.57 \\ 0.57 & 2.94 \end{bmatrix}$	0.28		
0.2586	[15.11 2.05]	0.70	$\begin{bmatrix} 0.60 & -0.04 \\ -0.04 & 0.67 \end{bmatrix}$	-0.06		
0.2482	[1.15 17.99]	0.75	$\begin{bmatrix} 1.39 & -1.00 \\ -1.00 & 2.15 \end{bmatrix}$	-0.58		
0.2486	[15.98 15.99]	0.77	$\begin{bmatrix} 1.12 & -0.3 \\ -0.3 & 1.05 \end{bmatrix}$	-0.27		

Table 1. Estimation of the mixture model parameters for the first synthetic data set.

Table 2. Estimation of the mixture model parameters for the second synthetic data set.

Default Mixture Model Parameters						
π_k	μ_k	β_k	C_k	$ ho_k$		
0.25 (N = 300)	[8 16]	0.60	$\begin{bmatrix} 3 & -2 \\ -2 & 4 \end{bmatrix}$	-0.58		
0.25 (N = 300)	[15 2]	0.85	$\begin{bmatrix} 2 & 0 \\ 0 & 2 \end{bmatrix}$	0.00		
0.50 (N = 600)	[1 3]	0.85	$\begin{bmatrix} 3 & 1 \\ 1 & 5 \end{bmatrix}$	0.35		
Estimated mixture model parameters						
π_k	μ_k	β_k	C_k	$ ho_k$		
0.2596	[7.89 16.05]	0.76	$\begin{bmatrix} 2.27 & -1.34 \\ -1.34 & 3.39 \end{bmatrix}$	-0.48		
0.2817	[14.90 2.12]	0.67	$\begin{bmatrix} 0.62 & -0.17 \\ -0.17 & 0.78 \end{bmatrix}$	-0.24		
0.4587	[0.98 3.11]	0.70	$\begin{bmatrix} 0.91 & 0.61 \\ 0.61 & 2.12 \end{bmatrix}$	0.44		
0.4587	[0.98 3.11]	0.70	$\begin{bmatrix} 0.11 & 0.10 \\ 0.91 & 0.61 \\ 0.61 & 2.12 \end{bmatrix}$	0.44		

We can see that the estimated parameters are close to the real ones. According to the correlation coefficient comparisons, the slopes of the real and estimated scatter matrices are also close. Due to the role of weight, the points essentially involved in parameter estimation are concentrated in data-intensive areas, leading to a reduced scale parameter *m*. However, the change in parameter β is not necessarily a decrease. Suppose that β is small in the default setting. In the big trailing part, the data points are distributed sparsely. The weight

estimated β will become larger instead. Figure 2 presents the parameter estimation results at different noise levels. Parameter estimation is carried out after the proportional addition of random uniform noise in the primary distribution area ([-5.25, 5.25]) of the original data. When σ is appropriately selected, the weights remove most of the noise and outliers, and the actual points involved in parameter estimation are mostly in the desired original data distribution. The final results show that the mixture models of the three different cases are very similar, which effectively proves the WMGGMM algorithm's robustness.

operation will remove most of these points, so the shape of data is changed, and the final



Figure 2. Estimation results of the mixture model parameters with different noise levels: (**a**) 10% noise, (**b**) 20% noise, (**c**) 30% noise.

The results of the mixture model for overlapping data are shown in Figure 3. The introduction of shape parameter β enables MGGD to improve the identification ability of data distribution peaks. Although the overall distribution of two components overlap, the WMGGMM algorithm can still accurately identify the parameters of each component when their centers do not coincide.



Figure 3. The results of parameter estimation in the case of component overlap.

Figure 4 displays the execution time of the WMGGMM algorithm under different conditions (the default number of components is four). The average value of five independent running times is taken as the final result, and the unit is seconds. Because most of the algorithm's parameters need to be estimated iteratively (fixed-point equations and Newton–Raphson method), the algorithm's running time is much higher than that of the standard EM algorithm. The anti-interference performance of the model is improved by sacrificing some efficiency. The time here can only be used as a relative reference since the algorithm's run time is influenced by computer performance, hardware and software acceleration, and randomness of parameter initialization.



Figure 4. Performance analysis of WMGGMM for 2-D data.

3.2. Point Cloud Registration Using WMGGMM

We consider the approach proposed in [23], in the case of GMM, to perform WMGGMMbased registration. Two mixture models are adopted to describe the target scene and the scene to be registered. Then, the difference between these two models is applied to update the registration parameters. Nevertheless, the algorithm proposed in [23] is still a method based on point to point in essence. The data pre-clustering is not considered, but each point in the data set is used as a mean to initialize a mixture component. This approach is equivalent to converting the entire data set into a mixture model with multiple simple components. However, this method of initializing the mixture model is not suitable in the point cloud with a large data volume. The subsequent L2 divergence simplifies the derivation based on the transformation model. However, the derivative optimization process is closely related to the expression of the specific model. If the transformation model is changed, all optimization processes need to be re-derived. Therefore, we propose a method based on WMGGMM to pre-cluster the target scene and the scene to be registered and extract the key features to form the mixture models. Then, the optimal registration parameters are found by using stochastic optimization through the KLD between the mixture models. The KLD of the two statistical models is defined as follows:

$$KLD(f||g) = \int_{x} f(x) \log \frac{f(x)}{g(x)} dx$$
(35)

Unfortunately, the KLD between the mixture models has no analytical expression, so several approximations have been proposed [47–49]. Compared with GMM, the KLD of the generalized Gaussian mixture model using the inner product of the components is too complicated due to the introduction of shape parameters. Even though a method for calculating the KLD of two MGGDs is proposed in [38], the matched bound and variational approximations are also unfeasible because the premise of this approach is that all MGGDs have a zero mean. However, the mean value of components is significant in the point cloud

registration scenario. Therefore, we finally use Monte Carlo sampling to calculate the KLD between two GGMMs:

$$KLD_{MC}(P_S||P_M) = \frac{1}{n} \sum_{i=1}^{n} [\log P_S(x_i) - \log P_M(x_i)]$$
(36)

where $P_S(x)$ and $P_M(x)$ are the probability density functions of the GGMM obtained from the target scene and the scene to be registered, respectively, and $\{x_i\}_{i=1}^n$ denotes n samples taken from $P_S(x)$. In the above formula, the sum of the probability of samples replaces the integral, and the Monte Carlo method is the only method that can approximate the actual value of KLD when there are enough sample points. The Gibbs sampling, one of the Markov Chain Monte Carlo (MCMC) sampling techniques, is applied in this process using n = 1000.

We assume that the point cloud transformation model is:

$$X = f_T(X_0, \Omega) \tag{37}$$

where X_0 and X are the point sets before and after the transformation, and $\Omega = \{\omega_1, ..., \omega_m\}$ is the parameter set of the transformation model. Then, we can express the point cloud registration problem in the following form:

$$\Omega^* = \underset{\Omega}{\arg\min} \{ KLD_{MC}[ggmm(\mathbf{S}) \| ggmm(f_T(\mathbf{M}, \Omega))] \}$$
(38)

where **S** is the target scene point set, **M** is the point set to be registered, and $ggmm(\cdot)$ denotes the obtained PDF of the GGMM from a data set using the WMGGMM algorithm. The complete point cloud registration algorithm using WMGGMM is shown in Algorithm 2. However, it is only a general framework and does not specify a concrete random optimization method. The stochastic optimization technique can be selected depending on needs, as long as the definition of the loss function in (38) is satisfied, and the optimization domain is Ω . The simulated annealing (SA) algorithm is used in this paper.

Algorithm 2 Point cloud registration algorithm based on WMGGMM and stochastic optimization.

Input: Scene set: *S*, Model set: *M*, Initial transformation parameter set: $\Omega^{(0)}$ Output: Optimal transformation parameter set: Ω^* Set: $KLD^{old} = 0$, $KLD^{new} = 0$, $\Omega^{old} = \Omega^{(0)}$ $ggmm_s = GGMM(S)$ Gibbs sampling on $ggmm_s$ to get: $X_{sample} = \{x_i\}_{i=1}^{1000}$ repeat $ggmm_m^{old} = ggmm(f_T(M, \Omega^{old}))$ $KLD^{old} = KLD_{MC}(ggmm_s || ggmm_m^{old})$ $\Omega^{new} = RandomGeneration(\Omega^{old})$ $ggmm_m = ggmm(f_T(M, \Omega^{new}))$ $KLD^{new} = KLD_{MC}(ggmm_s || ggmm_m^{new})$ if $KLD^{new} < KLD^{old}$ then $\Omega^{old} = \Omega^{new}$ end if until some stopping criterion is satisfied Return $\Omega^* = \Omega^{old}$

When the transformation model is rigid, i.e., $X = R_{\alpha}X_0 + t$, the optimization process can be further simplified, where R_{α} is the rotation matrix with an angle of α and t is a translation vector. The transformation of the data set is equivalent to the rigid change of the corresponding mixture model. We can obtain $\mu'_k = R_{\alpha}\mu_k + t$ and $\Sigma'_k = R_{\alpha}^T\Sigma_k R_{\alpha}$; the shape parameter β and scale parameter m are not affected during the transformation. In other words, we only calculate the mixture models of the target scene and the scene to be registered through WMGGMM at the beginning, and the parameter estimation does not need to be repeated during optimization. The transformation models used in the experiments below are all rigid transformations and all test data are generated based on the method in the previous section.

The unlikeness in sampling rate is a common challenge in point cloud registration. In general, the original scene will have a higher sampling rate to achieve more accurate modeling. In comparison, the scene to be registered may have a lower sampling rate due to the limitations of the sampling environment and tasks. Figure 5 shows the registration results of point sets with different sampling rates. The target scene (blue) has 1200 samples, while the scene to be registered (red) has 800. Although the sampling rate varies, the two scenes' data meet similar statistical distributions, so the method based on GGMM can still effectively register the two sets with rotational and translational errors of 1.5% and 4.3%, respectively.



Figure 5. Registration results of point clouds with different sampling rates. (**a**) original; (**b**) centralized; (**c**) registration.

The registration results at different noise levels are given in Figure 6. The outcomes show that our method can effectively remove the noise and outliers' interference and extract the main features of the point cloud for matching, and the algorithm has good robustness with average rotational and translational errors of 3.5% and 6.3%, respectively.



Figure 6. Registration results with 10% (top), 20% (middle), and 30% (bottom) noise levels. (**a**) original; (**b**) centralized; (**c**) registration.

In Figure 7, we designed a point cloud shaped like the Chinese character "zhi" with a symmetrical approximate center (rotational and translational errors of 6.1% and 9.3%, respectively). The distinction between it and the original graph after 180 degrees of rotation is only at one "point". Hence, it is straightforward to fall into the optimal local solution in registration optimization. However, in several experiments, only 6% fell into local optima. Compared with derivative optimization, stochastic optimization provides the ability to jump out of the local optimal solution.



Figure 7. The registration results when locally optimal solution exists. (**a**) original; (**b**) centralized; (**c**) registration.

4. Conclusions

This paper proposes a weighted multivariate generalized Gaussian mixture model and combines it with the stochastic optimization algorithm for point cloud rigid registration. This method requires enough samples in the registration scene to meet the minimum support of components. It extracts the data's mass features rather than the edge features (contour and shell), so it is suitable for substantial point clouds with a large data volume. The introduction of weights and the ability of generalized Gaussian distributions to describe peak data can effectively reduce the influence of noise and outliers and obtain the critical features of data-intensive regions. Experimental results attest that the algorithm has sufficient robustness. The stochastic optimization algorithm reduces the coupling between algorithm modules, intensifies the algorithm's expansibility, and provides a more potent global optimization capability. However, in the mixture model's parameter estimation process, almost all parameters need to be learned iteratively; therefore, some performance is sacrificed to enhance the algorithm's accuracy. In addition, this work only considered a 2D scenario. Future work will be committed to improve the parameter estimation approach to improve the algorithm's performance and extend the approach to 3D scenarios.

Author Contributions: Conceptualization, B.G., F.N. and N.B.; methodology, B.G., F.N. and N.B.; software, B.G.; validation, B.G. and F.N.; resources, N.B.; writing—original draft preparation, B.G; writing—review and editing, F.N. and N.B.; supervision, N.B.; funding acquisition, N.B.; All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by NSERC grant number RGPIN-6656-2017.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Not applicable.

Acknowledgments: The completion of this research was made possible thanks to the Natural Sciences and Engineering Research Council of Canada (NSERC).

Conflicts of Interest: The authors declare no conflict of interest.

References

- Zhu, H.; Guo, B.; Zou, K.; Li, Y.; Yuen, K.V.; Mihaylova, L.; Leung, H. A review of point set registration: From pairwise registration to groupwise registration. *Sensors* 2019, 19, 1191. [CrossRef] [PubMed]
- Hirose, O. A Bayesian Formulation of Coherent Point Drift. IEEE Trans. Pattern Anal. Mach. Intell. 2021, 43, 2269–2286. [CrossRef] [PubMed]
- Hirose, O. Geodesic-Based Bayesian Coherent Point Drift. IEEE Trans. Pattern Anal. Mach. Intell. 2023, 45, 5816–5832. [CrossRef] [PubMed]
- Zhang, Z.; Dai, Y.; Sun, J. Deep learning based point cloud registration: An overview. Virtual Real. Intell. Hardw. 2020, 2, 222–246. [CrossRef]
- Min, Z.; Wang, J.; Pan, J.; Meng, M.Q.H. Generalized 3-D Point Set Registration With Hybrid Mixture Models for Computer-Assisted Orthopedic Surgery: From Isotropic to Anisotropic Positional Error. *IEEE Trans. Autom. Sci. Eng.* 2021, 18, 1679–1691. [CrossRef]
- De Silva, V.; Roche, J.; Kondoz, A. Fusion of LiDAR and camera sensor data for environment sensing in driverless vehicles. *arXiv* 2018, arXiv:1710.06230.
- Giering, M.; Venugopalan, V.; Reddy, K. Multi-modal sensor registration for vehicle perception via deep neural networks. In Proceedings of the 2015 IEEE High Performance Extreme Computing Conference (HPEC), Waltham, MA, USA, 15–17 September 2015; pp. 1–6.
- Mastin, A.; Kepner, J.; Fisher, J. Automatic registration of LIDAR and optical images of urban scenes. In Proceedings of the 2009 IEEE Conference on Computer Vision and Pattern Recognition, Miami, FL, USA, 20–25 June 2009; pp. 2639–2646.
- Rosas-Cervantes, V.; Lee, S.G. 3D Localization of a Mobile Robot by Using Monte Carlo Algorithm and 2D Features of 3D Point Cloud. Int. J. Control. Autom. Syst. 2020, 18, 1–11. [CrossRef]
- 10. Lu, W.; Wan, G.; Zhou, Y.; Fu, X.; Yuan, P.; Song, S. Deepvcp: An end-to-end deep neural network for point cloud registration. In Proceedings of the IEEE International Conference on Computer Vision, Seoul, Korea, 27 October–2 November 2019; pp. 12–21.
- Rundo, L.; Tangherloni, A.; Militello, C.; Gilardi, M.C.; Mauri, G. Multimodal medical image registration using particle swarm optimization: A review. In Proceedings of the 2016 IEEE Symposium Series on Computational Intelligence (SSCI), Athens, Greece, 6–9 December 2016; pp. 1–8.
- 12. Chen, Y.; He, F.; Li, H.; Zhang, D.; Wu, Y. A full migration BBO algorithm with enhanced population quality bounds for multimodal biomedical image registration. *Appl. Soft Comput.* **2020**, *93*, 106335. [CrossRef]
- Collignon, A.; Vandermeulen, D.; Suetens, P.; Marchal, G. 3D multi-modality medical image registration using feature space clustering. In Proceedings of the International Conference on Computer Vision, Virtual Reality, and Robotics in Medicine, Nice, France, 3–6 April 1995; Springer: Berlin/Heidelberg, Germany, 1995; pp. 195–204.
- Sinko, M.; Kamencay, P.; Hudec, R.; Benco, M. 3D registration of the point cloud data using ICP algorithm in medical image analysis. In Proceedings of the 2018 IEEE ELEKTRO, Mikulov, Czech Republic, 21–23 May 2018; pp. 1–6.
- 15. El-Hakim, S.F.; Beraldin, J.A.; Picard, M.; Godin, G. Detailed 3D reconstruction of large-scale heritage sites with integrated techniques. *IEEE Comput. Graph. Appl.* **2004**, *24*, 21–29. [CrossRef]
- Zhao, X.; Zhang, C.; Wang, Y.; Yang, B. A hybrid approach based on MEP and CSP for contour registration. *Appl. Soft Comput.* 2011, 11, 5391–5399. [CrossRef]
- 17. Bermejo, E.; Cordón, O.; Damas, S.; Santamaría, J. Quality time-of-flight range imaging for feature-based registration using bacterial foraging. *Appl. Soft Comput.* **2013**, *13*, 3178–3189. [CrossRef]
- 18. Ma, J.; Jiang, J.; Liu, C.; Li, Y. Feature guided Gaussian mixture model with semi-supervised EM and local geometric constraint for retinal image registration. *Inf. Sci.* 2017, 417, 128–142. [CrossRef]
- Keller, M.; Lefloch, D.; Lambers, M.; Izadi, S.; Weyrich, T.; Kolb, A. Real-time 3D reconstruction in dynamic scenes using point-based fusion. In Proceedings of the 2013 IEEE International Conference on 3D Vision-3DV, Seattle, WA, USA, 29 June–1 July 2013; pp. 1–8.
- 20. Zhang, Z. Iterative point matching for registration of free-form curves and surfaces. *Int. J. Comput. Vis.* **1994**, *13*, 119–152. [CrossRef]
- 21. Conte, D.; Foggia, P.; Sansone, C.; Vento, M. Thirty years of graph matching in pattern recognition. *Int. J. Pattern Recognit. Artif. Intell.* **2004**, *18*, 265–298. [CrossRef]
- 22. Gao, W.; Tedrake, R. Filterreg: Robust and efficient probabilistic point-set registration using gaussian filter and twist parameterization. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Long Beach, CA, USA, 15–20 June 2019; pp. 11095–11104.
- Jian, B.; Vemuri, B.C. Robust point set registration using gaussian mixture models. *IEEE Trans. Pattern Anal. Mach. Intell.* 2010, 33, 1633–1645. [CrossRef] [PubMed]
- Tao, W.; Sun, K. Asymmetrical Gauss mixture models for point sets matching. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Columbus, OH, USA, 23–28 June 2014; pp. 1598–1605.
- Ravikumar, N.; Gooya, A.; Çimen, S.; Frangi, A.F.; Taylor, Z.A. Group-wise similarity registration of point sets using Student's t-mixture model for statistical shape models. *Med Image Anal.* 2018, 44, 156–176. [CrossRef] [PubMed]
- 26. Bouguila, N.; Fan, W. Mixture Models and Applications; Springer: Berlin/Heidelberg, Germany, 2020.

- 27. Sefidpour, A.; Bouguila, N. Spatial color image segmentation based on finite non-Gaussian mixture models. *Expert Syst. Appl.* **2012**, *39*, 8993–9001. [CrossRef]
- Hu, C.; Fan, W.; Du, J.; Bouguila, N. A novel statistical approach for clustering positive data based on finite inverted Beta–Liouville mixture models. *Neurocomputing* 2019, 333, 110–123. [CrossRef]
- Bouguila, N.; Elguebaly, T. A fully Bayesian model based on reversible jump MCMC and finite Beta mixtures for clustering. Expert Syst. Appl. 2012, 39, 5946–5959. [CrossRef]
- Fan, J.; Yang, J.; Ai, D.; Xia, L.; Zhao, Y.; Gao, X.; Wang, Y. Convex hull indexed Gaussian mixture model (CH-GMM) for 3D point set registration. *Pattern Recognit.* 2016, 59, 126–141. [CrossRef]
- Pascal, F.; Bombrun, L.; Tourneret, J.Y.; Berthoumieu, Y. Parameter estimation for multivariate generalized Gaussian distributions. IEEE Trans. Signal Process. 2013, 61, 5960–5971. [CrossRef]
- 32. Elguebaly, T.; Bouguila, N. Model-based approach for high-dimensional non-Gaussian visual data clustering and feature weighting. *Digit. Signal Process.* **2015**, *40*, 63–79. [CrossRef]
- 33. Franchini, S.; Charogiannis, A.; Markides, C.N.; Blunt, M.J.; Krevor, S. Calibration of astigmatic particle tracking velocimetry based on generalized Gaussian feature extraction. *Adv. Water Resour.* **2019**, *124*, 1–8. [CrossRef]
- Kubo, Y.; Takamune, N.; Kitamura, D.; Saruwatari, H. Blind Speech Extraction Based on Rank-Constrained Spatial Covariance Matrix Estimation With Multivariate Generalized Gaussian Distribution. *IEEE/ACM Trans. Audio Speech Lang. Process.* 2020, 28, 1948–1963. [CrossRef]
- Hristova, H.; Le Meur, O.; Cozot, R.; Bouatouch, K. Transformation of the multivariate generalized Gaussian distribution for image editing. *IEEE Trans. Vis. Comput. Graph.* 2017, 24, 2813–2826. [CrossRef]
- 36. Verdoolaege, G.; Scheunders, P. Geodesics on the manifold of multivariate generalized Gaussian distributions with an application to multicomponent texture discrimination. *Int. J. Comput. Vis.* **2011**, *95*, 265. [CrossRef]
- 37. Rami, H.; Belmerhnia, L.; El Maliani, A.D.; El Hassouni, M. Texture retrieval using mixtures of generalized Gaussian distribution and Cauchy–Schwarz divergence in wavelet domain. *Signal Process. Image Commun.* **2016**, *42*, 45–58. [CrossRef]
- Verdoolaege, G.; Rosseel, Y.; Lambrechts, M.; Scheunders, P. Wavelet-based colour texture retrieval using the Kullback–Leibler divergence between bivariate generalized Gaussian models. In Proceedings of the 16th IEEE International Conference on Image Processing (ICIP), Cairo, Egypt, 7–10 November 2009; pp. 265–268.
- 39. Channoufi, I.; Bourouis, S.; Bouguila, N.; Hamrouni, K. Image and video denoising by combining unsupervised bounded generalized gaussian mixture modeling and spatial information. *Multimed. Tools Appl.* **2018**, 77, 25591–25606. [CrossRef]
- Channoufi, I.; Bourouis, S.; Bouguila, N.; Hamrouni, K. Color image segmentation with bounded generalized gaussian mixture model and feature selection. In Proceedings of the 4th IEEE International Conference on Advanced Technologies for Signal and Image Processing (ATSIP), Sousse, Tunisia, 21–24 March 2018; pp. 1–6.
- Allili, M.S.; Bouguila, N.; Ziou, D. Finite generalized Gaussian mixture modeling and applications to image and video foreground segmentation. In Proceedings of the 4th IEEE Canadian Conference on Computer and Robot Vision (CRV'07), Montreal, QC, Canada, 28–30 May 2007; pp. 183–190.
- 42. Najar, F.; Bourouis, S.; Bouguila, N.; Belghith, S. Unsupervised learning of finite full covariance multivariate generalized Gaussian mixture models for human activity recognition. *Multimed. Tools Appl.* **2019**, *78*, 18669–18691. [CrossRef]
- Najar, F.; Bourouis, S.; Bouguila, N.; Belghith, S. A fixed-point estimation algorithm for learning the multivariate GGMM: Application to human action recognition. In Proceedings of the 2018 IEEE Canadian Conference on Electrical & Computer Engineering (CCECE), Quebec, QC, Canada, 13–16 May 2018; pp. 1–4.
- 44. Gebru, I.D.; Alameda-Pineda, X.; Forbes, F.; Horaud, R. EM algorithms for weighted-data clustering with application to audio-visual scene analysis. *IEEE Trans. Pattern Anal. Mach. Intell.* **2016**, *38*, 2402–2415. [CrossRef]
- 45. Agarwal, P.; Jleli, M.; Samet, B. Banach Contraction Principle and Applications. In *Fixed Point Theory in Metric Spaces*; Springer: Berlin/Heidelberg, Germany, 2018; pp. 1–23.
- Figueiredo, M.A.T.; Jain, A.K. Unsupervised learning of finite mixture models. *IEEE Trans. Pattern Anal. Mach. Intell.* 2002, 24, 381–396. [CrossRef]
- 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.
- 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.
- Cui, S.; Datcu, M. Comparison of Kullback–Leibler divergence approximation methods between Gaussian mixture models for satellite image retrieval. In Proceedings of the 2015 IEEE International Geoscience and Remote Sensing Symposium (IGARSS), Milan, Italy, 26–31 July 2015; pp. 3719–3722.

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.