Asymmetrical Gauss Mixture Models for Point Sets Matching

Wenbing Tao and Kun Sun
National Key Laboratory of Science and Technology on Multi-spectral Information Processing
School of Automation, Huazhong University of Science and Technology, China
wenbingtao@hust.edu.cn; sunkun.hust@gmail.com

Abstract

The probabilistic methods based on Symmetrical Gauss Mixture Model(SGMM) [4, 13, 8] have achieved great success in point sets registration, but are seldom used to find the correspondences between two images due to the complexity of the non-rigid transformation and too many outliers. In this paper we propose an Asymmetrical GMM(AGMM) for point sets matching between a pair of images. Different from the previous SGMM, the AGMM gives each Gauss component a different weight which is related to the feature similarity between the data point and model point, which leads to two effective algorithms: the Single Gauss Model for Mismatch Rejection(SGMR) algorithm and the AGMM algorithm for point sets matching. The SGMR algorithm iteratively filters mismatches by estimating a non-rigid transformation between two images based on the spatial coherence of point sets. The AGMM algorithm combines the feature information with position information of the SIFT feature points extracted from the images to achieve point sets matching so that much more correct correspondences with high precision can be found. A number of comparison and evaluation experiments reveal the excellent performance of the proposed SGMR algorithm and AGMM algorithm.

1. Introduction

Point sets matching between a pair of images is to define a set of points in each image and find the correspondences between them, which is one important problem in computer vision.

Most of point sets matching technologies are based on local feature descriptors. Mikolajczyk and Schmid [12] compared the performance of different kinds of local descriptors and concluded that the SIFT-based descriptors [11] perform best. The traditional SIFT feature matching is mainly based on the similarity of the SIFT features, which tends to produce many mismatches due to the variation of light, the change of illumination and the repeated local patterns in images [1]. Some post-processing strategies are usually used to reject the mismatches [10, 14, 16, 18]. However, the produced matches are often too sparse, especially serious in wide-baseline cases.

The point sets matching problem between two images can also be regarded as the non-rigid point sets registration problem if only the position information is used. There exist many algorithms to handle point set registration. An effective strategy is to consider the alignment of two point sets as a probability density estimation problem [4, 5, 8, 13]. The core idea for these techniques is to assume one point set as the data points $\boldsymbol{X}$ and the other one as the model points $\boldsymbol{Y}$. Among them a representative method is to use the Gauss Mixture Model(GMM) [4, 8, 13] to describe the model points. This GMM-based point sets registration framework is a Symmetrical GMM framework(SGMM), where all Gauss components are given the same weights. The SGMM has achieved great success in point set registration [8, 13], where the points in two sets have only position information but without available feature information. However, the SGMM is seldom directly used to find the correspondences between two images for three reasons: 1)The non-rigid transformation between two images is complex due to the complexity of the projection of the 3D depth scene to the 2D plane images; 2)There are usually too many outliers among the initial extracted feature points, *e.g.* SIFT, often up to or even more than eighty percent; 3)There often exist too large rotation and deformation in multi-view images, especially in wide baseline cases.

To overcome the limitations mentioned above, we propose an Asymmetrical GMM(AGMM) with variable weights for point sets matching. In the AGMM-based point sets matching framework, each Gauss component is given a different weight which is related to the feature similarity between the data point and model point. The motivation for doing so is threefold: 1) If the Gauss distribution centroid which one model point corresponds to is very similar to the data point in feature space it is reasonable to give a large weight to this GMM component so that the two points have more chance to form a correspondence; 2) The AGMM framework may converge faster and be more robust to noise or outliers since the feature similarity has provided a reasonable prior information for the match between two point sets; 3) By applying AGMM, the feature similarity and spatial coherence can be conveniently and naturally integrated in the identical GMM framework for point sets matching.

1.1. Related Works

Feature points matching across images is a challenging task when too many outliers and noise are included and the spatial transformation between two point sets is too nonrigid. Some of these problems can be addressed by integrating spatial consistency and feature similarity for matching points. One of the effective strategies is to formulate correspondence finding as a graph matching problem [2, 3]. An attributed graph is constructed where graph nodes denote the feature descriptors and edges denote the spatial relations between features. And then the corresponding graph matching problem is solved by some optimization algorithms. Although there are some efficient algorithms for graph matching, such approaches are still difficult to be applied to handle large numbers of features due to the complexity of the problems themselves.

Another strategy integrating spatial consistency and feature similarity for point sets matching is to simultaneously encode spatial consistency and feature similarity in a lower-dimensional subspace [6, 15]. In [15], Torki and Elgammal proposed a nonrigid feature matching framework by fusion of the feature descriptor and its spatial location. This is implemented by learning an embedded representation combining the feature similarity and spacial arrangement in a unified Euclidean embedding space. Hamid et.al [6] improved the works [15] by using approximate subspace learning with random projections instead of exact spectral decomposition so that the computational efficiency is significantly increased. Although the feature embedding methods can effectively improve the accuracy of matching, the ability to cope with outliers and noise is still limited. Furthermore, the sparsity of the matches was not remarkably amended.

To the best of our knowledge, it is the first time for us to propose the Asymmetrical GMM(AGMM) with variable weights for point sets matching where the feature similarity is effectively integrated into GMM-based point sets registration framework. The feature similarities are encoded as the weights of the Gauss components of GMM instead of the identical weights for all Gauss components in Symmetrical GMM(SGMM). Extensive experiments on some typical datasets show that AGMM is more robust in the face of excessive outliers and large distortion and rotation. Furthermore, much more reliable matches can be found by the proposed methods than feature-based matching method and the other state-of-the-art algorithms combining feature and position information.

1.2. The Contribution

1)The SGMM has been extensively applied to point sets registration and achieved great success until now, but very little attention is paid to the AGMM and hardly any investigation is involved in it. In this paper, we propose the AGMM for point sets matching. The SGMM can be regarded as the special case of the AGMM when there is no available prior feature information. In AGMM algorithm framework, the feature similarities are encoded as the weights of the Gauss components of GMM instead of the identical weights for all Gauss components in SGMM framework. If one data point is very similar to one model point in feature space their corresponding Gauss component will be awarded a relatively larger weight, otherwise it will be punished with a smaller weight.The embedding of the feature information in AGMM is testified faster convergence, better robustness to outliers, deformation and rotation and can be exploited to handle more complex non-rigid transformation than only position information used.

2) If we have obtained the initial match set by a certain match method, we can simplify the AGMM to an extreme situation, which can be used to remove mismatches. In this situation, the weight of the GMM component which corresponds to a match pair is given one while the other weights are given zero. When the algorithm converges the matches with low probabilities will be discarded. This mismatches rejection algorithm is demonstrated to outperform many state-of-the-art methods.

3) The traditional feature-based matching methods such as SIFT feature match suffer from the excessive sparsity of the match and low precision, which can be improved by the fusion of the feature and position information. However, the previous works in this area are concentrated on graph matching and embedding space representation. By applying AGMM, the feature similarity and spatial coherence can be conveniently and naturally integrated in the identical GMM framework for point sets matching. This strategy is testified extremely effective to find much more correct correspondences with higher accuracy than feature-based point sets matching methods. The AGMM algorithm is also demonstrated to outperform other matching methods combining the feature and position information in state-of-the-art.

2. Asymmetrical Gauss Mixture Models for Point Sets Matching

When using probability density estimation to solve the alignment problem of two point sets, one point set is considered as the GMM centroids and the other one represents the data points. Assume $\boldsymbol{X} = \{\boldsymbol{x_i} | \boldsymbol{x_i} \in \mathbb{R}^2, i = 1, \ldots, n\}$ and $\boldsymbol{Y} = \{\boldsymbol{y_j} | \boldsymbol{y_j} \in \mathbb{R}^2, j = 1, \ldots, m\}$ are respectively the SIFT points extracted from a pair of images. We consider the points in $\boldsymbol{Y}$ as m centroids of the GMM and the points in $\boldsymbol{X}$ as the data points generated by the GMM. The GMM probability density function is

$$p(\boldsymbol{x_i}) = \sum_{j=1}^{m} p_{ij} p(\boldsymbol{x_i}|j). \tag{1}$$

Suppose $\boldsymbol{\mathcal{C}}(\boldsymbol{Y})$ is the motion model of the points in $\boldsymbol{Y}$ and $\forall \boldsymbol{y_j} \in \boldsymbol{Y}, \boldsymbol{y_j}^* = \boldsymbol{y_j} + \boldsymbol{\mathcal{C}}(\boldsymbol{y_j})$ is the new location of the j^{th} Gauss component. Accounting for noise and outliers an additional uniform distribution is added. Denoting the weight of the uniform distribution as $1-\omega$, $0 \le \omega \le 1$, the mixture model is

$$p(\boldsymbol{x_i}|\boldsymbol{\Theta}, \omega) = \omega \sum_{j=1}^{m} p_{ij} p(\boldsymbol{x_i}|j) + (1-\omega) p(\boldsymbol{outlier}), \tag{2}$$

where

$$p(\boldsymbol{x_i}|j) = \frac{1}{(2\pi\sigma^2)^{\frac{D}{2}}} e^{-\frac{\left\|\boldsymbol{x_i}-\boldsymbol{y_j^*}\right\|^2}{2\sigma^2}} \tag{3}$$

is the j^{th} Gauss component with respect to $\boldsymbol{x_i}$. $p(\boldsymbol{outlier}) = \frac{1}{n}$ models the outliers and noise. $p_{ij}(i = 1, \ldots, n; j = 1, \ldots, m)$ is the mixture coefficient for all the Gauss components. In the previous works [5, 8, 13], each component in GMM is assumed to have the same weight and $p_{ij} = \frac{1}{m}$ for all GMM components. $\boldsymbol{\Theta} = \{\boldsymbol{\mathcal{C}}(\boldsymbol{Y}), \sigma\}$ is the motion parameters and D is the dimension of the point sets.

However, besides considering the consistence of the location of point sets, the feature similarity between point sets is also an important factor that influences the reliability of the correspondences. If one Gauss distribution centroid is very similar to one data point in feature space it is reasonable to give a large weight to this GMM component so that the point pair with similar features has more chance to form a correspondence. Therefore, in the proposed asymmetrical GMM the weights are defined as follows:

$$p_{ij} = \frac{e^{-\alpha\|descr(\boldsymbol{x_i}) - decsr(\boldsymbol{y_j})\|^2}}{\sum_{k=1}^{m} e^{-\alpha\|descr(\boldsymbol{x_i}) - decsr(\boldsymbol{y_k})\|^2}}. \tag{4}$$

$descr(\boldsymbol{x_i})$ and $descr(\boldsymbol{y_j})$ respectively denote the SIFT features of point $\boldsymbol{x_i}$ and $\boldsymbol{y_j}$. α is a control parameter.

Thus, the likelihood is a mixture model of distributions for inliers and outliers as follows:

$$\mathcal{L}(\boldsymbol{\Theta}, \omega) = \prod_{i=1}^{n} p(\boldsymbol{x_i}|\boldsymbol{\Theta}, \omega). \tag{5}$$

To effectively deal with point sets registration with more complex transformation, we use Tikhonov regularization to define the prior as follows:

$$p(\boldsymbol{\mathcal{C}}) = e^{-\frac{\lambda}{2}\|\boldsymbol{\mathcal{C}}\|^2_{\mathcal{H}}}, \tag{6}$$

where $\|\boldsymbol{\mathcal{C}}\|^2_{\mathcal{H}}$ is the norm of $\boldsymbol{\mathcal{C}}(\boldsymbol{y})$ in the Reproduction Kernel Hilbert Space(RKHS). $\boldsymbol{\mathcal{C}}$ can be solved using calculus of variation and the optimal form can be written as the linear combination of kernels $\boldsymbol{\mathcal{C}}(\boldsymbol{z}) = \sum_{j=1}^{m} \varphi_j \boldsymbol{\mathcal{K}}(\boldsymbol{z}, \boldsymbol{y_j}) = \boldsymbol{\mathcal{K}\Phi}$, where $\boldsymbol{\mathcal{K}}$ is a $m \times m$ kernel matrix. $\boldsymbol{\mathcal{K}}(\boldsymbol{y_i}, \boldsymbol{y_j}) = e^{-\frac{1}{2}\frac{\|(\boldsymbol{y_i}-\boldsymbol{y_j})\|^2}{\beta}}$, where β is a kernel bandwidth controlling the local structure.

According to (5) and (6) the posteriori probability $p(\boldsymbol{\Theta}, \omega|\boldsymbol{X}) \propto \mathcal{L}(\boldsymbol{\Theta}, \omega) p(\boldsymbol{\mathcal{C}})$. Thus, we can solve the maximum a posteriori (MAP) problem to estimate the parameters $\boldsymbol{\Theta}$ and ω. This is equivalent to minimizing the negative log-posterior $\varepsilon(\boldsymbol{\Theta}, \omega|\boldsymbol{X})$ as

$$\varepsilon(\boldsymbol{\Theta}, \omega|\boldsymbol{X}) = -\sum_{i=1}^{n} \log(p(\boldsymbol{x_i}|\boldsymbol{\Theta}, \omega)) + \frac{\lambda}{2}\|\boldsymbol{\mathcal{C}}\|^2_{\mathcal{H}}, \tag{7}$$

where $p(\boldsymbol{x_i}|\boldsymbol{\Theta}, \omega)$ is given by (2).

3. EM Algorithm

The Expectation-Maximization(EM) algorithm is an elegant algorithm to solve this MAP problem. We use the EM algorithm to find the optimal $\boldsymbol{\Theta}$ and ω. We first guess the values of parameters("old" parameter values) and then use the Bayes theorem to compute a posteriori probability distribution of the mixture components, which is the expectation or E-step of the algorithm. In the M-step, the "new" parameter values are then found by minimizing $\boldsymbol{Q}(\boldsymbol{\Theta}, \omega)$, the expectation of $\varepsilon(\boldsymbol{\Theta}, \omega)$. Omitting terms that are independent of $\boldsymbol{\Theta}$ and ω, $\boldsymbol{Q}(\boldsymbol{\Theta}, \omega)$ can be written as

$$\begin{aligned}
\boldsymbol{Q}(\boldsymbol{\Theta}, \omega) = {} & \frac{D}{2}\log\sigma^2 \sum_{i=1}^{n}\sum_{j=1}^{m} p^{old}(j|\boldsymbol{x_i}, \boldsymbol{\Theta}, \omega) \\
& + \frac{1}{2\sigma^2}\sum_{i=1}^{n}\sum_{j=1}^{m}\Big[p^{old}(j|\boldsymbol{x_i}, \boldsymbol{\Theta}, \omega)\big\|\boldsymbol{x_i} \\
& - \big(\boldsymbol{y_j} + \sum_{k=1}^{m}\varphi_j \boldsymbol{\mathcal{K}}(\boldsymbol{y_j}, \boldsymbol{y_k})\big)\big\|^2\Big] \\
& - \log\omega \sum_{i=1}^{n}\sum_{j=1}^{m} p^{old}(j|\boldsymbol{x_i}, \boldsymbol{\Theta}, \omega) \\
& - \log(1-\omega)\sum_{i=1}^{n} p^{old}(\boldsymbol{outlier}|\boldsymbol{x_i}, \boldsymbol{\Theta}, \omega) \\
& + \frac{\lambda}{2} trace(\boldsymbol{\Phi}^T \boldsymbol{\mathcal{K}} \boldsymbol{\Phi}).
\end{aligned} \tag{8}$$

E-step: Compute the posteriori probability distribution of mixture components by the Bayes theorem:

$$p(j|\boldsymbol{x_i}, \boldsymbol{\Theta}, \omega) = \frac{\frac{p_{ij}}{(2\pi\sigma^2)^{\frac{D}{2}}} e^{-\frac{\left\|\boldsymbol{x_i}-\boldsymbol{y_j^*}\right\|^2}{2\sigma^2}}}{\omega\sum_{k=1}^{m} p_{ik} p(\boldsymbol{x_i}|k) + (1-\omega)/n}, \tag{9}$$

where $i = 1, \ldots, n$ and $j = 1, \cdots, m$.

$$p(\boldsymbol{outlier}|\boldsymbol{x_i}, \boldsymbol{\Theta}, \omega) = \frac{1-\omega}{n\omega\sum_{k=1}^{m} p_{ik} p(\boldsymbol{x_i}|k) + (1-\omega)}. \tag{10}$$

Formula (9) indicates the probability the sample $\boldsymbol{x_i}$ matches with $\boldsymbol{y_j}$.The smaller the probability is, the less reliable the match is. Therefore, the probability of the matches can be used to filter those unstable matches after EM algorithm converges. Giving a threshold t_p, the matches whose probabilities are less than t_p can be regarded as mismatches to be removed.

M-step: Update parameters $\boldsymbol{\Theta}$ and ω by minimizing the energy function $\boldsymbol{Q}(\boldsymbol{\Theta}, \omega)$.

Solving $\frac{\partial \boldsymbol{Q}}{\partial \sigma} = 0$ by (8), we can compute σ by

$$\sigma^2 = \frac{\sum_{i,j=1}^{n,m} p^{old}(j|\boldsymbol{x_i}, \boldsymbol{\Theta}, \omega)\left\|\boldsymbol{x_i} - \boldsymbol{y_j^*}\right\|^2}{D\sum_{i,j=1}^{n,m} p^{old}(j|\boldsymbol{x_i}, \boldsymbol{\Theta}, \omega)}. \tag{11}$$

Similarly, solving $\frac{\partial \boldsymbol{Q}}{\partial \omega} = 0$, ω can be computed by

$$\omega = \left[\sum_{i,j=1}^{n,m} p^{old}(j|\boldsymbol{x_i}, \boldsymbol{\Theta}, \omega)\right] / \left[\sum_{i,j=1}^{n,m} p^{old}(j|\boldsymbol{x_i}, \boldsymbol{\Theta}, \omega) \right.$$
$$\left. + \sum_{i=1}^{n} p^{old}(\boldsymbol{\mathit{outlier}}|\boldsymbol{x_i}, \boldsymbol{\Theta}, \omega)\right]. \tag{12}$$

And let $\frac{\partial \boldsymbol{Q}}{\partial \boldsymbol{\Phi}} = 0$, we can get $\boldsymbol{\Phi}$ by

$$(diag(\boldsymbol{P}\mathbf{1})\boldsymbol{\mathcal{K}} + \lambda\sigma^2)\boldsymbol{\Phi} = \boldsymbol{PX} - diag(\boldsymbol{P}\mathbf{1})\boldsymbol{Y}, \tag{13}$$

where $\mathbf{1}$ is a column vector of all ones, $\boldsymbol{P}$ is the matrix with $\boldsymbol{P}(j, i) = p(j|\boldsymbol{x_i}, \boldsymbol{\Theta}, \omega)$.

4. Single Gauss Model for Mismatch Rejection

In GMM framework, the point $\boldsymbol{x_i}$ in $\boldsymbol{X}$ is generated by the Gauss mixture models centered at $\boldsymbol{y_j}(j = 1, \cdots, m)$, as shown in (1). In the proposed AGMM model(Section 2), the weight of each Gauss component is related to the feature similarity between the data point and the Gauss distribution centroid. We simply binarize the weights computed by (4) into 1 and 0 by only giving 1 to the largest weight and 0 to the others, which leads to the SGMR algorithm. This means, for each data point we have decided its match by feature similarity. This matching process is actually the SIFT-based feature matching.

Assume we have obtained the registration result between point sets $\boldsymbol{X}$ and $\boldsymbol{Y}$ by a certain algorithm. Let $\boldsymbol{\mathcal{S}}$ be the match set between $\boldsymbol{X}$ and $\boldsymbol{Y}$. If $(\boldsymbol{x_i}, \boldsymbol{y_j})$ is a match, we have $(\boldsymbol{x_i}, \boldsymbol{y_j}) \in \boldsymbol{\mathcal{S}}$. Some mismatches may be included in $\boldsymbol{\mathcal{S}}$. Therefore, p_{ij} in (1) can be rewritten as

$$p_{ij} = \begin{cases} 1, \ (\boldsymbol{x_i}, \boldsymbol{y_j}) \in \boldsymbol{\mathcal{S}} \\ 0, \ \text{else} \end{cases}. \tag{14}$$

According to (2), (3) and (14) $p(\boldsymbol{x_i}|\boldsymbol{\Theta}, \omega)$ takes the form

$$p(\boldsymbol{x_i}|\boldsymbol{\Theta}, \omega) = \frac{\omega}{(2\pi\sigma^2)^{\frac{D}{2}}} e^{-\frac{\left\|\boldsymbol{x_i} - \boldsymbol{y_j^*}\right\|^2}{2\sigma^2}} + \frac{1-\omega}{n} \tag{15}$$

for all $(\boldsymbol{x_i}, \boldsymbol{y_j}) \in \boldsymbol{\mathcal{S}}$.

The negative log-posterior $\varepsilon(\boldsymbol{\Theta}, \omega)$ of the posteriori probability can be obtained by (7)

$$\varepsilon(\boldsymbol{\Theta}, \omega) = -\sum_{i=1}^{n} \log(\frac{\omega}{(2\pi\sigma^2)^{\frac{D}{2}}} e^{-\frac{\left\|\boldsymbol{x_i} - \boldsymbol{y_j^*}\right\|^2}{2\sigma^2}}$$
$$+ \frac{1-\omega}{n}) + \frac{\lambda}{2}\|\boldsymbol{\mathcal{C}}\|_H^2. \tag{16}$$

We can still use EM algorithm to minimize (16) and the expectation of (16) is just like formula (8).

For $(\boldsymbol{x_i}, \boldsymbol{y_j}) \in \boldsymbol{\mathcal{S}}$, the $p(j|\boldsymbol{x_i}, \boldsymbol{\Theta}, \omega)$ can be computed by (9), (10), (14) and (15) as follows,

$$p(j|\boldsymbol{x_i}, \boldsymbol{\Theta}, \omega) = \frac{\omega e^{-\frac{\left\|\boldsymbol{x_i} - \boldsymbol{y_j^*}\right\|^2}{2\sigma^2}}}{\omega e^{-\frac{\left\|\boldsymbol{x_i} - \boldsymbol{y_j^*}\right\|^2}{2\sigma^2}} + \frac{(1-\omega)}{n}(2\pi\sigma^2)^{\frac{D}{2}}}, \tag{17}$$

otherwise $p(j|\boldsymbol{x_i}, \boldsymbol{\Theta}, \omega) = 0$. $p(\boldsymbol{\mathit{outlier}}|\boldsymbol{x_i}, \boldsymbol{\Theta}, \omega)$ can be written as

$$p(\boldsymbol{\mathit{outlier}}|\boldsymbol{x_i}, \boldsymbol{\Theta}, \omega) = \frac{(1-\omega)(2\pi\sigma^2)^{\frac{D}{2}}}{n\omega e^{-\frac{\left\|\boldsymbol{x_i} - \boldsymbol{y_j^*}\right\|^2}{2\sigma^2}} + (1-\omega)(2\pi\sigma^2)^{\frac{D}{2}}}. \tag{18}$$

5. The Implementation Details

The SGMR algorithm filters the mismatches in the initial match set by motion coherence of point sets [13]. When the algorithm converges a transformation is learned from the given match sets and a probability matrix $\boldsymbol{P}$ is obtained. Those matches with probabilities less than a probability threshold t_p are considered to be less consistent with the transformation and should be removed as mismatches. An appropriate value of the threshold t_p can bring good balance between precision and recall of the SGMR algorithm. Parameter β defines the model of the nonrigid transformation and λ controls the degree of the coherence constraint of point sets. In our experiments, we find our method insensitive to parameters t_p, λ and β. We empirically set $\beta = 3.5$, $\lambda = 5$ and $t_p = 0.3$ throughout this paper. The inlier ratio ω is initialized to 0.3 and parameter σ is initialized as $\sigma^2 = \frac{1}{m*n*D}\sum_{i=1}^{n}\sum_{j=1}^{m}\|\boldsymbol{x_i} - \boldsymbol{y_j}\|^2$, $\boldsymbol{\Phi} = \mathbf{0}$. Parameters ω and σ will be updated during the algorithm.

Similar to CPD [13], the AGMM algorithm aligns the point sets by iteratively alternating between the correspondence and the transformation estimation until convergence. Different from the SGMR algorithm, the AGMM algorithm just uses the information of the SIFT feature points extracted from two images, not including the match information. Since the SGMR algorithm is the special case of the AGMM algorithm, the same settings of the parameters work well for AGMM. The initially extracted two SIFT point sets are considered as data set and model set respectively. The weight of each Gauss component is decided by the feature similarities between the data points and model points, which is given by formula (4). The matches are decided according to the probability matrix $\boldsymbol{P}$ after the algorithm converges. The maximum element in each column vector of $\boldsymbol{P}$ decides a match. However, due to outliers and noise, the result may include many mismatches. There are two alternatives to filter the mismatches: (1) Filter unstable matches using the probability threshold t_p only. This is feasible when the scene is sim-

Figure 1: Columns from left to right: 4 image pairs (boat1,boat3), (graf1,graf4), (trees1,trees5) and (bikes1,bikes3) are selected to visualize the results.

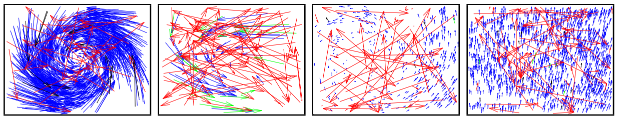
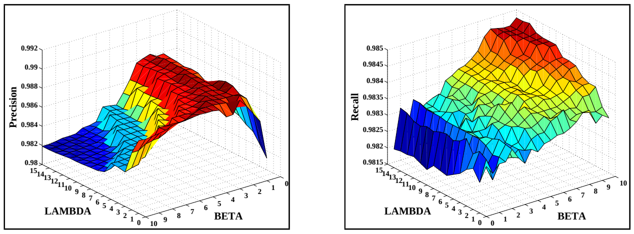


Figure 2: The mismatch eliminating results on the image pairs in Fig. 1 by SGMR. The red arrows(true negative) and black arrows (false negative) are the filtered mismatches, and the blue arrows(true positive) and green arrows(false positive) are the retained matches by SGMR.

ple. (2) Use fundamental matrix $\boldsymbol{F}$ and $t_p = 0.3$ to discard mismatches for a challenging scene. This combination is proved to work well in most cases. Besides, other excellent mismatch rejection methods can also be applied depending on the situation.

6. Experiments

In this section we will present experimental results on the application of our proposed SGMR algorithm and AG-MM algorithm to some data sets. The main data set used to test the proposed algorithms is the data set of Mikolajczyk *et al.* [12]. The reasons why we choose this dataset lie in two-fold:1) It offers ground-truth for the performance evaluation; 2) It contains challenging situations in matching such as illumination, rotation, view point changing and so on. The Mikolajczyk dataset includes eight groups of data, namely “bark”, “boat”, “graf”,“leuven”, “ubc”, “wall”, “trees” and “bikes”. Each group has six images corresponding to the same scene and forms five matching image pairs. In the experiments on the Mikolajczyk datasets, we use the method presented in [12] to identify the correct matches to evaluate the performance. Also some other data sets are designed to test the robustness of the proposed algorithms to rotation, distortion, outliers and noise. We use the “VLFeat toolbox” [17] with all the default parameters to extract SIFT features throughout this paper.



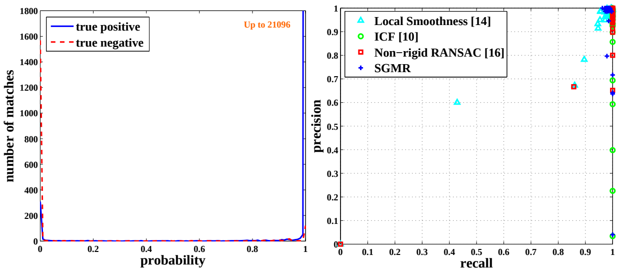
(a) (b)

Figure 3: The precision (a) and recall (b) rate of SGMR over all the forty image pairs [12] with different lambda and beta. As a tradeoff, we take $\lambda \in [4, 12]$ and $\beta \in [2, 6]$.

6.1. Mismatch Rejecting by SGMR Algorithm

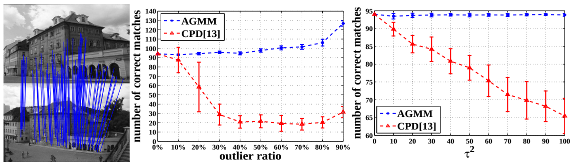
The Mikolajczyk dataset [12] is first used to evaluate the mismatch rejection performance of the proposed SGMR algorithm. The performance for mismatch removing is characterized by precision and recall. We first use four typical image pairs (boat1,boat3), (graf1,graf4), (trees1,trees5) and (bikes1,bikes3) in Fig. 1 to illustrate the mismatch removing performance of the proposed SGMR algorithm. A match is represented by a directed arrow in Fig. 2 to show coherence. We can see that the SGMR algorithm can filter most of the obvious mismatches with $t_p = 0.3$, which is shown in Fig. 2.

In Fig. 3 we evaluate the proposed SGMR algorithm with respect to the values of parameters β and λ on all the forty image pairs in Mikolajczyk dataset [12]. We find the results insensitive to the parameters. For a wide range of β and λ, both precision and recall do not change significantly. Comparatively, the algorithm is more sensitive to β since it reflects the inherent relationship of the model set. As a



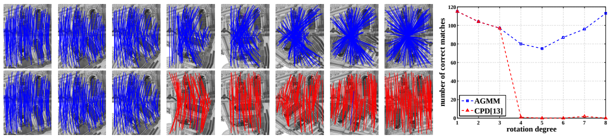
(a) (b)

Figure 4: (a) The distribution of SGMR’s $p(j|\boldsymbol{x_i}, \boldsymbol{\Theta}, \omega)$(17) for all the 40 image pairs in [12]. (b) The precision-recall of the Local Smoothness [14], ICF [10], Non-rigid RANSAC [16] and the proposed SGMR algorithm.



(a) (b) (c)

Figure 5: Robustness test: outliers and deformation. (a) The image pair “kampa” [9] and the marked 94 reliable matches(blue lines). (b) The number of correct matches with increasing outlier ratio. (c) The number of correct matches with increasing deformation.



(a) (b)

Figure 6: Robustness test: rotation. (a) The “semper” dataset [7]. From left to right: the results of our algorithm is displayed in the first row while the second row shows the results of CPD’s. We use the fundamental matrix to evaluate the results. The blue line indicates correct matches and the red line indicate false matches. (b) The number of correct matches with increasing rotation angle.

tradeoff, the algorithm performs best for $\lambda \in [4, 12]$ and $\beta \in [2, 6]$.

We now test the effect of different t_p on the mismatch rejection performance of the SGMR algorithm. Fig. 4(a) shows the distribution of the probability $p(j|\boldsymbol{x_i}, \boldsymbol{\Theta}, \omega)(17)$ for all the matches found using the forty image pairs in Mikolajczyk dataset [12]. The histogram bin width is set to 0.01. The blue line denotes the distribution of the true positive matches and the red line denotes the the distribution of the true negative matches. From the figure we see that almost all of the true negative matches are centralized in the section with $t_p < 0.02$ and almost all of the true positive matches are centralized in the section with $t_p > 0.98$. This means that parameter t_p can be set as any value between 0.02 and 0.98 to obtain satisfactory performance. Therefore, we can point out that the parameter t_p can effectively discard the mismatches and the performance of the SGMR is not sensitive to t_p.

The overall mismatch removing performance is evaluated by precision and recall. The precision-recall plot of the SGMR algorithm with $t_p = 0.3$ is given in Fig. 4(b). Several state-of-the-art mismatch rejection algorithms [10, 14, 16] are used to compare with the SGMR algorithm. As shown in Fig. 4(b), our proposed SGMR algorithm can get the best precision-recall trade-off among all the algorithms. And we find that the proposed SGMR algorithm is robust to large view angle, image rotation, and affine transformation since all these situations occur in Mikolajczyk dataset [12].

6.2. Point Sets Matching by AGMM algorithm

In this part, the performance of the proposed AGMM algorithm is tested. We first testify the robustness of the proposed AGMM algorithm to outliers, deformation and rotation and compare with CPD. In Fig. 5, one image pair “kampa” from [9] is chosen to carry out the outliers and deformation experiments. We first obtain a true correct match set of the “kampa” image pair, shown in Fig. 5(a). Then the outlier ratio is gradually increased to 90%. Outliers are randomly selected SIFT feature points on each image except for the given correct matches beforehand. We can see from Fig. 5(b) that the correct matches found by CPD gradually decreases as the outlier ratio increases while the proposed AGMM can always find all the correct matches, or even more (the reason is that the randomly added SIFT feature points may include a few inliers). In Fig. 5(c) the robustness test to deformation is shown. Deformation is generated by adding a Gauss white noise on each true correct match and the degree of deformation is controlled by the parameter τ. The results show that the robustness of the proposed AGMM algorithm is much better than CPD. For the experiments in both (b) and (c) of Fig. 5 we conduct 30 trials with the same settings and plot the mean and variance as well. The rotation experiment is shown in Fig. 6, where the “semper” dataset [7] is adopted. The “semper” dataset [7] contains 9 images of increasing rotation angle. We get eight image pairs by matching the first image to the other eight images respectively. For each image pair, we mark a set of true correct matches(notice that this set may be different for different image pairs). Then we restart the search of the correspondences between the point sets regardless of the existing match relationship between them by using only the position information of the two point sets to conduct the CPD algorithm [13] and using the position information together with the feature information of the point sets to conduct the proposed AGMM algorithm. The test results demonstrate that the proposed AGMM algorithm has better robustness to rotation than the CPD algorithm.

Another dataset that we use to evaluate the performance of the AGMM algorithm is still the Mikolajczyk dataset [12] for its convenient quantitative evaluation. For most of the image pairs in [12] the ratio of the outliers in the extracted SIFT points exceeds 80% and for some image pairs it exceeds 90%. Therefore, this dataset is challenging for GMM-based point sets registration algorithms.

The iterative process comparison for the CPD algorith-

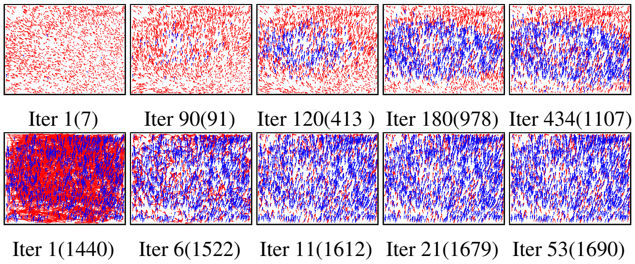


Figure 7: The iterative process for both CPD[13] and AGMM using the image pair "bikes1v3". The number of iterations and the found corresponding correct matches are shown under each figure. Correct matches are displayed in blue and false matches in red. Top row: the results of CPD. CPD iterates 434 times and finally finds 1107 correct matches. Bottom row: the results of AGMM. AGMM finds 1690 correct matches after only 53 iterations.

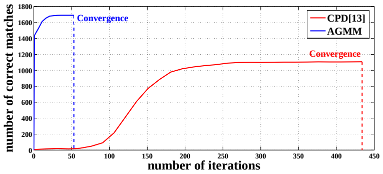


Figure 8: Plot of the number of correct matches versus the number of iterations for both CPD [13] and AGMM using the image pair "bikes1v3". The vertical dashed line indicates that the algorithm converges here.

m [13] and the proposed AGMM algorithm is shown in Fig. 7 and Fig. 8. We use the image pair "bikes1v3" in column 4 of Fig. 1 to illustrate the comparison process. The results show that the proposed AGMM algorithm converges much faster than the CPD algorithm and can find more correct matches. For all the forty image pairs in Mikolajczyk dataset [12] the average running time of AGMM and CPD is respectively about 90s and 630s. A PC, which is equipped with a 2.4GHz Intel i3 CPU and 4GB memory, is used.

As done for the SGMR algorithm, we also evaluate the proposed AGMM algorithm with respect to the values of parameters β and λ in Fig. 9 on all the forty image pairs in Mikolajczyk dataset [12]. We find the results also insensitive to the parameters like in the SGMR algorithm. Therefore, we can still set $\lambda \in [4, 12]$ and $\beta \in [2, 6]$ for the AGMM algorithm.

In Fig. 10, we visually illustrate the performance of the AGMM algorithm using the four image pairs in Fig. 1. We first discard matches(yellow arrows) that dissatisfy the epipolar geometry. Then $t_p = 0.3$ is used to eliminate unstable matches(green arrows). The red arrows are the mismatches evaluated by the method in [12] and the blue arrows are the final correct matches.

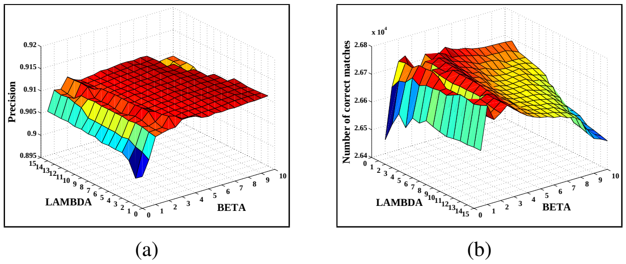


Figure 9: The precision rate(a) and number of correct matches(b) of AGMM over the forty image pairs [12] with different λ and β. As a tradeoff, we take $\lambda \in [4, 12]$ and $\beta \in [2, 6]$.

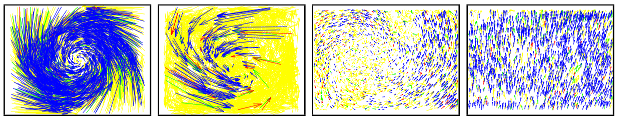
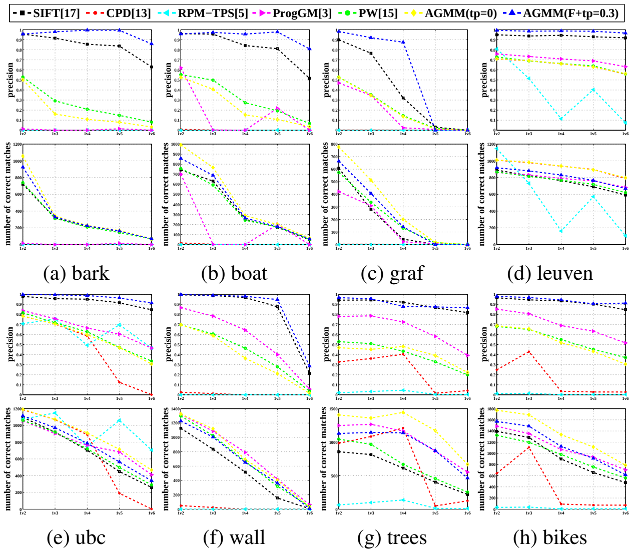


Figure 10: The illustration of the AGMM algorithm using the four image pairs in Fig. 1. See the text for details.



(a) bark (b) boat (c) graf (d) leuven

(e) ubc (f) wall (g) trees (h) bikes

Figure 11: Precision and number of correct matches for different methods: SIFT [17], CPD [13], RPM-TPS [5], ProgGM [3], PW [15] and AGMM algorithm. Two results, AGMM with $t_p = 0$ filtering(yellow line) and AGMM with $t_p = 0.3$ filtering(blue line) are plotted.

More comprehensive results are shown in Fig. 11. The results of all of the eight groups of image pairs in [12] by the AGMM algorithm and the other compared algorithms such as SIFT [11, 17], CPD [13], RPM-TPS [5], ProgGM [3] and PW [15] are illustrated. For each group of image pairs, the top is the precision and the bottom is the number of correct matches. Two results are given by the proposed AGMM algorithm, one is AGMM with $t_p = 0$ filtering and the other is AGMM with F+$t_p = 0.3$ filtering. The results show that the proposed AGMM algorithm can find much more correct matches with higher precision than SIFT match method. The CPD algorithm fails for most of the image pairs due to the lack of feature information in spite of its outstanding performance in point sets registration. Our proposed AGMM algorithm is also demonstrated to outperform the other compared algorithms in state-of-the-art. The AGMM algorithm with $t_p = 0$ filtering can find the most correct matches for almost all of forty image pairs. The AGMM algorithm with F+$t_p = 0.3$ filtering may sacrifice some correct matches but almost find the most correct matches with the highest precision among all the compared algorithms. Consider that CPD [13] and RPM-TPS [5] use only the position information of the point sets while our proposed AGMM algorithm and the other two compared algorithms PW [15] and ProgGM [3] use both the position information and feature information of the point sets. Therefore, to some degree, the comparison is unfair for the CPD [13] and RPM-TPS [5] algorithms since they can be effectively used in many other situations such as 2D and 3D point sets registration where the feature information is not available. Here the aim of our comparison just emphasizes two points: 1) The fusion of the position information and feature information is superior to the pure position information for point sets matching if the feature information is available; 2) Our proposed AGMM algorithm outperforms the other algorithms which combine the position information and feature information for point sets matching.

7. Conclusion

In this paper, we extend the classical SGMM-based point sets registration to point sets matching based on the AGMM. The earning for doing this includes three aspects: 1) the point pair with similar feature has more chance to form a correspondence, which directly leads to faster convergence than the SGMM-based method like CPD; 2) the robustness to outliers, deformation and rotation is extremely enhanced; 3) Much more correct matches can be found than the methods in state-of-the-art. One by-product of the AGMM is the Single Gauss Model for Mismatch Rejection (SGMR) algorithm, where the weights are binarized into $\{0, 1\}$. Thus, the AGMM framework is reduced to Gauss Model framework where only the given point in the data set has a Gauss distribution. A large number of qualitative and quantitative comparison experiments demonstrate good performance of the proposed SGMR and AGMM algorithms.

8. Acknowledgement

We would like to thank the reviewers for their constructive and valuable comments. We are grateful to Dr. Yongtao Wang for helpful discussions. The work is supported by National Natural Science Foundation of China(Grants 61371140 and 61273279).

References

[1] A. Albarelli, E. Rodol, and A. Torsello. Imposing semi-local geometric constraints for accurate correspondences selection in structure from motion: A game-theoretic perspective. *IJCV*, 97(1):36–53, 2012.

[2] T. Caetano, J. McAuley, L. Cheng, Q. V. Le, and A. Smola. Learning graph matching. *T-PAMI*, 31(6):1048–1058, 2009.

[3] M. Cho and K. M. Lee. Progressive graph matching: Making a move of graphs via probabilistic voting. In *CVPR*, pages 398–405, 2012.

[4] H. Chui and A. Rangarajan. A feature registration framework using mixture models. In *Mathematical Methods in Biomedical Image Analysis, IEEE Workshop on*, pages 190–197, 2000.

[5] H. Chui and A. Rangarajan. A new point matching algorithm for non-rigid registration. *Computer Vision and Image Understanding*, 89(2-3):114–141, 2003.

[6] R. Hamid, D. Decoste, and C. J. Lin. Dense non-rigid point-matching using random projections. In *CVPR*, pages 2914–2921, 2013.

[7] J. Heinly, E. Dunn, and J. M. Frahm. Comparative evaluation of binary features. In *ECCV*, pages 759–773, 2012.

[8] B. Jian and B. Vemuri. Robust point set registration using gaussian mixture models. *T-PAMI*, 33(8):1633–1645, 2011.

[9] K. Lebeda, J. Matas, and O. Chum. Fixing the locally optimized ransac. In *BMVC*, pages 95.1–95.11, 2012.

[10] X. Li and Z. Hu. Rejecting mismatches by correspondence function. *IJCV*, 89(1):1–17, 2010.

[11] D. G. Lowe. Distinctive image features from scale-invariant keypoints. *IJCV*, 60(2):91–110, 2004.

[12] K. Mikolajczyk and C. Schmid. A performance evaluation of local descriptors. *T-PAMI*, 27(10):1615–1630, 2005.

[13] A. Myronenko and X. B. Song. Point-set registration: Coherent point drift. *T-PAMI*, 32(12):2262–2275, 2010.

[14] D. Pizarro and A. Bartoli. Feature-based deformable surface detection with self-occlusion reasoning. *IJCV*, 97(1):54–70, 2012.

[15] M. Torki and A. Elgammal. One-shot multi-set non-rigid feature-spatial matching. In *CVPR*, pages 3058–3065, 2010.

[16] Q. H. Tran, T. J. Chin, G. Carneiro, M. S. Brown, and D. Suter. In defence of ransac for outlier rejection in deformable registration. In *ECCV*, pages 274–287, 2012.

[17] A. Vedaldi and B. Fulkerson. VLFeat: An open and portable library of computer vision algorithms. `http://www.vlfeat.org/`, 2008.

[18] J. Zhao, J. Ma, J. Tian, J. Ma, and D. Zhang. A robust method for vector field learning with application to mismatch removing. In *CVPR*, pages 2977–2984, 2011.