

A Robust Method for Vector Field Learning with Application to Mismatch Removing

Ji Zhao Jiayi Ma Jinwen Tian Jie Ma Dazhi Zhang
Institute for Pattern Recognition and Artificial Intelligence
Huazhong University of Science and Technology, Wuhan, China
{zhaoji84, jyma2010}@gmail.com jwttian@mail.hust.edu.cn

Abstract

We propose a method for vector field learning with outliers, called vector field consensus (VFC). It could distinguish inliers from outliers and learn a vector field fitting for the inliers simultaneously. A prior is taken to force the smoothness of the field, which is based on the Tikhonov regularization in vector-valued reproducing kernel Hilbert space. Under a Bayesian framework, we associate each sample with a latent variable which indicates whether it is an inlier, and then formulate the problem as maximum a posteriori problem and use Expectation Maximization algorithm to solve it. The proposed method possesses two characteristics: 1) robust to outliers, and being able to tolerate 90% outliers and even more, 2) computationally efficient. As an application, we apply VFC to solve the problem of mismatch removing. The results demonstrate that our method outperforms many state-of-the-art methods, and it is very robust.

1. Introduction

A vector field is a map that assigns each position $\mathbf{x} \in \mathbb{R}^P$ a vector $\mathbf{y} \in \mathbb{R}^D$ defined by a vector-valued function. Past empirical work has shown that, learning a vector-valued function exploiting the relationships between its components often leads to more accurate predictions than learning the outputs independently. However, it has received much less attention in machine learning.

Computing a function from sparse data by assuming an appropriate prior on the class of approximating functions is addressed in regularization theory. This problem is ill-posed since it has an infinite number of solutions. To make the result depend smoothly on the data, regularization techniques typically impose smoothness constraints on the approximating set of functions in a hypothesis space [7]. There is a particularly useful family of hypothesis spaces called reproducing kernel Hilbert spaces (RKHS) [1], each of which is

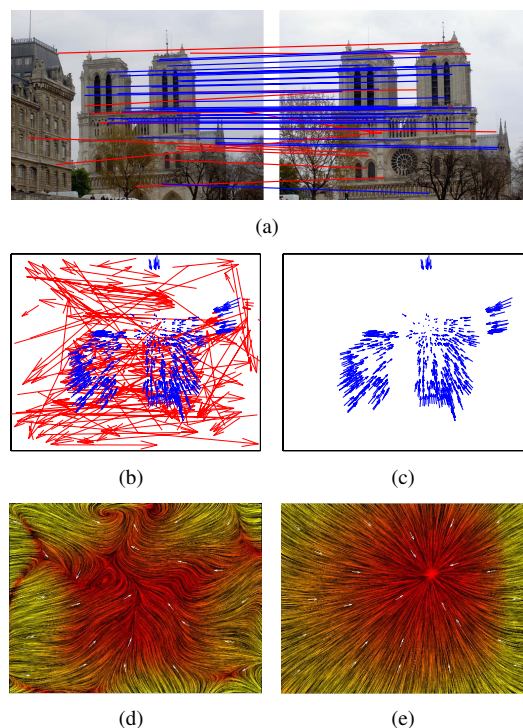


Figure 1. Mismatch removing and vector field learning. (a) An image pair and its putative matches. Blue and red lines represent correct matches and mismatches respectively. For visibility, only 50 randomly selected matches are presented. (b)(c) Vector field samples introduced by all matches and correct matches respectively. The head and tail of each quiver correspond to the positions of feature points in two images. (d)(e) Learned vector field using samples of (b) and (c) respectively. The visualization method is line integral convolution (LIC) [5], color indicates the magnitude of the vector at each point.

associated with a particular kernel, and the representer theorem for Tikhonov regularization in RKHS is widely applied. In this paper, we focus on RKHS of vector-valued functions. The vector field learning under the framework of vector-valued RKHS has been considered in [14], and they

generalized the representer theorem for Tikhonov regularization to the vector-valued setting. An extension work can be found in [2]. They studied a new class of regularized kernel methods for learning vector fields, which is based on filtering the spectrum of the kernel matrix.

There is a precondition for these methods that the given sparse data should not contain outliers. However, real-world observations are always perturbed by noise and outliers, sometimes the proportion of outliers is even very large. The existence of outlier could ruin the traditional vector field learning methods since they consider all samples as inliers. In this case, robust estimators should be developed to provide stable results. The main purpose of this paper is to learn the vector field from sparse data with outliers, as well as remove outliers. Compared to traditional methods, the difference is that we associate each sample with a latent variable which indicates whether it is an inlier, and then use expectation-maximization (EM) algorithm to find a maximum posterior (MAP) solution for the vector field by treating the latent variable as missing data.

Now considering figure 1, to illuminate our main idea, we take mismatch removing as an example. As shown in figure 1(a), blue lines denote inliers, red lines denote outliers. We first convert the matches into vector field training set which is shown in figure 1(b). The inliers are shown in figure 1(c). Using the traditional method, we obtain the vector field in figure 1(d) and (e) from the training set in figure 1(b) and (c) respectively. Obviously, the vector field in figure 1(d) is not the purpose, and the training set in figure 1(c) is not known in advance. Therefore, the problem is how to use the training set in figure 1(b) to estimate the vector field in figure 1(e), and this is the very goal in our work. There is another problem, considering figure 1(b) only, we may ask why the blue quivers indicate inliers, and the red quivers indicate outliers. This is because the blue quivers have “low frequency” in space domain, and they are consistent with the smoothness prior. However, if the ground truth itself contains the red quivers, which means the true vector field does not have “dominated frequency” in space domain, our method will not be suitable for this case, and this is beyond the scope of our work.

1.1. Related work

Yuille and Grzywacz proposed the motion coherence theory (MCT) for velocity field smoothing [21]. Myronenko and Song extended the MCT to point set registration and it is robust in the presence of outliers and noise [16]. These methods do not consider the interaction between the components of velocity field. Several recent works have been done on vector field learning. Micchelli and Pontil learned vector-valued functions under the framework of vector-valued RKHS [14]. Based on their work, Baldassarre *et al.* studied spectral filtering method for learning vector

fields [2]. Besides regularization method, vector field learning methods also include support vector regression [13] and sparse basis field method [8].

Our proposed vector field consensus (VFC) method is inspired by vector field learning [2] and robust model fitting [18]. The main contributions of our work include: i) we proposed a new vector field learning method called vector field consensus which can learn vector field from sparse samples with outliers; ii) we applied VFC to mismatch removing which is a fundamental problem in computer vision, and the results demonstrate that it outperforms many state-of-the-art methods. To the best of our knowledge, the vector field learning with outliers has not yet been studied.

2. Vector field learning with outliers

2.1. Problem formulation

Given a set of observed input-output pairs $S = \{(\mathbf{x}_n, \mathbf{y}_n) \in \mathcal{X} \times \mathcal{Y}\}_{n=1}^N$ by random sampling a vector field which contains some unknown outliers, where $\mathcal{X} \subseteq \mathbb{R}^P$ and $\mathcal{Y} \subseteq \mathbb{R}^D$ are input space and output space respectively, our purpose is to distinguish outliers from inliers and learn a mapping $\mathbf{f} : \mathcal{X} \rightarrow \mathcal{Y}$ to fit the inliers well, where $\mathbf{f} \in \mathcal{H}$, and we assume \mathcal{H} is a reproducing kernel Hilbert space (RKHS) [14].

In the following we make the assumption, without loss of generality, that for inliers, the noise is Gaussian with zero mean and uniform standard deviation σ ; and for outliers, the observations of output occur within a bounded region of \mathbb{R}^D , so the distribution is assumed to be uniform $\frac{1}{a}$, where a is just a constant (the volume of this region). Let γ be the percentage of inliers which we do not know in advance. Thus the likelihood is a mixture model of distributions for inliers and outliers:

$$p(\mathbf{Y}|\mathbf{X}, \boldsymbol{\theta}) = \prod_{n=1}^N p(\mathbf{y}_n|\mathbf{x}_n, \boldsymbol{\theta}) \\ = \prod_{n=1}^N \left(\frac{\gamma}{(2\pi\sigma^2)^{D/2}} e^{-\frac{\|\mathbf{y}_n - \mathbf{v}_n\|^2}{2\sigma^2}} + \frac{1-\gamma}{a} \right), \quad (1)$$

where $\boldsymbol{\theta} = \{\mathbf{f}, \sigma^2, \gamma\}$ is the set of unknown parameters, $\mathbf{X}_{N \times P} = (\mathbf{x}_1, \dots, \mathbf{x}_N)^T$, $\mathbf{Y}_{N \times D} = (\mathbf{y}_1, \dots, \mathbf{y}_N)^T$, $\mathbf{v}_n = \mathbf{f}(\mathbf{x}_n)$. Note that the uniform distribution function is nonzero only in a bounded region, here we omit the indicator function in it for clarity.

Considering the smoothness constraint, the prior of \mathbf{f} could be written as:

$$p(\mathbf{f}) \propto e^{-\frac{\lambda}{2} \|\mathbf{f}\|_{\mathcal{H}}^2}, \quad (2)$$

where $\lambda > 0$ is the regularization parameter, $\|\cdot\|_{\mathcal{H}}^2$ is the norm in the RKHS \mathcal{H} .

Given the likelihood (1) and prior (2), the posterior distribution $p(\boldsymbol{\theta}|\mathbf{X}, \mathbf{Y})$ could be estimated by applying Bayes

rule: $p(\boldsymbol{\theta}|\mathbf{X}, \mathbf{Y}) \propto p(\mathbf{Y}|\mathbf{X}, \boldsymbol{\theta})p(\mathbf{f})$. In order to optimally estimate $\boldsymbol{\theta}$ a MAP solution, $\boldsymbol{\theta}^*$, is made such that

$$\boldsymbol{\theta}^* = \arg \max_{\boldsymbol{\theta}} p(\mathbf{Y}|\mathbf{X}, \boldsymbol{\theta})p(\mathbf{f}), \quad (3)$$

with $\boldsymbol{\theta}^*$ corresponding to the estimate of the true $\boldsymbol{\theta}$. Thus the vector field \mathbf{f} will be obtained. However, seeking a good solution for \mathbf{f} implies weakening the influence of outliers. In this paper, we cope with this problem under an EM framework which we will discuss in the following subsections.

2.2. An EM solution

There are several ways to estimate the parameters of the mixture model, such as EM algorithm, gradient descent, variational inference, etc. The EM algorithm provides a natural framework for solving this problem. EM alternates two steps, an expectation step (E-step) and a maximization step (M-step). In the E-step, responsibilities for samples belonging to inlier are estimated based on the currently best estimate of vector field \mathbf{f} . In the M-step, a maximum likelihood vector field \mathbf{f} is estimated based on the responsibilities computed in the E-step. The E-step can be interpreted as inlier detection with a fixed vector field \mathbf{f} , whereas the M-step implements vector field learning under the assumption that the responsibilities are known.

We associate sample n with a latent variable $z_n \in \{0, 1\}$, where $z_n = 1$ represents Gaussian distribution and $z_n = 0$ represents uniform distribution. Following standard texts [4] and omitting terms that are independent of $\boldsymbol{\theta}$, the complete-data log posterior (i.e. the sum of log likelihood and log prior) is given by

$$\begin{aligned} \mathcal{Q}(\boldsymbol{\theta}, \boldsymbol{\theta}^{\text{old}}) &= -\frac{1}{2\sigma^2} \sum_{n=1}^N P(z_n = 1 | \mathbf{x}_n, \mathbf{y}_n, \boldsymbol{\theta}^{\text{old}}) \|\mathbf{y}_n - \mathbf{v}_n\|^2 \\ &\quad - \frac{D}{2} \ln \sigma^2 \sum_{n=1}^N P(z_n = 1 | \mathbf{x}_n, \mathbf{y}_n, \boldsymbol{\theta}^{\text{old}}) \\ &\quad + \ln \gamma \sum_{n=1}^N P(z_n = 1 | \mathbf{x}_n, \mathbf{y}_n, \boldsymbol{\theta}^{\text{old}}) \\ &\quad + \ln(1 - \gamma) \sum_{n=1}^N P(z_n = 0 | \mathbf{x}_n, \mathbf{y}_n, \boldsymbol{\theta}^{\text{old}}) \\ &\quad - \frac{\lambda}{2} \|\mathbf{f}\|_{\mathcal{H}}^2. \end{aligned} \quad (4)$$

This may be maximized by treating the z_n as missing data from the mixture model.

E-step: Denote $\mathbf{P} = \text{diag}(p_1, \dots, p_N)$, where the responsibility $p_n = P(z_n = 1 | \mathbf{x}_n, \mathbf{y}_n, \boldsymbol{\theta}^{\text{old}})$ can be computed

by applying Bayes rule:

$$p_n = \frac{\gamma e^{-\frac{\|\mathbf{y}_n - \mathbf{v}_n\|^2}{2\sigma^2}}}{\gamma e^{-\frac{\|\mathbf{y}_n - \mathbf{v}_n\|^2}{2\sigma^2}} + (1 - \gamma) \frac{(2\pi\sigma^2)^{D/2}}{a}}. \quad (5)$$

The posterior probability p_n is a soft decision, which indicates to what degree the sample n agrees with the current estimated vector field \mathbf{f} .

M-step: We determine the revised parameter estimate $\boldsymbol{\theta}^{\text{new}}$ as follows: $\boldsymbol{\theta}^{\text{new}} = \arg \max_{\boldsymbol{\theta}} \mathcal{Q}(\boldsymbol{\theta}, \boldsymbol{\theta}^{\text{old}})$. Considering \mathbf{P} is diagonal matrix, take derivative of $\mathcal{Q}(\boldsymbol{\theta})$ with respect to σ^2 and set it to zero, we obtain

$$\sigma^2 = \frac{\text{tr}[(\mathbf{Y} - \mathbf{V})^T \mathbf{P}(\mathbf{Y} - \mathbf{V})]}{D \cdot \text{tr}(\mathbf{P})}, \quad (6)$$

where $\text{tr}(\cdot)$ is the trace.

Take derivative of $\mathcal{Q}(\boldsymbol{\theta})$ with respect to γ and set it to zero, we obtain

$$\gamma = \frac{\text{tr}(\mathbf{P})}{N}. \quad (7)$$

The mixing coefficient γ for Gaussian component is given by the average responsibility which the Gaussian component takes for explaining the samples.

Maximization \mathcal{Q} with respect to \mathbf{f} is complex, and we will discuss it in the next subsection.

After the EM converges, it should make a decision which samples are inliers. With a predefined threshold τ , we obtain inlier set $T = \{n | p_n > \tau, n = 1, \dots, N\}$. When making such hard decision, the set of T is the so-called consensus set in Random Sample Consensus (RANSAC) [6], so we call our method *vector field consensus* (VFC).

2.3. Vector field regularization using matrix kernel

Considering the terms of objective function \mathcal{Q} in (4) that are related to \mathbf{f} , and multiply them by -1, we get an energy function:

$$\mathcal{E}(\mathbf{f}) = \frac{1}{2\sigma^2} \sum_{n=1}^N p_n \|\mathbf{y}_n - \mathbf{f}(\mathbf{x}_n)\|^2 + \frac{\lambda}{2} \|\mathbf{f}\|_{\mathcal{H}}^2. \quad (8)$$

Then the maximization of \mathcal{Q} with respect to \mathbf{f} is equivalent to minimize energy function $\mathcal{E}(\mathbf{f})$. This energy function is a vector-valued extension of Tikhonov regularization, and the first term could be seen as weighted empirical error.

Taking advantage of the equivalence of RKHS and kernel, we define the hypothesis space \mathcal{H} using matrix kernel. Matrix kernel is a symmetric matrix valued mapping $\Gamma : \mathbb{R}^P \times \mathbb{R}^P \rightarrow \mathbb{R}^{D \times D}$ that satisfies a positivity constraint.

Using the vector-valued representer theorem, the optimal \mathbf{f} has this form [2, 14]:

$$\mathbf{f}(\mathbf{x}) = \sum_{n=1}^N \Gamma(\mathbf{x}, \mathbf{x}_n) \mathbf{c}_n, \quad (9)$$

with the coefficient c_n determined by a linear system

$$(\tilde{\mathbf{\Gamma}} + \lambda\sigma^2\tilde{\mathbf{P}}^{-1})\mathbf{C} = \tilde{\mathbf{Y}}, \quad (10)$$

where kernel matrix $\tilde{\mathbf{\Gamma}}$ and $\tilde{\mathbf{P}}$ are $N \times N$ block matrices, each block is a $D \times D$ scalar matrix. The (i, j) -th block of $\tilde{\mathbf{\Gamma}}$ is $\mathbf{\Gamma}(\mathbf{x}_i, \mathbf{x}_j)$. $\tilde{\mathbf{P}} = \mathbf{P} \otimes \mathbf{I}_{D \times D}$, where \otimes denotes Kronecker product. $\mathbf{c}_n \in \mathbb{R}^{D \times 1}$, $\mathbf{C} = (\mathbf{c}_1^T, \dots, \mathbf{c}_N^T)^T$, $\tilde{\mathbf{Y}} = (\mathbf{y}_1^T, \dots, \mathbf{y}_N^T)^T$ are $D \times N$ dimensional vectors.

We also provide a fast implementation for VFC. Notice that the coefficient matrix in linear system (10) is positive definite, so the low rank matrix approximation could be used to reduce the complexity to linear as in [17, 16]. We call this implementation *FastVFC*.

We summarize the vector field consensus method in algorithm 1.

Algorithm 1: The Vector Field Consensus (VFC) Algorithm

Input: Training set $S = \{(\mathbf{x}_n, \mathbf{y}_n)\}_{n=1}^N$, kernel $\mathbf{\Gamma}$
Output: Vector field \mathbf{f} , inliers set T

- 1 Initialize λ, τ, γ ;
 - 2 Initialize $\mathbf{V} = \mathbf{X}$, $\mathbf{P} = \mathbf{I}_{N \times N}$;
 - 3 Set a to be the volume of the output space;
 - 4 Initialize σ^2 by formula (6);
 - 5 Construct kernel matrix $\tilde{\mathbf{\Gamma}}$ using the definition of $\mathbf{\Gamma}$;
 - 6 **repeat**
 - 7 *E-step:*
 - 8 Update $\mathbf{P} = \text{diag}(p_1, \dots, p_N)$ by formula (5);
 - 9 *M-step:*
 - 10 Update \mathbf{C} by solving linear system (10);
 - 11 Update \mathbf{V} by using $\mathbf{v}_n = \mathbf{f}(\mathbf{x}_n)$ and formula (9);
 - 12 Update σ^2 and γ by formulas (6) and (7);
 - 13 **until** \mathcal{Q} converges;
 - 14 Vector field \mathbf{f} is determined by formula (9);
 - 15 The inlier set is $T = \{n | p_n > \tau, n = 1, \dots, N\}$.
-

3. Removing mismatch by VFC algorithm

To validate the effectiveness of vector field consensus method, we apply it to mismatch removing problem. First of all, we show how the putative match set can be naturally modeled as vector field training set. Then we discuss some key issues when applying VFC to mismatch removing problem.

3.1. Vector field introduced by image pairs

Assume a match is comprised by a pair $(\mathbf{u}_i, \mathbf{u}'_i)$, where \mathbf{u}_i and \mathbf{u}'_i are positions of the two feature points in template image and reference image respectively. We made a linear re-scaling of the correspondences, so that the position in template and reference image have zero mean and

unit variance. Suppose the normalized match is $(\hat{\mathbf{u}}_i, \hat{\mathbf{u}}'_i)$. We convert the normalized match into a vector field sample by a transformation $(\hat{\mathbf{u}}_i, \hat{\mathbf{u}}'_i) \rightarrow (\mathbf{x}, \mathbf{y})$, where $\mathbf{x} = \hat{\mathbf{u}}_i$, $\mathbf{y} = \hat{\mathbf{u}}'_i - \hat{\mathbf{u}}_i$.

3.2. Kernel selection

By choosing different kernels, the norm in the corresponding RKHS encodes different notions of smoothness. Usually, for mismatch removing problem, the structure of the generated vector field is simple. We find decomposable kernel is adequate for solving this problem, then the vector field learning problem can be decomposed into D essentially independent scalar problems [2]. Suppose the kernel is:

$$\mathbf{\Gamma}(\mathbf{x}_i, \mathbf{x}_j) = \kappa(\mathbf{x}_i, \mathbf{x}_j)\mathbf{A}, \quad (11)$$

where scalar kernel κ encodes the similarity between the inputs, and matrix \mathbf{A} encodes the relationships between the outputs. The matrix kernel could exploit the relationships among the components of the vector field.

For scalar kernel κ , we choose Gaussian kernel as follows

$$\kappa(\mathbf{x}_i, \mathbf{x}_j) = e^{-\beta\|\mathbf{x}_i - \mathbf{x}_j\|^2}. \quad (12)$$

From the Fourier perspective, the regularization can be written in terms of a penalty in the frequency domain. By choosing an appropriately sized Gaussian filter we have the flexibility to control the range of filtered frequencies and thus the amount of spatial smoothness [16].

For the relationship matrix \mathbf{A} , we choose graph regularization [2]. A particular form is $\mathbf{A} = \omega\mathbf{1} + (1 - \omega D)\mathbf{I}$. In practice, we found that just an identity matrix could work quite well, so we set ω to 0 in this paper.

3.3. Computational complexity

For vector-valued Tikhonov regularization, the solution of $\mathbf{f}(\mathbf{x})$ is found by inverting a matrix whose dimension is $(DN) \times (DN)$. The time complexity is $O((DN)^3)$. For mismatch removing problem, we choose a decomposable kernel. By redefining a coordinate system, the vector field learning problem can be reduced to solving D scalar regularization problems[2]. This particular kernel can reduce the time complexity to $O(DN^3)$. Usually, $D \ll N$, and for mismatch removing problem, $D = 2$.

In our current implementation, we just use MATLAB “slash” operator, which implicitly use Cholesky decomposition to inverse matrix. For FastVFC, its total time complexity could be reduced to $O(mDN)$, where m is the iterative times for EM. Our experiments demonstrate that FastVFC is faster than Cholesky decomposition method, while with little performance degradation.

3.4. Implementation details

In practice, we update γ using a little different form from formula (7). After we obtain the posterior probability p_n in

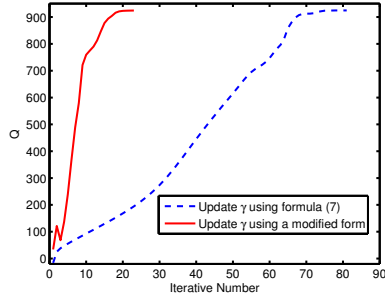


Figure 2. The change of complete-data log posterior Q during the EM iterations. (The experiment is performed on image pairs of *Mex* with ratio test threshold 1.0. The initial correct match percentage is 11.65%.)

each iteration, we estimate inlier set $T = \{n | p_n > \tau, n = 1, \dots, N\}$, and then we update gamma as $\gamma = |T|/N$, where $|\cdot|$ denotes the cardinality of the inliers. When EM converges, we fix γ as the current value and redo the EM until convergence. And for numerical stability, we compel $\gamma \in [0.05, 0.95]$. We find that this trick accelerates the convergence and is more robust when the inliers percentage is low.

When the initial correct match percentage is high, the two update methods of γ performs nearly the same. For low initial correct match percentage, figure 2 illuminates a typical process of convergence for 11.65% inliers. We can see that update using the previous trick is faster than updating using formula (7). Even though EM algorithm gives theoretical guarantees of obtaining local minimum only if we using formula (7), we found that in practice this trick does not hurt the performance, on the contrary, it makes the algorithm more efficient.

Now considering the linear system (10), the matrix inversion operation will cause some problem when the matrix \mathbf{P} is singular. For numerical stability, we cope with this problem by defining a lower bound ε . Diagonal elements of \mathbf{P} that below ε is set as ε . In this paper, we set ε as 10^{-5} .

Parameters initialization: There are mainly four parameters in VFC algorithm: β , λ , τ and γ . Parameters β and λ both reflect the amount of smoothness regularization. Parameter β determines how wide the range of interaction between samples. Parameter λ controls the trade-off between the closeness to the data and the smoothness of the solution. A detailed description of them can be found in [21]. Parameter τ is a threshold, which is used for deciding the correctness of a match. And parameter γ reflects our initial assumption on the amount of inliers in correspondence sets. In practise, we find our method is insensitive to parameters. We set $\beta = 0.1$, $\lambda = 3$, $\tau = 0.75$ and $\gamma = 0.9$ through this paper. The constant a is set as the area of the reference image after linear re-scaling.

4. Experimental setup

We test our method on dataset of Mikolajczyk *et al.* [15] and Tuytelaars *et al.* [19]. The image pairs in the first dataset are either of planar scenes or the camera position is fixed during acquisition. The images are, therefore, always related by a homography. The ground truth homographies are supplied by the dataset. The test data of Tuytelaars contains several wide baseline image pairs. For further details about the datasets we refer the reader to [15, 19].

We use the open source VLFEAT toolbox [20] to determine the initial matches of SIFT [12]. All parameters are set as the default values except the ratio test threshold t . Usually, the greater value of t indicates the smaller amount of matches with higher correct match percentage. The default value of t is 1.5, and the smallest value is 1.0, which is the nearest neighbor strategy.

We measure the performance of VFC for vector learning by an angular measure of error [3, 2] between the learned vector of VFC and the ground truth. If $\mathbf{v}_g = (v_g^1, v_g^2)$ and $\mathbf{v}_e = (v_e^1, v_e^2)$ are the ground truth and estimated fields, we consider the transformation $\mathbf{v} \rightarrow \tilde{\mathbf{v}} = \frac{1}{\|(v^1, v^2, 1)\|} (v^1, v^2, 1)$. The error measure is defined as $err = \arccos(\tilde{\mathbf{v}}_e, \tilde{\mathbf{v}}_g)$.

The match correctness is determined as follows. On the dataset of Mikolajczyk *et al.*, the authors use the overlap error ε_S to determine the match correctness. We use a similar criterion. We just reduce the scale of feature point to 1/3 of original scale, and we regard a match as correct match if $\varepsilon_S > 0$. In our experiment, we find this method is consistent with human's perception. On the dataset of Tuytelaars, we use a method combining subjectivity and objectivity. We first generate matches by using VLFEAT with the default parameters, which will generate high accuracy matches. And then we fit the fundamental matrix by RANSAC and use it to determine the match correctness. We further confirm them artificially. Though the judgment of correct match and mismatch seems arbitrary, we make the benchmark before performing experiments to ensure objectivity.

5. Experimental results

We test the performance of our proposed VFC and verify the validity of vector field learning on real images. The experiments are done from the following three aspects: (1) results on some wide baseline images; (2) vector field learning and mismatch removing performance on a dataset; (3) robustness.

5.1. Results on wide baseline images

We test VFC method on wide baseline image pairs, as shown in figure 3. One image pair is a structured scene, and the other image pair is a unstructured scene. The training

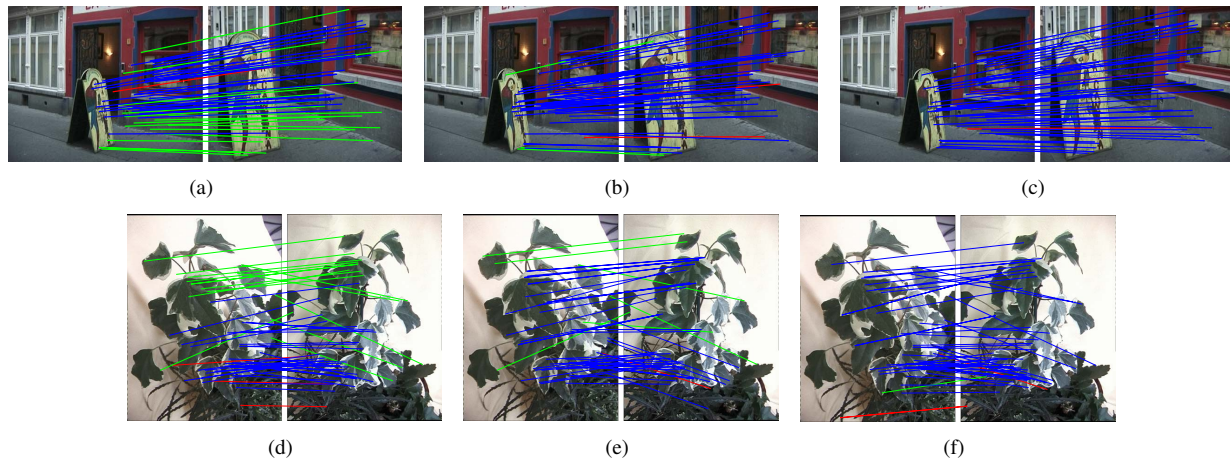


Figure 3. Experimental results for image pairs of *Mex* and *Tree*. (a-c) Results of ICF, GS and VFC for *Mex* respectively. The initial correct match percentage is 51.90%, and the precision-recall pairs are (96.15%, 60.98%), (93.83%, 92.68%) and (96.47%, 100.00%) respectively. (d-f) Results of ICF, GS and VFC for *Tree* respectively. The initial correct match percentage is 56.29%, and the precision-recall pairs are (92.75%, 68.09%), (97.62%, 87.23%) and (94.85%, 97.87%) respectively. The lines indicate inlier detection result (blue = true positive, green = false negative, red = false positive). For visibility, only 50 randomly selected correspondences are presented, and the true negatives are not shown. Best viewed in color.

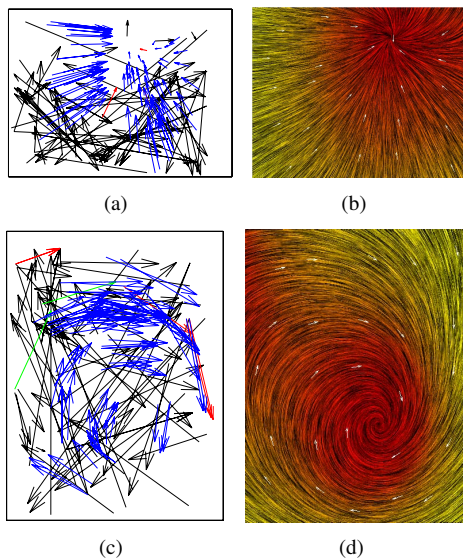


Figure 4. Vector field learning results for image pairs of *Mex* and *Tree*. (a)(b) Inlier detection result and the learned field for *Mex*. (c)(d) Inlier detection result and the learned field for *Tree*. The quivers indicate inlier detection result (blue = true positive, black = true negative, green = false negative, red = false positive). Best viewed in color.

samples of vector fields and the learned vector fields are shown in figure 4. We can see that even for complex vector fields, VFC can learn them from noisy samples.

The performance for mismatch removing is characterized by precision and recall. We compare our VFC algorithm with two state-of-the-art mismatch removing meth-

ods, Identifying point correspondences by Correspondence Function (ICF) [9] and Graph Shift (GS) [10, 11]. The ICF method uses support vector regression to learn a correspondence function pair which map points in one image to their corresponding points in another, and then reject the mismatches by checking whether they are consistent with the estimated correspondence functions. The GS method constructs affinity graph via spatially coherent correspondences, and then optimizes objective function by spectral method. We implemented ICF and set all parameters according to its default parameters. For GS, we implemented it based on the publicly available core code supplied by the authors. The parameters are set according to the original paper, and we try our best to tune some details. In all the experiment, three algorithms' parameters are fixed.

The performance of VFC compared to other two approaches is show in figure 3. The results suggest that VFC can distinguish inliers from outliers, and has the best trade-off between precision and recall.

5.2. Results on a dataset

In this subsection, we test our method for vector field learning and mismatch removing on the dataset of Mikolajczyk *et al.* We use all the 40 image pairs, and for each pair, we set the SIFT ratio test threshold t as 1.5, 1.3, 1.0 respectively. The cumulative distribution function of original correct match percentage is shown in figure 5(a). The initial average precision of all image pairs is 69.57%, and about 30 percent of the training sets have correct match percentage below 50%, so the dataset is challenging for mismatch removing.

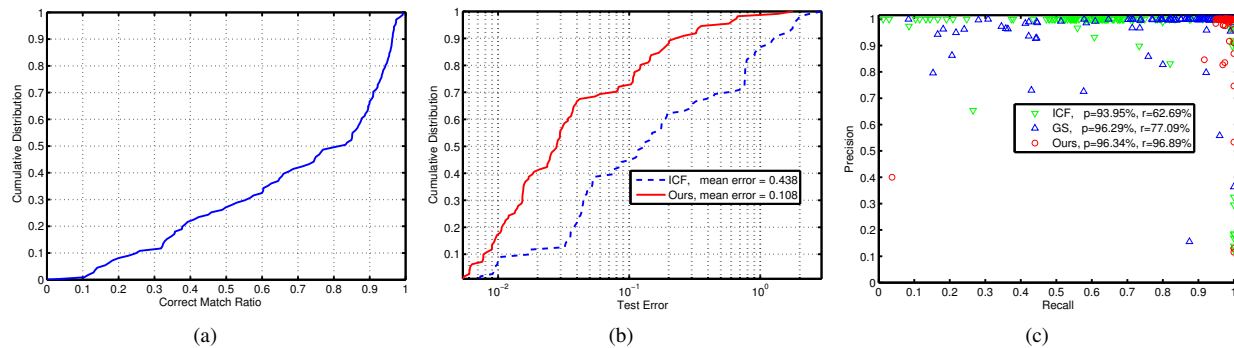


Figure 5. Experimental results on the dataset of Mikolajczyk *et al.* (a) Cumulative distribution function of original correct match percentage. (b) Cumulative distribution functions, test error of learned vector field comparison between ICF and VFC. (c) Precision-recall statistics for ICF, GS, and our method. Our method (red circles, upper right corner) has the best precision and recall overall.

The image pairs in this dataset satisfy homographies. As a result, the ground truth vector fields can be calculated precisely. The error between two vectors is defined in experimental setup. The test error of an image pair is estimated as this: after learning the vector field, we use it to test the displacement vectors of all pixels in the template image, and compare them to the ground truth, the mean error is then regarded as the test error. We compare VFC with ICF, since ICF implicitly estimates a two-dimensional vector field by estimating its components independently. Figure 5(b) gives the cumulative distribution functions of test error. It is obviously that the vector fields estimated by VFC have less error than ICF.

Figure 5(c) gives the precision-recall plot of three methods. The average precision-recall pairs are (93.95%, 62.69%), (96.29%, 77.09%) and (96.34%, 96.89%) for ICF, GS and VFC respectively. As shown, when the initial correct match percentage is high, all the three methods perform well. When mismatch percentage is high or the viewpoint change is large, ICF tends to select all samples as support vectors, and set all samples as inliers. Or on the contrary, it selects very few samples as support vectors, and set other samples as outliers. As a result, ICF is not robust for such cases and usually has high precision or recall, but not simultaneously. GS has high precision and low recall. Perhaps this is because GS can not estimate the factor for affinity matrix automatically and it is not affine-invariant. Our proposed method VFC has the best precision-recall trade-off. And we found that the mismatch removing capability of VFC is not affected by the large view angle, image rotation and affine transformation since these cases are all contained in the dataset. In fact, VFC performs well except that the initial correct matches is very few.

The average run time of VFC and FastVFC on this dataset is about 9.8s and 0.5s per image pair on a 2.5 GHz PC respectively. The FastVFC is much faster than the other two methods.

5.3. Robustness test

We test the robustness of VFC on two image pairs, *Graf* and *Church*, as shown in figure 6. The *Graf* pair has large affine deformation, and the *Church* pair is a wide baseline image pair. For each image pair, we generate six training sets by the following procedure: firstly, the ratio test threshold is set as 1.5, 1.3 and 1.0 respectively, and then we fix threshold to 1.0 and add 1000, 2000, 4000 random mismatches respectively.

We test our method and compare it to RANSAC, ICF and GS, as shown in table 1 and table 2. It can be seen that the performance of VFC is quite satisfactory, and it can tolerate 90% outliers and even more. As the correct match percentage decrease, the precision and recall of VFC decrease gradually. Still, the results are acceptable comparing to other three methods.

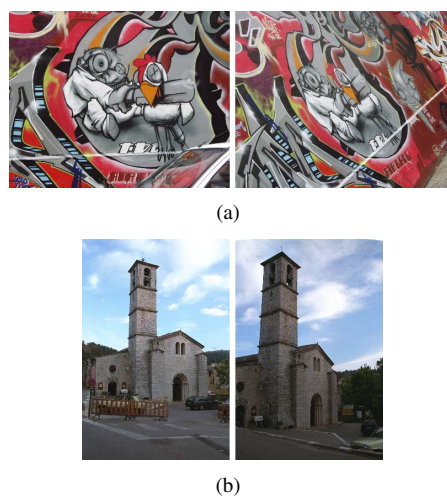


Figure 6. Image pairs used for robustness test. (a) *Graf*; (b) *Church*.

Table 1. Performance on figure 6(a). The pairs in the table are precision-recall pairs (%).

match percentage	37.74%	34.67%	13.44%	8.54%	6.26%	4.08%
RANSAC [6]	(75.95, 100.00)	(78.26, 96.43)	(67.65, 98.29)	-	-	-
ICF [9]	(89.80, 73.33)	(93.15, 60.71)	(18.54, 100.00)	(8.54, 100.00)	(6.26, 100.00)	(4.08, 100.00)
GS [10, 11]	(82.76, 80.00)	(85.86, 75.89)	(92.86, 44.44)	(92.11, 29.91)	(85.45, 20.09)	(85.71, 17.95)
VFC	(86.96, 100.00)	(91.57, 100.00)	(91.67, 98.72)	(91.34, 99.15)	(88.72, 73.93)	(86.60, 71.79)

Table 2. Performance on figure 6(b). The pairs in the table are precision-recall pairs (%).

match percentage	54.76%	37.25%	9.56%	5.32%	3.69%	2.28%
RANSAC [6]	(94.52, 100.00)	(83.50, 92.47)	(54.31, 89.17)	-	-	-
ICF [9]	(91.67, 63.77)	(100.00, 21.51)	(13.29, 100.00)	(5.39, 100.00)	(3.69, 100.00)	(2.28, 100.00)
GS [10, 11]	(91.78, 97.10)	(92.31, 90.32)	(84.44, 63.33)	(88.71, 45.85)	(86.05, 30.83)	(84.00, 17.50)
VFC	(98.33, 85.51)	(94.25, 88.17)	(90.76, 90.00)	(95.24, 83.33)	(86.96, 83.33)	(85.47, 83.33)

6. Conclusion

In this paper, we investigate a robust vector field learning method, called *vector field consensus* (VFC), and apply it to mismatch removing problem. It can detect outliers and fit a smooth vector field from a training set. VFC alternatively recovers the vector field and estimates the consensus set. The experiments on mismatch removing demonstrate that VFC is very robust, it can tolerate 90% mismatches and even more. The quantitative results on a dataset show that VFC outperforms state-of-the-art mismatch removing methods. We also provide an efficient implementation of VFC called FastVFC, which has linear time complexity.

Acknowledgement

We are grateful to Yuan Gao, Xiangru Li, Wenbing Tao, Yongtao Wang, Changcai Yang and Huabing Zhou for helpful discussions. The work is supported by National Natural Science Foundation of China (61074156).

References

- [1] N. Aronszajn. Theory of reproducing kernels. *Transactions of American Mathematical Society*, 68(3):337–404, 1950.
- [2] L. Baldassarre, L. Rosasco, A. Barla, and A. Verri. Vector field learning via spectral filtering. In *ECML*, 2010.
- [3] J. Barron, D. Fleet, and S. Beauchemin. Performance of optical flow techniques. *IJCV*, 12(1):43–77, 1994.
- [4] C. M. Bishop. *Pattern Recognition and Machine Learning*. Springer, 2006.
- [5] B. Cabral and L. C. Leedom. Imaging vector fields using line integral convolution. *Computer Graphics (SIGGRAPH '93 Proc.)*, 27:263–270, 1993.
- [6] M. A. Fischler and R. C. Bolles. Random sample consensus: A paradigm for model fitting with application to image analysis and automated cartography. *Communications of the ACM*, 24(6):381–395, 1981.
- [7] F. Girosi, M. Jones, and T. Poggio. Regularization theory and neural networks architectures. *Neural Computation*, 7(2):219–269, 1995.
- [8] S. Haufe, V. V. Nikulin, A. Ziehe, K.-R. Müller, and G. Nolte. Estimating vector fields using sparse basis field expansions. In *NIPS*, 2009.
- [9] X. Li and Z. Hu. Rejecting mismatches by correspondence function. *IJCV*, 89(1):1–17, 2010.
- [10] H. Liu and S. Yan. Common visual pattern discovery via spatially coherent correspondence. In *CVPR*, 2010.
- [11] H. Liu and S. Yan. Robust graph mode seeking by graph shift. In *ICML*, 2010.
- [12] D. Lowe. Distinctive image features from scale-invariant keypoints. *IJCV*, 60(2):91–110, 2004.
- [13] I. Macêdo and R. Castro. Learning divergence-free and curl-free vector fields with matrix-valued kernels. Technical report, Instituto Nacional de Matematica Pura e Aplicada, Brasil, 2008.
- [14] C. A. Micchelli and M. Pontil. On learning vector-valued functions. *Neural Computation*, 17(1):177–204, 2005.
- [15] K. Mikolajczyk, T. Tuytelaars, C. Schmid, A. Zisserman, J. Matas, F. Schaffalitzky, T. Kadir, and L. van Gool. A comparison of affine region detectors. *IJCV*, 65(1):43–72, 2005.
- [16] A. Myronenko and X. Song. Point set registration: Coherent point drift. *IEEE TPAMI*, 2010.
- [17] R. M. Rifkin. *Everything Old Is New Again: A Fresh Look at Historical Approaches in Machine Learning*. PhD thesis, MIT, 2002.
- [18] P. H. S. Torr and A. Zisserman. MLESAC: A new robust estimator with application to estimating image geometry. *CVIU*, 78(1):138–156, 2000.
- [19] T. Tuytelaars and L. van Gool. Matching widely separated views based on affine invariant regions. *IJCV*, 59(1):61–85, 2004.
- [20] A. Vedaldi and B. Fulkerson. VLFeat - An open and portable library of computer vision algorithms. In *MM*, 2010.
- [21] A. L. Yuille and N. M. Grzywacz. A mathematical analysis of the motion coherence theory. *IJCV*, 3:155–175, 1989.